

© 2023 Haofeng Sun

DESIGN AUTOMATION AND ANALYSIS OF POWER DISTRIBUTION
NETWORKS AND CHIPLET-TO-CHIPLET COMMUNICATION
INTERFACES IN 2.5D ADVANCED PACKAGING TECHNOLOGIES

BY

HAOFENG SUN

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois Urbana-Champaign, 2023

Urbana, Illinois

Adviser:

Professor José E. Schutt-Ainé

ABSTRACT

This thesis presents a comprehensive design automation workflow for modeling and analyzing the power distribution networks in 2.5D advanced packaging for heterogeneous integration applications. Challenges related to the routing schemes, 3D model construction time, and simulation complexity are identified, and temporary solutions are deliberated. Multiple test cases involving power distribution networks and high-speed via pairs are implemented with the automation workflow, and the results are analyzed. Finally, bunch of wires, an open source chiplet-to-chiplet interface, is introduced and the progress towards the design automation is discussed.

To my parents, for their love and support.

ACKNOWLEDGMENTS

I fondly recall grappling with Fourier transforms and Laplace transforms six years ago while enrolled in Professor Songbin Gong’s ECE210: Analog Signal Processing. Little did I envision that, six years later, I would serve as the TA for signal processing for over two years! Completing this master’s thesis in ECE ILLINOIS marks a significant milestone in my life and the culmination of my academic journey at the University of Illinois over six and a half years.

This achievement would not have been possible without the unwavering guidance and support of my adviser, Professor José Schutt-Ainé. I extend my sincere appreciation to him for welcoming me into the research group, teaching invaluable courses, offering insightful suggestions for my research, and providing constant encouragement. His passion and attitude towards academic research will undoubtedly leave a profound impact on my future career.

I am deeply grateful to Dr. Thong Nguyen for his exceptional talent and passion in research. His prompt and detailed explanations have been invaluable in guiding me through my research journey. I vividly remember the initial stages of my research project when I was unfamiliar with packaging theory and simulation tools. Dr. Nguyen’s patient guidance, useful references, and inspirational insights into novel research topics were instrumental in shaping my understanding. Our lengthy discussions about the automation script highlighted the importance of considering the bigger picture and optimizing code.

I express my appreciation to the members of the EM Lab who shared the same office space with me, including PhD candidate Bobi Shi, PhD candidate Yi Zhou, PhD candidate Juhitha Konduru, Xingjian (Victor) Shangguan, Dr. Yixuan (Nancy) Zhao, and Prof. Hanzhi Ma. Our time together and the enlightening conversations, both on research and personal topics, were truly memorable, especially during the most stressful periods. I will miss

each one of you.

Furthermore, I extend my thanks to all my students in the signal processing classes over two and a half years. Witnessing the brilliance in those classes and interacting with students who may shape the future of the hardware industry has been rewarding. Being a TA has transformed me from someone timid to one who confidently delivers review lectures in front of hundreds of students. Thank you all for being my students, and I hope you gained valuable insights from my labs and office hours.

Finally, my deepest gratitude goes to my parents for their unwavering love and support. People often underestimate what they can achieve in five years. Similarly, as an undergraduate, I could never have imagined teaching signal processing classes, interning at a major tech company, and completing this master's thesis. It underscores the importance of having faith in oneself. As for the next five years, only time will tell.

TABLE OF CONTENTS

LIST OF TABLES	viii
LIST OF FIGURES	ix
LIST OF ABBREVIATIONS	xi
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 BACKGROUND	4
2.1 Packaging Technologies	4
2.2 Network Parameters	8
CHAPTER 3 DESIGN AUTOMATION FOR PDN	19
3.1 Initiate SBU Class Object	20
3.2 Adding Patterns	25
3.3 Floor Plan	27
3.4 Building model in HFSS	29
CHAPTER 4 DESIGN CHALLENGES IN PDN AND POSSI- BLE SOLUTIONS	36
4.1 Routing Challenges	36
4.2 Timing Challenges	37
4.3 Solution	38
CHAPTER 5 TEST CASES	41
5.1 PDN with 4 Chiplets in a Single Voltage Domain	41
5.2 PDN with 4 Chiplets in Two Voltage Domains	43
5.3 Differential Via Pair	44
CHAPTER 6 BUNCH OF WIRES - A CHIPLET-TO-CHIPLET INTERFACE	54
6.1 BoW Standard Specification	54
6.2 Design Automation for BoW	56
6.3 Embedded Multi-die Interconnect Bridge (EMIB)	60

CHAPTER 7 CONCLUSION AND FUTURE WORK	63
7.1 Conclusion	63
7.2 Future Work	64
REFERENCES	65

LIST OF TABLES

3.1	Typical bump pitches at multiple levels	28
3.2	Typical dimensions for stackup	29
3.3	Typical dimensions for solder balls	31
4.1	Chiplet specifications	37
5.1	Number of bumps for each chiplet in test case 1	41
5.2	Skin depth for copper at different frequencies	44
5.3	Design parameters for the differential via pair model	45
6.1	BoW Signals at the C2C Interface	55

LIST OF FIGURES

2.1	Dual in-line package	5
2.2	Dual in-line package side view	5
2.3	Flip chip	6
2.4	3D packaging	7
2.5	2.5D packaging	7
2.6	Voltage and current conventions for 2-port network	8
2.7	Cascaded 2-ports	11
2.8	Traveling wave definitions of the 2-port network	12
2.9	Signal flow graph for a 2-port network	13
2.10	Signal flow graph for a 2-port network with load	14
2.11	Representing transmission line as distributed circuit	15
2.12	Circuit schematic of a transmission line in ADS	17
2.13	S-parameters a transmission line in Smith Chart	17
2.14	S-parameters a transmission line in magnitude and phase	18
3.1	PDN structure with 2.5D packaging	20
3.2	Workflow for PDN modeling automation	21
3.3	Micro-via structure	23
3.4	Anti-pad top view	24
3.5	Solder balls	25
3.6	Building larger chiplets	25
3.7	Bounding box for an example pattern	26
3.8	Common violations in adding patterns	27
3.9	Floorplan by shadowing	28
3.10	Obtaining pinmaps	29
3.11	Pinmaps with bumps reduction	30
3.12	Workflow for building model in HFSS	31
3.13	Handling metal layer with draw_layer=False	31
3.14	Antipads drawing cases	33
3.15	Example PDN model	34
4.1	Power islands from a reference design	37
4.2	Timing analysis for drawing micro-vias in HFSS	38
4.3	Routing solution for avoiding via collisions	39
4.4	Trace connections at the chiplet-level	40

5.1	Decoupling capacitor placement	42
5.2	PDN model for four chiplets in one power domain	47
5.3	Input impedance for the PDN model for four chiplets in one power domain	48
5.4	PDN model for four chiplets in two power domains	49
5.5	Input impedance for the PDN model for four chiplets in two power domains	50
5.6	Differential via pair model	50
5.7	Transmission S-parameters for the differential via pair model with nominal parameter values	51
5.8	ADS schematic for eye diagram simulation	51
5.9	Eye diagrams by sweeping the antipad radii	52
5.10	Eye diagrams by sweeping the via and pad radii in the core region	53
6.1	Big picture of the integration of the the C2C links and core of the chiplets	55
6.2	BoW Minimal Bidirectional Reference Link. Adapted from [1]	56
6.3	Vertical routing for BoW	56
6.4	Workflow for BoW modeling automation	57
6.5	Conventions for defining bound origins and directions	58
6.6	Example BoW stack with bounding boxes	59
6.7	BoW model with two chiplets	59
6.8	BoW 3D model with one slice for each chiplet	60
6.9	Cross sectional view of EMIB in package. Adapted from [2] . .	61
6.10	Side view of fabricated EMIB. Adapted from [3]	61
6.11	Insertion loss and far-end crosstalk for EMIB. Adapted from [4]	62

LIST OF ABBREVIATIONS

ADS	Advanced Design System
BGA	Ball Grid Array
BoW	Bunch of Wires
C4	Controlled Collapse Chip Connection
CAD	Computer Aided Design
CPU	Central Processing Unit
DSP	Digital Signal Processor
EDA	Electronic Design Automation
EM	Electromagnetic
EMI	Electromagnetic Interference
EMIB	Embedded Multi-die Interconnect Bridge
FEM	Finite Element Method
HFSS	High-Frequency Structure Simulator
HI	Heterogeneous Integration
ISI	Inter-Symbol Interference
KCL	Kirchhoff's Current Law
KVL	Kirchhoff's Voltage Law
LIM	Latency Insertion Method
ML	Machine Learning
PCB	Printed Circuit Board

PCIe	Peripheral Component Interconnect Express
PDN	Power Distribution Network
PI	Power Integrity
RF	Radio Frequency
Rx	Receiver
SFG	Signal Flow Graph
SI	Signal Integrity
SiP	System in Package
SoC	System on Chip
Tx	Transmitter
UCIE	Universal Chiplet Interconnect Express
VNA	Vector Network Analyzer

CHAPTER 1

INTRODUCTION

Currently, emerging technologies such as heterogeneous integration (HI) enable the combination of chiplets with different technologies, fabrication processes, functionalities, and passive components on a single package to form System-in-Package (SiP) designs [5]. In contrast to System on Chip (SoC) designs where necessary blocks of the system, such as the digital signal processor (DSP) and central processing unit (CPU), are integrated into a single chip, SiP designs offer better flexibility. They allow mixing and matching of components in analog, digital, and radio frequency (RF) domains, and expedite development and production by leveraging commercially available components and standard packaging techniques [6].

For chiplets to communicate with each other, it is not necessary for them to descend to the board level; instead, dedicated chiplet-to-chiplet (C2C) communication links, such as the Bunch-of-Wires (BoW) and Universal Chiplet Interconnect Express (UCIE), are integrated within the organic interposer of the 2.5D package. This integration not only streamlines inter-chiplet communication but also enables the realization of fine-pitch interconnects.

In previous research work, the impacts of various design parameters—including trace widths, bump pitches, via geometries, and stackup configurations—have been extensively studied for PDNs [7][8] and conventional high-speed links [9][10]. Also, in recent years, there have been studies around some design parameters related to the novel C2C interface and some general design guidelines are proposed [11][12]. Those investigations have revealed that increasing line spacing and widths, along with reducing metal layer thickness, would contribute to better designs in terms of SI performance [13]. However, given the expansive design space in the C2C link, it remains an imperative task to thoroughly investigate all possible design parameters to provide insightful guidelines for future applications on C2C links, ensuring the design reliability in diverse scenarios.

As SiP designs handle chiplets with different domains, it becomes necessary for the power distribution network (PDN) to be able to provide stable power supply to the individual chiplets with different voltage levels and frequencies from the printed circuit board (PCB) at the bottom of the package. Power integrity (PI) remains an active research topic to perform analysis on whether the desired voltage and current are delivered to the load and find solutions to improve the design to meet certain specifications. Some PI requirements include keeping the voltage ripples at the chip level lower than threshold, controlling ground bounce, and checking the electromagnetic interference (EMI).

For high-speed links, such as the peripheral component interconnect express (PCIe) used for the interface for the personal computer's motherboard and functional components, signal integrity (SI) needs to be taken into consideration. Signals transmitting at higher rates are more vulnerable to non-ideal distortions and losses caused by reflections, skin effect, crosstalks, jitter, and mutual couplings. With high density interconnect schemes, such as the embedded multi-die interconnect bridge (EMIB) introduced by Intel Corporation [2], crosstalks between adjacent signal lines will become more significant and, thus, require careful analysis.

To perform analysis on PDN or high-speed link structures, full wave simulators such as the High-Frequency Structure Simulator (HFSS) by Ansys need to be employed. Those simulators usually work by using the finite element method (FEM) where the big structures are first partitioned into small meshes and Maxwell's equations are solved to produce the S parameters [14]. Since a large number of partial differential equations are solved and boundaries conditions are matched, especially for large structures as packages, the software can take days to solve a single design, making it very impractical to analyze the influence of varying design parameters. Recently, there have been many successful attempts in using machine learning (ML) strategies to solve electromagnetic (EM) related design problems. A deep reinforcement learning was proposed to optimize decoupling capacitor number of positions in the PDN to satisfy the target impedance where the geometric information is encoded in matrix format [15], and a tandem neural network was employed to estimate design parameters from the desired channel performance [16]. Also, multiple surrogate modeling methods are compared for microwave circuit applications in terms of training time, accuracy, and linearity requirements [17].

However, for ML purposes, large sets of data (model S parameters in this case) are required to train the model and validate the model. Challenges still exist in chiplet floorplanning, via routing in multi-layer boards, and 3D structure modeling as they are labor-intensive. Therefore, a robust electronic design automation (EDA) workflow is needed to shorten the time for the lengthy design process.

This thesis is arranged as follows. Chapter 1 introduces the novel technologies and the challenges they might pose. Chapter 2 focuses on the evolution of packaging technologies and provides a physical interpretation about S parameters with an ideal transmission line example. Chapter 3 discusses the geometry of the PDN in 2.5D package and its automation workflow in python. Chapter 4 outlines the challenges associated with the automation and simulation of large PDN structures and proposes solutions in solving those difficulties. Chapter 5 shows some test cases for PDNs and high-speed links with the results analyzed. Chapter 6 explains the BoW standards for C2C communication, the progress towards the design automation for BoW and the challenges associated with EMIB. Chapter 7 concludes the current research work and suggests possible directions for future research.

CHAPTER 2

BACKGROUND

2.1 Packaging Technologies

Packaging has been an indispensable part of integrated circuits (ICs) over the years for the following reasons

1. Semiconductor components in the ICs are very sensitive to the moisture and dusts in the surrounding environment, so package can provide protection and shielding for the ICs.
2. Pins at the chiplet level typically have a higher density compared to the density at the board level. If the chiplets need to “talk” to the board, either for signals or powers, the package provides the vertical interconnects to stably connect the relevant pins.
3. For high-performance applications, ICs would generate excessive heat during operation, and a package helps to dissipate the heat to avoid overheating.

The package can be a bottleneck to the system performance, and sometimes chiplets can outperform the packaging capabilities. Packaging technologies have evolved greatly over the 30 years to meet with the growing demand for performance and functionality. Some key milestones in the packaging technologies will be discussed in the following subsections.

2.1.1 Dual In-line Package (DIP)

The dual in-line package (DIP) is one of the earliest standardized packages for the ICs and it is still being used nowadays in the instructional labs for easy mounting and debugging. The die is protected by a plastic or ceramic body

and it is connected to the two rows of lead frames through bond wires, making it convenient to mount on the socket via the pin-through-hole configuration. The cross-sectional view of a DIP is illustrated in Figure 2.2.

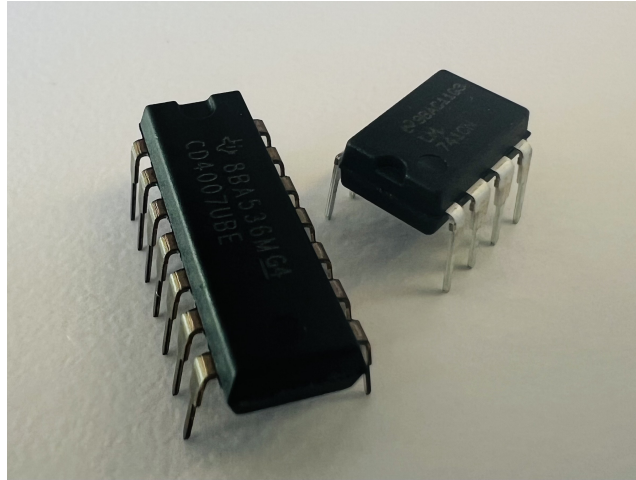


Figure 2.1: Dual in-line package

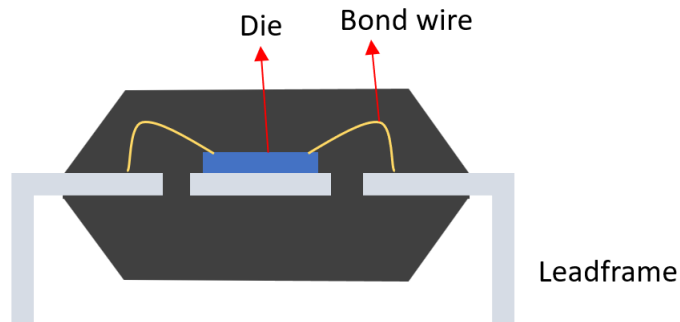


Figure 2.2: Dual in-line package side view

However, for DIP, the chip only occupies less than 20% of the whole package, leaving much space unused. Additionally, the bond wires and lead frames have large inductance, significantly increasing the impedance at high frequencies and thus degrading the SI and PI performance.

2.1.2 Flip Chip

Flip chip, also known as controlled collapse chip connection (C4), is another way to interconnect dice to the external circuits. For each pin on the dice,

the metal pad is metallized on the surface, and a solder ball is deposited for each pad. To connect to dice to the circuit board, the dice are first flipped so that the solder balls face the board, and the solder balls are positioned to align with the corresponding pins on the circuit board. Finally, the solder bumps are remelted, and electrically-insulating adhesive material is placed in the interconnect region as the underfill.

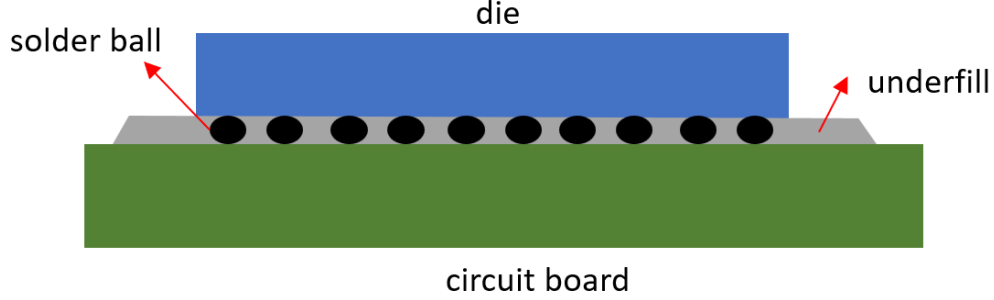


Figure 2.3: Flip chip

As the solder bumps are smaller than the bond wires and lead frame, the inductance associated with flip chip technology has a significant reduction compared with dual in-line package, yielding better SI and PI performance. Yet, since the dice are directly mounted on the circuit board by solder balls, the floorplan at the board level poses constraints on the arrangements of pins at the chip level, limiting design flexibility. Additionally, the short interconnects limit the chip performance in the thermal and mechanical perspectives.

2.1.3 Advanced Packaging: 3D Packaging

3D packaging is considered an advanced packaging technology as it employs a silicon interposer, and dice are stacked in the z-dimension to maximize the space usage and reduce the weight. A silicon interposer is used for high-density routing and connections between the corresponding pins in the dice and the circuit board, reducing the interconnection lengths and thus optimizing the propagation delay for signals. By stacking dice vertically with through-silicon-vias (TSVs), 3D packaging exceeds 100% silicon efficiency compared to conventional packages [18].

Nevertheless, there are some challenges related to 3D packaging: vertical

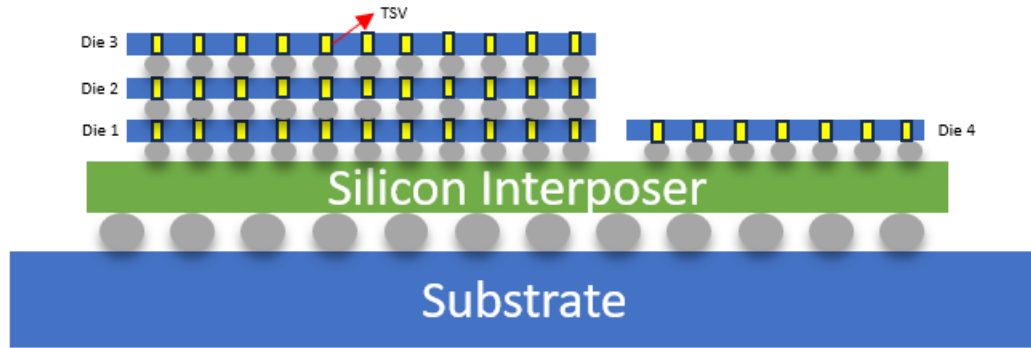


Figure 2.4: 3D packaging

stacking makes the heat dissipation difficult, high-density interconnects pose stress on the materials and fabrication, and the lack of design software makes the time to delivery long.

2.1.4 2.5D Packaging

As there are too many challenges with 3D packaging right now, although it offers lots of advantages compared to conventional packaging, 2.5D packaging will be the temporary solution as the trade-off between design complexity and performance. The 2.5D package is composed of a standard substrate and an interposer for fine-pitch interconnects. For heterogeneous integration, multiple chiplets can be placed on top of the interposer horizontally, rather than vertically in 3D packaging.

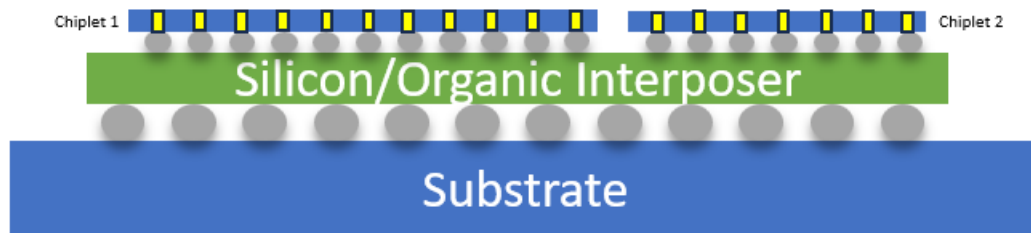


Figure 2.5: 2.5D packaging

The material for the interposer can be silicon or organic: a silicon interposer features a thinner stackup, finer pitch, and smaller via dimensions.

Researchers have shown that a design with a silicon interposer has less power consumption, shorter average wire lengths, and a smaller area compared with an organic interposer-based design. Yet, with an organic interposer, the DC impedance is smaller, and worst-case delay is shorter for PDN applications [19].

2.2 Network Parameters

An n -port network contains $2n$ terminals and n ports as the terminals are combined in pairs. Among those, the 2-port network is the most common one as it can be considered a “black-box” to model large circuits, including transmission lines, matching networks, transformers, and small-signal models for transistors. The currents I_1 and I_2 denote the currents that flow into one of the terminals and the voltages V_1 and V_2 denote the voltages between the two terminals of the two ports respectively. The conventions for defining the currents and voltages are shown in Figure 2.6.

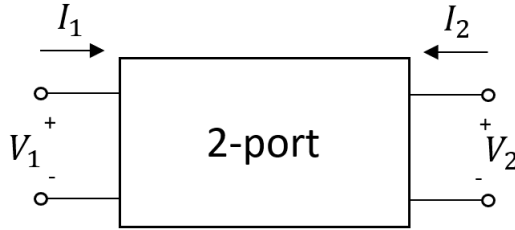


Figure 2.6: Voltage and current conventions for 2-port network

To characterize any 2-port networks as a “black-box” model with voltages and currents, network parameters are introduced by defining two of the four variables (I_1 , I_2 , V_1 , and V_2) as independent variables and the other two as dependent variables. With combinations of selecting two variables as independent, six sets of network parameters can be formulated. In the later parts of this section, Z-, Y-, and ABCD-parameters will be introduced. Also, the limitations of those parameters for high-frequency applications will be discussed and scattering (S) parameters will be introduced.

2.2.1 Z-parameters

For Z-parameters (impedance parameters), the currents are chosen to be the independent variables, and they are formulated as follow

$$\begin{aligned} V_1 &= Z_{11}I_1 + Z_{12}I_2 \\ V_2 &= Z_{21}I_1 + Z_{22}I_2 \end{aligned} \tag{2.1}$$

Z-parameters have the unit of impedance. To obtain the individual Z-parameters, the two currents need to be forced to 0 with one at a time, which means Z-parameters have the reference impedance of an open circuit.

$$\begin{aligned} Z_{11} &= \left. \frac{V_1}{I_1} \right|_{I_2=0} \\ Z_{21} &= \left. \frac{V_2}{I_1} \right|_{I_2=0} \\ Z_{12} &= \left. \frac{V_1}{I_2} \right|_{I_1=0} \\ Z_{22} &= \left. \frac{V_2}{I_2} \right|_{I_1=0} \end{aligned} \tag{2.2}$$

2.2.2 Y-parameters

For Y-parameters (admittance parameters), the voltages are chosen to be the independent variables, and they are formulated as follow

$$\begin{aligned} I_1 &= Y_{11}V_1 + Y_{12}V_2 \\ I_2 &= Y_{21}V_1 + Y_{22}V_2 \end{aligned} \tag{2.3}$$

Y-parameters have the unit of admittance. To obtain the individual Y-parameters, the two voltages need to be forced to 0 with one at a time, which

means Y-parameters have the reference impedance of a short circuit.

$$\begin{aligned}
Y_{11} &= \left. \frac{I_1}{V_1} \right|_{V_2=0} \\
Y_{21} &= \left. \frac{I_2}{V_1} \right|_{V_2=0} \\
Y_{12} &= \left. \frac{I_1}{V_2} \right|_{V_1=0} \\
Y_{22} &= \left. \frac{I_2}{V_2} \right|_{V_1=0}
\end{aligned} \tag{2.4}$$

2.2.3 ABCD-parameters

For ABCD-parameters, the variables at port 2 are chosen to be the independent variables, and they are formulated as follow

$$\begin{aligned}
V_1 &= AV_2 - BI_2 \\
I_1 &= CV_2 - DI_2
\end{aligned} \tag{2.5}$$

Since port 2 variables are selected to be the independent variables, ABCD-parameters do not have a fixed reference impedance. For example, for calculating elements A and C, the reference is an open circuit; and for calculating elements B and D, the reference is a short circuit.

$$\begin{aligned}
A &= \left. \frac{V_1}{V_2} \right|_{I_2=0} \\
B &= -\left. \frac{V_1}{I_2} \right|_{V_2=0} \\
C &= \left. \frac{I_1}{V_2} \right|_{I_2=0} \\
D &= -\left. \frac{I_1}{I_2} \right|_{V_2=0}
\end{aligned} \tag{2.6}$$

If two 2-port networks are cascaded (Figure 2.7) and the overall system needs to be characterized, ABCD-parameters are most useful because the ABCD-parameters of the cascaded 2-ports will be the matrix product of the individual ABCD-parameters by construction. There are formulas to convert back and forth from Y- or Z-parameters to ABCD-parameters for the ease

of characterizing multiple 2-ports in cascade.

$$ABCD_{\text{cascaded}} = ABCD_a \cdot ABCD_b \quad (2.7)$$

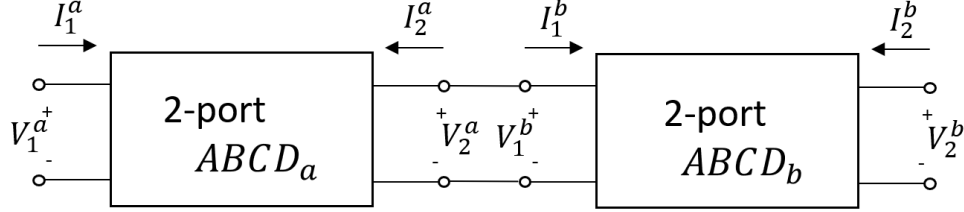


Figure 2.7: Cascaded 2-ports

2.2.4 S-parameters

Previously, it has been shown that Z-, Y-, and ABCD-parameters are useful to describe 2-port networks as “black-box” models. However, they are based on the reference impedance of short or open circuits, which are very hard to implement and may drive the active components to oscillate at microwave frequencies. S-parameters, on the other hand, use a finite resistance Z_0 as the reference (usually $50\,\Omega$), which can be easily realized at high frequencies. Also, due to the distributed nature of wires at high frequency, wave variables will be used to define S-parameters. With the traveling wave approach (Figure 2.8), the total voltages and currents can be expressed as the sum of the forward- and backward-traveling waves.

$$\begin{aligned} V_1 &= E_{i1} - E_{r1} \\ V_2 &= E_{i2} - E_{r2} \\ I_1 &= \frac{E_{i1} - E_{r1}}{Z_0} \\ I_2 &= \frac{E_{i2} - E_{r2}}{Z_0} \end{aligned} \quad (2.8)$$

Then, it would be useful to define another set of normalized voltage wave

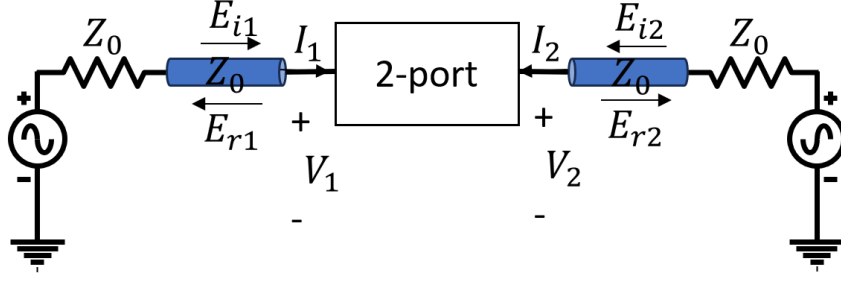


Figure 2.8: Traveling wave definitions of the 2-port network

variables (a_1, a_2, b_1, b_2) . The variables ‘ a ’s denote the incident waves and the variables ‘ b ’s denote the reflected waves.

$$\begin{aligned} a_1 &= E_{i1}/\sqrt{Z_0} & a_2 &= E_{i2}/\sqrt{Z_0} \\ b_1 &= E_{r1}/\sqrt{Z_0} & b_2 &= E_{r2}/\sqrt{Z_0} \end{aligned} \quad (2.9)$$

The S-parameters are defined based on the normalized wave variables and the incident waves are the independent variables.

$$\begin{aligned} b_1 &= S_{11}a_1 + S_{12}a_2 \\ b_2 &= S_{21}a_1 + S_{22}a_2 \end{aligned} \quad (2.10)$$

To obtain the individual S-parameters of a 2-port network, the incident wave variables a_1 and a_2 will be forced to zero. For forcing a_i to be zero, no excitation should be provided at port i and port i should be matched to the characteristic impedance of the reference transmission line (Z_0).

$$\begin{aligned} S_{11} &= \left. \frac{b_1}{a_1} \right|_{a_2=0} \\ S_{12} &= \left. \frac{b_1}{a_2} \right|_{a_1=0} \\ S_{21} &= \left. \frac{b_2}{a_1} \right|_{a_2=0} \\ S_{22} &= \left. \frac{b_2}{a_2} \right|_{a_1=0} \end{aligned} \quad (2.11)$$

For an intuitive understanding about the S-parameters, S_{ij} stands for the ratio of the powers incident at port j and received at port i . For example, S_{11} in a 2-port network means the input reflection coefficient when port 2 is matched to Z_0 and likewise S_{22} means the output reflection coefficient when port 1 is matched to Z_0 .

A signal flow graph (SFG) is a useful graphical representation of n -port networks where nodes are the wave variables, and branches are the S-parameters. An example of SFG for a 2-port network with both ports unterminated is shown in Figure 2.9.

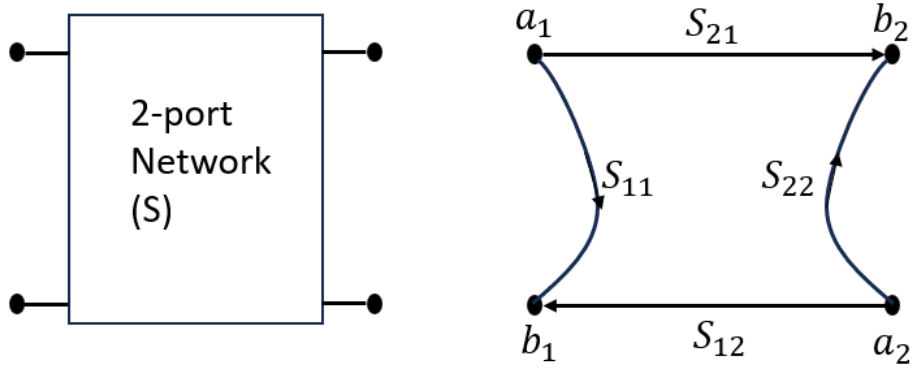


Figure 2.9: Signal flow graph for a 2-port network

The SFG will come in handy when multiple n -port networks are cascaded, and the new set of cascaded S-parameters needs to be found from the individual n -port S-parameters. Mason's rule can be used to calculate the transfer functions in the SFG.

$$T = \frac{P_1[1 - \sum L(1)^{(1)} + \sum L(2)^{(1)} - \dots] + P_2[1 - \sum L(1)^{(2)} + \dots] + \dots}{1 - \sum L(1) + \sum L(2) - \sum L(3) + \dots} \quad (2.12)$$

where

1. A first-order loop is the product of the branches that start and end at the same node in which no node is crossed more than once.
2. A second-order loop is the product of any two non-touching first-order loops.
3. A third-order loop is the product of any three non-touching first-order loops.

4. T is the ratio of the dependent variable over the independent variable.
5. P_k 's are the various paths connecting the two variables in T .
6. $L(j)^{(k)}$ is a loop of order j that does not touch path k .

As an example, if port 2 of a 2-port network is terminated with a load impedance Z_L , the SFG for the 2-port can be constructed as shown in Figure 2.10. The load impedance is represented by the reflection coefficient at the load $\Gamma_L = \frac{Z_L - Z_0}{Z_L + Z_0}$. In the SFG, there are two paths that connect variable b_1 and b_s : $S_{21}\Gamma_L S_{12}$ and S_{11} . There is only one first order loop $S_{22}\Gamma_L$. Therefore, the input reflection coefficient for this 2-port can be derived from Mason's rule as

$$\begin{aligned}\Gamma_{in} &= \frac{b_1}{b_s} \\ &= \frac{S_{11}(1 - S_{22}\Gamma_L) + S_{21}\Gamma_L S_{12}}{1 - S_{22}\Gamma_L} \\ &= S_{11} + \frac{S_{21}\Gamma_L S_{12}}{1 - S_{22}\Gamma_L}\end{aligned}$$

and the input impedance of the 2-port network can be obtained by

$$Z_{in} = Z_0 \frac{1 + \Gamma_{in}}{1 - \Gamma_{in}} \quad (2.13)$$

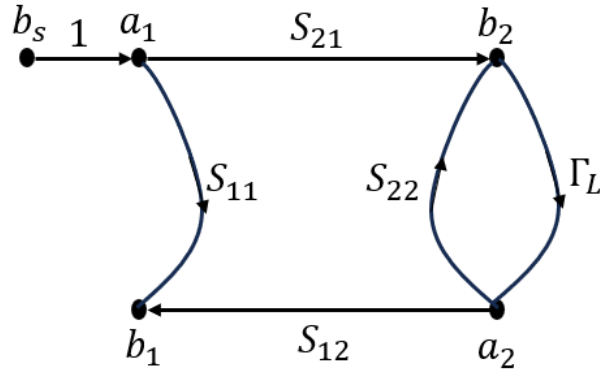


Figure 2.10: Signal flow graph for a 2-port network with load

2.2.5 Transmission Line

At lower frequencies, where the structural dimensions are significantly smaller than the wavelength, the lumped model can be effectively used with Kirchhoff's current law (KCL) and Kirchhoff's voltage law (KVL). However, at higher frequencies, where the structural dimensions become comparable to the wavelength, a distributed model should be used and wires should be considered as transmission lines.

An ideal lossless transmission line can be considered as a distributed circuit consisting of an array of per unit length inductors ($\mathcal{L}$) and capacitors ($\mathcal{C}$), illustrated in Figure 2.11. The characteristic impedance of the line is related to the per unit length inductance and capacitance by

$$Z_0 = \sqrt{\frac{l}{c}} \quad (2.14)$$

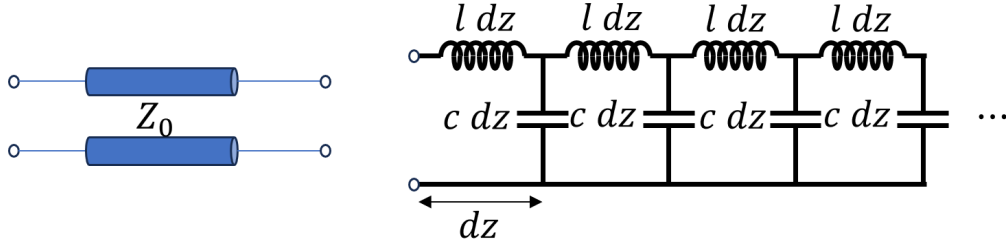


Figure 2.11: Representing transmission line as distributed circuit

By applying KCL and KVL on a small segment (dz) of the transmission line, one can obtain the transmission line equations

$$\begin{aligned} \frac{\partial V(z,t)}{\partial z} &= -l \frac{\partial I(z,t)}{\partial t} \\ \frac{\partial I(z,t)}{\partial z} &= -c \frac{\partial V(z,t)}{\partial t} \end{aligned} \quad (2.15)$$

For an ideal transmission line with characteristic impedance Z_C and length d , its S-parameters, measured by a vector network analyzer (VNA) with

characteristic impedance Z_0 , are given by

$$\begin{aligned} S_{11} &= \frac{(1 - X^2)\Gamma}{1 - \Gamma^2 X^2} \\ S_{21} &= \frac{(1 - \Gamma^2)X}{1 - \Gamma^2 X^2} \end{aligned} \quad (2.16)$$

Here, the propagation constant is defined as $X = e^{-j\beta d}$ and the reflection coefficient as $\Gamma = \frac{Z_C - Z_0}{Z_C + Z_0}$. Given the reciprocity and symmetry of a piece of transmission line, $S_{11} = S_{22}$ and $S_{21} = S_{12}$.

When the characteristic impedance of the transmission line is matched to the reference impedance of the VNA, there will be no reflection at the interface of the transmission line and the VNA, i.e, $\Gamma = 0$. In this case, the S-parameters can be simplified to

$$\begin{aligned} S_{11} &= S_{22} = 0 \\ S_{21} &= S_{12} = X = e^{-j\beta d} \end{aligned} \quad (2.17)$$

which mean if an excitation is sent from port 1, everything is transmitted to port 2 with some phase shift (time delay), and nothing is reflected. This confirms with the definition of ideal transmission lines.

An circuit in Advanced Design System (ADS) is set up to validate the S-parameters of an ideal transmission line shown above. The width and length of the copper trace are optimized to achieve a characteristic impedance of around 50Ω based on the substrate definition, so the reflection will be minimal. The S-parameters simulation is performed and the frequency is swept from 100 MHz to 6 GHz.

The S-parameters from the simulation are presented in the Smith Chart in Figure 2.13 and rectangular plots in Figure 2.14. From the results, the transmission line exhibits perfect reciprocity and symmetry; there is almost no reflection at both ports and almost all the power incident at one port is transmitted to the other port. The finite difference from the theoretical values may arise due to the small mismatch between the two characteristic impedance and the dielectric loss. The phase of S_{21} is linear as expected, ignoring the discontinuities at 180° and -180° as they wrap around.

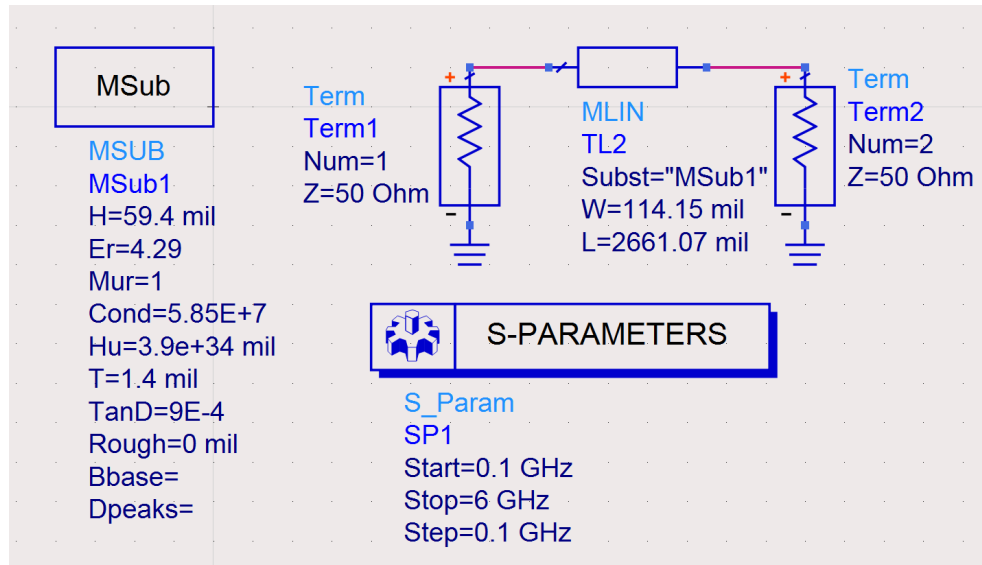


Figure 2.12: Circuit schematic of a transmission line in ADS

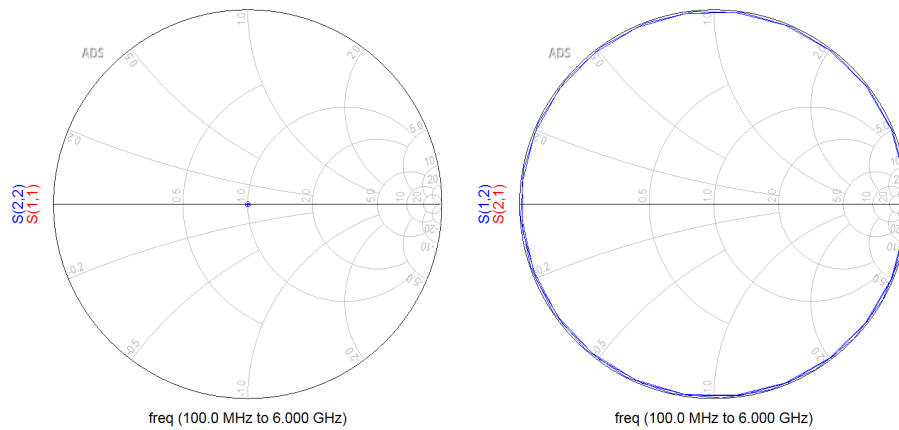


Figure 2.13: S-parameters a transmission line in Smith Chart

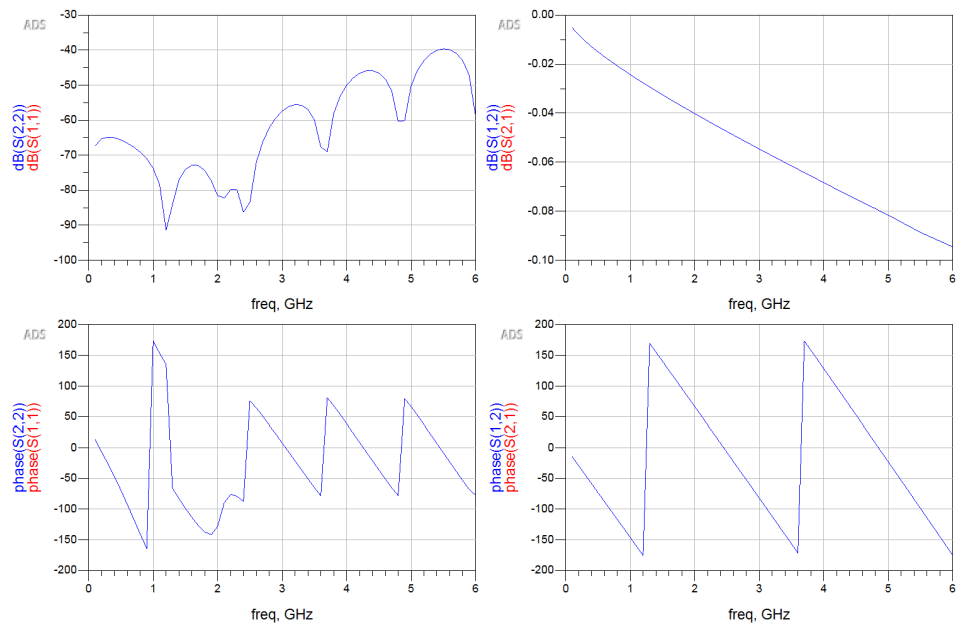


Figure 2.14: S-parameters a transmission line in magnitude and phase

CHAPTER 3

DESIGN AUTOMATION FOR PDN

With heterogeneous integration, chiplets operating at different domains can be integrated together on a single package, so a PDN becomes essential for ensuring a stable power supply to all the chiplets. This chapter delves into the discussion of the design automation for PDN with 2.5D packaging.

The 2.5D package is made of multi-layer substrate and the interposer structures. The stackup contains alternating metal layers and dielectric layers where the metal layers can be assigned to power or signal nets. The substrate is divided into three regions: bottom build-up, core, and top build-up. The core, situated between the two build-up regions, processes larger layer thicknesses and via dimensions than those in the build-up region to provide enhanced mechanical support. The interposer, on the other hand, only contains the core and top build-up. The solder bumps at the interface of circuit board and package substrate are called ball grid arrays (BGAs) and those at the interface of substrate and interposer are called micro-BGAs (μ BGAs). As one of the goal for the package is to connect the corresponding pins on the board-level and chip-level, via routings are needed in the package to account for the different pitches and positions. As a convention for the rest of the text, black stands for signal net, yellow denotes ground net, and orange represents power net. The general structure for the PDN and the via routings in the package are shown in Figure 3.1.

An automation script has been developed to model the PDN in Ansys HFSS with the help of the PyAEDT library in Python. The PyAEDT library enables direct interaction with the API in Ansys electronic desktop to draw 3D components, perform boolean operations on objects, set up excitations, and alter simulation setting. Firstly, the sequential build-up (SBU) object has to be initiated to save critical parameters into the class object. Subsequently, for each pattern in each chiplet, the **pattern_adder()** function is called to save pattern information into the class object, provided there are

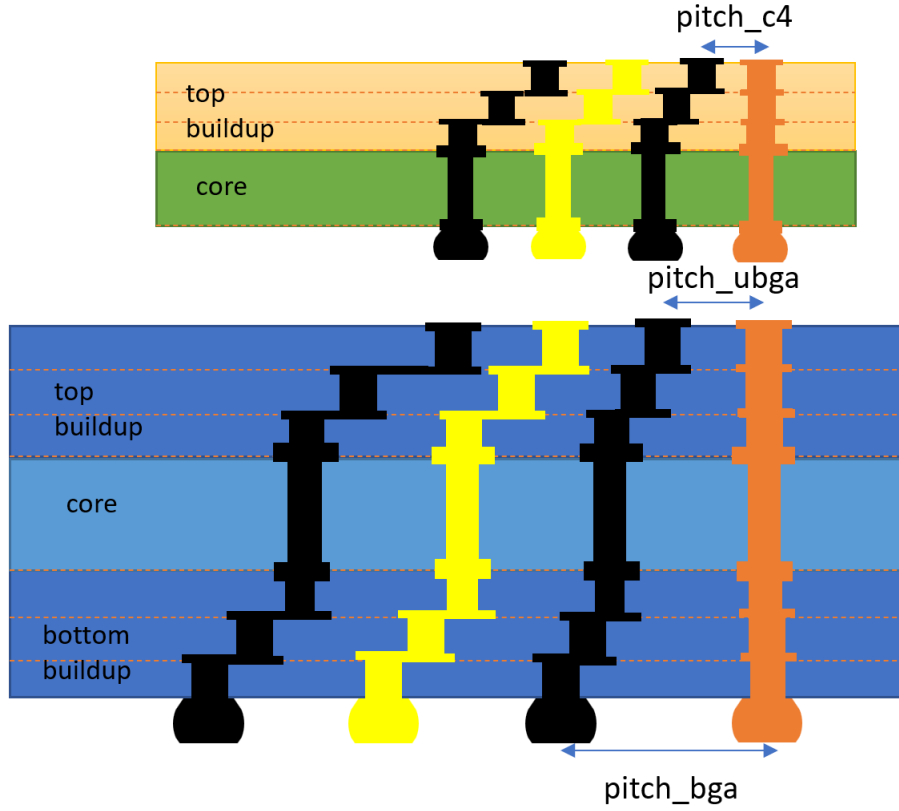


Figure 3.1: PDN structure with 2.5D packaging

no violations. Following this, the **floorplan()** function is called to calculate the pin locations at the 3 different levels based on the pattern information at the chip-level. Finally, with all the information needed, the **build_model()** function is called to construct the PDN model in HFSS and solve for S-parameters. The detailed descriptions of each step will be expounded upon in the following sections.

3.1 Initiate SBU Class Object

When initiating the SBU object, the primary objective is to save the essential geometric parameters related to the stackup and via dimensions into the object. These parameters are crucial for subsequent functions, facilitating seamless retrieval for floorplanning and the construction of the 3D model.

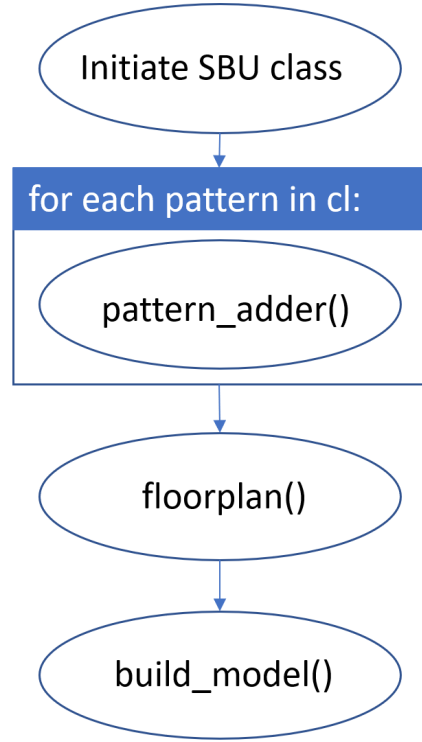


Figure 3.2: Workflow for PDN modeling automation

3.1.1 Stackup

Stackup encapsulates all the vital information needed to draw the multi-layer structures for the substrate and the interposer. First of all, the widths and lengths of the substrate and interposer need to be defined in the unit of μm while the heights can be determined by the layer definitions. Layer definitions need to specify the following information with the layers ordered from bottom to top:

1. Layer name: A unique identifier for the layer which will be used as the key to the stackup dictionary. For example, the first metal layer in the bottom build-up can be named as "bu1-M1".
2. Thickness: The thickness of the layer in the unit of μm .
3. Material: The material of the layer. Note that the name of the material must match one of the material names in Ansys; if not, it is necessary to define the material in Ansys beforehand.

4. Draw plane: A boolean variable that defines whether the plane needs to be drawn, considering that certain planes are reserved for traces. Note that this parameter is only needed for metal layers, as dielectric layers will always be drawn.
5. Netname: The name of the net that the layer belongs to. This parameter applies exclusively to metal layers as well.
6. AEDT variable name: The name of the layer thickness variable to define in HFSS. This allows for easier retrieval of thickness later for spatial calculations.

An example definition of the stackup for the bottom build-up in the substrate is provided below.

```
bot_bu = {  
  "bu1_M1": {  
    "type": "metal",  
    "thickness": 15,  
    "material": "copper",  
    "draw_plane": false,  
    "netname": "s",  
    "aedt_varname": "subst_bu_metal_t"},  
  "bu1_d1": {  
    "type": "dielectric",  
    "thickness": 30,  
    "material": "FR4_epoxy",  
    "aedt_varname": "subst_bu_dielec_t"},  
  "bu1_M2": {  
    "type": "metal",  
    "thickness": 15,  
    "material": "copper",  
    "draw_plane": true,  
    "netname": "ga",  
    "aedt_varname": "subst_bu_metal_t"}  
}
```

3.1.2 Via Geometries

Vias play a pivotal role multi-layer boards as they provide vertical interconnects for signals to propagate from one layer to the other. The giant

via making connection from the bottom of the board to the top can be deconstructed into multiple micro-vias where each of them is connecting two adjacent layers. A micro-via (Figure 3.3) is made of two cylindrical pads that are at the level of metal planes and a cone-shaped via body that goes through the dielectric region. To model the micro-vias closely to fabricated ones, the radii of the top via pad and body are larger than the bottom ones.

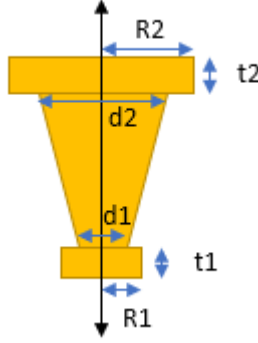


Figure 3.3: Micro-via structure

The parameters needed to model the micro-vias are top pad radius, top via body radii, bottom pad radius, and bottom via body radius. The thicknesses of the two via pads have to be the same as the thicknesses of the metal layers they are on. As these parameters are different from the core region to the build-up region and from the package substrate to the interposer, separate definitions are needed.

When the net of the via is the same as the net of the metal layer the pad will land on, the via pad will be electrically connected to the metal layer. Conversely, when the net of the via is not matched to the net of the metal layer the pad will land on, an anti-pad should be drawn to prevent the different signals from mixing into each other. The anti-pad works in a way that a larger cylinder is drilled from the solid metal plane so the via pad will not be in contact with the plane. The top-view of an anti-pad is shown in Figure 3.4.

As for most of the cases, the top radii are scaled versions of the bottom radii and anti-pad dimensions are scaled versions of the pad radii. Hence, users are only required to enter the bottom pad and via body radii and the rest will be calculated by the script. For example, if the user inputs $15\mu m$

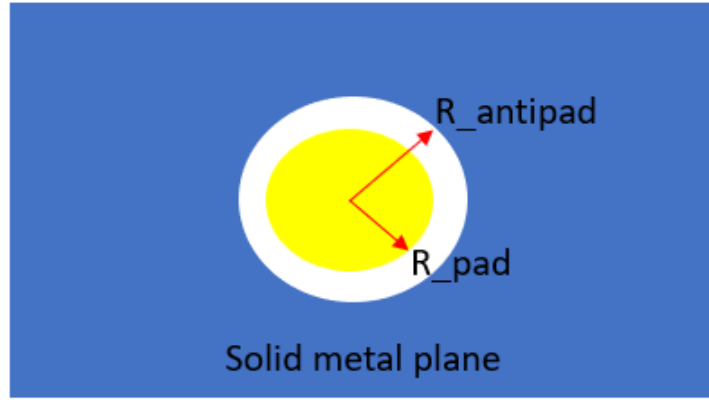


Figure 3.4: Anti-pad top view

for the bottom pad radius and 1.1 as the scaling constant, then the top pad radius can be calculated by $15 * 1.1 = 16.5\mu m$. An example of the via geometry definitions for the package substrate are shown below.

```
R_substr =
{"Rvia":{"core_v":[50,],"core_p":[75,],
        "buildup_v":[15,], "buildup_p":[20,]},
"Rapad":{"Rapad_buildup_top":[140,],
          "Rapad_buildup_bottom":[140,],
          "Rapad_core_top":[140,],
          "Rapad_core_bottom":[140,]}}
```

3.1.3 Solder Balls

BGAs are used to establish contact between the circuit board and the package substrate, while μ BGAs are used to establish contact between the substrate and the interposer. Both of them can be modeled as spheres with partial top and bottom cut-offs. Therefore, users need to specify the sphere radius R , the height from the center of the sphere to the top $h1$, and the height to the bottom $h2$ as depicted in Figure 3.5.

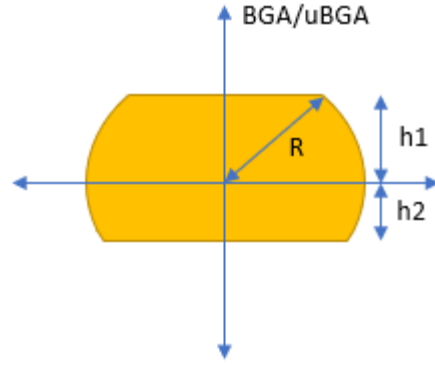
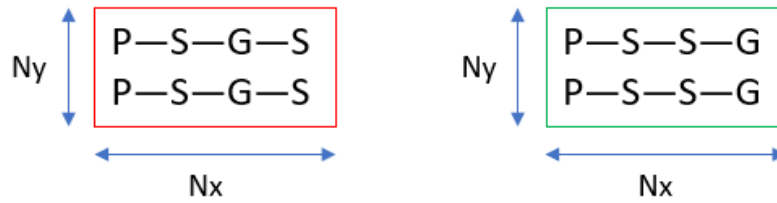


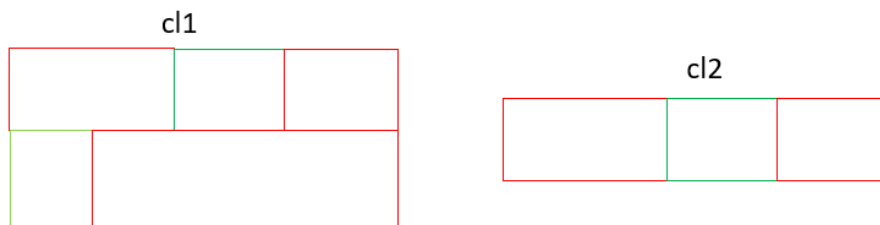
Figure 3.5: Solder balls

3.2 Adding Patterns

Patterns represent the arrangement of power, ground, and signal bumps on a chiplet and they will be repeated in the x- and y-directions to create a large rectangular region. In practice, various patterns with distinct instances in x- and y-directions are combined to make an actual chiplet. For example, two different patterns are defined as building blocks in Figure 3.6a, and their combinations of them are employed to make two different chiplets in Figure 3.6b.



(a) Defining patterns



(b) Building chiplets from patterns

Figure 3.6: Building larger chiplets

To define a pattern, users need to specify the chiplet to which pattern

belongs to, the ordering of nets in the building block as a list, the coordinates of the origin of the pattern's bounding box (lower left corner of the box), the pitch at the chiplet-level, the number of patterns in the x- and y-directions, and the direction in which the pattern extends in. The bounding box of the pattern starts at the bound origin and is padded some distance from the boundary bumps. Figure 3.7 provides some definitions for a pattern with sequence power-signal-ground-signal, extending in the x-direction and repeating once in the x-direction and twice in the y-direction.

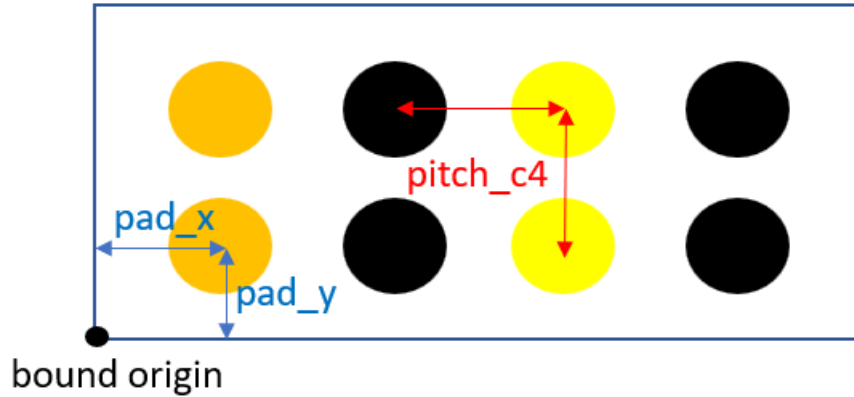


Figure 3.7: Bounding box for an example pattern

When multiple patterns are amalgamated to form a chiplet, the individual patterns must neither overlap nor be fragmented to comply with the enforced pitch at the chiplet-level. Therefore, bounding boxes of the patterns need to be aligned, which means they can be combined into a larger bounding box. In the examples shown below, the first big pattern in a solid red bounding box is created by integrating two smaller patterns. However, when the next pattern (dashed red box) is defined by the user, Figure 3.8 shows several violations to the rules mentioned above. In such cases, the script will raise an error and cease execution. Otherwise, the pattern information will be incorporated into the SBU object for floorplanning and drawing.

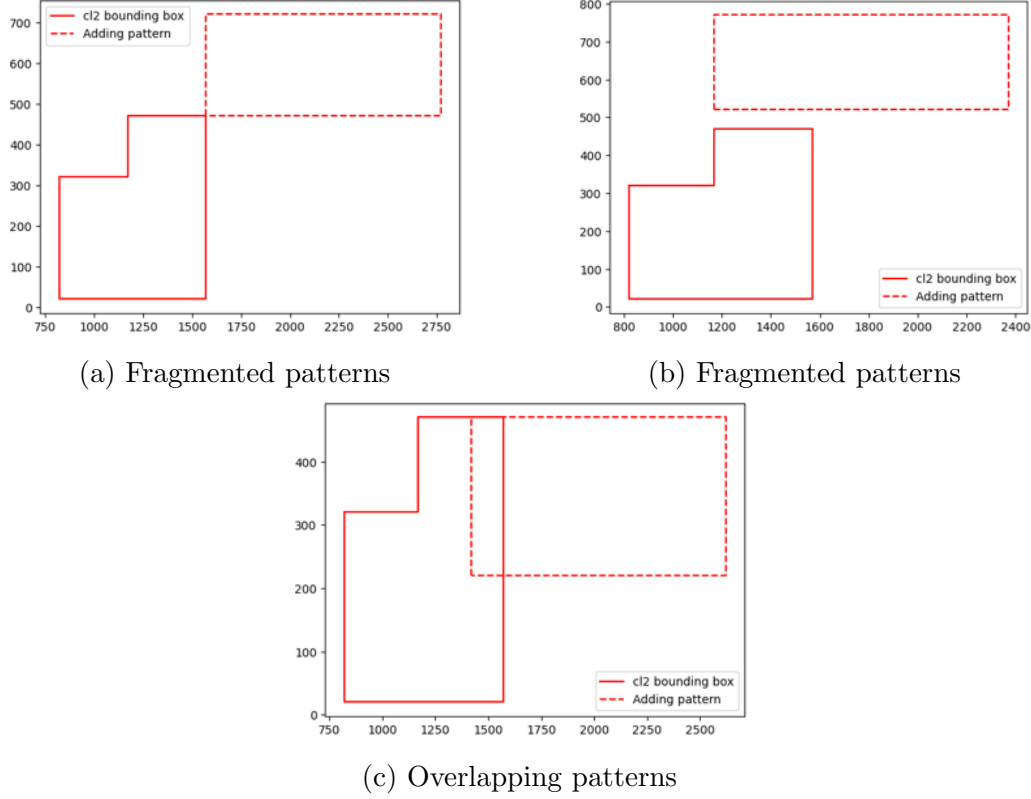


Figure 3.8: Common violations in adding patterns

3.3 Floor Plan

After defining all the patterns at the chiplet-level and saving the relevant information, the next step involves performing the floorplanning, which entails calculating the pinmaps at the chiplet-level and projecting them onto the μ BGA-level and BGA-level. The simplest approach to perform floorplanning is through shadowing, which means the first c4 bump is projected to the lower levels, and the patterns extend in both directions following the pitches as shown in Figure 3.9. The pinmaps at each level are obtained net by net for higher efficiency. For example, for the pattern “power-signal-ground-signal” that extends in the x-direction, power bumps need to move four pitches in the x-direction to reach the next power bump and one pitch in the y-direction to reach the next one, whereas signal bumps only need to move two pitches in the x-direction to reach the next one. This is illustrated in Figure 3.10. The typical bump pitches in a 2.5D package with an organic interposer are shown in Table 3.1.

Given that the pitch at the chiplet-level is significantly larger than the

Table 3.1: Typical bump pitches at multiple levels

	Board-level	μ BGA-level	Chiplet-level
Bump pitch	$1000\mu m$	$400\mu m$	$50\mu m$

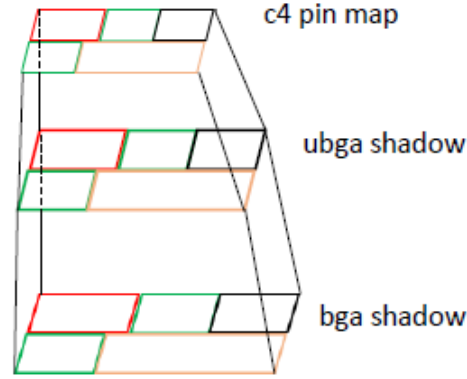


Figure 3.9: Floorplan by shadowing

pitch at the board-level, the occupied area at the two levels will be largely disproportionate, resulting in the waste of package area if no optimized routing algorithm is available. Therefore, a reduction ratio used between the adjacent bump levels will be applied as an industry standard to ensure approximately the same area among all levels. In the following example, the L-shape chiplet is formed by two patterns “v1” and “v2-ground”. From c4-level to μ BGA-level, the reduction ratio is (4,4), meaning the number of patterns at the μ BGA-level is the number of patterns at the c4-level divided by 4 in both directions; from μ BGA-level to BGA-level, the reduction ratio is (2,2). The resulting pinmaps are shown in Figure 3.11.

In most cases, a single chiplet will contain multiple voltage domains to provide all the transistors with the voltages to ensure they are operating in the correct region. Thus, the PDN has to handle cases where multiple different powers enter the metal layer assigned to power. It would be incorrect to have all different power vias electrically connected to the power layer as the powers would be mixed. Also, it would be impractical to draw antipads for all the power vias as the interconnects for the same power net have to be connected. As an industry solution, solid metal layers will be replaced by clusters of power islands assigned to each voltage domain.

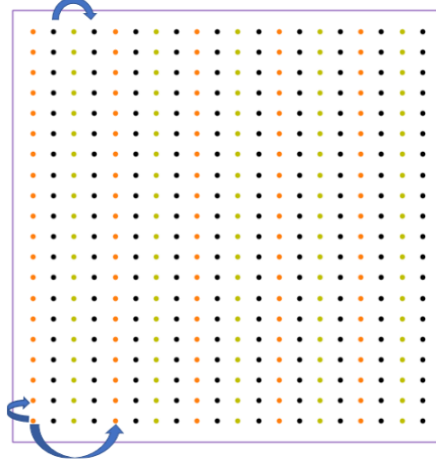


Figure 3.10: Obtaining pinmaps

3.4 Building model in HFSS

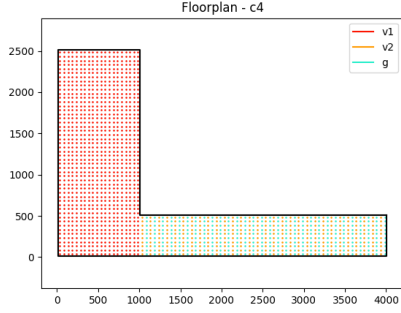
After completing the previous three steps, geometric parameters, pattern information, and pinmaps at all levels are calculated and saved into the class object. In this step, that information will be retrieved to build the package PDN model in Ansys HFSS. The general workflow is outlined in Figure 3.12.

3.4.1 Drawing Stackup

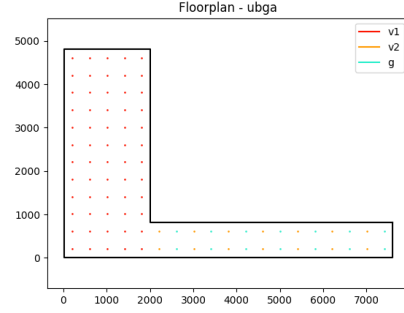
The stackup can be drawn for the substrate and interposer using exactly the same procedure. The typical thicknesses of the stackup are shown in Table 3.2. Before drawing, the elevation (the z-coordinate of the lower surface) of each layer is calculated from the stackup definition dictionary. When a metal layer with the draw_plane keyword set to False, the dielectric layers above and beneath it will be expanded by $0.5 \times \text{metal_thickness}$ to account for the elevation difference, as shown in Figure 3.13.

Table 3.2: Typical dimensions for stackup

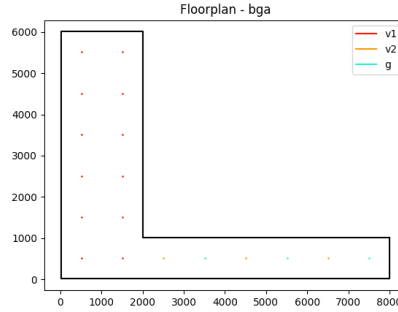
Type	BU dielectric	Core dielectric	BU metal	Core metal
Substrate	$15\ \mu m$	$600\ \mu m$	$15\ \mu m$	$30\ \mu m$
Interposer	$5\ \mu m$	$63\ \mu m$	$2\ \mu m$	$5\ \mu m$



(a) Pinmaps at c4-level



(b) Pinmaps at μ BGA-level



(c) Pinmaps at BGA-level

Figure 3.11: Pinmaps with bumps reduction

Since all the layer thicknesses are defined as variables in HFSS, the elevations for each layer are also defined as HFSS variables as a summation of all previous layer thicknesses. This approach would allow modifying the layer thicknesses later for studying without needing to redraw the entire model.

3.4.2 Drawing Solder Balls

With the solder ball structures shown in Figure 3.5, they can be modeled in HFSS by first drawing a sphere with radius R , and then two boxes at appropriate positions are subtracted from the sphere. To draw the solder ball, the relative coordinate system should be positioned at the center of the lower surface of the bump. Users can set the working coordinate system or specify the coordinate system by either entering the new origin or the name for an existing coordinate system. The function will return the solder ball 3D object for the union later. An example command for drawing the solder ball is shown below.

```

1 draw substrate stackup
2 draw interposer stackup
3 for each chiplet
4     for each pattern in chiplet
5         for each net in pattern
6             for each triplet of bump locations
7                 draw BGA ball
8                 draw vertical interconnect through substrate
9                 draw uBGA ball
10                draw vertical interconnect through interposer
11                unite object
12 draw and define excitations
13 assign boundary
14 specify analysis setup
15 solve

```

Figure 3.12: Workflow for building model in HFSS

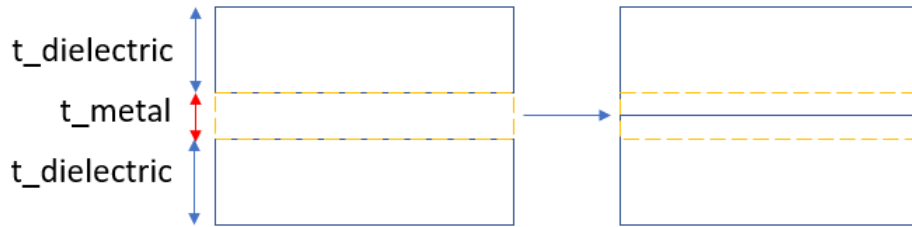


Figure 3.13: Handling metal layer with draw_layer=False

```

bump_obj = draw_solderball(hfss,R,h1,h2,unit,partname="BGA",
                           ref_coord=(200,100,0,"ref_cs"))

```

The typical dimensions of the solder balls are shown in Table 3.3.

Table 3.3: Typical dimensions for solder balls

Type	$R(\mu m)$	$h1(\mu m)$	$h2(\mu m)$
BGA	360	200	200
μ BGA	90	50	50

3.4.3 Drawing Vertical Interconnects

The **draw_vertical_interconnect_through_SBU()** function is the most important and complicated function when building the 3D model. When given a pair of bump locations, the function will try to draw a series of vias

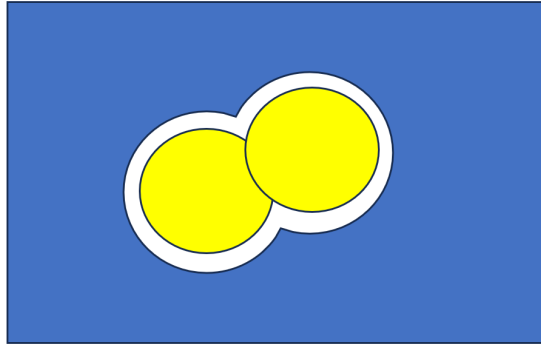
through the package to connect those two points. In the script, the horizontal shift between the bottom and top is divided evenly into the shift on metal layers where shiftings are possible. Since the core via and the two microvias connected to it are in the same xy-plane for better mechanical support, the number of possible horizontal shifts for an N -layer board is $N - 4$. The function would draw the vias from the bottom to the top and when each via is to be drawn, it will

1. Check for whether a horizontal shift is needed. If a shift is needed relative to the previous via, generate a new relative coordinate where the origin is the center of the via.
2. Check if an antipad is needed. If needed, draw the antipad.
3. Draw the via.

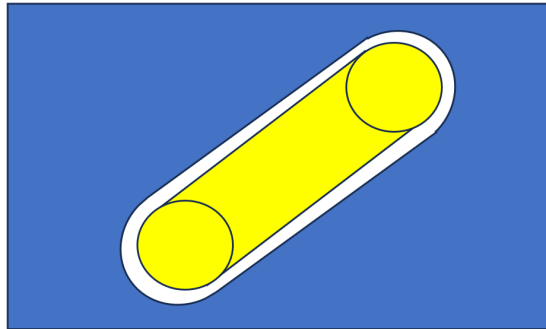
It is important to note that the antipad is needed when the net of the via is different from the net of the metal layer. When the nets are the same, the via pads will overlap with the metal plane, and it would not be a problem in solving, as HFSS can handle the material overriding. When an antipad is needed and the shifted distance is less than the diameter of the via pad, the top pad of the bottom via and the bottom pad of the top via will be connected in the antipad region as shown in Figure 3.14a, allowing the signal to propagate. On the other hand, when the antipad is needed and the shifted distance is larger than the diameter of the via pad, the two via pads will not touch due to the antipad, then there is no path for the signal. Therefore, in this case, a trace will be drawn to connect the two via pads with another antipad drawn for the trace as shown in Figure 3.14b.

After drawing all the vias to form one vertical interconnect between the two bumps, the function will return a list of all the 3D objects drawn. Since the interposer has the exact structure compared to the substrate except for the bottom build-up, the same function is used to draw the interconnect between BGA and μ BGA as well as between μ BGA and C4. All the components of one vertical interconnect from BGA to C4 will be united into a single 3D object when the drawing is done.

An example PDN model with the pattern “power-ground” repeating itself twice is built with the python command shown below, and the model pictures



(a) Antipads with via pads touching



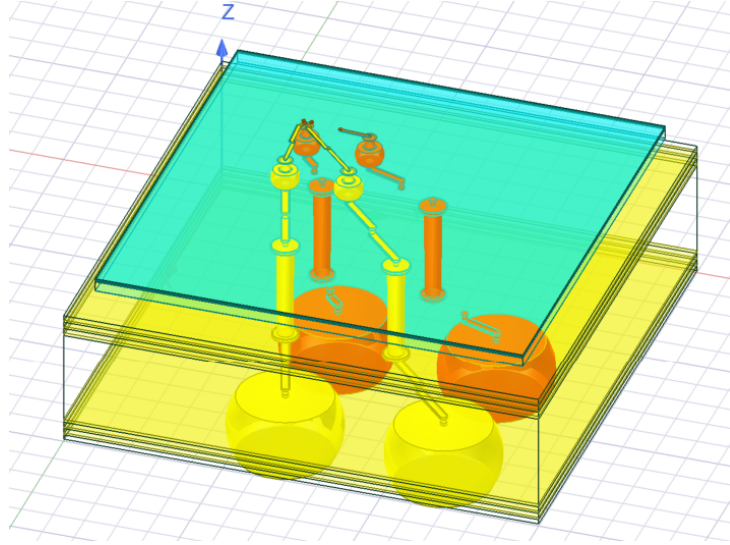
(b) Antipads with a trace connecting via pads

Figure 3.14: Antipads drawing cases

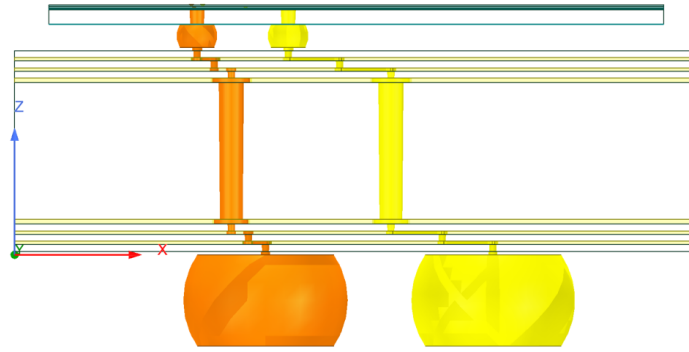
are shown in Figure 3.15. The yellow region is the substrate, and the cyan region is the interposer.

```
pads = pitch_c4/2
p1 = bupkg.pattern_adder("c11",{"Nx":1,"Ny":2,"pattern":["v1",
    "g"],"direction":"x"},(600,
    600),pads,pitch_c4)

bupkg.ubga_pinmap(pitch_ubga,c4_ubga_ratio=(1,1),map_type="
    shadow")
bupkg.ubga_pinmap(pitch_bga,c4_ubga_ratio=(1,1),level="bga",
    map_type="shadow")
bupkg.floorplan(fdraw_bump=True)
bupkg.build_model(finterp=True)
```



(a) Example PDN trimetric view



(b) Example PDN side view

Figure 3.15: Example PDN model

3.4.4 Setting Up Simulation

When the whole model is drawn, the simulation needs to be set up, including defining excitations, boundaries, and the analysis setup. To obtain S-parameters, ports need to be defined and there are usually two types of ports for package-level simulation: lumped port and wave port. The lumped port has to be used inside the model, whereas the wave port has to be located at the boundary of the object. For example, if the start of the transmission line segment is inside the structure, users must use a lumped port for the excitation. For using a wave port, de-embedding is available, which means the reference planes can be shifted, assuming the material and cross-sections are exactly the same. In this case, the S-parameters of long uniform structures

can be obtained by just simulating a small segment, saving simulation time and computational resources without losing accuracy. An airbox surrounding the model should be drawn as the radiation boundary to create an open model.

CHAPTER 4

DESIGN CHALLENGES IN PDN AND POSSIBLE SOLUTIONS

In the previous chapter, the general workflow for the design automation for PDN is outlined and discussed in detail. However, the floorplanning strategy of projection by shadowing only works for simple designs where the routing is not too complicated. For chiplets with industry standards, the number of bumps in one chiplet can easily exceed 1000, resulting in a huge demand for computational resources for both drawing the model and simulation. This chapter will discuss the cases that cannot be systematically handled by the automation script and propose some solutions to address those.

4.1 Routing Challenges

As the current strategy for floorplanning is projection by shadowing and for routing is dividing the horizontal shift needed evenly by the number of shifts possible, it cannot be guaranteed that the vias from different nets will not collide, which will sometimes make the design invalid. The likelihood of via collisions will be higher especially when the pitches at the chiplet-level and board-level are drastically different. Since the whole model is drawn with the automation script and the micro-via dimensions are much smaller compared to the package model, such design violations can only be identified with a scrupulous check. Advanced routing algorithms to avoid via collisions are possible, but they would be really time-consuming to study and implement.

In the case of multi-voltage domains, the solid metal planes for powers will be replaced with power islands allocated for each power domain. At the board-level, the bumps are usually grouped by the individual power domain, but they need to route in the package to reach the target chiplet pin in advanced industry applications, as one chiplet can require multiple powers to function as desired. One such example design is shown in Figure 4.1 where

the highlighted region is the power island for VSS, and others are for VCDC [20].

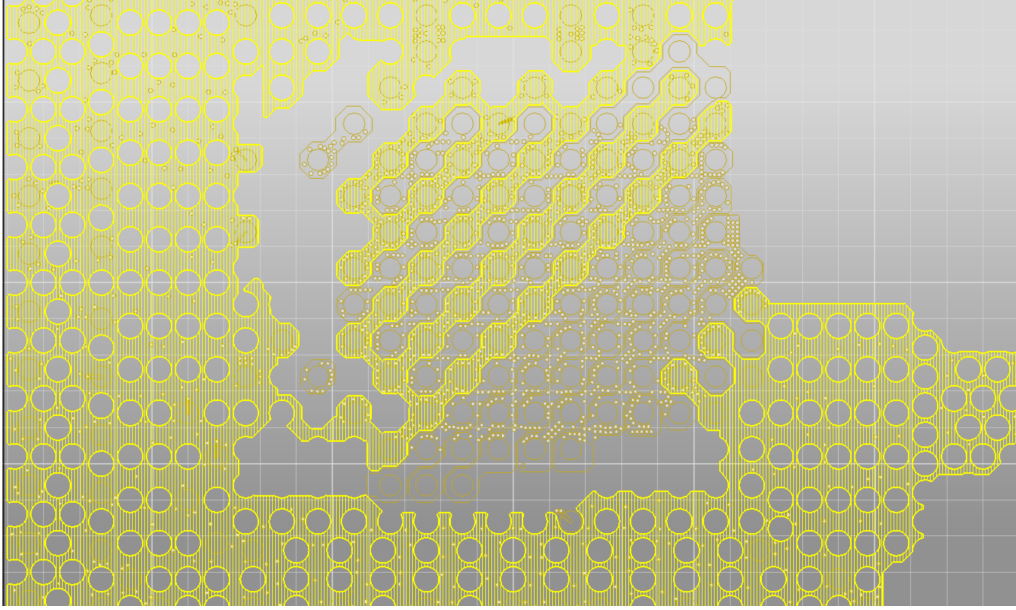


Figure 4.1: Power islands from a reference design

4.2 Timing Challenges

Considering the specifications of chiplet from recent research works as listed in Table 4.1 [21][22], the number of chiplet bump can be calculated by dividing the chiplet dimension by the pitch. Even for the smaller chiplet among the two, the number of bumps at the chiplet-level is 60×120 or 30×60 if signal bumps are ignored for PDN purposes.

Table 4.1: Chiplet specifications

	Chiplet 1	Chiplet 2
Supply Voltage	0.3V, 0.8V, 1.2V	0.62V, 0.8V
Clock Frequency	4 GHz	1.5 GHz
Dimension	4.4×6.2 mm	3×6 mm
Bump Pattern	P-S-G-S	P-S-G-S

An experiment for modeling the PDN in HFSS demonstrated that the time required to draw each micro-via would increase exponentially with the model

size, as shown in Figure 4.2. This implies that if the model is too large, the modeling part itself cannot be finished within a reasonable amount of time due to the memory usage. Since each interconnect has unique shifts and makes an antipad with different layers, it is infeasible to employ the 3D object duplication approach.

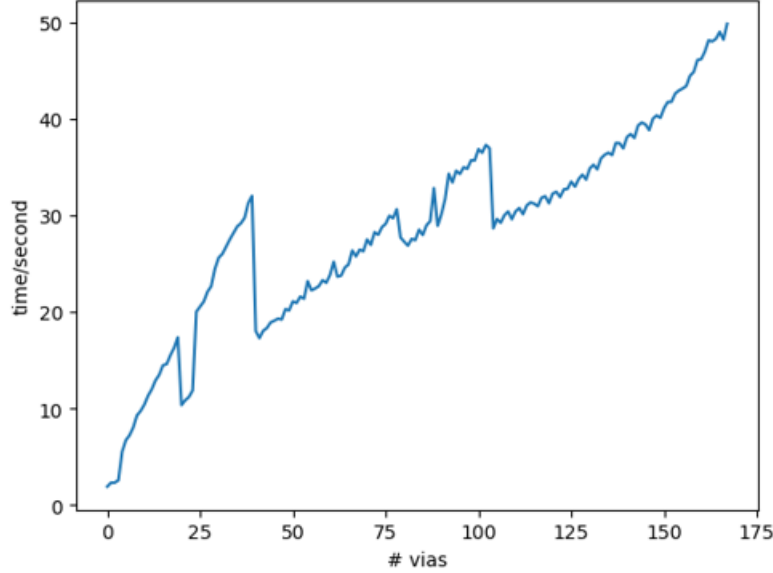
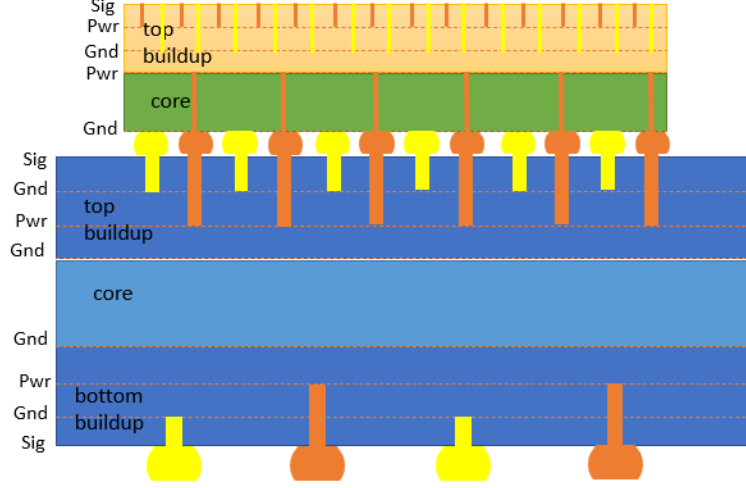


Figure 4.2: Timing analysis for drawing micro-vias in HFSS

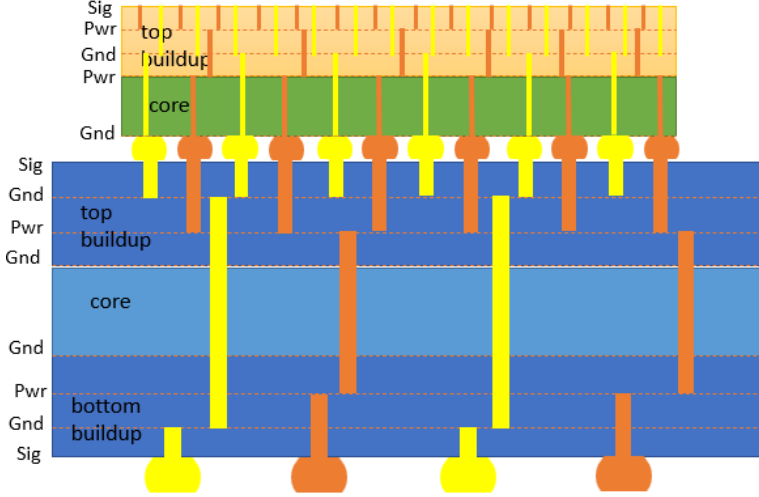
4.3 Solution

With the two major issues with the current routing algorithm and modeling limitations of HFSS, large chiplet PDN models cannot be accurately built and analyzed. As a solution to avoid via collisions, bumps are first connected vertically to the nearest power or ground layers as shown in Figure 4.3a. Then, long vertical vias are utilized to electrically connect the top and bottom of the layers in the substrate and interposer respectively, as shown in Figure 4.3b. It would be a simple search problem to find the feasible locations to place those long vias for connection where bump locations for the other nets are considered obstacles. It has to be guaranteed that the center position of the connecting via is at least $2 * R_{antipad}$ away from the vias for other nets.

In practice, whole metal planes are not available at the chiplet-level and



(a) Connecting bumps to the nearest power or ground layers



(b) Using long vias to connect the bottom and top layers

Figure 4.3: Routing solution for avoiding via collisions

only metal islands are available. In this scenario, only bumps directly above the islands will have vias connecting downwards, while others are connected to those bumps with via connections through traces. The top view of the chiplet in this scenario is shown in Figure 4.4 where the gray area represents the power island.

If wave ports are used as excitation, a large rectangular sheet parallel to the xy-plane needs to cover all the bumps of interest, and the ground bumps should be identified as references. Even though the traces belong to the same net as the vias, they should not be united because each power bump should be a distinct port for the purpose of studying the difference in performance

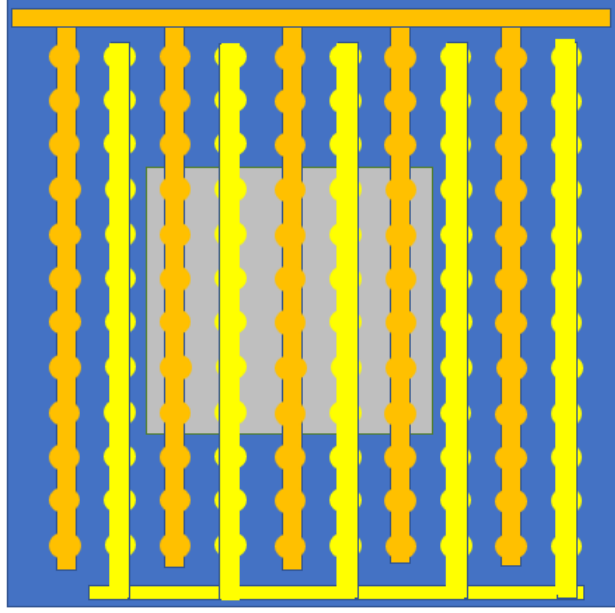


Figure 4.4: Trace connections at the chiplet-level

based on bump locations. Therefore, traces will also be considered as ports if the trace connections are on the top layer of the interposer. To solve this issue, trace connections can be done on one layer below the top layer, and vias are used to connect the bumps to the traces. Since the via connections here are vertical, it is possible to use the duplicate object feature in HFSS to reduce the time in drawing.

On the other hand, for using lumped ports, a small rectangular sheet needs to be drawn for each pair of power and ground bumps, where ground bumps serve as the reference. In this scenario, trace connections should also be done at the layer below the top layer for correctly setting up the ports.

By experimenting with different simulation excitations, it is observed that using terminal wave ports would take a very long time for the simulation to complete, and there is a limitation on the number of signal conductors associated with each wave port. However, if modal lumped ports are employed with the integration ling pointing from the signal conductor to the ground conductor, the simulation can be completed much faster with reasonable results.

CHAPTER 5

TEST CASES

With the general automation workflow for the PDN and the solutions for the challenges with building large PDN models in HFSS, three test cases are built and the results are analyzed in this chapter.

5.1 PDN with 4 Chiplets in a Single Voltage Domain

In the first test case, a PDN is built in a 2.5D package for 4 chiplets in the same voltage domain, and the bump patterns alternate between power and ground. For PDN applications, it is not necessary to look at the S-parameters; instead, the input impedance of the PDN as a function of frequency needs to be carefully analyzed because a PDN with large impedance will result in fluctuations in the supply voltage, causing some critical components to malfunction. Also, the stability of the PDN can be analyzed by the voltage droop in the time domain. Decoupling capacitors can be placed near the ICs to reduce the PDN impedance by storing and releasing the electrical energy. The purpose of this test case is to find the input impedance seen from the board-side and chip-side and try to place the decoupling capacitors at the chiplet-level in order to minimize the impedance. In this scenario, the PDNs for four chiplets with different aspect ratios are modeled with the number of bumps in each chiplet shown in Table 5.1.

Table 5.1: Number of bumps for each chiplet in test case 1

Chiplet number	Number of bumps
1	12×12
2	12×8
3	12×6
4	8×8

To further reduce the time in modeling and simulation, only the bumps directly above the islands (covering around 30% of the total bumps) and at the edges of the chiplets are defined as ports, and other bumps are represented with traces. The candidate locations for the die-side discrete decoupling capacitors are close to the chiplets and they are connected to the power and ground bumps through traces, as shown in Figure 5.1 in a top view [23].

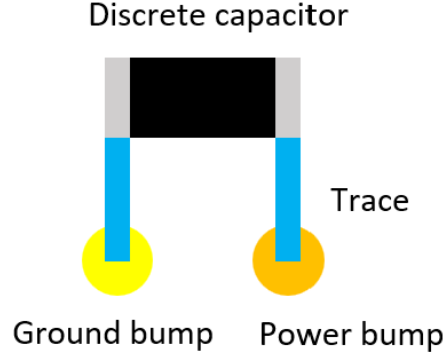


Figure 5.1: Decoupling capacitor placement

The bump reduction ratio from C4-level to μ BGA-level is (4,4), and from μ BGA-level to BGA-level is (2,2) so that the areas occupied by the three levels are roughly proportional. The complete model of the PDN for four chiplets in a single power domain is shown in Figure 5.2 where the blue regions are the islands in the interposer. Lumped ports are used for the excitation with modal solution type, which means a lumped port is defined for each pair of power and ground bumps with the integration line pointing from the power bump to the ground bump at the chiplet-level and BGA-level. After the full-wave simulation, 140-port S-parameters are obtained.

To find the input impedance based on the S-parameters, the bumps not being analyzed should be terminated, and a signal flow graph needs to be employed to derive the input reflection coefficient. It can be reasonably assumed that the bumps are terminated with $100\text{ m}\Omega$ as it roughly matches the impedance looking down to the board or looking up to the chiplet. Under this assumption, using a characteristic impedance of $100\text{ m}\Omega$ when setting up the ports will greatly simplify the SFG because $S(N, N)$ is the input reflection coefficient at port N when all other ports are matched to Z_0 . Therefore, the

input impedance at port N can be found by

$$Z_{in,N} = Z_0 \frac{1 + S(N, N)}{1 - S(N, N)} \quad (5.1)$$

Figure 5.3a shows the magnitude of the input impedance at a BGA port, and Figure 5.3b shows the magnitude of the input impedance at a C4 port of this PDN model for one power domain. It can be observed that the input impedance for the C4 port is smaller than that of the BGA port and the impedance of the BGA port has resonance and anti-resonance at around 1 GHz due to the effects of the traces.

5.2 PDN with 4 Chiplets in Two Voltage Domains

In the second test case, a PDN is built in a 2.5D package for the same 4 chiplets as in case 1 in two different power domains with the same alternating power and ground patterns. Chiplets 1 and 3 are in the same power domain, while chiplets 2 and 4 are in another power domain. Practically, each chiplet can have multiple voltage domains, but due to the current limitations of the floorplanning algorithm, each chiplet only has one voltage domain, and the islands for the two different domains are separated.

The bump reduction ratio from C4-level to μ BGA-level and from μ BGA-level to BGA-level are the same as the previous test case with one voltage domain. The complete model of the PDN for four chiplets in two power domains is shown in Figure 5.4, where the blue regions are the islands in the interposer. After the full-wave simulation with lumped ports, 126-port S-parameters are obtained.

Similarly, the input impedance in this test case is analyzed with Equation 5.1. Figure 5.5a shows the magnitude of the input impedance at a BGA port, and Figure 5.5b shows the magnitude of the input impedance at a C4 port of this PDN model for one power domain. The overall impedance patterns are matched to the case with one voltage domain, but the input impedance at the BGA port is larger than the previous case with two resonance peaks in the frequency range from 1 MHz to 2 GHz.

5.3 Differential Via Pair

The automation workflow is versatile, addressing not only PDNs but also various signal transmission schemes. It is critical to recognize that signals, usually transmitted at the rates in the tens of Gigabits per second (Gbps), cannot be connected through the copper planes like interconnects for power and ground. This limitation is mainly attributed to the skin effect [24]. At higher frequencies, the electric currents tend to be distributed mostly around the surface of the conductor and the current amplitude will decrease exponentially with the depth into the conductor. The skin depth (δ) denotes the distance from the conductor surface at which the amplitude reduces to $\frac{1}{e}$ of the peak amplitude. The skin depths for the copper at different frequencies are listed in Table 5.2. Consequently, considering the skin depth and the thickness of the copper planes, signal vias cannot directly contact the solid copper plane; instead, they should be routed with traces.

Table 5.2: Skin depth for copper at different frequencies

Frequency	Skin depth (μm)
50 Hz	9220
10 kHz	652
100 kHz	206
1 MHz	65.2
100 MHz	6.52
1 GHz	2.06

In this test case, a differential via pair model is constructed for high-speed link applications. There are two vertical interconnects for the signals and grounds, respectively, within a package substrate featuring 5 metal layers assigned to the ground net. Whenever the signal via intersects with the ground plane, an antipad will be drawn around the via. Wave ports are placed at the top and bottom of the model for the excitations with the ground planes serving as references. The complete model in HFSS is shown in Figure 5.6.

The parameters of the differential via pair model are defined inside HFSS as variables, allowing for easy parameter sweeping to identify which sets of parameters has significant impact on the overall SI performance. The list of design parameters and their ranges are detailed in Table 5.3 [25].

Table 5.3: Design parameters for the differential via pair model

Parameter	Symbol	Unit	Min	Max	Nominal
Core via diameter	d_{core}	μm	75	200	100
Buildup via diameter	d_{bu}	μm	30	70	30
Core pad radius	R_{core}	μm	50	200	75
Buildup pad radius	R_{bu}	μm	20	140	20
Core antipad radius	$R_{apad,core}$	μm	50	200	140
Buildup antipad radius	$R_{apad,bu}$	μm	35	200	140

When these parameters are set to the nominal values, the transmission S-parameters for the two vertical vias are shown in Figure 5.7. Even at 40 GHz , S_{21} is larger than -1 dB , indicating the channel has extremely low loss and is suitable for transmitting high-frequency signals with low distortion.

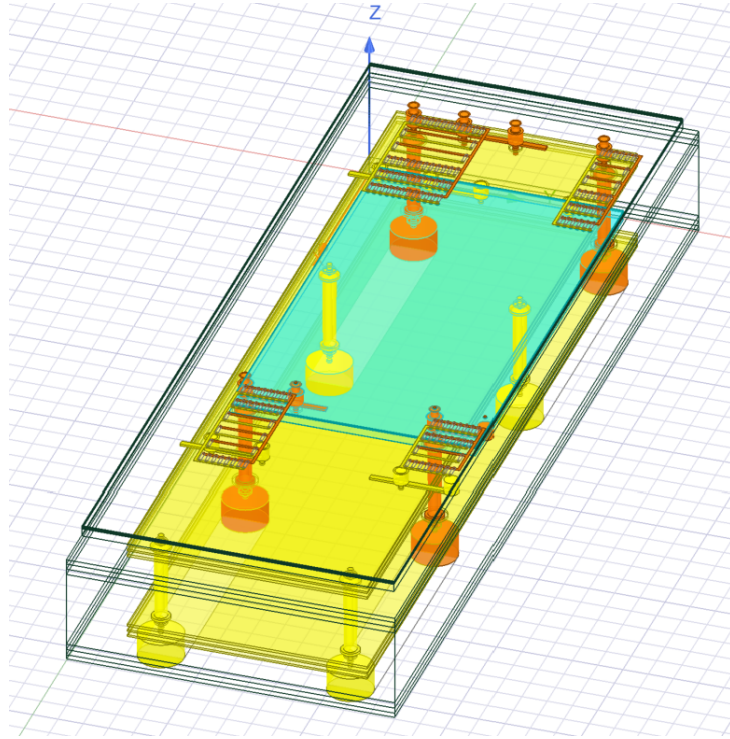
The eye diagram is a useful tool to evaluate the high-speed channel's performance and it is easy to see the inter-symbol interference (ISI), clock jitter, and crosstalk effects from the eye diagram. It can be generated by overlaying the different bit transitions (0 to 1, 0 to 1, 1 to 0, and 1 to 1) on a single plot using the output of the channel when the input is a bit sequence. There have been other studies about generating the eye diagram faster at the cost of losing some accuracy, such as the statistical method [26] and fast transient simulation via latency insertion method (LIM) [27].

An ADS circuit schematic, as depicted in Figure 5.8, is designed to perform an eye diagram simulation based on the model S-parameters obtained from the HFSS full-wave simulation. In this test case, the transmitter sends signals at the bit rate of 20 Gbps with 10 ps rise time. Also, a differential signaling convention is employed to enhance noise suppression. It is expected to see that level 1 aligns with 0.5 V , and level 0 aligns with 0 V because there are voltage divisions between the source impedance and the load impedance.

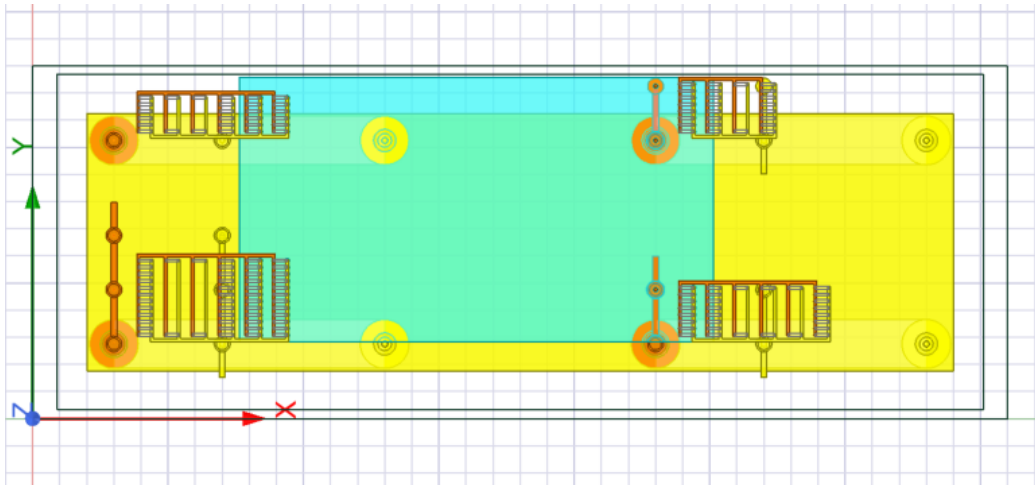
Initially, several simulations are performed by sweeping the antipad radii while keeping other parameters at the nominal values. The eye diagram when $R_{apad,core} = R_{apad,bu} = 140\text{ }\mu m$ is shown in Figure 5.9a; the eye diagram when $R_{apad,core} = R_{apad,bu} = 90\text{ }\mu m$ is shown in Figure 5.9b; and the eye diagram when $R_{apad,core} = 90\text{ }\mu m$ and $R_{apad,bu} = 35\text{ }\mu m$ is displayed in Figure 5.9c. Comparing these plots reveals that decreasing the antipad radii would deteriorate the channel performance since reducing the gap between different conductors would increase the mutual coupling and crosstalk. In addition,

the antipad radii in the core region exert a more significant impact on the channel performance compared to those in the buildup region.

Subsequently, the pad and via radii in the core region are swept while keeping other parameters at nominal values. The eye diagram when $d_{core} = 100\ \mu m$ and $R_{core} = 75\ \mu m$ is shown in Figure 5.10a; the eye diagram when $d_{core} = 150\ \mu m$ and $R_{core} = 110\ \mu m$ is presented in Figure 5.10b; and the eye diagram when $d_{core} = 75\ \mu m$ and $R_{core} = 50\ \mu m$ is displayed in Figure 5.10c. Results indicate that varying the pad and via dimensions in the core region does not have a significant impact on the channel performance, as the eye heights are almost the same among the three cases. However, increasing the via and pad radii in the core region does slightly degrade the channel performance because it increases the impedance mismatch between the core and buildup regions.

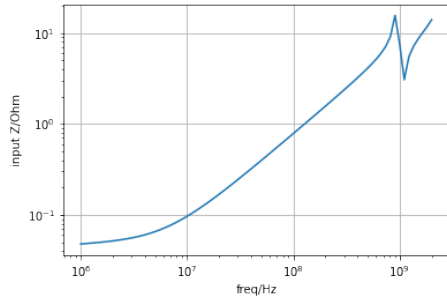


(a) PDN for four chiplets in one power domain with trimetric view

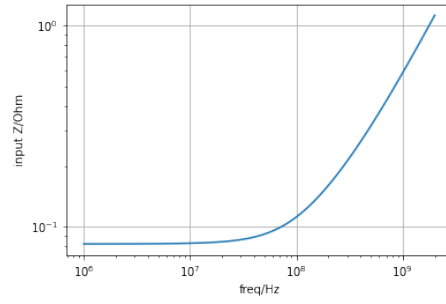


(b) PDN for four chiplets in one power domain with top view

Figure 5.2: PDN model for four chiplets in one power domain

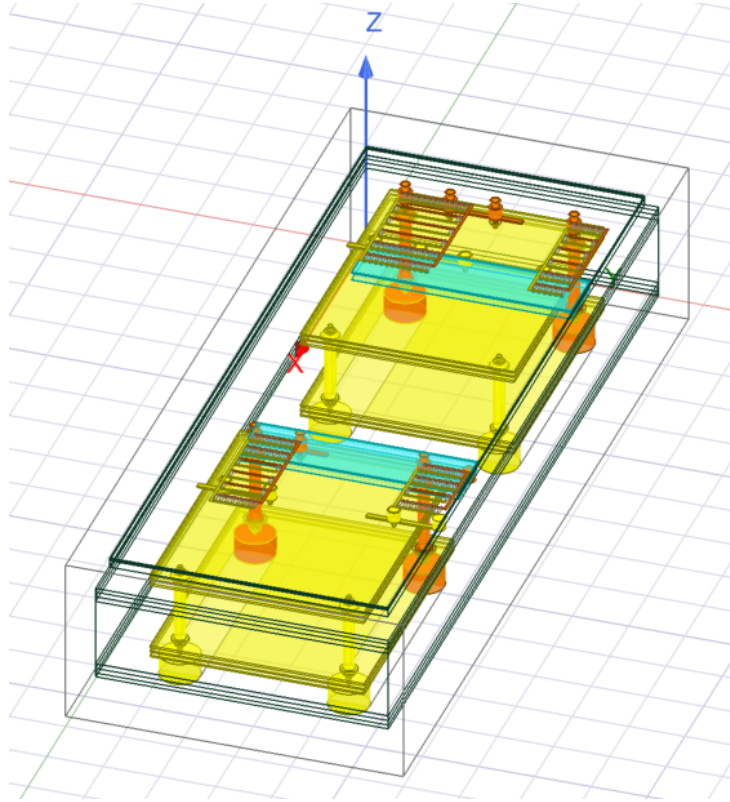


(a) Input impedance at a BGA port with single power domain

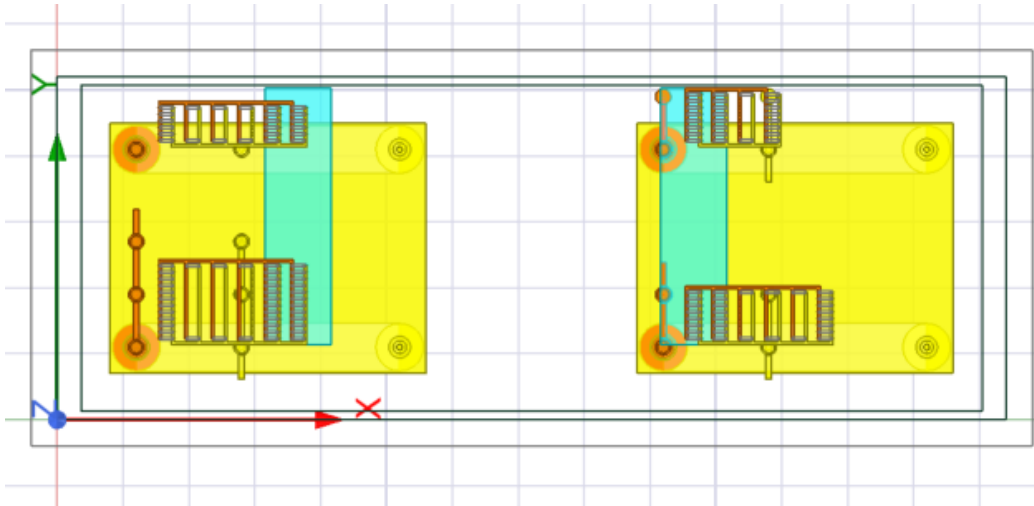


(b) Input impedance at a C4 port with single power domain

Figure 5.3: Input impedance for the PDN model for four chiplets in one power domain

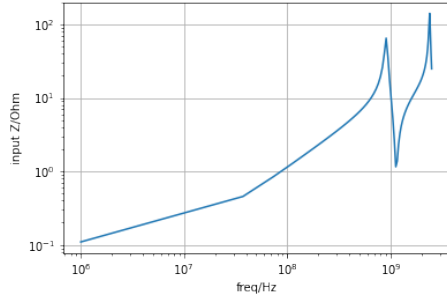


(a) PDN for four chiplets in two power domains with trimetric view

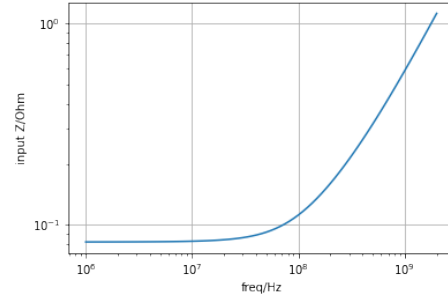


(b) PDN for four chiplets in two power domains with top view

Figure 5.4: PDN model for four chiplets in two power domains



(a) Input impedance at a BGA port with two power domains



(b) Input impedance at a C4 port with two power domains

Figure 5.5: Input impedance for the PDN model for four chiplets in two power domains

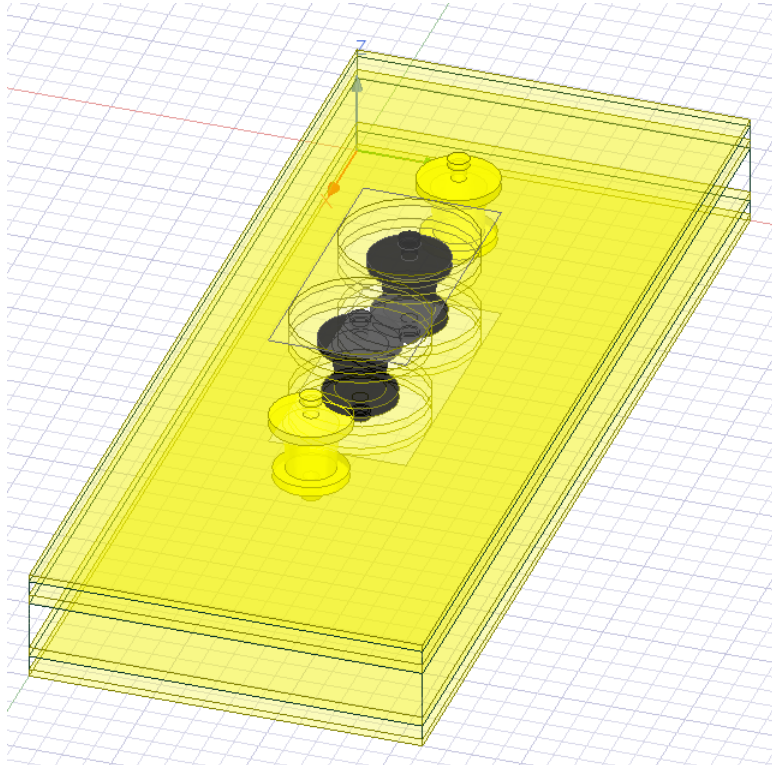


Figure 5.6: Differential via pair model

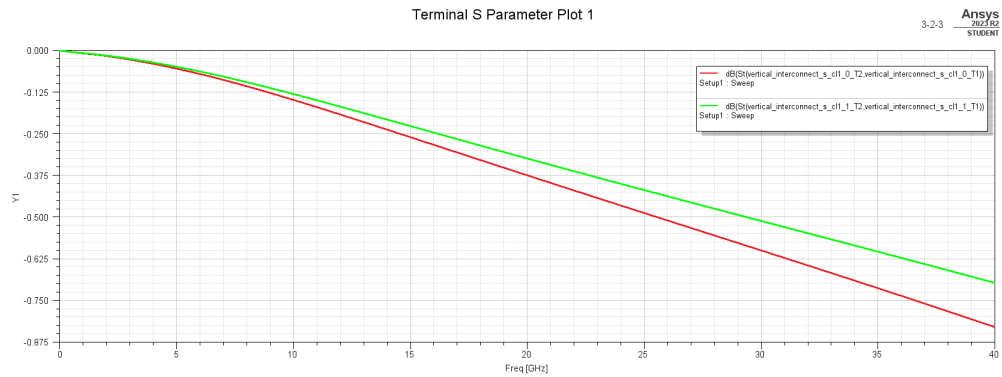


Figure 5.7: Transmission S-parameters for the differential via pair model with nominal parameter values

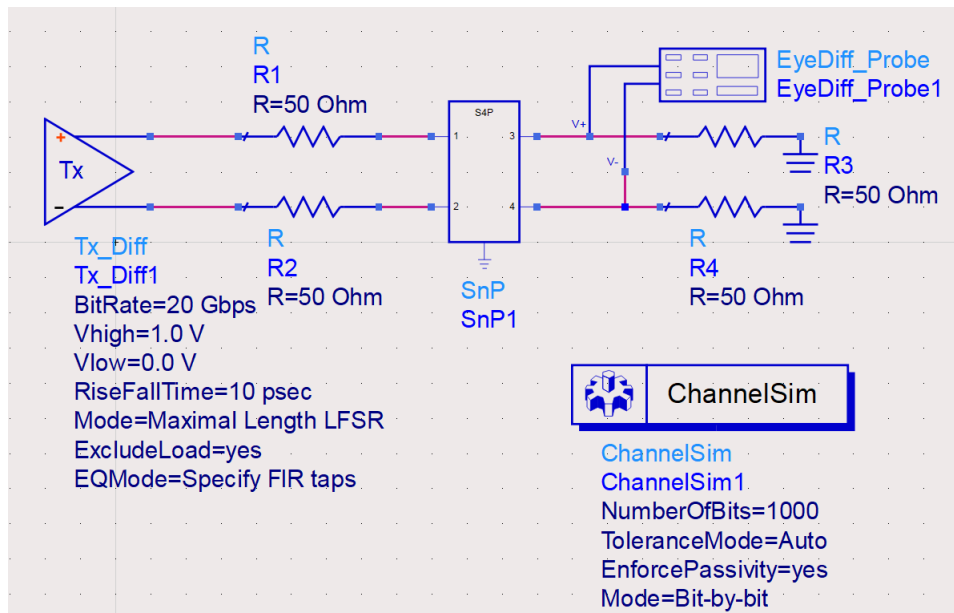
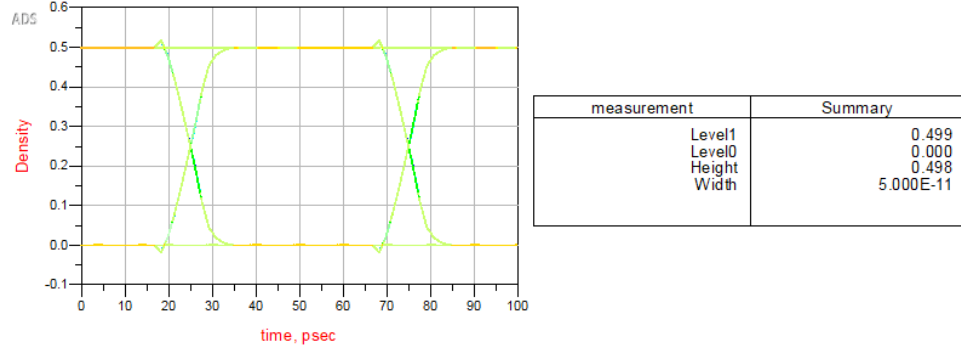
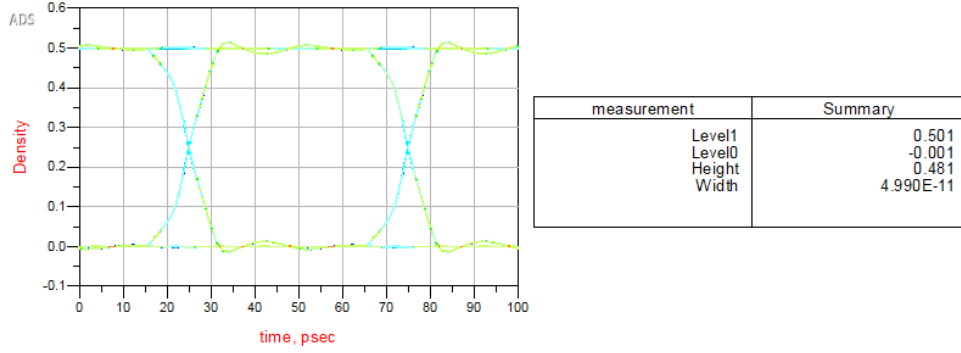


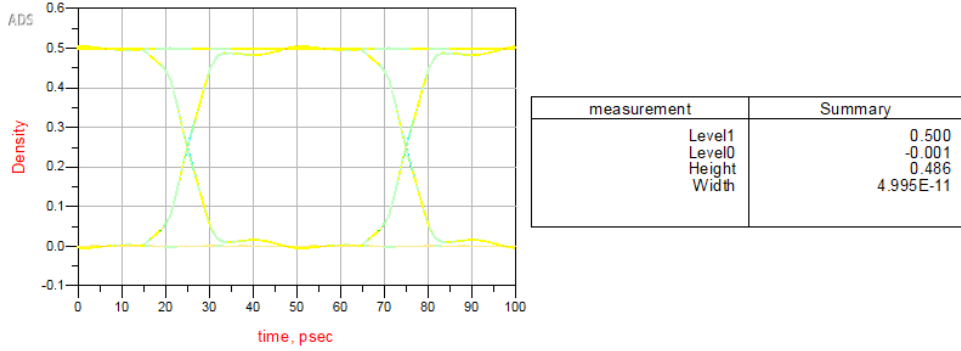
Figure 5.8: ADS schematic for eye diagram simulation



(a) Eye diagram when $R_{apad,core} = R_{apad,bu} = 140 \mu m$

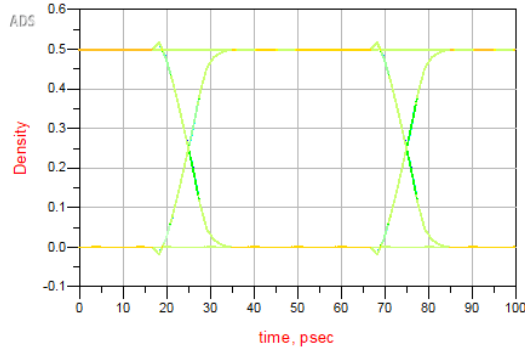


(b) Eye diagram when $R_{apad,core} = R_{apad,bu} = 90 \mu m$



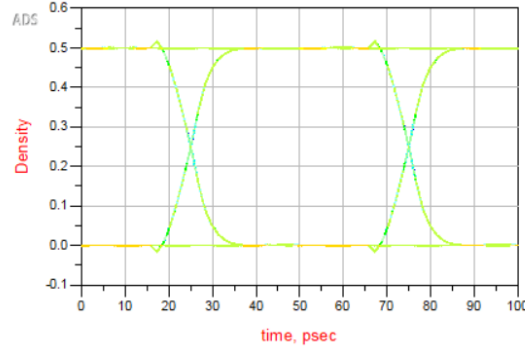
(c) Eye diagram when $R_{apad,core} = 90 \mu m$ and $R_{apad,bu} = 35 \mu m$

Figure 5.9: Eye diagrams by sweeping the antipad radii



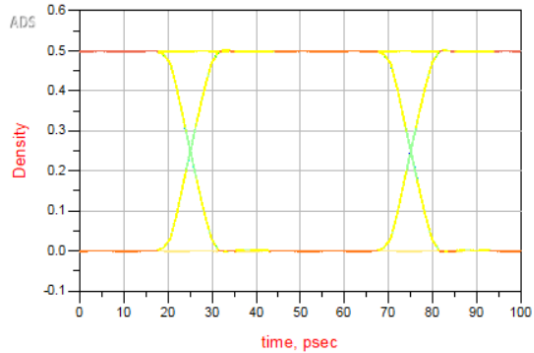
measurement	Summary
Level1	0.499
Level0	0.000
Height	0.498
Width	5.000E-11

(a) Eye diagram when $d_{core} = 100 \mu m$ and $R_{core} = 75 \mu m$



measurement	Summary
Level1	0.499
Level0	2.134E-4
Height	0.497
Width	4.995E-11

(b) Eye diagram when $d_{core} = 150 \mu m$ and $R_{core} = 110 \mu m$



measurement	Summary
Level1	0.499
Level0	0.000
Height	0.499
Width	4.995E-11

(c) Eye diagram when $d_{core} = 75 \mu m$ and $R_{core} = 50 \mu m$

Figure 5.10: Eye diagrams by sweeping the via and pad radii in the core region

CHAPTER 6

BUNCH OF WIRES - A CHIPLET-TO-CHIPLET INTERFACE

The previous chapters elucidate the methodology and test cases concerning the PDNs and the high-speed links for the communication between the chiplets and the board. However, for chiplets to communicate with each other, it is not necessary for them to descend to the board-level; but instead, dedicated chiplet-to-chiplet (C2C) communication links, such as the Bunch-of-Wires (BoW) and Universal Chiplet Interconnect Express (UCIE), are integrated within the organic interposer of the 2.5D package. This integration not only streamlines inter-chiplet communication but also enables the realization of fine-pitch interconnects.

In the broader context of integrating C2C links onto the chiplet-level, the core regions of the chiplets contain the PDNs and signal links to communicate with the board, while the C2C regions are positioned at the peripherals of the chiplets to communicate with other chiplets, as shown in Figure 6.1. This chapter will briefly introduce the BoW standard and the automation involved in building the structure.

6.1 BoW Standard Specification

Bunch of Wires is an open-source specification that defines the bump patterns and arrangements for C2C links [1][28]. In each slice, there can be 18 to 20 bumps belong to single-ended signals, differential clocks, optional forward error correction, and optional auxiliary signal, as specified in Table 6.1. A stack can be formed by combining multiple slices associated with one chiplet to increase I/O density, and a link can be established by one or multiple stacks between two chiplets for communication purposes. The slice can function as both a transmitter (Tx) and receiver (Rx), but a minimal bidirectional link (Figure 6.2) should contain at least two slices for each chiplet, with one slice

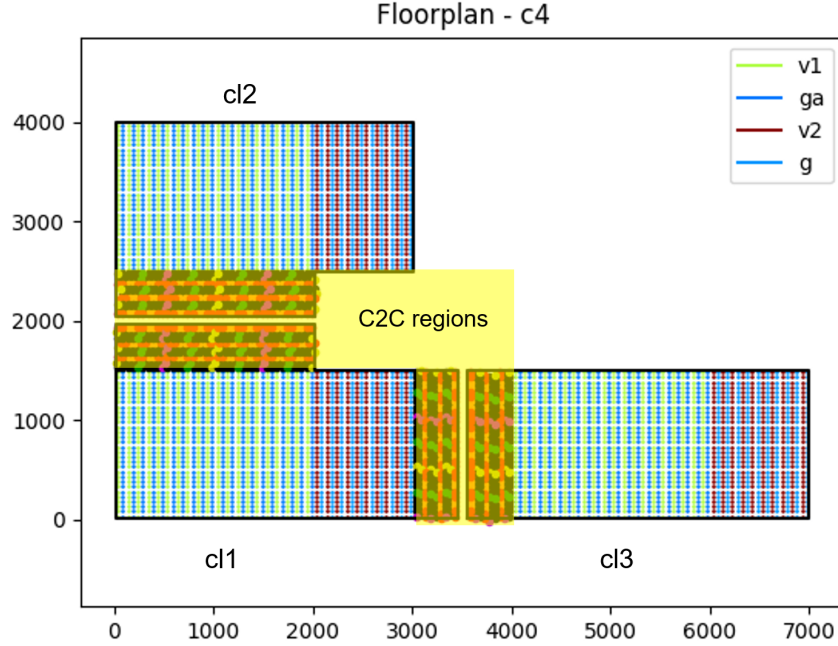


Figure 6.1: Big picture of the integration of the the C2C links and core of the chiplets

functioning as Tx and the other as Rx. A number of power and bumps are located at the edge of the slice that will descend to the board-level for power delivery.

Table 6.1: BoW Signals at the C2C Interface

Function	No. wires	Signal name
Data	16	D0-15
Clock	2	CLK+, CLK-
Forward error correction (FEC)	0/1	FEC
Auxiliary	0/1	AUX

The bumps in a BoW slice follow a staggered pattern, where the first bump in the second row is shifted half a pitch in the x-direction and one pitch in the y-direction from the first bump in the first row. This bump arrangement allows for straight trace connections between slices for two different chiplets, as shown in Figure 6.2, which facilitates horizontal routing. For vertical routing, the slices in the same row will use the same layer in the interposer for trace connection, and the bumps should be connected to vias that go vertically down. The vertical routing scheme is illustrated in Figure 6.3.

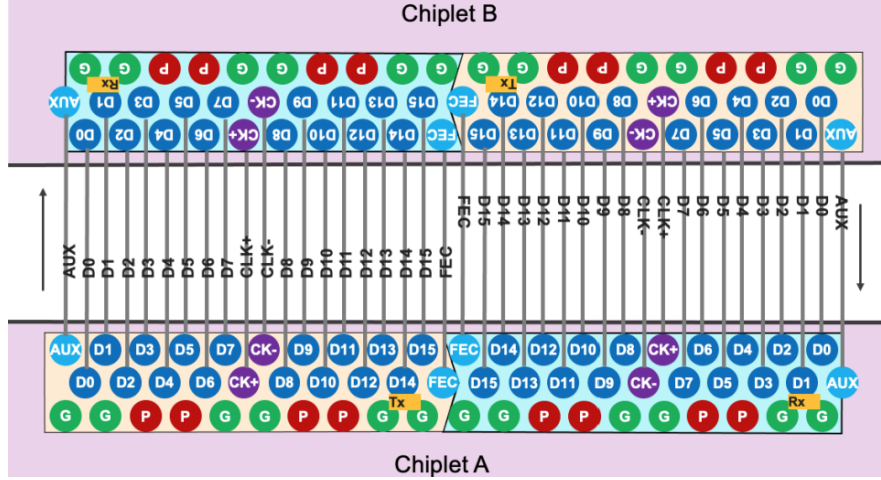


Figure 6.2: BoW Minimal Bidirectional Reference Link. Adapted from [1]

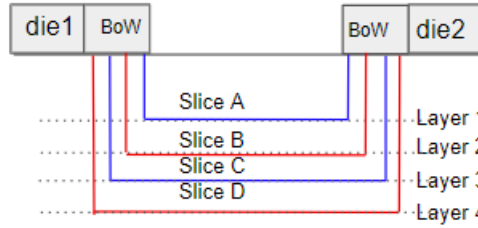


Figure 6.3: Vertical routing for BoW

In recent research works, SI and PI simulations have been performed regarding the BoW interface. It has been demonstrated that adding power planes in the package and performing optimization on the series impedance on each data wire would improve the signal SNR at the Rx. Most importantly, the study confirms that utilizing BoW for C2C interface would enable heterogeneous integration of chiplets in advanced packaging with decent performance [12].

6.2 Design Automation for BoW

The design automation workflow for BoW (Figure 6.4) mirrors the workflow for PDNs. Indeed, the automation for both cases have to be integrated into a single Python class for floorplanning and building the entire model, including

the PDNs and BoW C2C interfaces. The only difference is that floorplanning is not necessary for BoW, as those bumps do not need to descend to the board-level, not even the μ BGA-level; bump connections are made in the interposer, so only obtaining the bump locations at the chiplet-level would suffice.

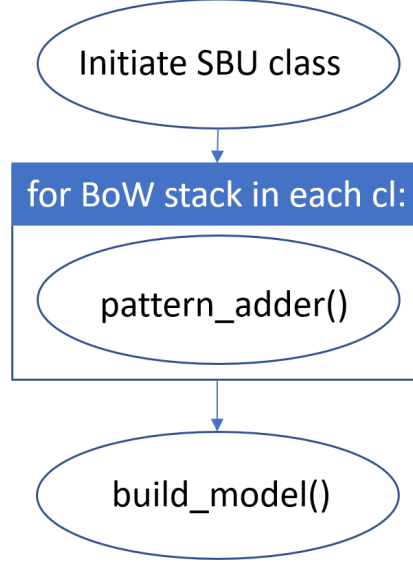


Figure 6.4: Workflow for BoW modeling automation

6.2.1 Getting BoW Bump Locations at the Chiplet-level

For each stack associated with one chiplet, users need to specify the number of slices in the x- and y-direction, the bound origin of the first slice, and the direction in which the bump pattern extends as a two-element array. The first element in the direction specifies the direction from the bound origin to the power and ground bumps, while the second element specifies the direction from the bound origin to the data bumps. The conventions for defining the bound origins and the directions are illustrated in Figure 6.5 with the minimal bidirectional link. Similar conventions can be derived for the links when all the slices are rotated by 90 degrees, which is useful when the trace connections go horizontally.

Once the bound origin and direction of the first slice in the stack are defined, this information for the rest of the slices in the stack can also be

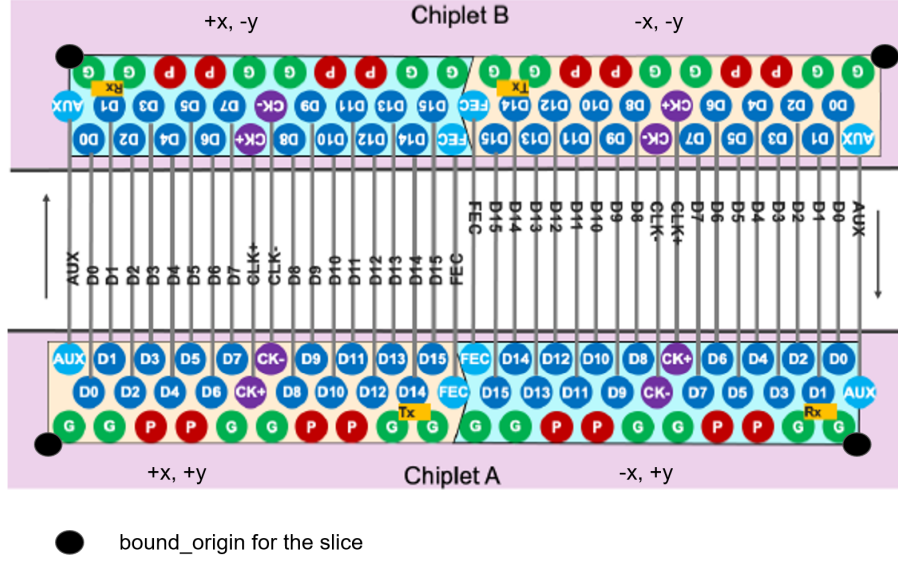


Figure 6.5: Conventions for defining bound origins and directions

derived following the trend, and the bump locations can be calculated by following the pitch pitches at the chiplet-level and the staggered pattern given by the BoW specification. Note that the bump nearest to the bound origin of the slice is $(\text{pitch_c4}/2, \text{pitch_c4}/2)$ away from the bound origin.

An example BoW stack is implemented with 5 slices in the x-direction, 3 slices in the y-direction, and the direction of the first slice is $[“+x”, “+y”]$. The plot of bump locations at the chiplet-level is shown in Figure 6.6. The bounding boxes for the slices partially overlap because the bounding box is half pitches away from the bumps on the edge and due to the staggered nature of BoW.

6.2.2 Drawing Interconnects for BoW

For the complete BoW model with two chiplets, the interconnects for the data bumps will be routed in the interposer, while the interconnects for the power and ground bumps will descend to the board-level for power delivery, as shown in Figure 6.7.

For drawing the interconnects for power and ground bumps, the exact procedure for PDNs will be applied to perform floorplanning on the three different levels and drawing the model. For data bumps, users need to specify

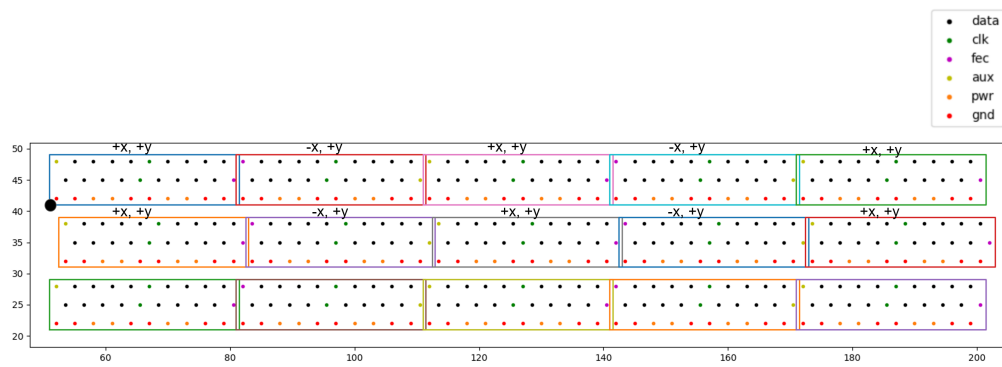


Figure 6.6: Example BoW stack with bounding boxes

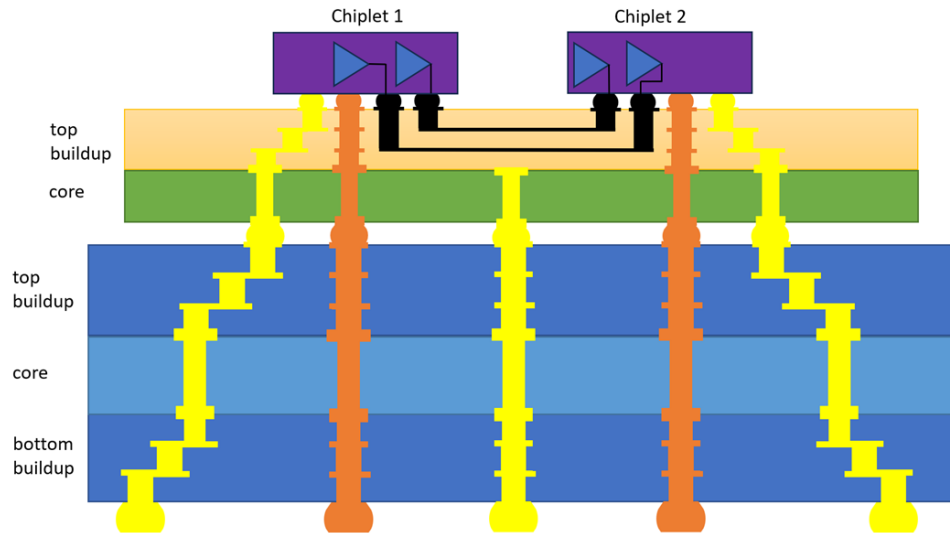


Figure 6.7: BoW model with two chiplets

the name of the slice for the first chiplet being connected to the name of the slice for the second chiplet, and the name of the layer in the interposer for trace routing as a list, with an example shown below.

```
cl_connection=["cl1","cl2","cl1_bow1","cl2_bow1","ibu_M3"]
```

The function **draw_io_interconnect()** will take a pair of bump locations at the chiplet-level, draw microvias that connect from the top of the interposer to the designated layer in the interposer, and draw the trace to connect the vias pads at the designated layer.

The example BoW structure with one slice for each chiplet, built with the automation script, is shown in Figure 6.8.

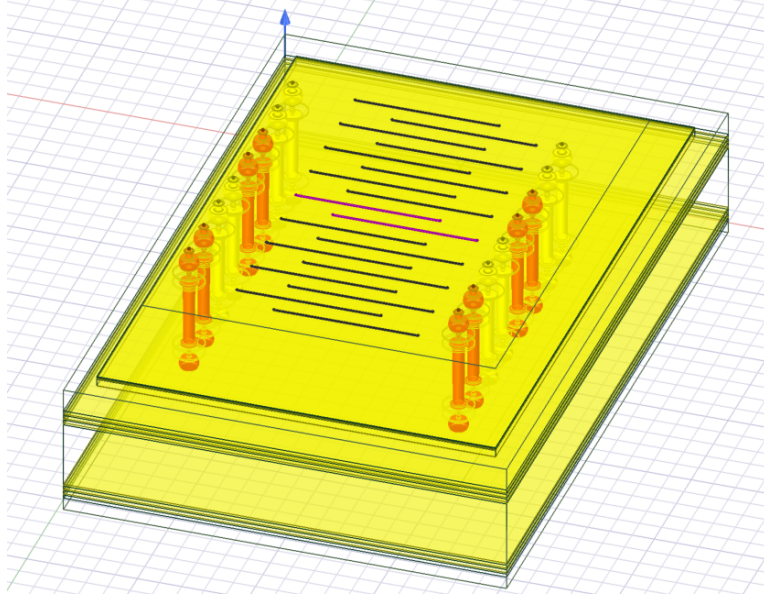


Figure 6.8: BoW 3D model with one slice for each chiplet

6.3 Embedded Multi-die Interconnect Bridge (EMIB)

Besides drawing the trace connections for BoW in the organic interposer of the 2.5D package, these horizontal interconnects can also be positioned in a special silicon region in the substrate called EMIB. EMIB is a thin silicon bridge that is embedded within the top two layers of the package substrate, allowing localized high-density interconnects without affecting other regions in the package [2]. The cross-section of a package with traces for C2C links in EMIB is shown in Figure 6.9.

Within the EMIB region, there are typically four metal layers, with two intended for trace routing and the others for ground as shielding. The trace width, spacing between adjacent traces, and the metal and dielectric layer thicknesses are all $2\mu m$ [3], with Figure 6.10 showing the side view of a fabricated EMIB. Recent studies have compared the insertion loss and far-end crosstalk between simulation and measurement on a pair of 4mm-long data interconnects in EMIB (Figure 6.11) [4]. It is found that the insertion loss is larger than 1 dB at low frequencies due to the large resistance of the traces caused by their small thickness.

However, since the EMIB technology, including its material properties, via geometries, and specific routing schemes, remains largely proprietary to Intel

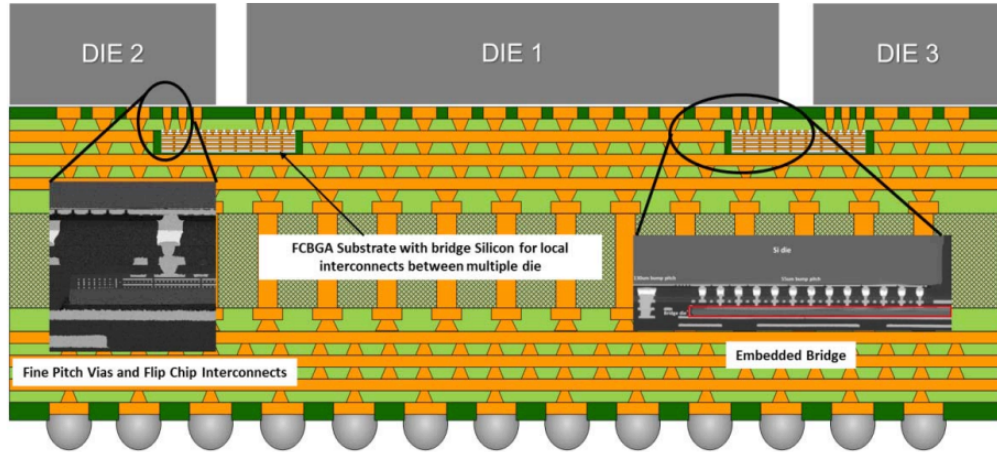


Figure 6.9: Cross sectional view of EMIB in package. Adapted from [2]

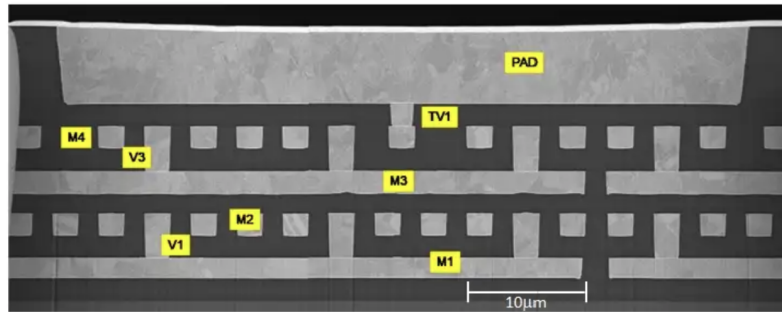


Figure 6.10: Side view of fabricated EMIB. Adapted from [3]

Corporation, it is difficult to perform modeling and analysis with BoW traces routed in EMIB.

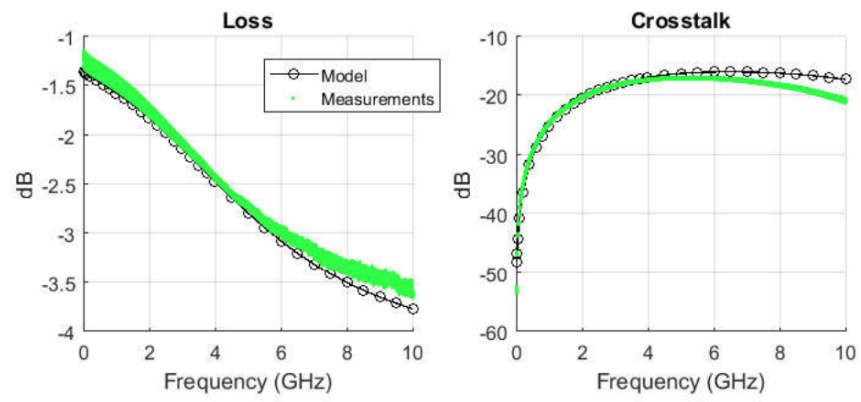


Figure 6.11: Insertion loss and far-end crosstalk for EMIB. Adapted from [4]

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this thesis, a comprehensive design automation workflow for PDNs is presented, covering initializing geometric parameters, adding chiplet patterns, floorplanning on three levels, building the 3D model in HFSS, and performing EM simulation. Through extensive testings, it is verified that the basic workflow performs well when the model size is moderate, and the pitches at the chiplet-level are not drastically smaller than those at the board-level. For larger and more realistic PDN models, the proposed solutions for via routing and excitation selection will ensure that the model can be built and solved within a reasonable amount of time.

In the test cases involving large PDN models for four chiplets of different sizes, it is observed that the input impedance looking from the BGA port exhibits resonance at around 1 GHz , which agrees with the typical impedance behavior for PDNs found in other literature [29]. With the differential via pair model, it is demonstrated that the signal would only experience low distortion when transmitting at 15 Gbps in this channel, based on the eye diagrams. In addition, by sweeping some design parameters, it is found that the antipad radius is one of the most crucial parameters in designing the channel, as larger antipads would reduce the coupling between conductors belonging to different nets.

Bunch of wires is a promising C2C interface standard to integrate with the PDNs of the chiplets, aligning with the needs for heterogeneous integration. The staggered bump patterns facilitate trace routing, and the slice arrangements provide flexibility in designs. The traces for BoW can be routed in the EMIB region of the package to achieve localized ultra-high-density interconnections with good SI performance.

7.2 Future Work

For large PDN models, the current approach of connecting the bumps to the nearest ground or power layer and then using long vias to the bottom and top layers of a net serve as only temporary solutions for the routing problem. In reality, metal planes are usually divided into smaller islands for multi-power domain applications, and each chiplet may contain multiple powers for normal operation. This necessitates a more robust algorithm for finding the feasible routes between the two bump positions in 3D space, with the existing vias and pads as obstacles.

Concerning the EM simulations in HFSS, it has been observed that using modal lumped ports would save a lot of time in solving the S-parameters compared to using wave ports. However, the S-parameters obtained by the two different ports exhibit slight discrepancies, and it is unclear which one yields more realistic results. To address for this, a test structure with identical geometries and materials can be fabricated, and a VNA can be used to measure the S-parameters of the model for comparison with simulation results. In addition, there is a limitation on the number of signal conductors that can touch a single wave port. Therefore, the results obtained by using a large wave port to cover all the bumps and using multiple smaller wave ports to cover the same number of bumps should be compared to provide insights into future excitation setup.

In future work, several case studies can be carried out to explore the routing layouts for some commercially available C2C links. The investigations should explore the routing possibilities for a single C2C link, considering both single-layer and dual-layer configurations. Also, for scenarios involving two C2C links, the study should explore the options of utilizing separate layers for each link's routing or amalgamating part of them on the same layer for spatial efficiency. When multiple links share a layer for routing, the design allows for flexibility in selecting which signal lines from the two links should share the same layer, while others occupy a distinct layer. Notably, for recent heterogeneous integration applications, different chiplets may exhibit diverse pitches, both in the core and I/O regions. This variation in pitches poses challenges for routing in the same layer, prompting an in-depth analysis of the design performance with various multi-layer routing strategies in comparison to the uniform pitch scenario.

REFERENCES

- [1] “Bunch of wires (BoW) phy specification - open compute project,” 2023. [Online]. Available: <https://www.opencompute.org/documents/bow-specification-v2-0d-1-pdf>
- [2] R. Mahajan, R. Sankman, N. Patel, D.-W. Kim, K. Aygun, Z. Qian, Y. Mekonnen, I. Salama, S. Sharan, D. Iyengar, and D. Mallik, “Embedded multi-die interconnect bridge (EMIB) – a high density, high bandwidth packaging interconnect,” in *2016 IEEE 66th Electronic Components and Technology Conference (ECTC)*, Aug. 2016.
- [3] “Intel’s EMIB packaging technology – a deep dive,” 2021. [Online]. Available: <https://semiwiki.com/semiconductor-manufacturers/intel/298674-intels-emib-packaging-technology-a-deep-dive/>
- [4] Z. Qian, J. Xie, and K. Aygün, “Electrical analysis of EMIB packages,” in *2018 IEEE 27th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*, Oct. 2018.
- [5] “Heterogeneous integration roadmap 2023 edition,” 2023. [Online]. Available: <https://eps.ieee.org/technology/heterogeneous-integration-roadmap/2023-edition.html>
- [6] T. Zhu, Y. Liang, Y. Cai, L. Lan, and X. Li, “A test method of sip module,” in *2014 15th International Conference on Electronic Packaging Technology*, Aug. 2014.
- [7] Z. O. Zaw, X. C. Wei, E.-X. Liu, E.-P. Li, and L.-W. Li, “Efficient analysis for multilayer power-ground planes with multiple vias and signal traces in an advanced electronic package,” in *2008 IEEE-EPEP Electrical Performance of Electronic Packaging*, Oct. 2008.
- [8] R. Mishra, Z. Hussain, and R. S. Hegde, “Power delivery network optimization in high speed bga package,” in *2018 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS)*, Dec. 2018.

- [9] R. W. Y. Chang, K. Y. See, and E. K. Chua, “Comprehensive analysis of the impact of via design on high-speed signal integrity,” in *2007 9th Electronics Packaging Technology Conference*, Dec. 2007.
- [10] N. K. H. Huang and J. Lai, “Comprehensive optimization of differential via design for high-speed applications,” in *2023 IEEE Symposium on Electromagnetic Compatibility Signal/Power Integrity (EMC+SIPI)*, July 2023.
- [11] T. Kim, S. Moon, C. Jo, S. Nam, and Y. Lee, “Psi design solutions for high speed die-to-die interface in chiplet applications,” in *2022 IEEE 72nd Electronic Components and Technology Conference (ECTC)*, May 2022.
- [12] R. Krishna, A. O. Watanabe, J. W. Golz, R. Bonam, F. R. Libsch, E. Rosenbaum, and A. Kumar, “Signal and power integrity design and analysis for bunch-of-wires (BoW) interface for chiplet integration on advanced packaging,” in *2023 IEEE 66th Electronic Components and Technology Conference (ECTC)*, May 2023.
- [13] S. Sitaraman, S. Verhaverbeke, S. Banna, M. Sowwan, L. Jiang, E. M. Bazizi, and B. Ayyagari-Sangamalli, “Optimization of 2.5d organic interposer channel for die and chiplets,” in *2022 IEEE 72nd Electronic Components and Technology Conference (ECTC)*, May 2022.
- [14] J.-M. Jin, *Theory and Computation of Electromagnetic Fields*. Wiley-IEEE Press, 2010.
- [15] L. Zhang, L. Jiang, J. Juang, Z. Yang, E.-P. Li, and C. Hwang, “Decoupling optimization for complex pdn structures using deep reinforcement learning,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 71, pp. 3773–3783, Mar. 2023.
- [16] H. Ma, E.-P. Li, Y. Wang, B. Shi, J. Schutt-Aine, A. Cangellaris, and X. Chen, “Channel inverse design using tandem neural network,” in *2022 IEEE 26th Workshop on Signal and Power Integrity (SPI)*, May 2022.
- [17] T. Nguyen, B. Shi, H. Ma, E.-P. Li, X. Chen, A. C. Cangellaris, and J. Schutt-Ainé, “Comparative study of surrogate modeling methods for signal integrity and microwave circuit applications,” *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 11, pp. 1369–1379, July 2021.
- [18] J.-Q. Lu, “3-d hyperintegration and packaging technologies for micro-nano systems,” *Proceedings of the IEEE*, vol. 97, pp. 18–30, Jan. 2009.

- [19] J. Kim, V. C. K. Chekuri, N. M. Rahman, M. A. Dolatsara, H. Torun, M. Swaminathan, S. Mukhopadhyay, and S. K. Lim, "Silicon vs. organic interposer: PPA and reliability tradeoffs in heterogeneous 2.5d chiplet integration," in *2020 IEEE 38th International Conference on Computer Design (ICCD)*, Oct. 2020.
- [20] "Packaging benchmark suite." [Online]. Available: <http://www.packaging-benchmarks.org/>
- [21] M.-S. Lin, T.-C. Huang, C.-C. Tsai, K.-H. Tam, C.-H. Hsieh, T. Chen, W.-H. Huang, J. Hu, Y.-C. Chen, S. K. Goel, C.-M. Fu, S. Rusu, C.-C. Li, S.-Y. Yang, M. Wong, S.-C. Yang, and F. Lee, "A 7nm 4GHz Arm®-core-based CoWoS® Chiplet Design for High Performance Computing," in *2019 Symposium on VLSI Circuits*, June 2019.
- [22] J. Oh, S. K. Lee, M. Kang, M. Ziegler, J. Silberman, A. Agrawal, S. Venkataramani, B. F. and Michael Guillorn, J. Choi, W. Wang, S. Mueller, S. Ben-Yehuda, J. Bonanno, N. Cao, R. C. and Chia Yu Chen, M. Cohen, O. Erez, T. Fox, G. Gristede, H. Haynie, V. Ivanov, S. Koswatta, S.-H. L. and Martin Lutz, G. Maier, A. Mesh, Y. Nustov, S. Rider, M. Schaal, M. Scheuermann, X. Sun, N. Wang, F. Yee, C. Zhou, V. Shah, B. Curran, V. Srinivasan, P.-F. Lu, S. Shukla, K. Gopalakrishnan, and L. Chang, "A 3.0 TFLOPS 0.62V Scalable Processor Core for High Compute Utilization AI Training and Inference," in *2020 Symposium on VLSI Circuits*, June 2020.
- [23] J. Hwang, J. Jeong, H. Choi, J. S. Pak, H. Lee, M. M. Kim, and I. Kim, "Analysis on optimal chip floorplanning considering various types of decoupling capacitors in package PDN," in *2022 IEEE 72nd Electronic Components and Technology Conference (ECTC)*, July 2022.
- [24] X.-D. Cai and G. Costache, "Time domain analysis of a through hole via with time dependent skin effect model," in *Proceedings of 1995 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference*, July 1995.
- [25] O. W. Bhatti, O. Akinwande, and M. Swaminathan, "Uncertainty quantification with invertible neural networks for signal integrity applications," in *2022 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, July 2022.
- [26] B. Shi, Y. Zhou, T. Nguyen, and J. E. Schutt-Aine, "Statistical method for eye diagram simulation in a high-speed link nonlinear system," in *2022 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, Dec. 2022.

- [27] Y. Zhou, B. Shi, Y. Zhao, and J. E. Schutt-Aine, “Fast eye diagram simulation based on latency insertion method,” in *2022 IEEE Electrical Design of Advanced Packaging and Systems (EDAPS)*, Dec. 2022.
- [28] S. Ardalan, R. Farjadrad, M. Kuemerle, K. Poulton, S. Subramaniam, and B. Vinnakota, “An open inter-chiplet communication link: Bunch of wires (BoW),” *IEEE Micro*, vol. 41, pp. 51–60, Jan. 2021.
- [29] F. de Paulis, R. Cecchetti, C. Olivieri, and M. Buecker, “Genetic algorithm pdn optimization based on minimum number of decoupling capacitors applied to arbitrary target impedance,” in *2020 IEEE International Symposium on Electromagnetic Compatibility Signal/Power Integrity (EMCSI)*, July 2020.