Low-Power Methodologies for High-Performance and Yield-Enhanced 3D Interconnects

Dissertation zur Erlangung des akademischen Grades Doktor-Ingenieur (Dr.-Ing.) im Fach Elektrotechnik und Informationstechnik

Lennart Bamberg

1. Gutachter:	Prof. Dr. Alberto García-Ortiz
2. Gutachter:	Prof. Dr. Thilo Pionteck
Eingereicht am:	04.02.2020
Tag des Promotionskolloquiums:	24.03.2020

Universität Bremen

To all my teachers

(those who are at times,
have been, and will be)

Abstract

The through-silicon vias (TSVs) that make up the vertical interconnects between the dies in modern 3D integrated circuits (ICs), today, prevent 3D integration from being a viable solution for a wide range of mass products, despite the enormous advantages the integration style promises. These advantages include a better form factor, the possibility of heterogeneous integration, and a reduction in the critical metal-wire lengths. One reason why TSVs are a roadblock for 3D integration is their low manufacturing yield. The other is the fact that TSVs entail dramatic parasitic capacitances. As a result of the large capacitances, TSVs are a severe threat to the power consumption and the performance (*i.e.*, maximum achievable frequency) of a 3D IC.

Relying just on advances in the manufacturing to provide TSVs with better power-performance metrics is not a promising strategy as the capacitances scale poorly due to increasing coupling effects with decreasing TSV geometries. Hence, TSV-related power and performance issues are best addressed on higher levels of abstractions. High-level techniques exploit the fact that both metrics strongly depend on the transmitted bit patterns (*i.e.*, data words) besides the parasitics. Nevertheless, this thesis shows that existing high-level techniques only result in somewhat limited improvements in the TSV performance (maximum 12 %), while even resulting in a drastically increased TSV power consumption. This poor gain of existing methods is because they are derived based on physically imprecise models for the TSV metrics.

To overcome this limitation, a set of physically precise high-level models for the TSV power consumption and performance are presented in this thesis. Based on these models, a low-power technique is contributed, which can reduce the TSV power consumption in modern 3D ICs by over 40 %. Despite its drastic power savings, the method results in negligible implementation costs.

Additionally, two optimization techniques are presented, which improve the TSV power consumption and performance simultaneously. The methods improve the TSV performance by up to 65 % while providing power savings of 17 %, and this at lower costs compared to the best previous technique. This fact underlines the substantial superiority of the proposed methods compared to all previous ones.

Moreover, a low-power technique is presented, which also improves the manufacturing yield of TSVs. The method reduces the TSV-related defect rate of 3D ICs by a factor of 17 ×, while additionally providing an improvement in the interconnect power consumption of modern 3D ICs by over 30 %.

Besides the TSVs, the metal-wire parasitics are a critical concern for 3D integration—even if less when compared to 2D integration. Thus, the proposed methods furthermore allow for an improvement in the metal-wire power consumption and performance to the same extent as the provided improvements for TSVs. This further boosts the efficiency of the proposed techniques.

Kurzfassung

Die Integration von Halbleiter-Schaltkreisen in die dritte Dimension hat enormes Potential, digitale Systeme substanziell zu verbessern. Im Gegensatz zu traditionellen integrierten Schaltkreisen (ICs) benötigen 3D-ICs vertikale Verbindungen zwischen gestapelten Silizium-Substraten. Hierfür werden vorwiegend TSVs verwendet. Jedoch sind TSVs der Hauptgrund dafür, dass 3D-Integration heute nur in wenigen kommerziellen Produkten zum Einsatz kommt. Ein Grund ist die geringe Fertigungsausbeute von TSVs; ein anderer die Tatsache, dass TSVs massive parasitäre Kapazitäten aufweisen. Infolge dieser großen Kapazitäten können TSVs den Energieverbrauch eines 3D-ICs erheblich erhöhen. Auch limitieren die Kapazitäten die Performance von 3D-ICs.

Es ist keine vielversprechende Strategie sich lediglich darauf zu verlassen, dass neue Prozessverfahren die Probleme von TSVs beheben. Dies begründet sich darin, dass selbst eine drastische Reduktion der TSV-Dimensionen die parasitären Kapazitäten nur unwesentlich verkleinert. Daher müssen die TSV-Probleme auf höheren Abstraktionsebenen angegangen werden. Optimierungsverfahren, die auf höheren Abstraktionsebenen hergleitet werden, nutzen aus, dass die Performance, der Energieverbrauch und die Fehlertoleranz von TSVs nicht nur von den parasitären Kapazitäten abhängig sind, sondern auch von den übertragenen Bitmustern. Diese Arbeit zeigt, dass bestehende Optimierungsverfahren nur zu einer Erhöhung der TSV-Performance von maximal 12 % führen und den Energieverbrauch sogar erheblich erhöhen. Dies lässt sich darauf zurückführen, dass die Verfahren auf physikalisch ungenauen Modellen für die TSV-Metriken beruhen.

Um dieses Problem zu überwinden, werden in dieser Arbeit eine Reihe physikalisch präziser, aber dennoch abstrakter Modelle für den Energieverbrauch und die Performance von TSVs vorgestellt. Basierend auf diesen Modellen wird anschließend ein Optimierungsverfahren präsentiert, welches den TSV-Energieverbrauch um bis zu 45 % senkt. Trotz dieser drastischen Energieeinsparungen hat die Methode vernachlässigbare Implementierungskosten.

Außerdem werden zwei Optimierungsmethoden vorgestellt, die den TSV-Energieverbrauch sowie gleichzeitig die Performance verbessern. Experimentelle Auswertungen der Verfahren zeigen, dass diese die Performance von TSVs um bis zu 65 % optimieren können und dennoch eine Reduktion im Energieverbrauch von 17 % ermöglichen—und all das zu geringeren Kosten als vorherige Methoden. Darüber hinaus wird eine Technik vorgestellt, die neben dem Energieverbrauch auch den Herstellungsertrag von TSVs verbessert. Diese Optimierungsmethode reduziert die TSV-bezogene Defektrate von modernen 3D-ICs um den Faktor 17 und senkt gleichzeitig den Energieverbrauch um über 30 %.

Zudem sind die präsentierten Verfahren so entwickelt, dass sie zusätzlich eine Optimierung für die horizontalen Metallleitungen in 3D-ICs bewirken. Damit wird auch dem oft signifikanten Einfluss langer Metallverdrahtungen auf den Energieverbrauch und die Performance des Gesamtsystems Rechnung getragen.

Publications as Lead Author

Original articles (A) in peer-reviewed international journals

[A1] Lennart Bamberg and Alberto García-Ortiz. High-level energy estimation for submicrometric TSV arrays. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 25(10):2856–2866, 2017.

[A2] Lennart Bamberg, Amir Najafi, and Alberto García-Ortiz. Edge effects on the TSV array capacitances and their performance influence. *Elsevier Integration*, 61:1–10, 2018.

[A3] Lennart Bamberg, Amir Najafi, and Alberto García-Ortiz. Edge effect aware low-power crosstalk avoidance technique for 3D integration. *Elsevier Integration*, 69:98–110, 2018.

[A4] Lennart Bamberg, Jan Moritz Joseph, Thilo Pionteck, and Alberto Garcia-Ortiz. Crosstalk optimization for through-silicon vias by exploiting temporal signal misalignment. *Elsevier Integration*, 67:60–72, 2019.

[A5] Lennart Bamberg and Alberto García-Ortiz. Coding-based low-power through-silicon-via redundancy schemes for heterogeneous 3-D SoCs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 27(10):2317–2330, 2019.

[A6] Lennart Bamberg co-shared with Jan Moritz Joseph, Imad Hajjar, Robert Schmidt, Thilo Pionteck, and Alberto García-Ortiz. Simulation environment for link energy estimation in networks-on-chip with virtual channels. *Elsevier Integration*, 2019. (In press; available online).

Original papers (P) in proceedings of peer-reviewed international conferences

[P1] Lennart Bamberg, Ami Najafi, and Alberto García-Ortiz. Edge effect aware crosstalk avoidance technique for 3D integration. In *International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 1–8. IEEE, 2017. **(Awarded Best Paper)**.

[P2] Lennart Bamberg, Robert Schmidt, and Alberto García-Ortiz. Coding approach for low-power 3D interconnects. In *Design Automation Conference (DAC)*, pages 1–6. ACM/IEEE, 2018.

[P3] Lennart Bamberg and Alberto García-Ortiz. Exploiting temporal misalignment to optimize the interconnect performance for 3D integration. In *International Symposium on Power and Timing Modeling,*

Optimization and Simulation (PATMOS), pages 214–221. IEEE, 2018. **(Awarded Best Paper)**.

[P4] Lennart Bamberg, Jan Moritz Joseph, Robert Schmidt, Thilo Pionteck, and Alberto García-Ortiz. Coding-aware link energy estimation for 2D and 3D networks-on-chip with virtual channels. In *International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 222–228. IEEE, 2018.

[P5] Lennart Bamberg, Lingjun Zhu, Sai Pentapati, Da Eun Shim, Alberto Garcia-Ortiz, and Sung Kyu Lum. Macro-3D: A physical design methodology for face-to-face-stacked heterogeneous 3D ICs. In *Design, Automation & Test in Europe Conference (DATE)*, pages 1–6. IEEE, 2020. (Accepted for publication).

Publications as Coauthor

Original coauthored articles (CA) in peer-reviewed international journals

[CA1] Alberto García-Ortiz, Lennart Bamberg, and Amir Najafi. Low-power coding: trends and new challenges. *ASP Journal of Low Power Electronics*, 13(3):356–370, 2017.

[CA2] Jan Moritz Joseph, Lennart Bamberg, Dominik Ermel, Behnam Razi Perjikolaei, Anna Drewes, Alberto García-Ortiz, and Thilo Pionteck. NoCs in heterogeneous 3D SoCs: Co-design of routing strategies and microarchitectures. *IEEE Access*, 7:135145–135163, 2019.

[CA3] Sai Surya Kiran Pentapati, Lingjun Zhu, Lennart Bamberg, Da Eun Shim, Alberto García-Ortiz, and Sung Kyu Lim. A logic-on-memory processor-system design with monolithic 3D technology. *IEEE Micro*, 39(6):38–45, 2019.

[CA4] Amir Najafi, Lennart Bamberg, Ardalan Najafi, and Alberto García-Ortiz. Integer-value encoding for approximate on-chip communication. *IEEE Access*, 7:179220–179234, 2019.

[CA5] Amir Najafi, Lennart Bamberg, and Alberto García-Ortiz. Misalignment-aware energy modeling of narrow buses for data encoding schemes. *Elsevier Integration*, 72:58–65, 2020.

[CA6] Jan Moritz Joseph, Dominik Ermel, Lennart Bamberg, Alberto García-Oritz, and Thilo Pionteck. Application-specific soc design using core mapping to 3d mesh nocs with nonlinear area optimization and simulated annealing. *Technologies*, 8(1):1–10, 2020.

[CA7] Jan Moritz Joseph, Lennart Bamberg, Imad Hajjar, Behnam Razi Perjikolaei, Alberto García-Ortiz, and Thilo Pionteck. Ratatoskr: An open-source framework for in-depth power, performance and area analysis in 3D NoCs. *ACM Transactions on Modeling and Computer Simulation*, pages 1–22, 2019. (Under Review).

Original coauthored papers (CP) in proceedings of peer-reviewed international conferences

[CP1] Amir Najafi, Lennart Bamberg, Ardalan Najafi, and Alberto García-Ortiz. Energy modeling of coupled interconnects including intrinsic misalignment effects. In *International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 262–267. IEEE, 2016.

[CP2] Jan Moritz Joseph, Lennart Bamberg, Sven Wrieden, Dominik Ermel, Alberto Garcia-Oritz, and Thilo Pionteck. Design method for asymmetric 3D interconnect architectures with high level models. In *International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pages 1–8. IEEE, 2017.

[CP3] Jan Moritz Joseph, Lennart Bamberg, Sven Wrieden, Dominik Ermel, Alberto García-Ortiz, and Thilo Pionteck. Design method for asymmetric 3D interconnect architectures with high level models. In *International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pages 1–8. IEEE, 2017.

[CP4] Amir Najafi, Lennart Bamberg, Ardalan Najafi, and Alberto García-Ortiz. Misalignment-aware delay modeling of narrow on-chip interconnects considering variability. In *International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pages 1–4. IEEE, 2018.

[CP5] Jan Moritz Joseph, Lennart Bamberg, Gerald Krell, Imad Hajjar, Alberto García-Ortiz, and Thilo Pionteck. Specification of simulation models for NoCs in heterogeneous 3D SoCs. In *International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pages 1–8. IEEE, 2018.

[CP6] Jan Moritz Joseph, Dominik Ermel, Lennart Bamberg, Alberto García-Ortiz, and Thilo Pionteck. System-level optimization of network-on-chips for heterogeneous 3D system-on-chips. In *International Conference on Computer Design (ICCD)*, pages 1–4. IEEE, 2019.

[CP7] Jan Moritz Joseph, Dominik Ermel, Tobias Drewes, Lennart Bamberg, Alberto García-Oritz, and Thilo Pionteck. Area optimization with non-linear models in core mapping for system-on-chips. In *International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pages 1–4. IEEE, 2019.

[CP8] Amir Najafi, Lennart Bamberg, Guillermo Payá Vayá, and Alberto Garcia-Ortiz. A coding approach to improve the energy efficiency of approximate NoCs. In *International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pages 1–8, 2019.

[CP9] Lingjun Zhu, Lennart Bamberg, Sai Pentapati, Kyungwook Chang, Francky Catthoor, Dragomir Milojevic, Manu Komalan, Brian Cline, Saurabh Sinha, Xiaoqing Xu, Alberto García-Ortiz, and Sung Kyu Lim. High-performance logic-on-memory monolithic 3D Arm Cortex-A processor designs. In *International Symposium on Low Power Electronics and Design (ISLPED)*, pages 1–6. ACM/IEEE, 2020. (Under Review).

Acknowledgments

The present thesis would not exist as it is without the support of many people. However, to mention everybody individually, would exceed the reasonable limit for the length of an "Acknowledgment" section, by far. Thus, I would like to use these introductory sentences to apologize to everybody who is not explicitly mentioned: Your support is not forgotten!

My supervisor throughout this doctoral research project, Prof. Alberto García-Ortiz, shaped the content of this thesis fundamentally through his guidance. Prof. García always provided all the support that was needed for me to grow and become an independent scientist. His door is always open for his students—a fact that helped me overcome many obstacles on the sometimes rocky path toward a successful doctoral research project. Moreover, his passion for research founded my motivation to dive deep into research and pursue a Ph.D. degree back when I was an undergraduate student, taking my first courses with Prof. García.

I thank Prof. Thilo Pionteck and Jan Moritz Joseph, from the University of Magdeburg, with whom I have collaborated through a research project funded by the *German Research Foundation* (DFG). Prof. Pionteck kindly agreed to serve as a second examiner for my thesis. This service and his general guidance have been particularly useful. I enjoyed the active cooperation with Jan Moritz Joseph, which resulted in many exciting research outcomes. Hopefully, we will have the chance to continue this cooperation in the future.

Furthermore, I want to thank Prof. Sung Kyu Lim from the Georgia Institute of Technology (Atlanta, United States of America), even though huge parts of the research I conducted with his guidance are already beyond the scope of this thesis. Prof. Lim kindly agreed to take over my supervision during my stay as an invited research scholar at his institution. Throughout my whole stay in America, and ever since, he guided me like one of his peers, not only in my research but also on a personal level.

Lingjun Zhu, who was a colleague of mine at the Georgia Institute of Technology, must be mentioned here explicitly. He did not only help me with my research in Atlanta, but he also became an everlasting friend who made my stay in Atlanta truly enjoyable. I am looking forward to our joint work in the future. As I had Lingjun and others at the Georgia Institute of Technology, I had great colleagues at the University of Bremen. I am happy to call each of them my friends today. Ardalan Najafi, Amir Najafi, Robert Schmidt, Jakob Döring, Andreas Beering, Wanli Yu, Yanqiu Huang, Behnam Razi, and Kerstin Janssen, thanks for all the laughs, the support, and the distraction that is sometimes needed to get new motivation. Ardalan, evenings with you have proven to me that sometimes having a drink with a colleague can help to progress in your research more than working a long night shift.

Our office manager at the University of Bremen, Kerstin Janssen, deserves individual credit. Kerstin, the commitment to your job and the support you give everybody, at any time, even outside your office hours—irrespective of whether he is an undergraduate student or a Professor—is truly inspiring. You have contributed so much to the work of everybody in our team.

I would like to thank Prof. Francky Catthoor from IMEC (Leuven, Belgium) and Prof. Steffen Paul from the University of Bremen for serving as the third and fourth examiner for this dissertation. Research grants and scholarships provided by the DFG, the *German Academic Exchange Service* (DAAD), the *German Federal Ministry of Education and Research* (BMBF), the University of Bremen, and the Georgia Institute of Technology, supported this research project—especially my stay in Atlanta.

Moreover, I would like to thank all my friends and family who were at my side all the time. Your support was vital for this thesis. Thanks for understanding when I could not join a dinner, a holiday, or any other joint activity because I wanted to keep working on my research. Again there is one person I want to mention explicitly: Jonathan Weering, the support you provided, especially in the difficult time before moving to Atlanta, is far beyond what one expects from a friend. Thank you!

Lastly, I would like to thank my parents and Ankie Sophie Jakob, who deserve all the credit. I cannot at all imagine having pursued a Ph.D. or even a Bachelor's degree without the strong support my parents provided, irrespective of the situation. Ankie Jakob was always at my side during my time as a Ph.D. student. Not only did she (almost always ☺) understand when I did not return home from work before night, or when I went to the office on Sundays, she even supported me as much as she could in every possible manner. Furthermore, having her at my side filled most days with joy—even complicated ones.

Table of Contents

Specific Mathematical Symbols

$\neg$	bit-wise Boolean negation.
$\circ$	Hadamard-product operator (for matrices).
$\langle\rangle$	Frobenius inner product of two matrices.
$\mathbb{B}^n$	vector of n binary numbers.
$\mathbb{B}^{n\times m}$	matrix with n rows and m columns of binary numbers.
$\mathbb{B}$	set of binary numbers: $\{0, 1\}$.
$\mathbb{E}\{\}$	expectation operator.
$\mathbb{F}_2$	binary Galois field 2.
$\mathbb{N}$	set of natural numbers: $\{0, 1, 2, 3, \dots\}$.
$\mathcal{L}\{\}$	Laplace transform.
$\oplus$	Boolean XOR operator.
$\overset{!}{=}$	demanded equality.
$\overset{\text{def}}{=}$	defined as being equal to.
$\sim$	proportionality.
$\max\operatorname{diag}()$	maximum diagonal entry of a matrix.
$\max_{k,\,i}()$	maximum entry of a discrete-time vector over all cycles.
spark()	spark of a matrix.
max()	maximum matrix of two matrices (entry-by-entry).
$\log_2()$	base-2 logarithm.
mod()	modulo operation.

Specific Units

GE	gate equivalent.
pp	percentage point.

Specific Symbols Used in Multiple Chapters

Δb_i	clock-cycle-based self switching of b_i.
$\Delta C_{i,j}$	change in $C_{i,j}$ with increasing p_i and p_j.
$\mathbf{\Delta C}$	matrix of the $\Delta C_{i,j}$ values.
α_i	switching activity of b_i.
$\delta_{i,j}$	clock-cycle-based crosstalk factor for b_i and b_j.
$\gamma_{i,j}$	switching correlation of b_i and b_j.
$\mathbb{S}_{\mathbf{I}_\pi,n}$	set of all valid $n \times n$ permutation matrices.
ρ	correlation of subsequent data words.

σ	standard deviation of the data words.
b_i	clock-cycle-based bit signal.
C_{c0}	ground capacitance of a corner TSV.
C_{c1}	coupling capacitance of a corner TSV and a directly adjacent TSV.
C_{c2}	coupling capacitance of a corner TSV and an indirectly adjacent TSV.
C_{d}	coupling capacitance of diagonally adjacent TSVs.
C_{e0}	ground capacitance of an edge TSV.
C_{e1}	coupling capacitance of two directly adjacent edge TSVs.
C_{e2}	coupling capacitance of two indirectly adjacent edge TSV.
$C_{\mathrm{eff},i}$	clock-cycle-based effective capacitance of interconnect i.
$\hat{C}_{\mathrm{eff}}$	maximum possible effective capacitance for all interconnects.
$\bar{C}_{\mathrm{eff},i}$	mean effective capacitance of interconnect i.
$\vec{C}_{\mathrm{eff}}$	clock-cycle-based vector of the $C_{\mathrm{eff},i}$ values.
$C_{\mathrm{G},i,j}$	$C_{i,j}$ value for p_i and p_j equal to 0.
$\mathbf{C}_{\mathrm{G}}$	matrix of the $C_{\mathrm{G},i,j}$ values.
$C_{i,j}$	coupling capacitance between interconnect i and j for $i \neq j$; ground capacitance of interconnect i for $i = j$.
$\mathbf{C}$	matrix of the $C_{i,j}$ values.
$C_{\mathrm{mw,c}}$	coupling capacitance of adjacent metal wires.
$C_{\mathrm{mw,g}}$	ground capacitance of a metal wire.
C_{n}	coupling capacitance of directly adjacent middle TSVs.
$C_{\mathrm{R},i,j}$	$C_{i,j}$ value for p_i and p_j equal to 0.5.
$\mathbf{C}_{\mathrm{R}}$	matrix of the $C_{\mathrm{R},i,j}$ values.
d_{min}	minimum TSV pitch.
f	clock frequency (*i.e.*, $1/T_{\mathrm{clk}}$).
f_{s}	significant frequency of the TSV signals.
$\mathbf{I}_{\pi}$	permutation matrix.
$\mathbf{I}_{\pi,\mathrm{perf\text{-}opt}}$	performance-optimal net-to-TSV assignment expressed as a permutation matrix.
l_{tsv}	TSV length.
$M \times N$	TSV-array shape.
P	interconnect power consumption.
p_i	probability of b_i being 1.
R_{D}	driver equivalent resistance.

r_{tsv}	TSV radius.
$\mathbf{S}$	clock-cycle-based switching matrix of the bit values..
$\mathbf{S}_{\mathbb{E}}$	mean switching matrix of the bit values..
$\mathbf{S}_{\text{wc}}$	worst-case switching matrix of the bit values..
T_{clk}	cycle duration of the clock.
$T_{\text{D},0}$	driver-induced offset in the signal propagation delay.
$T_{\text{edge},i}$	delay on the signal edges on b_i relative to the rising clock edges.
t_{ox}	TSV-oxide thickness.
T_{pd}	interconnect signal-propagation delay.
$\hat{T}_{\text{pd}}$	maximum interconnect signal-propagation delay.
V_{dd}	power-supply voltage.
$w_{\text{dep},i}$	depletion-region width of TSV_i.
W/L_{min}	transistor sizing.

Acronyms

2.5D	2.5-dimensional.
2D	two-dimensional.
3D	three-dimensional.
ADC	analog-to-digital converter.
AND	logical conjunction.
ASIC	application-specific integrated circuit.
BEOL	back-end of line.
BP	bus partitioning.
BPSK	binary phase-shift keying.
CAC	crosstalk-avoidance code.
CBI	classical bus invert.
CMOS	complementary metal-oxide-semiconductor.
CODEC	coder-decoder circuit.
CPU	central processing unit.
DAC	digital-to-analog converter.
DRAM	dynamic random-access memory.
DSP	digital signal processor.
EDA	electronic design automation.
EM	electromagnetic.
FEOL	front-end of line.
FinFET	fin field-effect transistor.
FPF	forbidden-pattern free.
FPGA	field-programmable gate array.
FTF	forbidden-transition free.
IC	integrated circuit.
ILD	inter-layer dielectric.
ITRS	*International Technology Roadmap for Semiconductors*.
KOZ	keep-out zone.
LPC	low-power code.
LSB	least significant bit.
LUT	lookup table.
MAE	maximum absolute error.
MIV	monolithic inter-tier via.

MOS	metal-oxide-semiconductor.
MOSFET	MOS field-effect transistor.
MSB	most significant bit.
NAND	logical non-conjunction.
NMAE	normalized maximum absolute error.
NoC	network on chip.
NOR	logical non-disjunction.
NRMSE	normalized root-mean-square error.
NVM	non-volatile memory.
P/G	power or ground.
PTM	*Predictive Technology Model*.
QAM16	16-point quadrature amplitude modulation.
QAM64	64-point quadrature amplitude modulation.
RAM	random-access memory.
RC	resistance-capacitance.
RF	radio frequency.
RGB	red-green-blue.
RLC	resistance-inductance-capacitance.
RMS	root mean square.
RMSE	root-mean-square error.
RR	repair register.
RS	repair signature.
RTL	register-transfer level.
Rx	receiver.
SA	stuck-at.
SA0	stuck-at-0.
SA1	stuck-at-1.
SoC	system on chip.
SOI	silicon on insulator.
SRAM	static random-access memory.
TSV	through-silicon via.
Tx	transmitter.
VLSI	very-large-scale integration.
VSoC	vision system on chip.
XNOR	logical exclusive non-disjunction.
XOR	logical exclusive disjunction.

Part I: Introduction & Background

CHAPTER 1

Introduction and Overview

Interconnects are the bottleneck of current, as well as future, very-large-scale integration (VLSI) circuits. As technology advances, the transistors in a chip become faster and more power-efficient. In contrast, the physical connections between the components scale rather poorly [1–3]. Thus, although once negligible, today, interconnects can be responsible for over 50 % of the overall power consumption of modern digital systems [4]. Moreover, the long global interconnects mainly determine the maximum possible operation frequency of today's digital systems [1, 3].

A promising solution to this problem is the stacking of multiple silicon substrates, resulting in three-dimensional (3D) integrated circuits (ICs). The 3D organization reduces the average as well as the maximum length of an interconnect. Thus, 3D integration offers vast potential to improve the performance and the power consumption. Furthermore, integrating the components into the third dimension brings additional advantages. The 3D organization shrinks the system's footprint, which enables scaling beyond the foreseen end of Moore's law [5]. Another key feature of 3D integration is that it allows for heterogeneous integration. This feature of 3D integration is essential since the

silicon technology of each substrate of the stack can be fully optimized only for its integrated components, which boost performance, power efficiency, and robustness [6].

However, along with the many advantages, there are still several critical challenges that need to be addressed before 3D integration becomes feasible for a wide range of commercial mass products. Firstly, the 3D organization requires a new set of tools enabling electronic design automation (EDA) for the layout generation and verification, due to the added (third) dimension for standard-cell placement and routing [7]. Furthermore, the dissipation of the heat that is building up within the stack is a design concern for 3D ICs [8]. However, the key challenge is that, in contrast to traditional two-dimensional (2D) ICs, 3D ICs require the fabrication of vertical electrical connections through the silicon substrates in order to communicate in-between the individual tiers of the stack.

Typically, through-silicon vias (TSVs) are used to establish these vertical inter-tier connections, as they result in a high system reliability at relatively low manufacturing costs since they enable the stacking of individual, prefabricated "2D" dies. A TSV is formed by etching a cylindrical hole into a substrate, which is filled with copper or another conducting material. Except for the TSV manufacturing, the individual dies of a TSV-based 3D IC can be manufactured using the process steps known from traditional 2D-IC manufacturing. Finally, the pre-fabricated dies are stacked using a mechanical die-on-die bonding technique.[1]

The main disadvantage of TSVs is their relatively large size: Partially due to the immaturity of the involved process steps; and partially since they have to traverse the full substrate in the vertical direction, which is generally not thinned below 50 μm, in order to guarantee enough mechanical stability (*e.g.*, for bonding) and proper noise isolation between the individual dies.

Hence, the size of a TSV can be several hundred times larger than that of a transistor whose feature size can be below 10 nm in modern wafers. This significant area overhead, paired with a relatively low TSV manufacturing yield due to the immaturity of the involved process steps, strongly demands the use of vertical TSV interconnects only in bundles and on a global level (*i.e.*, to connect larger circuit blocks). Thus, such as global metal wires are typically routed in buses[2], TSVs are bundled together in regular rectangular arrays rather than used in isolation. The main advantage of clustering global interconnects is that the commonality of the individual signal paths of a bundle is maximized, without lengthening the individual interconnects unnecessarily (which would worsen the already critical interconnect power consumption and the system's

[1] Stacking refers here to a bonding process which also realizes low-resistive electrical connections between the device levels of the dies.

[2] A bus is defined as a set of long parallel global interconnects.

routability). This is of particular importance for modern 3D systems where the individual blocks contain wide input/output signals to ensure a high data throughput (*e.g.*, for 3D vision SoCs or 3D DRAMs [9]). Furthermore, to bundle TSVs in rectangular arrays facilitates their manufacturing and thus affects the critical yield beneficially.

However, particularly when clustered together, TSVs entail large parasitic capacitances, which is an issue that becomes even worse with ongoing device scaling (*i.e.*, TSV parasitics scale worse than the ones of active-circuit elements). The TSV parasitics can heavily impair the power consumption of 3D ICs. Although not as drastic, the signal propagation delay is also affected negatively by the large TSV parasitic capacitances (in contrast to the power consumption, the propagation delay also depends on the interconnect resistance, which is extremely low for TSVs due to their large diameter).

Previous work has shown that the TSVs are the limiter for the performance improvement achieved by 3D integration and often result in an even increased power consumption compared to a 2D counterpart implementation [10]. Hence, the TSV power consumption, performance (*i.e.*, maximum propagation delay), and yield are major concerns for modern 3D ICs and consequently caught the attention of a wide range of academic as well as industrial experts (*e.g.*, [9–38]).

Many previous works that address the impact of TSVs on the design quality of 3D ICs deal with quantifying the sizes of the TSV parasitic elements and analyzing their impact on common design metrics such as the power consumption or the performance [10–20]. Some works moreover propose manufacturing techniques to suppress TSV noise due to their large coupling capacitances, which also improves the performance through an optimized TSV signal-propagation delay [16–20]. On the downside, these manufacturing techniques significantly increase the production costs and further impair the already critical TSV manufacturing yield—possibly pushing TSV-based 3D integration even further away in time from being a viable technique for a wide range of commercial products.

Consequently, research has been recently conducted on crosstalk-avoidance codes (CACs) for TSV arrays, which are derived on higher abstraction levels (*i.e.*, bit-level and above) [21–23]. These CAC techniques improve the TSV performance without relying on advanced manufacturing techniques by exploiting the pattern-dependent nature of the propagation delay over VLSI interconnect structures outlined in [39]. In detail, the techniques aim to improve the TSV performance by strictly avoiding transitions in the transmitted patterns that result in a signal-propagation delay that exceeds a certain threshold value. On the downside, existing 3D CACs result in a drastic increase in the critical TSV power consumption by up to 50 % due to their high overhead costs.

Issues caused by the low manufacturing yield of TSVs have also been addressed on higher abstraction levels through TSV testing methods used in combination with TSV redundancy schemes [9, 25–38]. For TSV yield enhancement, the

required TSVs are logically grouped into sets of equal size, and at least one redundant TSV is added to each set.[3] The testing method is used to check the TSVs for defects after manufacturing. Subsequently, the test results are interpreted to identify the faulty TSVs. If the maximum number of broken TSVs in a set does not exceed the number of redundant TSVs per set, the redundancy scheme is used to repair the links in order to enhance the overall manufacturing yield. Albeit not as dramatically as existing 3D CACs, existing redundancy schemes also increase the critical TSV-related power consumption due to the added redundancy and the induced active-circuit elements.

Hence, while techniques to improve the TSV performance and yield have been recently proposed, the power-consumption issues are still completely unaddressed, despite the tremendous importance of achieving a low TSV power consumption. However, we clearly cannot just rely on advances in the TSV manufacturing to provide the required low TSV power consumption. Not only because high-level approaches promise higher power savings [40], but also to boost the process of making 3D integration suitable for a wide range of products, as new manufacturing techniques typically further impair the already critical TSV manufacturing yield.

1.1. Objective

The main objective of this doctoral research project is to achieve a low 3D-interconnect power consumption, while providing a high general 3D-interconnect quality, by designing efficient optimization techniques that are based on higher abstraction levels.

In detail, the objective is threefold. The first aspect is to systematically derive abstract, universally valid, and yet precise high-level models to estimate the TSV power consumption and performance. Such models are the key enabler for the demanded optimization techniques for low-power 3D interconnects.

The second goal of this thesis is to present optimization techniques which simultaneously improve the TSV and the metal-wire quality, rather than techniques that are exclusively efficient for TSV structures. Through-silicon vias are used to establish the vertical (inter-die) interconnects in 3D ICs, but metal wires are still needed to establish the horizontal (intra-die) interconnects. Hence, 3D interconnects spanning over multiple dies are typically made up of several metal-wire and TSV segments. Thus, by making the proposed low-power techniques effective for TSVs as well as metal wires, the methods can improve the quality of each element of an arbitrary 3D-interconnect structure. This means a major advantage compared to low-power techniques that only optimize the TSV quality, as even the parasitics of shorter metal wires dominate over the ones of active circuit elements in aggressively scaled technology nodes [41]. Fur-

[3] This, again, demands for regular TSV arrangements such as arrays.

thermore, such hybrid low-power techniques are even efficient for interconnects in a 3D IC that do not span over multiple dies and consequently do not contain TSVs. Hence, the proposed techniques are efficient for arbitrary interconnects in a 3D IC, and thus can be integrated without an in-depth knowledge of the exact physical layout.

The third objective of this doctoral research project is to derive high-level techniques which reduce the 3D-interconnect power consumption and yet improve their performance (*i.e.*, the maximum propagation delay of TSVs and metal wires) and the overall TSV manufacturing yield, as previous research has shown that these two metrics are also of great importance for 3D integration. Such optimization techniques will boost the process of making 3D stacking beneficial for a broad set of commercial systems as they overcome power issues of 3D ICs, while moreover providing a high system performance, as well as an improved overall manufacturing yield.

1.2. Working Hypothesis

The working hypothesis of this thesis is that the demanded high-level optimization techniques can only significantly improve the TSV quality if the techniques are derived based on physically precise, universally valid, and yet abstract models for the pattern-dependent TSV power consumption and performance.

This hypothesis is based on the following observation. Existing high-level CAC techniques are based on an abstract model for the TSV-array capacitances since the propagation delay is determined by the signal switching (crosstalk) over the coupling capacitances [21–23]. The used capacitance model is a slightly extended version of the well-established model for metal wires [39]. Hence, specific physical phenomena that newly arise for TSV arrays are not captured by the previously used TSV capacitance model. An in-depth analysis of the existing CAC techniques for modern TSV structures—conducted in the scope of the present thesis—reveals that all existing techniques actually result in a TSV-performance improvement that is at least 50 % lower than previously reported. This is due to the disregarded TSV-bundle-specific physical phenomena.

This mitigation in the performance gains, paired with an induced drastic increase in the power consumption, clearly make existing 3D-CAC techniques impractical for most real applications. A physically precise and yet abstract model for the TSV capacitances is the only way to overcome this substantial limitation of existing high-level techniques.

1.3. Research Approach

Consistent with the working hypothesis, the present thesis follows a bottom-up approach. This means that the first chapters start at the lowest level of abstraction (*i.e.*, transistor and circuit level) to systematically derive models

to estimate the pattern-dependent power consumption and performance of 3D interconnects on higher abstraction levels (*i.e.*, bit-level and above), which are then used to derive the low-power techniques. Such an elaborated approach is imperative to capture TSV-array-specific physical phenomena in the high-level models—required to overcome the substantial limitations of existing approaches due to the over-simplistic underlying high-level models.

Hence, starting with the standard formulas for the voltage-current relationships of parasitic elements, mathematical formulas for a precise calculation of the power consumption and performance of general VLSI interconnect structures are derived. The derived formulas are subsequently transferred to formulas that depend on the switching of the logical bit values on the interconnects rather than the actual voltage waveforms in order to increase the abstraction level. A scaleable and universally valid high-level model to estimate the technology-dependent parameters of the formulas (*i.e.*, the capacitances) is derived in the sequel. This model captures the arising physical phenomena in TSV arrays in an abstract, scaleable, and yet universally valid way. To allow for an even further increase in the abstraction level, models to estimate the bit-level statistics (required for the estimation of the power consumption) by means of abstract word-level and data-flow characteristics are discussed in the following.

Afterward, the new insights provided by the derived high-level models, mainly due to the captured new physical phenomena, are systematically exploited in order to derive efficient optimization techniques. To obtain optimization techniques that improve the TSV as well as the metal-wire quality, the proposed optimization techniques are designed based on optimization techniques that have proven to be efficient for traditional 2D ICs. Sophisticated data-encoding techniques that effectively improve the metal-wire power consumption and performance in 2D ICs are well known (see, [CA1, 39]). Hence, these techniques are partially reused for the optimization techniques presented throughout this dissertation. However, the techniques are modified by means of the insights provided by the proposed high-level models such that they also effectively optimize the TSV power consumption, performance, and yield. Thereby great care is taken to ensure that the modifications are designed in a way that the implementation costs of the techniques are not increased significantly, while the gains for metal wires are retained. This approach results in effective low-power methodologies for arbitrary 3D-interconnect structures that also tackle interconnect-related performance and manufacturing-yield issues.

1.4. Outcomes

This doctoral research project has two main outcomes. First, precise high-level models for the power and performance estimation of 3D interconnects; and second, a set of highly efficient optimization techniques.

In detail, this doctoral research project provides the following concrete contributions to the state-of-the-art, which partly have been presented to the scientific community through a wide range of internationally renowned and peer-reviewed journals and conference proceedings:

1. A set of high-level models in which a deeper understanding of the physical phenomena that shape the TSV power consumption and performance is encapsulated. The proposed models allow for a precise estimation of the pattern-dependent power consumption and performance of 3D interconnects on higher abstraction levels. Such models are not only imperative for the derivation of efficient 3D-interconnect optimization techniques but also for a fast and yet precise design-space exploration of competing architectures for 3D integration. The contributed high-level models have been partly presented to the scientific community through [A1, A2, A6, P4].

2. A high-level optimization technique which can drastically improve the 3D-interconnect power consumption at negligible costs. The technique systematically exploits the new insights provided by the proposed high-level models and the intrinsic bit-level characteristic of typical data streams found in most 3D ICs. Despite its negligible implementation costs, the proposed technique allows for a drastic power-consumption improvement. Furthermore, the technique maximizes the efficiency of traditional low-power codes for arbitrary 3D-interconnect structures. This optimization technique has been presented to the scientific community through [P2].

3. Two high-level optimization techniques that improve the 3D-interconnect performance, while simultaneously providing an improvement in their power consumption. Both optimization techniques are again derived based on the contributed high-level models. The first technique is a coding technique that follows the same fundamental idea as existing CAC techniques. However, by being derived based on the contributed physically precise high-level models, the TSV performance improvement is increased by more than a factor of $5\times$ compared to the best previous technique. Moreover, while previous CAC techniques only allow for an improved TSV performance (at an increased power consumption), the proposed technique simultaneously improves the power consumption as well as the performance of both structures, TSVs and metal wires. Thereby, the technique enables a drastic improvement in the quality of arbitrary 3D-interconnect structures, made up of varying metal-wire and TSV segments. Despite these drastically higher gains, the proposed CAC technique results in lower overhead costs than the best previous technique.

Since the proposed and previous CAC techniques are only applicable in case of a temporal alignment between the signal edges, an optimization technique is moreover contributed, which simultaneously improves the TSV performance and power consumption for arbitrary scenarios of temporal misalignment between the signal edges at low costs. The two optimization techniques have been presented to the scientific community through [A3, A4, P1, P3].

4. An optimization technique that improves the critical TSV manufacturing yield, while still providing a drastic improvement in the power consumption of TSVs and metal wires at relatively low costs. Again, the insights provided by the contributed high-level models are exploited for the derivation of this fourth contributed optimization method. Furthermore, the technique exploits technological heterogeneity between the dies of a 3D IC. This optimization technique has been presented to the scientific community through [A5].

1.5. Thesis Outline

This thesis is organized in four main parts: The present introductory part; a first core part in which the high-level models are derived and evaluated; a second core part in which the optimization techniques are developed and evaluated; and a final concluding part. In detail, the four parts are structured as follows:

I **Introduction**
The remaining Chapter 2 of the first part includes the background of the research problem addressed by this doctoral research project.

II **Modeling**
The high-level models to estimate the pattern-dependent power consumption and performance of TSV-based 3D interconnects are presented in this second part. In Chapter 3, generally valid formulas to precisely estimate the 3D-interconnect power consumption and performance, depending on the bit-level characteristic of the transmitted signals and the interconnect capacitances, are presented. Afterward, a high-level model to estimate the required capacitances of TSV arrays on higher abstraction levels—encapsulating complex physical phenomena—is presented in Chapter 4. In the last Chapter 5 of this second part, methods to estimate the required bit-level statistics for common data-transmission scenarios are presented.

III **Optimization**
The proposed low-power techniques to improve the 3D-interconnect quality

are presented in this third part. First, the technique to reduce the TSV power consumption at negligible costs is presented in Chapter 6. Afterward, the CAC technique, which effectively improves the power consumption and performance of arbitrary 3D-interconnect structures in the case of temporally aligned signal edges, is presented in Chapter 7. In the following Chapter 8, the optimization technique, which improves the performance and power consumption of TSV-based interconnects for arbitrary scenarios of temporally misaligned signal edges, is presented. At last, the optimization technique, which improves the manufacturing yield and the power consumption of arbitrary 3D interconnects at the same time, is presented in Chapter 9.

***IV* Finale**

In the final part of this thesis, an overall conclusion is drawn. Furthermore, the impact of this doctoral research project on future work is discussed.

CHAPTER 2

Background

The background to the problem addressed by this dissertation is outlined in this chapter. First, the interconnect bottleneck in nanometric VLSI systems and primary solutions, integrated into most of today's system on chips (SoCs), are briefly summarized. Afterward, the main reasons why 3D integration is widely acknowledged as a promising solution for the challenges we are facing for the next generations of SoCs—in which existing techniques alone will not be sufficient—are outlined. In the sequel, manufacturing techniques and challenges for the realization of 3D ICs are discussed. Thereby, it is outlined that TSV-based 3D integration is the most superior approach today due to the more mature involved manufacturing steps. Afterward, the design challenge of TSV-based 3D integration, which is addressed by this doctoral research project, is outlined in depth in Section 2.4. Finally, the chapter is concluded in Section 2.5. Smaller parts of this chapter are published in "Elsevier Integration" as an invited but peer-reviewed article (see [A4]).

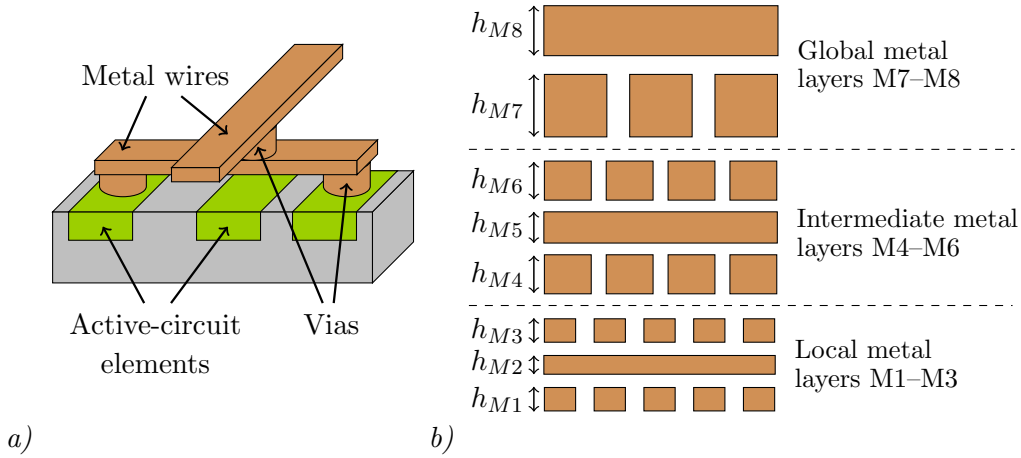


Fig. 2.1: Metal-wire interconnects: *a)* Connections to the active circuits in the FEOL using metal wires and vias (simplified); *b)* Cross-view through an eight-metal-layer BEOL (not showing vias) with wires in adjacent layers being routed in orthogonal directions. Note that the isolating dielectric of the BEOL is not shown in the figure.

2.1. Interconnects—A Never-Ending Challenge?

Once negligible, the on-chip interconnects become more and more the bottleneck of integrated circuits with ongoing technology scaling. While transistors still become significantly faster and more power-efficient with each new technology node, the parasitics of the interconnects scale rather poorly [3]. The reason for this is that more densely spaced interconnects result in drastically increased coupling capacitances between them, which is a threat to the power consumption and the signal propagation delay [1]. Furthermore, the small interconnect cross-section, required to connect aggressively scaled active circuit elements, impair the interconnect resistances, which again affects the delay negatively.

In a 2D IC, interconnections between the active circuit elements in the front-end of line (FEOL), are realized through metal wires and vias patterned in the back-end of line (BEOL), as shown in Figure 2.1a. To reduce the interconnect bottleneck, the BEOL of modern 2D ICs consists of multiple metal layers that are grouped into three main tiers: Local, intermediate, and global, as illustrated in Figure 2.1b [42]. Each tier typically consists of multiple layers. Wires in adjacent layers are routed in orthogonal directions as it drastically reduces the coupling capacitances between wires of different layers [43]. Metal vias establish low-resistive connections between wires in adjacent layers at intersections. Local metal wires in the bottom metal layers (*i.e.*, closest to the FEOL) have the smallest cross-section. They are needed to connect nearby active-circuit elements that often have a pitch in the nanometer range. Intermediate wires in

the next higher metal layers have a larger cross-section than the local wires and are used to connect components that are slightly further away. Global wires in the topmost layers have the highest cross-section and are used to connect components that are particularly far away. The key idea of increasing the wire cross-section for critical longer routes is to decrease the resistance and thereby the signal propagating delay. However, an expansion in the wire cross-section has an adverse impact on the power consumption as it increases the parasitic capacitances.

The interconnect bottleneck has also been addressed from the material/manufacturing perspective, for example, by using low-κ dielectrics as the insulator between the metal wires and vias, which decreases the parasitic capacitances. Furthermore, the usage of copper as the conductor material for wires and vias is well established today as it results in particularly low interconnect resistances [44–46].

Placing stable power or ground lines in between the wires helps to reduce the performance degradation due to the crosstalk over the coupling capacitances without the need for advances in the BEOL manufacturing [47]. The same applies to increasing the wire width (decreases resistance) and spacing (decreases coupling capacitances) for long critical paths [47]. However, these techniques reduce the effectively available routing resources and thus have an adverse effect on the routability, which often is a major concern in today's aggressively scaled technology nodes. Other techniques that do not impair routability as drastically require extra active-circuit elements, such as repeater insertion to reduce the interconnect delay at the expense of an increase in the power and area requirements [47]. Techniques integrated at the bit-level and the register-transfer level, such as data encoding, typically result in the most significant improvements in the metal-wire power consumption and performance of all existing approaches [CA1, 39]. However, such techniques can result in a significant circuit overhead, which is why they have to be integrated wisely.

The previously summarized techniques have prolonged an ongoing performance improvement gained from technology scaling over the last decades. However, all techniques do not allow to reduce/scale the maximum and mean wire length beyond technology-scaling. Thus, they cannot be seen as ultimate solutions to overcome the interconnect bottleneck. Despite all efforts, the global interconnects are responsible for the largest fraction of the power consumption and are the limiting factor for the maximum clock speed in most of today's SoCs [3]. Hence novel techniques are required to improve the interconnect quality in future SoCs.

A promising approach, based on higher abstraction levels, is to integrate on-chip network structures that realize a packet-based/hop-to-hop global communication (similar to Internet) between larger circuit blocks. Integrating such a network on chip (NoC) drastically reduces the maximum wire length—even

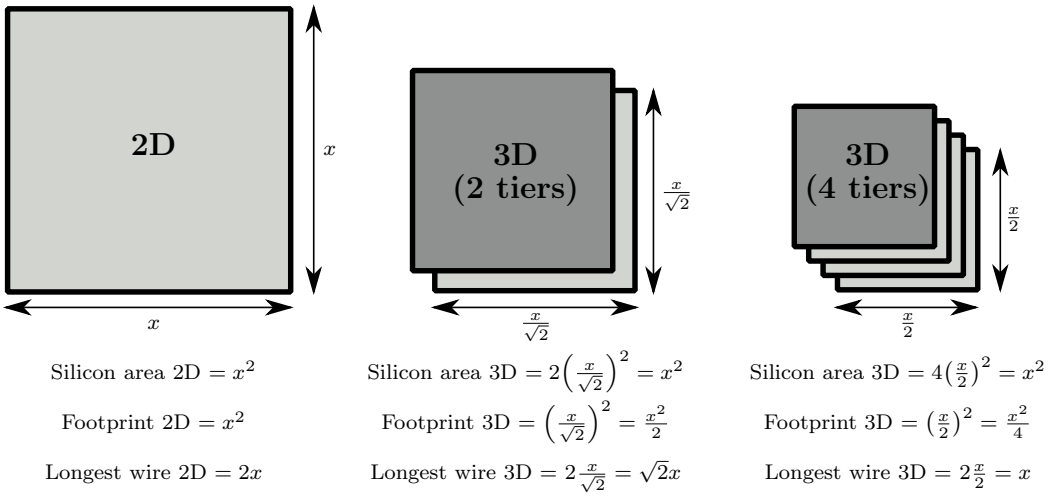


Fig. 2.2: Impact of 3D integration on the wire lengths and the system footprint.

if the number of integrated processing elements is increased concurrently [48]. Moreover, using optical interconnects instead of metal wires is a promising approach for the global communication in future SoCs, which has to be investigated on all levels of abstraction [49]. This is also true for the third emerging technique, which is to integrate circuits into the third dimension, in order to reduce the metal-wire lengths through a reduced IC footprint. Each of the three approaches is not intended to be a replacement for the other two, but rather all three have to be seen as synergetic. This dissertation mainly contributes to the third approach, 3D integration, which is outlined in depth in the following two sections.

2.2. Benefits of 3D Integration

The possibility of integrating circuits into the third dimension is investigated more and more for two main reasons. One reason is the resulting decrease in the wire lengths over an equivalent 2D-integrated system [50]. Figure 2.2 is used to illustrate the reason for this better. Consider a 2D IC with a quadratic floorplan that has a side length of x. Hence, the available silicon/substrate area, as well as the footprint of the IC, is x^2. Since wires are only routed in two directions that are orthogonal to each other (*i.e.*, Manhattan routing), the maximum possible wire length in an IC is equal to the sum of the two side lengths of the rectangular floorplan (here $2x$).

Now consider an alternative implementation as a 3D IC made up of two stacked substrates. With a footprint that is half as big, the same substrate area is available in this 3D IC as in the 2D IC. Thus, the side lengths of the floorplan can be theoretically reduced by a factor of $\sqrt{2}\times$ to $x/\sqrt{2}$, without increasing

a) Memory-logic stack [CA3] *b)* CPU-FPGA-DRAM stack [51]

Fig. 2.3: Examples of heterogeneous 3D integration: *a)* Memory-logic 3D integration; *b)* Memory-FPGA-CPU 3D integration.

the integration density in the substrates with respect to the substrate of the 2D IC. When the number of tiers in the 3D IC is again doubled to four, the side lengths, and hence the wire lengths, can be further reduced by $\sqrt{2}\times$ to $x/2$, while still providing the same substrate area for integration. Hence, the maximum and mean wire length is reduced by about a factor of $\sqrt{NT}\times$, where NT is the number of tiers of the 3D system (*i.e.*, physically stacked substrates). Thus, 3D integration results in a continuous scaling in the wire lengths with an increasing number of tiers.

Please note that, in this motivational example, overhead costs for vertical interconnects—required to establish inter-tier connections—are not considered. Thus, only if the power consumption, performance, or area of the vertical interconnects does not cancel out the promised wire-length savings gained from the 3D organization, 3D integration is a viable solution for the interconnect bottleneck. Hence, efficient vertical interconnects between the substrates are imperative to obtain the promised interconnect-related power and performance gains from integration into the third dimension.

Another key feature of 3D integration—often seen as even more promising than the implied wire-length reductions—is that it enables heterogeneous integration. In a 3D IC, the electrical characteristics of the individual tiers/substrates can be fine-tuned in a way that each layer is particularly efficient for the integration of a specific kind of component. Furthermore, components that are physically located in different tiers are no longer constrained by being fully process-compatible with each other. This feature brings numerous advantages such as a decreased design complexity for full-custom components, a higher system performance, and a lower power consumption [6].

An already well-established example of heterogeneous integration is memory-on-logic stacking, where some dies are only dedicated to the integration of memory components (*e.g.*, SRAM cells), while other ones are dedicated to the integration of semi-custom logic blocks made up of standard cells as illustrated

in Figure 2.3a. Through such a heterogeneous 3D integration, the memory cells are no longer constrained by being process compatible with logic cells, and *vice versa*, as they are located in different substrates. Already for a two-tier 3D system, memory-on-logic 3D integration shows to improve the performance of a placed and routed multi-core processor system by 36.8 % compared to a 2D baseline design [CA3]. Other works such as [51] advocate increasing the degree of heterogeneity by adding an FPGA layer between a standard-logic layer (used to integrate a CPU) and a memory layer, as illustrated in Figure 2.3b. This organization promises a power-consumption reduction by up to 47.5 % versus a baseline 2D system [51].

Also, mixed-signal SoCs benefit from heterogeneous 3D integration. In contrast to logic components, sensors and other analog or mixed-signal components typically do not benefit from using ultimately scaled technology nodes. Thus, in 3D-integrated mixed-signal SoCs, one or more substrates can be optimized for the integration of digital components by using an aggressively scaled technology, while other substrates are optimized for mixed-signal and analog components by using a less aggressively scaled technology. Hence, heterogeneous 3D integration promises significantly better power, performance, area, and cost metrics than homogeneous 2D or 3D integration for a broad set of systems.

2.3. Manufacturing Techniques for 3D ICs

In this section, the two main possibilities to realize a true 3D IC from the manufacturing perspective are reviewed: TSV-based and monolithic 3D integration.[1] Also, the advantages and disadvantages of both approaches from the manufacturing perspective are briefly outlined. However, the focus is placed more on TSV-based 3D integration in this section, as it is the integration style that is addressed by this dissertation.

2.3.1. TSV-Based 3D Integration

The idea of TSV-based 3D integration is to stack pre-fabricated "2D dies", as illustrated in Figure 2.4. In this 3D-integration style, the vertical interconnections through the substrates are realized by through-silicon vias (TSVs). Stacking pre-fabricated "2D dies" brings the tremendous advantage that only mature manufacturing techniques from traditional 2D-IC manufacturing are required, except for the TSV fabrication and the die-to-die bonding. Hence, arbitrary FEOL and BEOL manufacturing techniques known from 2D-IC manufacturing can be reused for TSV-based 3D integration. Consequently, TSV-based

[1] Integration methods in which the vertical interconnects between the tiers are realized off-chip (*e.g.*, through wire-bonding or a passive interposer) do not provide a high vertical throughput. Thus, such techniques are referred to as 3D-packing or 2.5D-integration, rather than 3D-integration, strategies [50].

*Not to scale

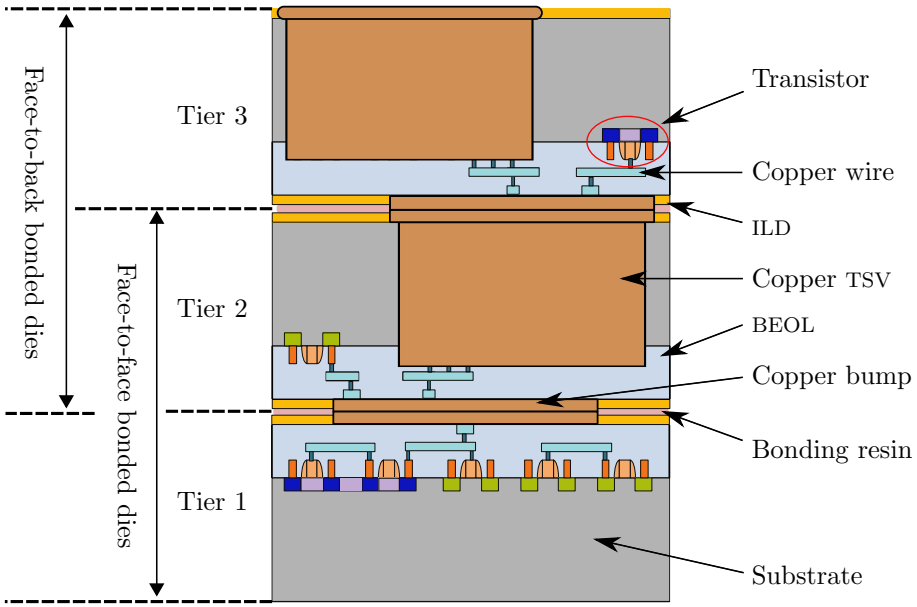


Fig. 2.4: Cross-view illustration of a three-tier TSV-based 3D IC. The first two dies are face-to-face bonded, and the third die is bonded on top of the second one in a face-to-back manner.

3D ICs can be manufactured with well-established transistor technologies such as silicon on insulator (SOI), FinFET, or traditional planar CMOS. This fact accelerates the process of making 3D ICs more efficient than 2D counterparts. Also, it increases the manufacturing yield compared to sequential/monolithic approaches where the 3D stack is grown on top of a single handle substrate.

In a stacked 3D IC, TSVs are not needed for the interconnects between the first two tiers. The first two pre-fabricated dies can be electrically connected through their top-most metal layers using a face-to-face bonding technology, as illustrated in Figure 2.4. This only requires bonding bumps, which are typically made from copper due to the inherent advantage of compatibility with the metal wires and vias in the BEOLs of the dies. As the bonding resin, epoxies and polymers are commonly used due to their excellent adhesive properties. From the third die onward, a face-to-back bonding is needed, which connects the top metal layer of the added die with the substrate backside of the previously bonded die. For a face-to-back bonding, TSVs are needed to establish low-resistive electrical connections between elements located in different dies, as shown in Figure 2.4. In contrast to metal wires and vias, a TSV occupies an area of the substrate, which consequently cannot be used for active-circuit elements. Hence, TSVs increase substrate-area requirements.

TSV Manufacturing

In the following, the manufacturing steps for the fabrication of TSVs are briefly reviewed. Three main TSV-manufacturing variants exist: Via-first, via-last, and via-middle [50]. In the via-first process, TSVs are formed in the substrate before the active circuits (*i.e.*, FEOL) and the metal layers (*i.e.*, BEOL). Via-first TSV manufacturing has the advantage that it generally results in the shortest TSVs. However, for the manufacturing of the FEOL, very high temperatures are required. This is a threat to via-first TSVs. Consequently, via-first TSVs must have strong thermal reliability, which typically forbids copper as the conductor material. However, copper TSVs are desirable due to the compatibility with standard BEOL fabrication steps.

The second variant, via-last TSVs, are manufactured after the FEOL and the BEOL. Via-last manufacturing has the advantage that the TSVs only have to withstand the manufacturing stress caused by the bonding and wafer thinning. On the downside, via-last TSVs are the longest, and the TSV etching must-be performed through several metal and dielectric layers besides the substrate. Furthermore, via-last TSVs have a lower thermal budget for manufacturing as the previously fabricated metal layers in the BEOL must be preserved.

Thus, the predominant approach today is to use via-middle TSVs. As illustrated in Figure 2.5, via-middle TSVs are fabricated after the FEOL but before the BEOL. Hence, first, the active circuits in the substrate and the pre-metal dielectric are fabricated, subsequently the TSVs, and finally, the BEOL.

Through-silicon vias are formed by etching a cylindrical hole in the substrate, which is then filled with copper surrounded by a dielectric liner to isolate the TSV conductor from the doped, and thus conductive, substrate. For this purpose, the *Bosch* process is used, which was initially invented to manufacture micro-electro-mechanical systems (MEMS) [50,52]. The *Bosch* process applies an etching and a silicon-dioxide (insulator) deposition in successive time intervals in the range of seconds. Afterward, the etched hole with the insulator is filled with the TSV conductor material.

Typically, TSVs are formed as blind vias before bonding, which are exposed during a wafer-thinning step (*e.g.*, wet etching), as illustrated in Figure 2.5. The advantages of this approach are the compatibility with well-established manufacturing techniques and a simplified wafer handling [53]. However, this has the severe disadvantage that the following wafer-thinning and bonding process steps induce stress on the formed TSVs, impairing their manufacturing yield. Nevertheless, a thinning of the wafers or dies before the TSV manufacturing is typically not an alternative as this requires several processing steps with a thin wafer, which makes manufacturing significantly more difficult.

The wafer is typically thinned before bonding, which often demands a temporary bonding of a carrier/wafer-handle after thinning to increase the

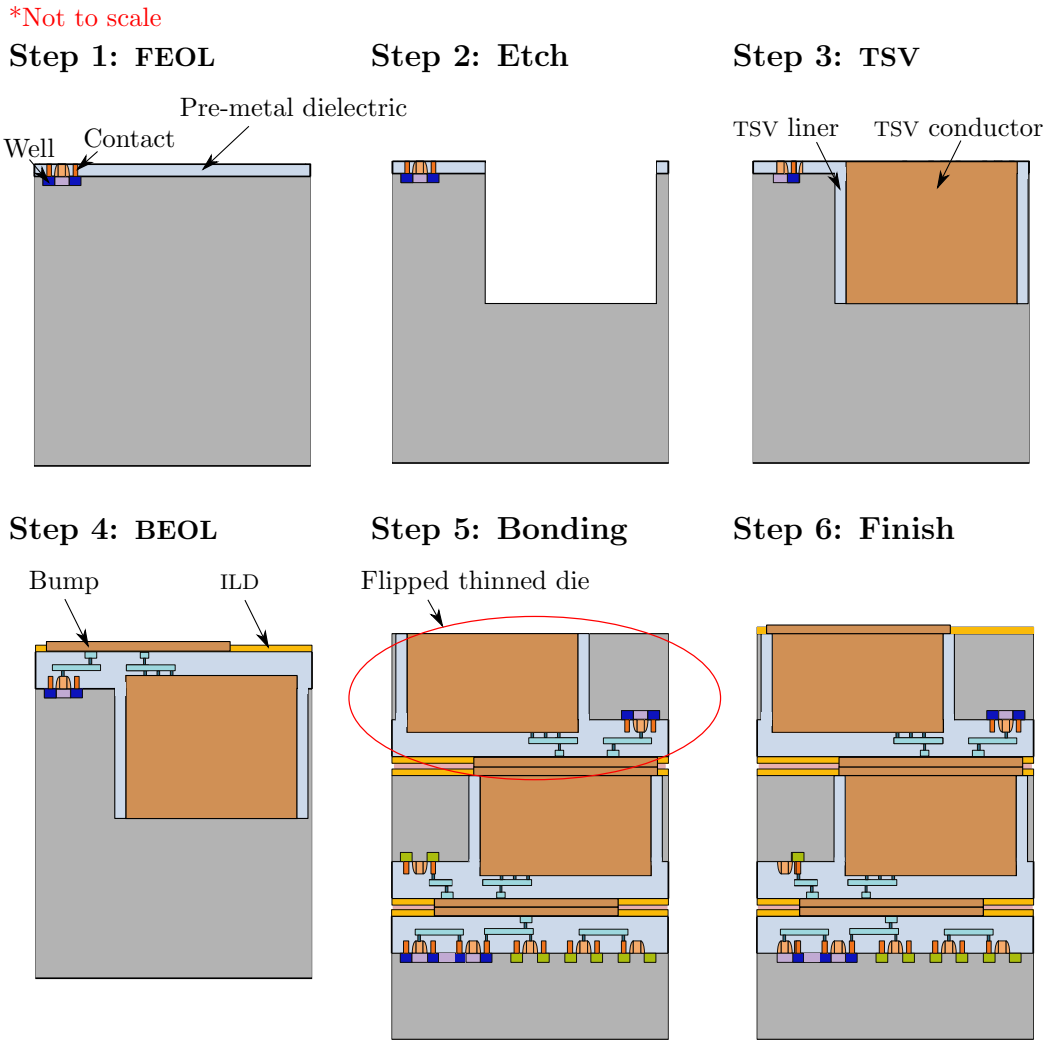


Fig. 2.5: Basic (simplified) steps to fabricate a 3D IC with via-middle TSVs.

mechanical stability during the bonding stage [50]. After thinning and bonding, an inter-layer dielectric (ILD) is deposited on top of the stack together with copper bumpers. These bumpers are required for the electrical connections with the next die that will be bonded to the stack. For the last die of the stack, the bumpers are extended by solder balls, required to bond the 3D IC onto a printed circuit board in a flip-chip manner.

TSV Manufacturing Challenges

While metal wires and vias, as well as active-circuits elements, are fabricated with mature, and thus efficient, manufacturing techniques; in stacked 3D ICs, the fabrication of the TSVs is challenging.

The first critical challenge is to achieve a high TSV manufacturing yield [9]. Even-though correctly manufactured TSVs enable a relatively reliable high-bandwidth communication between the dies (compared with other approaches such as inductive coupling through the dies), they have the drawback of being fabricated with a poor manufacturing yield due to the immaturity of the involved process steps.

In the following, the major TSV manufacturing defects are briefly summarized. Four main TSV defect types exist: Voids, delamination at the interface, material impurities, and TSV-to-substrate shorts [24–26]. Improper TSV filling or stress during bonding can result in voids or cracks in the TSV conductor. The second defect type, TSV delamination, is caused by a spatial misalignment between a TSV and its bumper during the bonding stage. Material impurities mitigate the conductivity of the TSV channel. A TSV-to-substrate short arises due to a pinhole in the oxide liner, which usually isolates the TSV from the conductive substrate.

Since the occurrence of TSV defects can dramatically reduce the overall manufacturing yield, TSVs have to be integrated efficiently (*i.e.*, sparsely). This requirement typically demands to use TSVs only on a global level in the form of clustered arrays (*i.e.*, to connect larger circuit blocks), as it reduces the overall number of required TSVs. Moreover, using regular arrays facilitates TSV manufacturing due to the more regular patterning for the *Bosch* process and the TSV filling. Furthermore, redundancy techniques can be integrated to increase the overall manufacturing yield if the TSVs are clustered together [9].

However, the immaturity of the TSV manufacturing steps does not only has a critical impact on the manufacturing yield but also on the silicon-area requirements. *Globalfoundries* has presented its via-middle TSV manufacturing processes, which are used for their 14-nm and 20-nm wafers, through a comprehensive set of publications (*e.g.*, [54, 55]). Pictures of TSVs manufactured by *Globalfoundries*, are shown in Figure 2.6.

A close look at the figures reveals that the TSVs are extremely large compared to the metal wires in the BEOL and the transistors in the FEOL. The radius of the commercially available TSVs is about 2.5 μm for the 14-nm technology and 3 μm for the 20-nm technology. Furthermore, the fabricated TSVs are 50 μm long/deep after die thinning for both technology nodes. Also for the successive TSV-manufacturing node of the foundry, the TSV depth is kept at 50 μm, but the radius is further reduced to 1.5 μm [56]. The reason for the steady TSV depth likely is the fact that an even more aggressively thinned substrate dramatically decreases the yield. Moreover, a too aggressively thinned substrate leads to a critical increase in the inter-die noise and thermal coupling.

The reason for the large TSV radius—determining the reduction in the available substrate area for active-circuit components—is that TSVs cannot yet be manufactured with a large aspect ratio (*i.e.*, the ratio of the via length

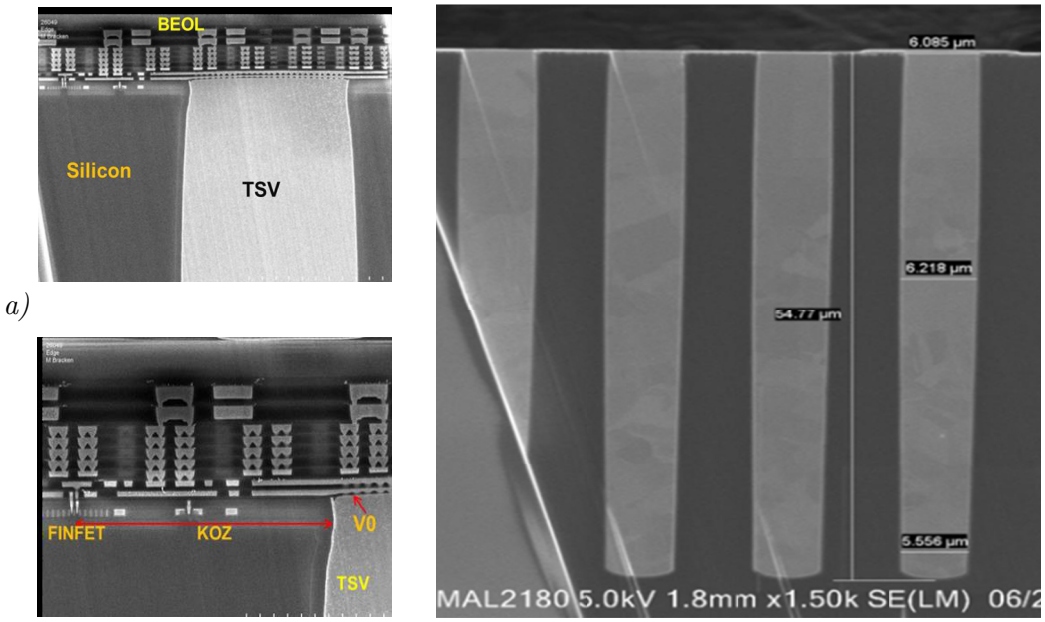


Fig. 2.6: *Globalfoundries* via-middle TSV process: *a)* Cross-view of the top of a fabricated TSV with a radius of 2.5 μm and a depth of 50 μm and the full BEOL stack for the 14-nm node; *b)* FinFETs device at the edge of the TSV KOZ for the 14-nm node; *c)* Full cross-view of a fabricated via-middle TSV with a radius of 3 μm and a depth of 55 μm for the 20-nm node before wafer thinning (note that after wafer thinning the TSV depth is reduced from 55 μm to 50 μm). Pictures taken from [54, 55].©IEEE

over its diameter). For the year 2018, the minimum possible aspect ratio of a global TSV is $^{20}/_{1}$, according to *International Technology Roadmap for Semiconductors* (ITRS) predictions [41]. This maximum ratio implies a TSV radius of at least 1.25 μm for a 50 μm thin substrate. Furthermore, since the die-to-die bonding is a mechanical process, its alignment is relatively poor (typically in the micrometer range), which is another limiter for the minimum TSV or bumper size.

To put the substrate area occupation of a TSV into a better perspective, the quadratic substrate footprint of a TSV (*i.e.*, $4\,r_{\text{tsv}}^2$) with a 2.5-μm radius—integrated into 14-nm wafers [55]—is compared to the footprint of the NAND cell (drive strength 1×) and the full-adder cell of the publicly available 15-nm standard-cell library *NanGate15* [57]. Furthermore, the footprint of the two standard cells is compared to the footprint of an aggressively scaled global TSV with a 1-μm radius.

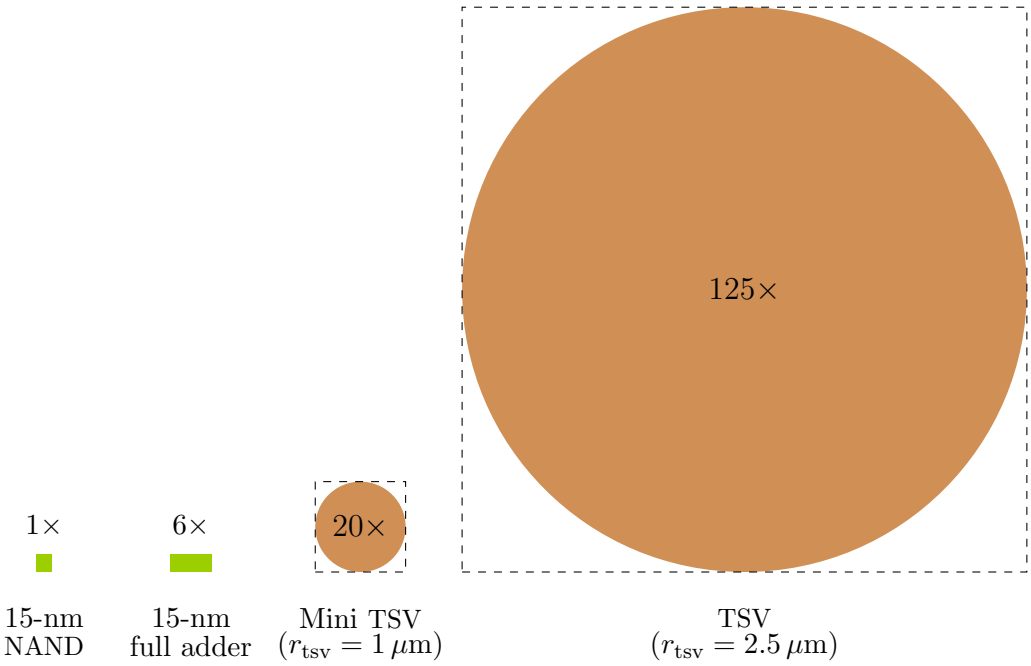


Fig. 2.7: Substrate footprints of modern TSVs and standard cells from the 15-nm library *NanGate15*. The considered radius, r_{tsv}, of a standard TSV is equal to the one demonstrated in commercial 14-nm wafers [55]. For the "mini TSV", the radius is equal to the minimum radius of a global TSV predicted by the ITRS for the year 2018 [41].

In Figure 2.7, the results of the comparison are illustrated to scale. The substrate area occupation of a commercial TSV is about 125 and 21 times bigger than for a NAND cell and a full-adder cell, respectively. Even for the aggressively scaled radius, the TSV footprint is still more than three times larger than the footprint of the 15-nm full-adder cell. Furthermore, a TSV is surrounded by a so-called keep-out zone (KOZ), which is an area where no active circuit can be placed in the substrate, resulting in a further reduction in the effectively available area for logic cells. The reason for the KOZ is that the copper filling of a via-middle TSV results in high thermal stress around the TSV, which can impair previously fabricated FEOL structures [54]. For the 14-nm FinFET transistors used in [55], the TSV KOZ is about the size of the TSV diameter, as shown in Figure 2.6b.

In summary, two main TSV-manufacturing challenges need to be mastered for TSV-based 3D integration: First, to achieve a higher TSV yield, and second, to reduce the TSV area occupation. To palliate both issues from the design perspective requires to use TSVs only on a global level and in the form of regular array arrangements.

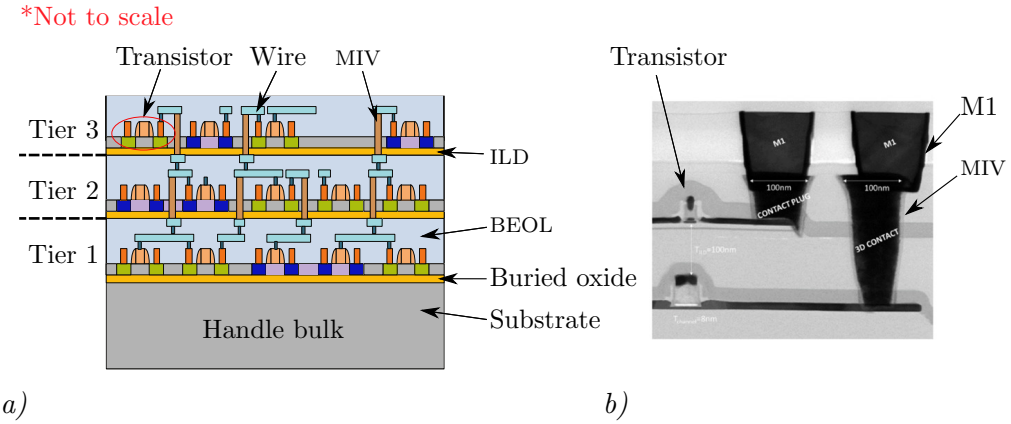


Fig. 2.8: Monolithic 3D integration: *a)* Cross-view illustration of a three-tier circuit. *b)* Cross-view of a small manufactured monolithic 3D circuit from [61].©IEEE

2.3.2. Monolithic 3D Integration

In this subsection, the concept of monolithic 3D integration for future 3D ICs is briefly discussed alongside its advantages and disadvantages. Monolithic 3D integration is an emerging alternative to TSV-based 3D integration that results in a much lower area occupation for the vertical interconnects through the substrates. A monolithic 3D system is fabricated sequentially on a single handle bulk/substrate, one tier after another, instead of stacking pre-fabricated 2D dies [58–60]. The structure of a monolithic 3D IC is illustrated in Figure 2.8.

First, the active circuits and the metallization in the bottom tier (handle bulk) of a monolithic 3D IC are processed with standard techniques, which are also used to manufacture 2D ICs. Second, an inter-layer dielectric (ILD) is added on top of the metallization, followed by a direct-bonding of a top substrate. Afterward, this pure substrate is aggressively thinned down, for example, through a wet-etching process. The resulting SOI structure is used to integrate the active circuit elements of the second tier. However, to fabricate the active components in the higher tiers (*i.e.*, second and above), only a lower thermal budget (*ca.* 500° C) is available to preserve already manufactured components in the metal layers of lower tiers. After forming the active circuits of the second die, the associated metallization is created. Thereby, the *Bosch* process is used to etch the vertical interconnects between the tiers. The vertical inter-tier interconnects in a monolithic 3D IC are commonly referred to as monolithic inter-tier vias (MIVs). To build more tiers on top, the manufacturing steps are repeated from step two, the ILD formation.

The previously summarized sequential flow offers much smaller and thus denser vertical inter-tier interconnects compared to TSV-based 3D integration. While TSVs are huge compared to active circuit components, MIVs can be

manufactured with a radius as small as 50 nm, which is in the range of the radius of a modern local metal via. Thus, an MIV is even significantly smaller than modern logic cells. The reason for this is the small thickness of the substrates from the second tier onward and the absence of mechanical via bonding. These two facts result in MIV sizes and pitches that are typically only bounded by the stepper resolution of the lithography machine.

Even-though monolithic 3D integration promises to exploit the full potential of 3D integration in terms of wire-length reductions, it is still an immature technology with an inferior manufacturing yield—even if compared to TSV-based 3D integration. The reason for this low yield is that, in contrast to TSV-based 3D integration, monolithic 3D integration requires to fundamentally develop new processing steps to form active circuits and metal layers. Not only because of the lower thermal budget for the upper tiers but also because of the vastly different technological parameters [61]. Hence monolithic 3D integration is considered as still several years away from mass production. For this reason, this dissertation addresses TSV-based 3D integration.

2.4. TSV Capacitances—A Problem Resistant to Scaling

Extensive research has been conducted on how to accelerate the process of overcoming TSV-related yield and area issues through techniques derived on higher-levels of abstraction (*e.g.*, [9, 37, 38, 62–64]). However, another major issue of TSV structures is not yet adequately addressed: TSVs entail large parasitic capacitances that can be a crucial bottleneck of modern and future 3D ICs, especially with regards to the power consumption. This TSV-related issue is outlined in this section.

For this purpose, the parasitics of modern global TSVs are compared to the parasitics of modern standard cells. The investigated standard cells belong to two libraries that were also previously used to compare the impact of TSVs on the 3D-layout quality (*i.e.*, power-performance-area metrics) [65]. One library is based on the 22-nm *Predictive Technology Model* (PTM) and the other one on the 16-nm PTM. The libraries also consider metal-wire and TSV parasitics. However, the used TSV dimensions in the sub-micron range are too optimistic, as revealed by the commercially available TSV structures. For example, the smallest analyzed TSV radius in [65] is 50 nm, which is 50× smaller than the radius of the TSVs integrated into commercial 14-nm wafers (2.5 μm).

2.4.1. Model to Extract the TSV Parasitics

To obtain TSV parasitics for realistic geometrical TSV dimensions and arbitrary array shapes, a parameterizable *Python* script is used, which, when executed, builds a 3D model of a rectangular TSV array in the *Ansys Electromagnetics*

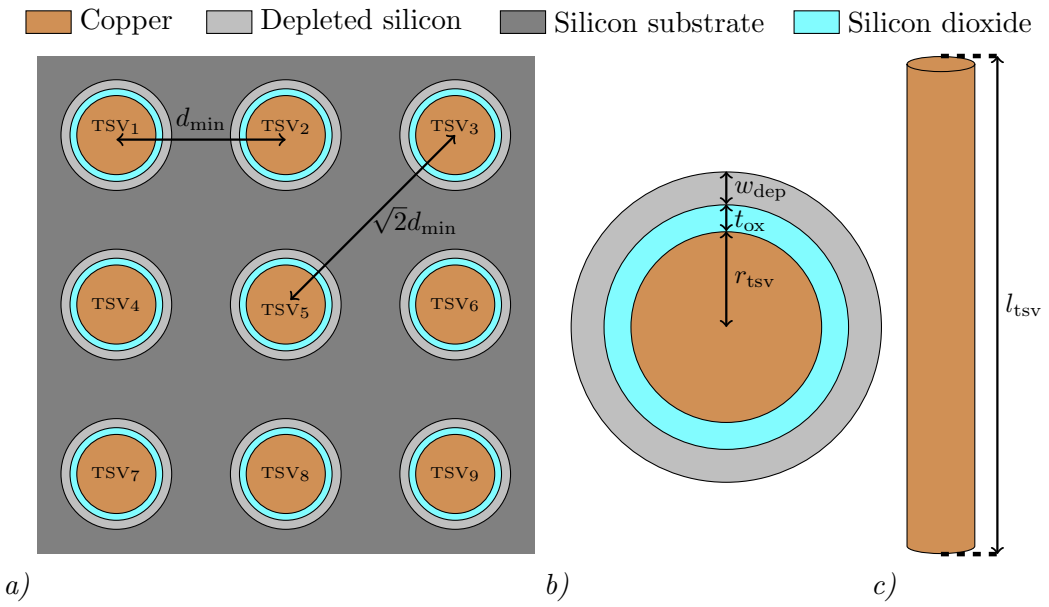


Fig. 2.9: Parameterisable 3D model of a TSV array used for parasitic extraction: *a)* Substrate top view for a 3×3 array; *b)* TSV cross-view; *c)* TSV conductor side view.

Suite. Such a 3D model enables us to extract the TSV-array parasitics by means of the electromagnetic (EM) field solver *Q3D Extractor*, which uses the method of moments [66].[2]

The TSVs in the 3D model are homogeneously placed in an $M \times N$ array and indexed as TSV_i, where the column location of the TSV in the array is equal to i modulo N, and the row location is equal to $\lceil i/N \rceil$. Both M and N can be arbitrarily defined through two parameters in the script. An exemplary model instance of a 3×3 TSV array is depicted in Figure 2.9. The minimum pitch between the centers of directly adjacent TSVs is set by a parameter represented by $d_{\min}$ in this thesis. Thus, diagonally adjacent TSVs in the model have a pitch of $\sqrt{2}d_{\min}$. The length, l_{tsv}, and radius, r_{tsv}, of the cylindrical copper TSVs are parameterizable as well. A substrate area with an x/y-expansion of at least $2\,d_{\min}$, in which no other TSV is located, is surrounding the TSV array in the 3D model.[3]

In the 3D model, the TSVs mainly traverse the silicon substrate, which has an electrical conductivity that is defined by parameter σ_{subs}. Unless stated

[2]Parasitic extractions for this TSV-array model using *Q3D Extractor* are used extensively in most chapters of this thesis.

[3]Due to the large TSV depth, compared to the depth of an n-well or a p-well (see Figure 2.6), nearby active circuits in the substrate have a negligible impact on the TSV parasitics.

otherwise, a typical p-doped (Boron) substrate, biased at 0 V, with a dopant concentration of about $1.35 \times 10^{15}\,\mathrm{cm}^{-3}$, is considered in this thesis. Such a substrate has a conductivity of about $10\,{}^{\mathrm{S}}/_{\mathrm{m}}$ due to the doping. For direct-current insulation from the conductive substrate, TSVs are surrounded by SiO_2 dielectrics of thickness t_{ox}. The ITRS does not report values for the thickness of these SiO_2 liners. In the 3D model, a TSV-liner thickness of $0.2r_{\mathrm{tsv}}$ is considered, being a realistic value according to existing manufacturing nodes.

The TSVs start and terminate in quadratic copper pads located in thin dielectric layers. These copper pads model bonding bumps at one side of the TSVs, and quadratic landing pads on the other side.[4] However, the copper pads and the thin dielectric layers have a secondary impact on the parasitics. Thus, the quadratic bumps, as well as landing pads, have a fixed side length of $2r_{\mathrm{tsv}}$ and a thickness of 50 nm in all analyses conducted throughout this thesis. This makes the TSV model reciprocal (*i.e.*, it is irrelevant which TSV side is the signal source and which the signal sink). The dielectrics used to isolate the copper pads from the conductive substrate have a constant thickness of 0.25 μm.

A TSV, its dielectric, and the substrate form a metal-oxide-semiconductor (MOS) junction. Consequently, a TSV is surrounded by a depletion region. For typical p-doped substrates, the width of the depletion region increases with an increase in the mean voltage on the related TSV, which further isolates the TSV from the conductive substrate. In contrast, the depletion-region width shrinks with an increasing mean TSV voltage for n-doped substrates [11].

The exact formulas from [11], are applied to determine the widths of the depletion regions surrounding the TSVs (represented by $w_{\mathrm{dep},i}$), which depend on the mean TSV voltages.[5] A depletion region is represented in the 3D model by a fully depleted substrate area (*i.e.*, electrical conductivity equal to zero), as in previous works [11, 14–16].

The parasitics of a TSV array show a slight frequency dependence, as shown in Chapter 4 of this thesis. Hence, TSV parasitics must always be extracted for a given significant frequency, f_{s}. According to the findings from [67], the significant frequency of a digital signal can be estimated using the driver-dependent mean rise/fall time of the TSV signals, T_{rf}:

$$f_{\mathrm{s}} \approx \frac{1}{2T_{\mathrm{rf}}}. \tag{2.1}$$

With the parameterizable TSV model, a parasitics extraction for various TSV-array structures is possible. The used *Q3D Extractor* employs a quasi-static field approximation for parasitic extraction, providing a high accuracy for reasonable significant frequencies. For all structures analyzed throughout this thesis, the

[4] Quadratic landing pads are used to connect TSVs with metal wires [65].

[5] In Appendix C, the formal method to determine the depletion-region widths is outlined.

Table 2.1.: Extracted parasitics for densely spaced global TSVs.

Large ($\mathbf{r_{tsv} = 2.5\,\mu m}$)			**Typical** ($\mathbf{r_{tsv} = 2\,\mu m}$)			**Small** ($\mathbf{r_{tsv} = 1\,\mu m}$)		
R_{tsv} [Ω]	C_{tsv} [fF]	$\kappa_{\mathrm{coup,tsv}}$ [%]	R_{tsv} [Ω]	C_{tsv} [fF]	$\kappa_{\mathrm{coup,tsv}}$ [%]	R_{tsv} [Ω]	C_{tsv} [fF]	$\kappa_{\mathrm{coup,tsv}}$ [%]
0.08	31.62	99.45	0.10	31.35	99.61	0.27	29.21	99.74

relative error within the quasi-static approximation is smaller than 0.2 % [68]. Even for an analysis of frequencies as high as 40 GHz (*i.e.*, $T_{\mathrm{rf}} \approx 13$ ps), this error remains below 2 %.

2.4.2. Analysis

A TSV radius of 2.5 μm (demonstrated for a commercial 14-nm wafer [54]), is considered in this work as a "large" value. Furthermore, a shrunk radius of 2 μm is chosen as a "typical" value. Thereby, even a conservative analysis of the impact of the TSV parasitics in a 16-nm or a 22-nm transistor technology is presented in the following. To moreover analyze the effect of an aggressive TSV scaling, "small" TSVs with the minimum possible radius predicted by the ITRS for the year 2018 (*i.e.*, 1 μm) are considered as well. The TSV depth is fixed to 50 μm. Hence, the depth does not scale in accordance with [54–56].

Considered are densely spaced TSVs, which minimize the area overhead due to a TSV array. Thus, the pitch between directly adjacent TSVs, d_{min}, is chosen as $4\,r_{\mathrm{tsv}}$ in accordance with the minimum possible value predicted by the ITRS for all manufacturing nodes [41]. The array dimensions are varied in this analysis between 3×3 and 10×10 to consider arrays with small, as well as large, amounts of TSVs.

Lumped resistance-inductance-capacitance (RLC) parasitics of the individual arrays are extracted with the *Q3D Extractor* for a significant signal frequency of 6 GHz (*i.e.*, $T_{\mathrm{rf}} \approx 83$ ps). The resulting maximum overall capacitance of a TSV, C_{tsv}, as well as the maximum TSV resistance, R_{tsv}, are subsequently compared for the three TSV radii. In Table 2.1, the results are reported which tend to be independent of the array shape, $M \times N$. Moreover, the table also includes the coupling ratio, $\kappa_{\mathrm{coup,tsv}}$, defined as the ratio of the sum of all coupling capacitances of a TSV over its accumulated capacitance value (a coupling capacitance is a capacitance between two different TSVs). To set the reported TSV capacitances in relation with the capacitances of logic cells, Table 2.2 includes the input capacitances of representative standard cells reported in [65].

The two tables reveal that to switch the logical value on a typical TSV requires, on average, 122 to 132 times more charge than to toggle an input of a standard two-input NAND gate for the 22-nm technology. These factors even increase to 133× and 144× for the 16-nm technology. The accumulated TSV

Table 2.2.: Input capacitances of 22-nm and 16-nm standard cells [65].

Cell	**22 nm** C_{in} [fF]	**16 nm** C_{in} [fF]
NAND 1×	0.24	0.22
XOR 1×	0.55	0.45
D flip-flop 1×	0.41	0.26
Inverter 4×	0.69	0.56
Full adder	1.31	1.36

Table 2.3.: Worst-case wire parasitics for the 22-nm and the 16-nm technology reported per unit wire length.

Wire	**22 nm**			**16 nm**		
	R'_{mw} [Ω/μm]	C'_{mw} [fF/μm]	κ_{mw} [%]	R'_{mw} [Ω/μm]	C'_{mw} [fF/μm]	$\kappa_{\text{coup,mw}}$ [%]
M1	9.43	0.11	65.20	23.92	0.09	48.18
M4	3.39	0.12	46.88	11.3	0.11	51.44
M8	0.38	0.16	47.13	0.95	0.15	48.91

capacitance is also more than 20 times larger than the input capacitances of the full-adder cells. Even scaling the TSV radius to the smallest possible value (*i.e.*, 1 μm), which implies a drastic increase in the manufacturing efforts, does not significantly close this gap. The scaling by a factor of 2× only reduces the TSV capacitances by less than 7 %. Thus, just relying on advances in the TSV radius due to improved manufacturing techniques, does not help to overcome problems due to the large TSV parasitic capacitances. In fact, with ongoing technology scaling for the transistors too, TSV parasitics likely become even more critical. The only approach that can overcome this issue is to aggressively shrink the die/substrate thickness, and thereby the TSV depth. However, this typically has unacceptable drawbacks for the manufacturing and the reliability, as outlined in Subsection 2.3.1.

Coupling capacitances contribute more than 99 % to the total capacitance of a TSV for all three analyzed radii. Thus, the coupling capacitances are the primary design concern. The coupling can also be a critical issue for metal wires [39], but the coupling capacitances of the metal wires are not reported in [65]. However, the dimensions for the eight-track metal stack per die are given. These values can be entered into the PTM interconnect tool [69] to obtain resistance and capacitance values for the local (M1), the intermediate (M4), and the global (M8) wires in the worst case (*i.e.*, minimum-spaced wires). Doing this results in the metal-wire parasitics reported in Table 2.3. The

results reveal that the coupling coefficients of metal wires are critical, but still significantly smaller than those of TSVs. Coupling effects are much more dominant for TSVs due to the increased number of adjacent aggressors in 3D, and the doped substrate TSVs traverse. Additionally, the coupling capacitances of metal wires, and thereby C'_{mw} and $\kappa_{\mathrm{coup,mw}}$, can be effectively reduced by increasing the line spacing, which does not work well for TSVs as shown in [17]. However, even without wire spacing, the overall capacitance of a small TSV is higher than the maximum capacitance of a global metal wire with a length of 182 μm. This value increases to over 243 μm when compared with local or intermediate wires. Hence, the capacitance per unit length of a TSV is much larger than for a metal wire. That the capacitances of local and intermediate wires as short as 3 μm already dominate over the input capacitances of standard cells, is another strong piece of evidence that parasitic TSV capacitances, such as metal-wire capacitances, are a serious concern.

Thus, the TSVs are a severe threat to the power consumption of a 3D IC since the parasitic capacitances, combined with the switching of the transmitted bits, determine the interconnect power consumption as shown in [39]—especially since this TSV issue shows to be resistive to scaling. Consequently, to address this challenge on higher abstraction levels, by exploiting the bit-pattern dependent nature of the power consumption, is of particular importance.

A positive of TSVs is their negligibly low resistance due to their large radius and the relatively short length/depth, especially when compared to long metal wires. For all three analyzed radii, the TSV resistance is significantly below 1 Ω, while the resistance of a long metal wire often is in the kΩ range. The signal-propagation delay of an interconnect for a transmission of random bit patterns can be estimated through its RC constant (*i.e.*, the product of the total interconnect-path capacitance and resistance) [70]. The total path resistance of an interconnect segment is estimated by the sum of the equivalent resistance of the driver's pull-up/pull-down path in the conductive state and the resistance of the actual interconnect.

To quantify the range of the driver's equivalent resistance, the current through the pull-up path of an inverting driver as a function of the transistor sizing is analyzed with the *Spectre* circuit simulator. For this purpose, a *Spectre* inverter circuit is built twice, once out of the 22-nm, and once out of the 16-nm PTM transistors. Ground is set at the input and the output of the inverter (beginning of pull-up phase) and the drain-source current, I_{ds}, through the p-channel MOS field-effect transistor (MOSFET) is measured. The quotient of the power-supply voltage and the current (*i.e.*, $^{V_{\mathrm{dd}}}/_{I_{\mathrm{ds}}}$) is used to estimate the effective channel resistance.

In Figure 2.10, the resulting resistances are plotted over $^{W}/_{L_{\mathrm{min}}}$ (the width of the transistor channel over its length). The equivalent resistance of the p-channel MOSFET is in the kΩ range for both technologies. It is about 40 kΩ

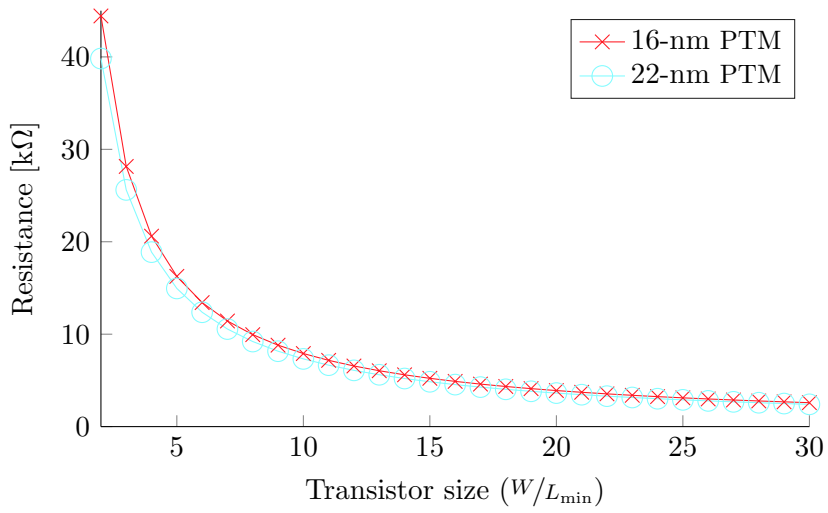


Fig. 2.10: Equivalent channel resistance of the 22-nm and the 16-nm PTM p-channel MOSFET transistor over the transistor size.

for small transistors and decreases toward 2.5 kΩ for very large transistor widths. This is orders of magnitude bigger than the resistance of TSVs and shorter metal wires. Thus, for a 3D interconnect made up of two drivers, a TSV, and short wires in-between, the total interconnect-path resistance is mainly determined by the drivers. In contrast, for long 2D interconnects, also the metal-wire resistance contributes significantly to the overall path resistance. Hence, the interconnect bottleneck in terms of timing can already be mitigated with today's TSV-manufacturing techniques; but, there still is plenty of room for improvement.

In summary, it was shown in this section that the large TSV capacitances are a severe threat to the power consumption of a 3D IC, which shows to be a problem that is resistive to scaling. Hence, even for future smaller TSV dimensions, a 3D implementation can even result in a higher power consumption compared to a logically equivalent 2D implementation. Furthermore, the large TSV capacitances affect the performance/timing negatively. Nevertheless, a 3D system will, in most cases, still result in a better performance than a 2D system due to the small resistance of a TSV.

2.5. Conclusion

The background of this doctoral research project was outlined in this chapter. First, it was outlined why integration into the third dimension is acknowledged as one of the most promising solutions to overcome the metal-wire bottleneck of modern digital systems. Furthermore, other promising key-features of 3D integration were discussed, such as the possibility of heterogeneous integration.

Today's most promising approach to realize a 3D IC is to stack pre-fabricated dies using TSVs for the inter-die interconnects. However, not only the low manufacturing yield of the TSVs is a roadblock for 3D integration, but also the negative impact of TSVs on the system's power consumption and performance due to their large parasitic capacitances. Through-silicon-via scaling will not effectively reduce the parasitic capacitances as shown experimentally in this chapter. Consequently, TSV-related power and performance issues should be addressed on higher abstraction levels by exploiting the dependence of the two metrics on the transmitted bit patterns. This builds a strong motivation for the present thesis.

Part II: Modeling

CHAPTER 3

High-Level Formulas for the 3D-Interconnect Power Consumption and Performance

In this chapter, generally valid high-level formulas to estimate the interconnect power consumption and performance are systematically derived. An abstract and yet accurate model, mapping abstract pattern properties to power consumption and performance quantities, is required to derive, and rapidly evaluate, techniques to improve the 3D-interconnect power consumption and performance on higher abstraction levels, which is the main objective of this doctoral research project.

In addition to the main objective of this thesis, the presented high-level formulas enable the estimation of the power requirements and the performance of a system at early design stages, without the need for complex circuit simulations. For this purpose, system simulators are typically used, which are implemented in high-level languages such as *SystemC* [71, 72]. The interconnect architecture of modern SoCs has a throughput of several GB/s. This data rate makes it almost impossible to perform circuit simulations to determine the application-specific power requirements of the interconnects. A *Spectre* circuit simulation for

the transmission of only 3,500 bit-patterns/data-samples with a throughput of 1.1 GB/s over a single global 3×3 TSV array, already results in a run time of about 1,140 s on a *Linux Xeon E5–2630* machine with 256 GB of RAM.[1] In contrast, a high-level estimation only takes a few seconds, independent of the pattern count and rate. For system simulations with data sets containing millions of data words, this translates to a speedup that is significantly higher than 1×10^6, just due to a high-level formula for the interconnect power consumption.

The remainder of this chapter is structured as follows. First, the formulas to estimate the interconnect power consumption and performance are derived in Section 3.1 and Section 3.2, respectively. Afterward, matrix formulations for both formulas are provided in Section 3.3, which enable a systematic derivation and evaluation of optimization techniques in Part III of this thesis. In the following Section 3.4, the derived formulas are validated through circuit simulation for a modern global TSV array. Finally, the chapter is concluded.

Core parts of this chapter are published in the "IEEE Transactions on Very Large Scale Integration (VLSI) Systems" as a regular paper (see [A1]).

3.1. High-Level Formula for the Power Consumption

Formulas to estimate the dynamic power consumption of 2D interconnects are well known (*e.g.* [CA1, 39, 73]).[2] However, previous formulas consider only two adjacent neighbors for each interconnect and are consequently not applicable for 3D interconnects as a TSV in an array arrangement is surrounded by up-to eight adjacent TSVs. In this subsection, a formula that overcomes these limitations is derived.

Figure 3.1 is considered as the starting point to derive the formula. The energy extracted from the driver of the i^{th} interconnect in the k^{th} cycle is expressed as

$$E_{\text{e},i}[k] = \int_{(k-1)T_{\text{clk}}}^{kT_{\text{clk}}} V_{\text{dd}}\, i_i(t)\mathrm{d}t, \tag{3.1}$$

where V_{dd} is the power-supply voltage, $i_i(t)$ is the current that flows through the driver at time t, and T_{clk} is the clock period. If the binary input b_i is zero, the current i_i, and therefore also the energy extracted from the supply, is zero for neglected leakage effects. Since $i_i = \sum_l i_{i,l}$ (Kirchhoff's law), the contribution of each capacitance, $C_{i,j}$, can be analyzed individually in the remainder.

First, the scenario of no temporal misalignment between the signal edges at

[1] All run times reported in this thesis are for the execution of a single thread, a utilization below 5 % for all cores, and more than 80 % available main-memory space.

[2] As shown in [39], leakage effects are negligible for traditional VLSI interconnects. This still holds for TSV-based interconnects, as shown in the evaluation section of this chapter.

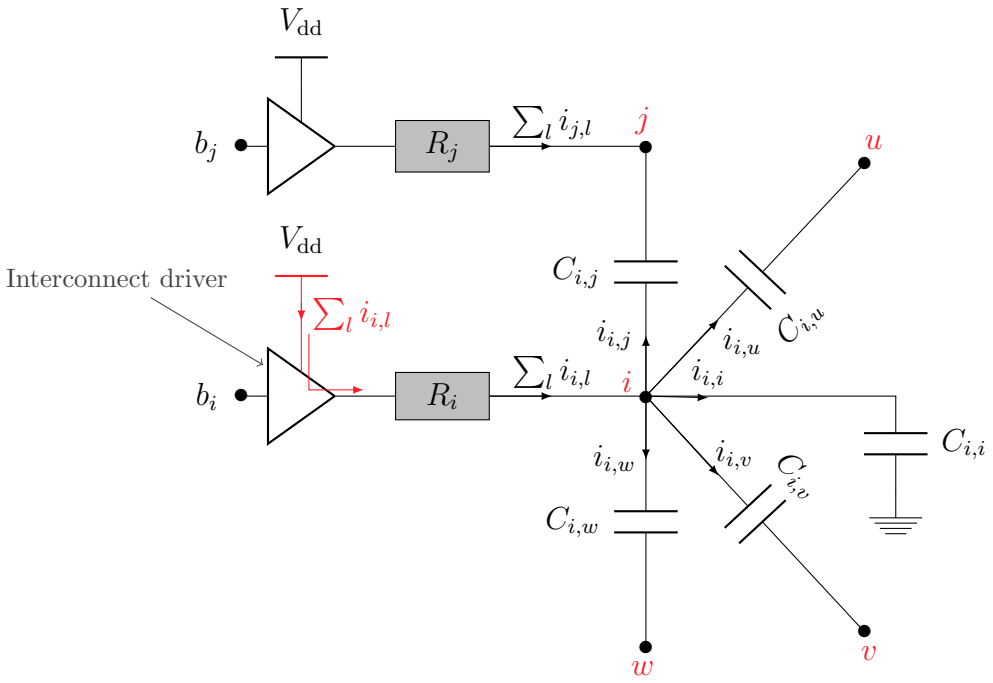


Fig. 3.1: Equivalent circuit of an exemplary interconnect structure used for the derivation of the formulas for the interconnect power consumption.

the input ports of the drivers of the i^{th} and the j^{th} interconnect is considered. In other words, the inputs b_i and b_j switch exactly at the beginning of the cycle (in case of an event on both signals). By considering the current-voltage relation of capacitances, the energy extracted from the i^{th} driver due to the coupling capacitance between the interconnects i and j, $C_{i,j}$, is calculated in this case as follows:

$$\begin{aligned} E_{\mathrm{e},i,j}[k] &= b_i[k] \int_{(k-1)T_{\mathrm{clk}}}^{kT_{\mathrm{clk}}} V_{\mathrm{dd}} C_{i,j} \frac{\mathrm{d}\left(v_i(t) - v_j(t)\right)}{\mathrm{d}t} \mathrm{d}t \\ &= V_{\mathrm{dd}} C_{i,j} b_i[k] (V_{\mathrm{dd}} b_i[k] - V_{\mathrm{dd}} b_j[k] - V_{\mathrm{dd}} b_i[k-1] + V_{\mathrm{dd}} b_j[k-1]) \\ &= V_{\mathrm{dd}}^2 C_{i,j} b_i[k] (\Delta b_i[k] - \Delta b_j[k]), \end{aligned} \tag{3.2}$$

where $v_i(t)$ is the voltage between the i^{th} interconnect and the ground at time t, which is equal to the binary value b_i times V_{dd} after the switching on the interconnect is completed (*i.e.*, at the end of a cycle). $b_i[k]$ and $b_i[k-1]$ are the binary values transmitted over the i^{th} interconnect in the actual, k^{th}, and the previous, $(k-1)^{\mathrm{th}}$, clock cycle, respectively. $\Delta b_i[k]$ is equal to $b_i[k] - b_i[k-1]$ and thus indicates the switching of the logical/binary value on the interconnect. It is equal to 1 for a logical 0 to logical 1 transition, 0 for no transition in the logical value, and -1 for a logical 1 to logical 0 transition. Analogously, one

can calculate the energy extracted over the driver of the j^{th} interconnect due to the same capacitance $C_{i,j}$:

$$E_{\text{e},j,i}[k] = V_{\text{dd}}^2 C_{i,j} b_j[k](\Delta b_j[k] - \Delta b_i[k]). \tag{3.3}$$

Consequently, the total extracted energy because of $C_{i,j}$ is expressed as

$$\begin{aligned} E_{\substack{\text{e},i,j\\ \text{e},j,i}} =& E_{\text{e},i,j} + E_{\text{e},j,i} \\ =& V_{\text{dd}}^2 C_{i,j}(b_i[k] - b_j[k])(\Delta b_i[k] - \Delta b_j[k]). \end{aligned} \tag{3.4}$$

Equation (3.4) does not hold anymore if a temporal misalignment between the edges of the two input signals exists. Temporal misalignment implies here that one of the two inputs, b_i or b_j, switches later than the other in case of an event/edge on both signals in the current cycle. Exemplary, the scenario, in which the signal edges on the i^{th} input b_i are delayed by $T_{\text{edge},i}$ against the edges on b_j, is considered in the following. If $T_{\text{edge},i}$ is greater than the rise/fall time of the input drivers, plus the delay over the interconnects, Equation (3.2) has to be modified to

$$\begin{aligned} E_{\text{e},i,j}[k] = b_i[k{-}1] &\int_{(k-1)T_{\text{clk}}}^{(k-1)T_{\text{clk}}+T_{\text{edge},i}} V_{\text{dd}} C_{i,j} \frac{-\mathrm{d}v_j(t)}{\mathrm{d}t} \mathrm{d}t \\ &+ b_i[k] \int_{(k-1)T_{\text{clk}}+T_{\text{edge},i}}^{kT_{\text{clk}}} V_{\text{dd}} C_{i,j} \frac{\mathrm{d}v_i(t)}{\mathrm{d}t} \mathrm{d}t \\ &= V_{\text{dd}}^2 C_{i,j}(b_i[k]\Delta b_i[k] - b_i[k{-}1]\Delta b_j[k]). \end{aligned} \tag{3.5}$$

$E_{e,j,i}$ can still be calculated with Equation (3.3) if the edges on b_i are delayed. Thus, the total extracted energy because of $C_{i,j}$ is in this case

$$E_{\substack{\text{e},i,j\\ \text{e},j,i}}[k] = V_{\text{dd}}^2 C_{i,j}\big(\Delta b_i[k](b_i[k] - b_j[k]) + \Delta b_j[k](b_j[k] - b_i[k{-}1])\big). \tag{3.6}$$

The formulas simplify if the dissipated energy is considered instead of the extracted energy. On average, the two energy quantities are equal. Thus, it is irrelevant which quantity is used in the following to derive a formula for the mean power consumption. The dissipated energy is the difference between the differential stored energy and the extracted energy:

$$E_{i,j}[k] = E_{\substack{\text{e},i,j\\ \text{e},j,i}}[k] - \Delta E_{\text{stored},i,j}[k], \tag{3.7}$$

where the difference between the energies stored in $C_{i,j}$ after and before the

transitions can be expressed as

$$\Delta E_{\text{stored},i,j} = \frac{1}{2} C_{i,j} \left(V_{i,j}^2[k] - V_{i,j}^2[k-1] \right) \tag{3.8a}$$

$$= \frac{V_{\text{dd}}^2 C_{i,j}}{2} \Big((b_i[k] - b_j[k])^2 - (b_i[k-1] - b_j[k-1])^2 \Big) \tag{3.8b}$$

$$= \frac{V_{\text{dd}}^2 C_{i,j}}{2} \left(b_i[k] - b_j[k] + b_i[k-1] - b_j[k-1] \right) \left(\Delta b_i[k] - \Delta b_j[k] \right). \tag{3.8c}$$

Here, $V_{i,j}[k-1]$ and $V_{i,j}[k]$ are the voltage drops over $C_{i,j}$ before and after the transitions, respectively. Equation (3.8c) can be derived from Equation (3.8b) by applying the third binomial formula. By substituting, Equation (3.8c) and (3.4) in Equation (3.7) a formula to calculate the energy dissipation due to $C_{i,j}$ in case of temporally aligned signal edges is obtained:

$$E_{i,j}[k] = \frac{V_{\text{dd}}^2 C_{i,j}}{2} (\Delta b_i[k] - \Delta b_j[k])^2. \tag{3.9}$$

For misaligned/skewed input signals, Equation (3.4) instead of Equation (3.6) has to be substituted, resulting in

$$E_{i,j}[k] = \frac{V_{\text{dd}}^2 C_{i,j}}{2} (\Delta b_i^2[k] + \Delta b_j^2[k]). \tag{3.10}$$

Analyzing Equation (3.10) reveals that, in the presence of temporal misalignment, it is irrelevant which interconnect changes its level first when the dissipated energy is considered instead of the extracted energy (*i.e.*, the formula is distributive for i and j as $E_{i,j}$ is always equal to $E_{j,i}$). The only important issue is whether the edges show a sufficient temporal misalignment or not. Another advantage of using the dissipated energy is that, generally, we do not have to differentiate between a charging and a discharging capacitance. Consequently, the formulas for the dissipated energy exhibit a significantly lower complexity than the ones for the extracted energy.

By substituting a 0 for $\Delta b_j[k]$ in Equation (3.9) or (3.10), the well-known formula for the dissipated energy due to the self/ground capacitance, $C_{i,i}$, between node i and the (logically stable) ground is obtained:

$$E_{i,i}[k] = \frac{V_{\text{dd}}^2 C_{i,i}}{2} \Delta b_i^2[k]. \tag{3.11}$$

Summarized, the overall dissipated energy in clock cycle k due to an interconnect structure containing n lines, for just perfectly temporal aligned signals

edges, is equal to

$$E[k] = \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} \Big(C_{i,i} \Delta b_i^2[k] + \sum_{j=i+1}^{n} C_{i,j} (\Delta b_i[k] - \Delta b_j[k])^2 \Big) \tag{3.12a}$$

$$= \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} \Big(C_{i,i} \Delta b_i^2[k] + \sum_{j=i+1}^{n} C_{i,j} \big(\Delta b_i^2[k] + \Delta b_j^2[k] - 2 \Delta b_i \Delta b_j[k] \big) \Big) \tag{3.12b}$$

$$= \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} \Big(C_{i,i} \Delta b_i^2[k] + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \left(\Delta b_i^2[k] - \Delta b_i[k] \Delta b_j[k] \right) \Big), \tag{3.12c}$$

From Equation (3.12c), a formula to quantify the energy consumption associated with an individual interconnect, symbolized as E_i, is derived. This formula is, for example, required to identify interconnects that contribute the most to the power consumption and thus strongly demand an optimization.

Defining $E \stackrel{\mathrm{def}}{=} \sum_i E_i$, the following formula is obtained for perfectly temporal aligned edges:

$$E_i[k] = \frac{V_{\mathrm{dd}}^2}{2} \left(C_{i,i} \Delta b_i^2[k] + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \left(\Delta b_i^2[k] - \Delta b_i[k] \Delta b_j[k] \right) \right). \tag{3.13}$$

While the energy consumption due to the self-capacitance of an interconnect, $C_{i,i}$, is determined only by the self switching of the associated input, Δb_i, the power consumption due to a coupling capacitance of the interconnect, $C_{i,j}$, additionally depends on the switching on the logical value on the j^{th} interconnect, Δb_j, if the lines switch temporally aligned.

Compared to the scenario where only the i^{th} interconnect toggles (*i.e.*, $\Delta b_j[k] = 0$), the contribution of capacitance $C_{i,j}$ to E_i is doubled when b_j toggles in the opposite direction (*i.e.*, $\Delta b_i[k] \Delta b_j[k] = -1$), and vanishes if it toggles in the same direction (*i.e.*, $\Delta b_i[k] \Delta b_j[k] = 1$). This effect is commonly referred to as crosstalk or coupling switching of the interconnects, while sometimes also the term Miller effect is used.

The energy dissipated in clock cycle k due to an n-bit interconnect in case of a significant temporal misalignment between the switching times of all input

signal pairs is expressed through Equation (3.10) and Equation (3.11) as

$$E[k] = \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} \left(C_{i,i} \Delta b_i^2[k] + \sum_{j=i+1}^{n} C_{i,j} \big(\Delta b_i^2[k] + \Delta b_j^2[k] \big) \right) \tag{3.14a}$$

$$= \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} \left(C_{i,i} + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \right) \Delta b_i^2[k] \tag{3.14b}$$

$$= \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} C_{i,j} \Delta b_i^2[k]. \tag{3.14c}$$

Hence, for misaligned edges, the energy dissipation associated with the i^{th} interconnect is independent of the switching on the other lines (*i.e.*, no crosstalk effects):

$$E_i[k] = \frac{V_{\mathrm{dd}}^2}{2} \sum_{j=1}^{n} C_{i,j} \Delta b_i^2[k]. \tag{3.15}$$

Effective Capacitance

Introducing the concept of an effective lumped capacitance for each interconnect has greatly simplified the derivation and evaluation of high-level optimization techniques for 2D interconnects in [39]. Thus, the concept of the effective capacitance is extended in this paragraph in a way that it can be reused for 3D interconnects.

The effective capacitance, $C_{\mathrm{eff},i}$, is defined as a single ground capacitance connected to the i^{th} line that, if recharged, results in the same energy consumption as the real circuit considering complex coupling effects. The energy that is dissipated if a ground capacitance C_{g} is recharged is equal to $(V_{\mathrm{dd}}^2 C_{\mathrm{g}})/2$. Thus, the effective capacitance is formally defined as follows.

Definition 3.1.1. Effective capacitance

$$E_i \stackrel{\mathrm{def}}{=} \frac{V_{\mathrm{dd}}^2}{2} C_{\mathrm{eff},i}, \tag{3.16}$$

which implies that

$$E \stackrel{\mathrm{def}}{=} \frac{V_{\mathrm{dd}}^2}{2} \sum_{i=1}^{n} C_{\mathrm{eff},i}. \tag{3.17}$$

Hence, the effective capacitance of the i^{th} interconnect for temporally aligned

edges, derived from Equation (3.13), is expressed as

$$C_{\text{eff},i}[k] = C_{i,i}\Delta b_i^2[k] + \sum_{\substack{j=1 \\ i\neq j}}^{n} C_{i,j}\left(\Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k]\right). \tag{3.18}$$

For completely misaligned edges, the formula for the effective capacitance (derived from Equation (3.15)) is

$$C_{\text{eff},i}[k] = \sum_{j=1}^{n} C_{i,j}\Delta b_i^2[k]. \tag{3.19}$$

In the same way, one can derive a formula for the effective capacitance in the case that some signal pairs switch temporally aligned, while some switch temporally misaligned:

$$C_{\text{eff},i}[k] = \sum_{j=1}^{n} C_{i,j} \begin{cases} \Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k] & \text{for aligned } \Delta b_i \Delta b_j \text{ and } i\neq j \\ \Delta b_i^2[k] & \text{else.} \end{cases} \tag{3.20}$$

This more complex formula is, for example, required to estimate the energy dissipation in scenarios where one-half of the signals switch temporally aligned with the rising clock edge while the other half switches with the falling clock edges, resulting in 50 % temporally misaligned signal pairs.

Power Consumption

In general, the critical concern is not the energy dissipation in a specific clock cycle, but the mean power consumption, represented by P. This quantity can be derived through the expected value of the dissipated energy.

Definition 3.1.2. Expected value

For a stochastic variable X, representing continuous values, with a cumulative density function described by $f_{\text{cd}}(x)$, the expected value is defined as:[3]

$$\mathbb{E}\{X\} = \int_{-\infty}^{\infty} x \cdot f_{\text{cd}}(x)\,\mathrm{d}x. \tag{3.21}$$

The expected value of a stochastic variable X with a finite number of finite outcomes is defined as

$$\mathbb{E}\{X\} = \sum_{i=1}^{m} x_i \cdot p_{x_i}, \tag{3.22}$$

[3]This integral does not necessarily exist.

where $x_1, x_2, \ldots, x_m$ are the possible outcomes occurring with the respective probabilities p_{x_1}, p_{x_2}, ..., p_{x_m}.

Since the power-supply voltage, as well as the capacitance values, are considered as constant, the mean energy dissipation for perfectly temporal aligned signal edges is derived from Equation (3.12) as follows:

$$\bar{E} = \mathbb{E}\left\{\frac{V_{\mathrm{dd}}^2}{2}\sum_{i=1}^{n}\left(C_{i,i}\Delta b_i^2 + \sum_{\substack{j=1\\ i\neq j}}^{n} C_{i,j}\left(\Delta b_i^2 - \Delta b_i \Delta b_j\right)\right)\right\} \tag{3.23a}$$

$$= \frac{V_{\mathrm{dd}}^2}{2}\sum_{i=1}^{n}\left(C_{i,i}\cdot\mathbb{E}\{\Delta b_i^2\} + \sum_{\substack{j=1\\ i\neq j}}^{n} C_{i,j}\left(\mathbb{E}\{\Delta b_i^2\} - \mathbb{E}\{\Delta b_i \Delta b_j\}\right)\right). \tag{3.23b}$$

In the same way, the formula for the mean energy dissipation in case of just completely temporal misaligned signal edges is derived from Equation (3.14):

$$\bar{E} = \frac{V_{\mathrm{dd}}^2}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} C_{i,j}\cdot\mathbb{E}\{\Delta b_i^2\}. \tag{3.24a}$$

From these two formulas, a simple formula is derived which can be used for all cases in which binary input signal pairs switch either perfectly aligned or perfectly misaligned:

$$\bar{E} = \frac{V_{\mathrm{dd}}^2}{2}\sum_{i=1}^{n}\left(C_{i,i}\alpha_i + \sum_{\substack{j=1\\ i\neq j}}^{n} C_{i,j}\left(\alpha_i - \gamma_{i,j}\right)\right). \tag{3.25}$$

In this equation, α_i is the toggle/switching activity of b_i (*i.e.*, $\mathbb{E}\{\Delta b_i^2\}$), which is equal to the probability of a transition in the binary value in a clock cycle. The second statistical parameter $\gamma_{i,j}$ is used to capture the coupling effects between the interconnect pair over which b_i and b_j are transmitted. If there is no temporal misalignment between the signal edges on b_i and b_j,

$$\gamma_{i,j} = \mathbb{E}\{(\Delta b_i \Delta b_j)\}. \tag{3.26}$$

For a sufficient misalignment between the signals b_i and b_j,

$$\gamma_{i,j} = 0. \tag{3.27}$$

$\gamma_{i,j}$ can take any real value in the range of -1 to 1. It is equal to: -1 if the signals b_i and b_j switch at the same time, but in the opposite direction, in every

clock cycle; 1 if the signals switch at the same time, and in the same direction, in every clock cycle; and 0 if the signals never switch at the same time or completely uncorrelated (in terms of the switching directions). Generally, $\gamma_{i,j}$ is bigger than zero if the two signals switch with a higher probability in the same direction than in the opposite direction, and smaller than zero if they are more likely to switch in the opposite direction (for temporally aligned edges). Hence, $\gamma_{i,j}$ is referred to as the switching correlation of the bits in the remainder of this thesis. However, note that a $\gamma_{i,j}$ value does not represent a correlation in the traditional form defined by Pearson.

Introducing the concept of a mean effective capacitance ($\bar{C}_{\text{eff},i}$) for each interconnect—used to estimate the power consumption—simplifies the derivation of the optimization techniques in Part III of this dissertation.

Definition 3.1.3. Mean effective capacitance

$$\bar{E}_i \stackrel{\text{def}}{=} \frac{V_{\text{dd}}^2}{2}\bar{C}_{\text{eff},i} \quad \text{with} \quad \bar{E} \stackrel{\text{def}}{=} \sum_i \bar{E}_i. \tag{3.28}$$

Comparing this definition with Equation (3.25) shows that the mean effective capacitance of the i^{th} interconnect can be expressed as

$$\bar{C}_{\text{eff},i} = C_{i,i}\alpha_i + \sum_{\substack{j=1 \\ i\neq j}}^{n} C_{i,j}\,(\alpha_i - \gamma_{i,j}). \tag{3.29}$$

Since energy is the integral of power over time, the mean power consumption is calculated by means of

$$P = \frac{\bar{E}}{T_{\text{clk}}} = \frac{V_{\text{dd}}^2 f}{2}\sum_{i=1}^{n} \bar{C}_{\text{eff},i}, \tag{3.30}$$

In this equation, f is the frequency used for signal transmission equal to the inverse of the clock period T_{clk}.

3.2. High-Level Formula for the Propagation Delay

The reciprocal of the maximum signal-propagation delay for any line is the common performance metric of an interconnect structure, as it quantifies the maximum rate at which bit-patterns can be transmitted error-free. Typically, the signal propagation delay is defined as the time difference between the point at which the input of the interconnect driver changes its logical value (*i.e.*, the node voltage crosses $V_{\text{dd}}/2$) and the time at which the node at the far end of

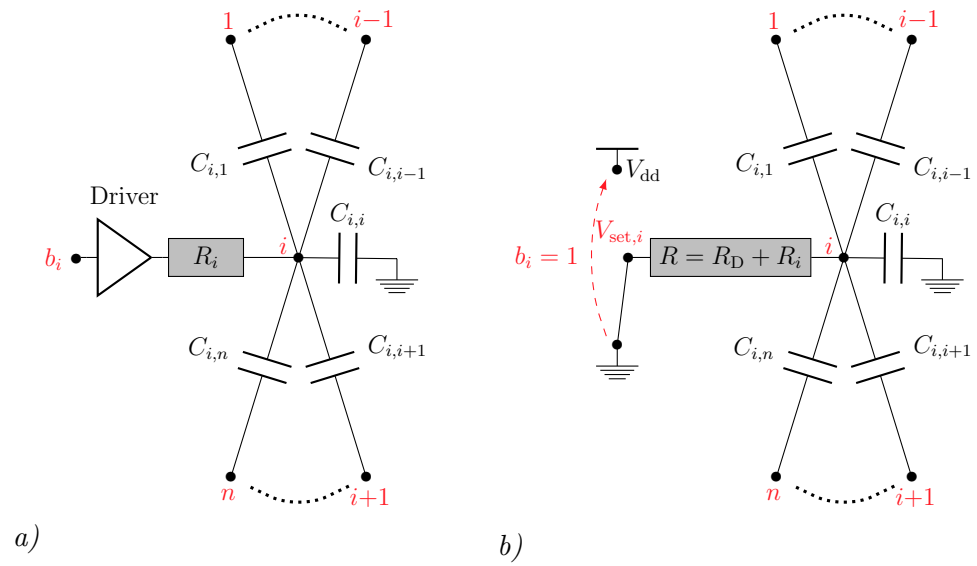


Fig. 3.2: Equivalent circuit of an interconnect structure with: *a)* an actual driver; *b)* a simplified linear driver model for the derivation of a formula for the signal propagation delay.

the interconnect changes its logical value accordingly.

In this section, a formula to estimate this propagation delay is derived by means of the simplified lumped-RC equivalent circuit shown in Figure 3.2b. The actual non-linear driver is modeled as a simple ideal switch and a resistance, R_D, representing the on-resistance of the pull-up or pull-down channel in the conductive state. Furthermore, the binary input value, b_i, is used in the model to control the switch and thereby the set potential, $V_{\mathrm{set},i}$, for the interconnect in the current cycle, k. $V_{\mathrm{set},i}[k]$ is equal to V_dd and 0 V (ground potential) for $b_i[k]$ equal to 1 and 0, respectively.

First, a formula for the output-voltage waveform, $v_i(t')$, of an interconnect as a function of the input switching, $\Delta\vec{b}[k] = [\Delta b_1, \ldots, \Delta b_n]^\mathrm{T}$, is derived. Thereby, t' refers to the time relative to the k^th rising clock edge (*i.e.*, $t' = t - k \cdot T_\mathrm{clk}$). Kirchhoff's current law on the interconnect output node i results in the following equation:

$$\frac{V_{\mathrm{set},i}[k] - v_i(t')}{R} = C_{i,i}\frac{\mathrm{d}v_i(t')}{\mathrm{d}t'} + \sum_{\substack{i=1 \\ j\neq i}}^{n} C_{i,j}\left(\frac{\mathrm{d}v_i(t')}{\mathrm{d}t'} - \frac{\mathrm{d}v_j(t')}{\mathrm{d}t'}\right), \tag{3.31}$$

where R is the sum of the equivalent resistance of the driver, R_D, and the interconnect resistance, R_i. $V_{\mathrm{set},i}$ is the set voltage for the interconnect.

By definition, the line only has a propagation delay when the logical value, transmitted through $V_{\text{set},i}$, is toggling compared to the previous clock cycle (*i.e.*, $\Delta b_i^2[k] = 1$). Thus, this particular scenario is considered in the following. In this case,the voltage change on any other line j, ${}^{\mathrm{d}v_j(t')}/_{\mathrm{d}t}$, can be approximated as $\eta_{i,j} \cdot {}^{\mathrm{d}v_i(t')}/_{\mathrm{d}t}$, with $\eta_{i,j} \in \{-1, 0, 1\}$ [39].[4] For perfectly temporal aligned edges on b_i and b_j, $\eta_{i,j}$ is equal to: 1 if b_j switches in the same direction as b_i; 0 if b_j is stable; and -1 if b_j toggles in the opposite direction as b_i. Hence,

$$\begin{aligned}\frac{\mathrm{d}v_i(t')}{\mathrm{d}t'} - \frac{\mathrm{d}v_j(t')}{\mathrm{d}t'} &= (1 - \eta_{i,j}) \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'} \\ &= (1 - \Delta b_i[k]\Delta b_j[k]) \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'}. \end{aligned} \tag{3.32}$$

By substituting Equation (3.32) in (3.31), the differential equation for solely temporal aligned edges is rewritten as

$$\frac{V_{\text{set},i}[k] - v_i(t')}{R} = \left(C_{i,i} + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \left(1 - \Delta b_i[k]\Delta b_j[k]\right) \right) \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'}. \tag{3.33}$$

For completely temporal misaligned edges on b_i and b_j, v_j is stable while v_i transitions from logical 0 to logical 1, implying ${}^{\mathrm{d}v_j}/_{\mathrm{d}t'} = 0$ in the delay analysis of the i^{th} line. Hence, if all edges occur temporally misaligned, Equation (3.31) simplifies to

$$\begin{aligned}\frac{V_{\text{set},i}[k] - v_i(t')}{R} &= \left(C_{i,i} + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \right) \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'} \\ &= \sum_{j=1}^{n} C_{i,j} \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'}. \end{aligned} \tag{3.34}$$

Since the formulas were derived under the assumption that $b_i^2[k]$ is equal to 1, Equation (3.33) can be rewritten as follows:

$$\frac{V_{\text{set},i}[k] - v_i(t')}{R} = \left(C_{i,i}\Delta b_i^2[k] + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \left(\Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k]\right) \right) \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'}, \tag{3.35}$$

[4] This assumption perfectly holds for drivers with perfectly matched rising and falling transitions for the individual interconnects.

In the same way, Equation (3.34) is transferred to

$$\frac{V_{\mathrm{set},i}[k] - v_i(t')}{R} = \sum_{j=1}^{n} C_{i,j} \Delta b_i^2[k] \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'}. \tag{3.36}$$

Comparing these two formulas with Equation (3.18) and (3.19), shows that, for both cases, aligned and misaligned signal edges, the differential equation can be simplified by using the concept of the effective capacitance:[5]

$$\frac{V_{\mathrm{set},i}[k] - v_i(t')}{R} = C_{\mathrm{eff},i}[k] \frac{\mathrm{d}v_i(t')}{\mathrm{d}t'}. \tag{3.37}$$

In the following, the Laplace transform, represented by $\mathcal{L}\{\}$, is used to solve this differential equation. Multiplying, both sides with R and then transforming the equation into the frequency domain results in

$$\frac{V_{\mathrm{set},i}[k]}{s} - \mathcal{L}\{v_i\}(s) = RC_{\mathrm{eff},i} \left(s\mathcal{L}\{v_i\}(s) - v_i(0) \right), \tag{3.38}$$

where $v_i(0)$ is the potential of the output node at the beginning of the clock cycle ($t' = 0$) equal to $b_i[k-1]V_{\mathrm{dd}}$. Furthermore, the set voltage can be expressed as $b_i[k]V_{\mathrm{dd}}$. Thus, the equation can be modified to

$$\mathcal{L}\{v_i\}(s) = \frac{b_i[k]V_{\mathrm{dd}} + sRC_{\mathrm{eff},i}[k]b_i[k-1]V_{\mathrm{dd}}}{s(1 + RC_{\mathrm{eff},i}s)}. \tag{3.39}$$

Finally, the inverse Laplace transform is applied to obtain $v_i(t')$:

$$v_i(t') = V_{\mathrm{dd}} \left(b_i[k] - \Delta b_i[k] e^{-\frac{t'}{RC_{\mathrm{eff},i}[k]}} \right). \tag{3.40}$$

For a logical 0 to logical 1 transition on the interconnect (*i.e.*, $\Delta b_i[k] = 1$) as well as for a logical 1 to logical 0 transition (*i.e.*, $\Delta b_i[k] = -1$), the rise and fall time for the node-voltage $v_i(t')$ to reach χV_{dd} and $1 - \chi V_{\mathrm{dd}}$, respectively, can be derived from Equation (3.40) as

$$T_{\mathrm{delay},i}(\chi)[k] = -\ln(1-\chi) RC_{\mathrm{eff},i}[k]. \tag{3.41}$$

The time to propagate the 50 % switching point is $T_{\mathrm{delay},i}(0.5)[k]$ equal to $0.69RC_{\mathrm{eff},i}[k]$. For the derivation of this formula, an ideal/simplified driver model was considered. To partially account for non-ideal driver effects, a driver-specific constant $T_{\mathrm{D},0}$ has to be added to the delay. Hence, the signal

[5] The same formula can be derived for the case in which temporally aligned and misaligned signal edges occur simultaneously.

propagation delay to reach the 50 % switching point is

$$T_{\mathrm{pd},i}[k] = 0.69RC_{\mathrm{eff},i}[k] + T_{\mathrm{D},0}\Delta b_i^2[k]. \tag{3.42}$$

Although this formula is derived under the assumption that b_i toggles, it is even valid for the scenario that b_i is stable. Clearly, in this case, the propagation delay is zero, which is equal to the output of the formula for $\Delta b_i[k]$ equal to 0 (resulting in $C_{\mathrm{eff},i}[k]$ equal to 0).

The performance is quantified by the reciprocal of the maximum propagation delay for all cycles and interconnects, as it equal to the maximum rate at which bit patterns can be transmitted over the interconnect structure error-free. This delay metric, represented by $\hat{T}_{\mathrm{pd}}$ in this thesis, can be estimated through the following equation:

$$\hat{T}_{\mathrm{pd}} = \max_{k,i}\left(T_{\mathrm{pd},i}[k]\right) = R\hat{C}_{\mathrm{eff}} + T_{\mathrm{D},0}, \tag{3.43}$$

assuming that one line eventually toggles, resulting in a non-zero entropy for the transmitted data (*i.e.*, $\max_{k,i}(\Delta b_i^2[k]) = 1$). In this equation, $\hat{C}_{\mathrm{eff}}$ is the maximum possible effective-capacitance value of all interconnects for all possible bit-pattern transitions.

In conclusion, not only the energy consumption of an interconnect is proportional to the pattern-dependent effective capacitance, but also the signal-propagation delay is. Thus, decreasing the effective capacitance values can improve the performance as well as the power consumption of VLSI interconnects. This correlation between the metrics enables us to design high-level techniques that effectively improve the 3D-interconnect power consumption and performance by optimizing the transmitted bit patterns.

3.3. Matrix Formulations

Matrix formulations for the derived formulas to estimate the power consumption and the propagation delay are presented in this section. These formulas do not only simplify the notation and allow for a more straightforward model implementation with *Matlab* or the *Python* package *Numpy*, but they also are the key enabler for the optimization techniques presented and evaluated in Part III of this thesis.

Through the sum of the mean effective capacitances—directly proportional to the dynamic power consumption (Equation (3.30))—the interconnects can be optimized toward a low power-consumption due to the pattern-dependent nature of this metric. The sum of the mean effective capacitances can be

expressed through the Frobenius inner product of two matrices:[6]

$$\sum_{i=1}^{n} \bar{C}_{\text{eff},i} = \sum_{i=1}^{n} \left(C_{i,i}\alpha_i + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j} \left(\alpha_i - \gamma_{i,j}\right) \right) = \langle \mathbf{S}_{\mathbb{E}}, \mathbf{C} \rangle. \tag{3.44}$$

Here, $\mathbf{C}$ is the capacitance matrix with capacitance $C_{i,j}$ on entry (i, j). Matrix $\mathbf{S}_{\mathbb{E}}$ contains the switching statistics. In detail, the entries of this switching matrix depend on the switching activities and correlations of the binary values transmitted over the interconnects:

$$S_{\mathbb{E},i,j} = \begin{cases} \alpha_i & \text{for } i = j \\ \alpha_i - \gamma_{i,j} & \text{for } i \neq j. \end{cases} \tag{3.45}$$

Hence, the mean power consumption of an interconnect structure can also be formulated using a matrix instead of a sum notation:

$$P = \frac{V_{\text{dd}}^2 f}{2} \langle \mathbf{S}_{\mathbb{E}}, \mathbf{C} \rangle. \tag{3.46}$$

In the formula for the cycle-based interconnect propagation delay (*i.e.*, Equation (3.42)), the effective capacitance is the pattern-dependent metric that allows for optimization at higher abstraction levels. Instead of using Equation (3.20) n times to calculate the effective capacitances of all interconnects individually, the vector containing all effective capacitances can be expressed in a single matrix notation:

$$\vec{C}_{\text{eff}}[k] = \text{diag}(\mathbf{S}[k] \cdot \mathbf{C}), \tag{3.47}$$

where the i^{th} vector entry is equal to the effective capacitance of the i^{th} line, $C_{\text{eff},i}[k]$. "diag()" is a function returning the diagonal of a matrix as a vector. The switching matrix $\mathbf{S}[k]$ is defined as follows:

$$S_{i,j}[k] = \begin{cases} \Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k] & \text{for aligned } \Delta b_i \Delta b_j \text{ and } i \neq j \\ \Delta b_i^2[k] & \text{else.} \end{cases} \tag{3.48}$$

Consequently, the overall maximum propagation delay for all interconnects—and thus the performance of the whole interconnect structure—can be estimated

[6] The Frobenius inner product, represented by $\langle\rangle$, of two real-valued matrices $\mathbf{A}$ and $\mathbf{B}$ is equal to the sum of the element-wise multiplication ($\sum_{i,j} A_{i,j} B_{i,j}$).

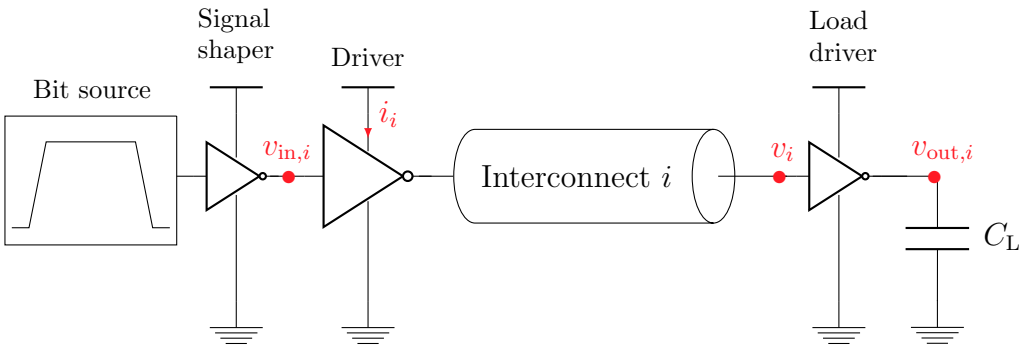


Fig. 3.3: Setup to measure the power consumption and propagation delay of interconnects through circuit simulations.

by means of the following matrix equation:

$$\begin{aligned}\hat{T}_{\text{pd}} &= R\hat{C}_{\text{eff}} + T_{\text{D},0} \\ &= R\left(\max_k \left(\text{diag}(\mathbf{S}[k] \cdot \mathbf{C})\right)\right) + T_{\text{D},0}.\end{aligned} \tag{3.49}$$

The two derived matrix formulations for the mean-sum and the cycle-based effective capacitance values—*i.e.*, Equation (3.44) and (3.47)—are of particular importance in this thesis as they are extensively used for most optimization techniques presented in Part III of this thesis.

3.4. Evaluation

The previously derived formulas for the interconnect power consumption and performance are validated in this section through a comparison with the results obtained for precise, but computationally complex, circuit simulations. For this purpose, first, a setup is constructed, which allows determining the pattern-dependent power consumption and propagation delay of interconnects by means of circuit simulations with *Cadence Spectre*.

The setup for each individual line of the interconnect structure under analysis is illustrated in Figure 3.3. Each interconnect is driven by an inverter made up of one n-channel and one p-channel MOSFET stemming from the 22-nm PTM. The W/L_{min} ratios of the two transistor channels define the strength of the interconnect drivers. For the analysis in this chapter, W/L_{min} of the n-channel and the p-channel MOSFETs are 12 and 20, respectively. At the far end, each TSV terminates in a smaller 22-nm PTM inverter acting as a driver for a capacitive load of 1 fF. The transistor channels of this load driver have widths that are by a factor of 3 × smaller than the respective channel widths of the interconnect driver.

Bit sources ($V_{\mathrm{dd}} = 1\,\mathrm{V}$) with a transition time of 10 ps and a bit duration of 1 ns (*i.e.*, $f_{\mathrm{clk}} = 1\,\mathrm{GHz}$) are used to generate the stimuli for the circuit simulations. The transmitted bit sequence and the delay of the signal edges, relative to the rising clock edges (at $k \cdot T_{\mathrm{clk}}$, with $k \in \mathbb{N}$), can be arbitrarily defined for each line. To achieve a realistic shape of the input-voltage waveforms of the TSVs, $v_{\mathrm{in},i}$, additional inverters are added, which are sized like the ones charging the capacitive load at the interconnect termination.

The energy consumption of the i^{th} interconnect for a specific time frame is measured by integrating the current that flows into the respective driver, i_i, over the considered time window. By dividing this energy consumption by the duration of the time frame, the mean power consumption is obtained. The difference between the simulation time at which the driver input voltage, $v_{\mathrm{in},i}$, crosses $V_{\mathrm{dd}}/2$, and the simulation time at which the interconnect output voltage, v_i, crosses $V_{\mathrm{dd}}/2$ accordingly is the measured signal-propagation delay of the i^{th} interconnect in the respective cycle.

In the analysis presented in this section, a 3×3 TSV array with a TSV radius and minimum pitch of 1 μm and 4 μm, respectively, is considered as the interconnect structure. These geometrical TSV dimensions correspond with the minimum global TSV dimensions for the year 2018, predicted by the ITRS [41]. A TSV length/depth of 50 μm is chosen to model a commonly thinned substrate. To extensively validate the derived formulas, all possible switching scenarios for the TSV located in the array middle (*i.e.*, TSV_5) are investigated. Furthermore, each scenario is analyzed twice: Once for perfectly temporal aligned signal edges, and once for completely temporal misaligned edges on TSV_5 compared to the other TSVs. To analyze temporally misaligned edges, the input of the driver of TSV_5 switches by 0.5 ns earlier/later than the inputs of the remaining drivers.

By means of the *Q3D Extractor* and the parameterisable 3D model presented in Subsection 2.4.1, the TSV parasitics are obtained. A mean voltage of 0.5 V ($V_{\mathrm{dd}}/2$) on each TSV is considered for the parasitic extraction since 0 and 1 bits are equally distributed in the analyzed pattern set. For the setup in this analysis, the significant frequency for TSV parasitic extraction is 6 Ghz. The resulting 3π-RLC equivalent *Spectre* circuit is integrated into the setup to model the TSVs in the circuit simulations. Furthermore, contact resistances of 100 Ω are added between the drivers and the TSVs in accordance with [74].

Besides the equivalent circuit, the capacitance matrix—required to perform the high-level estimation—is also generated out of the *Q3D Extractor* results. The effective TSV load capacitance due to the drivers (*ca.* 1 fF) is added to the self/ground capacitance of each TSV (*i.e.*, the diagonal entries of the capacitance matrix). Afterward, by analyzing the abstract bit-level switching of the considered pattern set, the power consumption is predicted with the derived formula. To quantify the accuracy of the power formula for an extensive set of

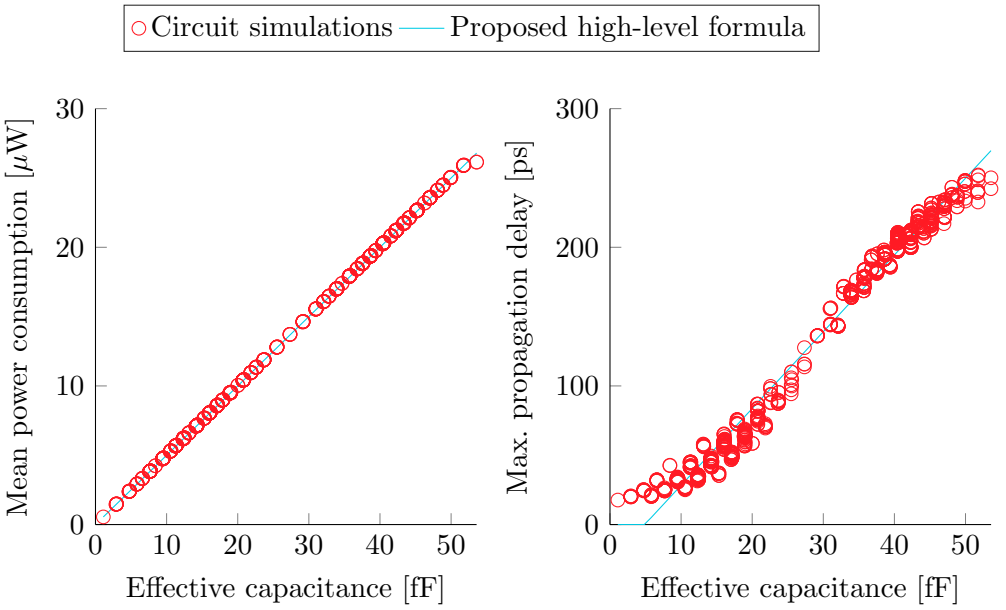


Fig. 3.4: Mean power consumption and maximum propagation delay of a TSV located in the middle of a 3×3 array over the effective capacitance according to circuit simulations and the derived high-level formulas.

switching scenarios, not only the mean power consumption of the TSV over all cycles is analyzed, but the individual mean power-consumption quantities for each possible effective capacitance value in the respective cycles.

The driver-dependent values R and $T_{\mathrm{D},0}$ need to be determined once before the propagation delay can be estimated through Equation (3.42). Here, a simple linear least-squares fit for the propagation delay of the middle TSV_5 as a function of the effective capacitance $C_{\mathrm{eff},5}$ is performed. Thereby, rising edges on the TSVs are considered as they result in a higher propagation delay due to the lower channel conductivity of the pull-up path (*i.e.*, *p*-channel MOSFET). The maximum propagation delay for each possible effective-capacitance value of TSV_5 is subsequently analyzed to quantify the accuracy of the derived formula for the performance estimation.

In Figure 3.4, the results of the analysis are plotted. As shown in the figure, the derived high-level formula enables a close to perfect estimation of the power consumption, despite the neglected inductance, internal-driver, and leakage effects. Thus, these effects are negligible for the TSV power consumption like they are for the metal-wire power consumption [39]. The overall root-mean-square error (RMSE) for the power estimation, normalized by the overall mean power consumption, is as low as 0.4 %. Only for larger effective capacitances, the model shows a slightly noticeable error due to the intrinsic misalignment effect for transitions that entail a large power consumption [CP1]. The main

reason for the intrinsic misalignment, here, is a mismatch in the equivalent on-resistances of the pull-up and pull-down path of the TSV driver due to the moderate increase in the width of the p-channel compared to the n-channel, which does not fully compensate for the lower hole mobility. Nevertheless, the normalized maximum absolute error (MAE) is 2.5 % for all estimated power quantities. Despite the high accuracy, the high-level estimation, performed with *Matlab*, requires a 115,000 times lower execution time than the circuit simulations on the *Linux Xeon E5–2630* machine.

For the estimation of the propagation delay, the derived formula results in an overall normalized root-mean-square error (NRMSE) of 3.9 %. Thus, the derived formula for the propagation delay is not as accurate as the one for the power consumption. Especially for very small effective-capacitance (and thus delay) values, the estimates are not reliable. Nevertheless, the strong general correlation between the effective capacitance, the power consumption, and the propagation delay is validated by the results. Furthermore, the performance is either way determined only by the maximum occurring delay. Considering only the switching scenarios that result in a propagation delay bigger than 50 % of the maximum possible one, the formula estimates all delay values with a relative MAE in the 10 % range. Thus, the accuracy of the derived delay formula is sufficient for the scope of this work as none of the later proposed optimization technique increases the interconnect performance by more than a factor of 2 ×.

The lower accuracy for the delay estimation is mainly due to the simplified driver model. Actually, the driver's equivalent pull-up and pull-down resistance R_{D}, as well as the offset factor $T_{\mathrm{D},0}$, depend on the size of the capacitive load (here, $C_{\mathrm{eff},i}$). A more accurate piece-wise-linear model for R_{D} and $T_{\mathrm{D},0}$ as a function of the capacitive load, could be extracted from the non-linear delay model (NLDM) liberty file of the used standard cell technology. Afterward, a precise delay estimation could be performed even for switching scenarios, which result in a particularly low propagation delay. Moreover, inductance effects have a minor impact on the propagation delay and could be considered to increase the accuracy of the delay formula further. However, including complex driver or inductance effects is not desired for the objective of this thesis as it would profoundly complicate the systematic derivation of optimization techniques without revealing a significantly higher optimization potential.

In summary, the results validate the derived formulas for the interconnect metrics. The derived formula for the power consumption is close to perfect, while the one for the propagation delay should only be used for performance (*i.e.*, maximum-delay) estimation as long as a non-linear driver model does not extend it.

3.5. Conclusion

In this chapter, abstract formulas for the power consumption and the signal propagation delay (required for a performance estimation) of 3D interconnects were derived systematically. Both formulas were not only derived for the case of perfectly temporal aligned signal edges on the interconnects, but also for the case of a substantial temporal misalignment (skew) between the signal edges. An important finding is that the derived formulas reveal a strong correlation between the interconnect power consumption and the interconnect performance through the pattern-dependent effective capacitance. This fact theoretically allows for an optimization of both metrics on higher abstraction levels.

The derived formulas have been validated against precise circuit simulations, considering a modern 3×3 array of global TSVs and 22-nm drivers. For both cases, temporally aligned and misaligned edges, as well as all possible pattern transitions, the power consumption is estimated with the derived formula with a normalized RMSE and MAE as low as 0.4 % and 2.5 %, respectively. A propagation-delay estimation with the proposed formula is not as accurate as the power estimation (NRMSE 3.9 %). However, for the purpose of this doctoral research project (*i.e.*, the estimation of the pattern-dependent performance), the formula still provides a more than sufficient accuracy.

The presented formulas for the 3D-interconnect power consumption and performance are essential for the derivation and fast evaluation of the optimization techniques presented in Part III of this thesis. However, for a power and performance estimation with the derived formulas, knowledge about the interconnect capacitances is required. Moreover, the bit-level statistics of the transmitted data are required for a precise power estimation. Hence, methods to precisely estimate the 3D-interconnect capacitances and the bit-level statistics are discussed in the following two chapters of this thesis.

CHAPTER 4

High-Level Estimation of the 3D-Interconnect Capacitances

In the previous Chapter 3, formulas were presented, which allow for a fast estimation of the power consumption and the performance of 3D interconnects on higher abstraction levels. However, these formulas require a knowledge of the 3D-interconnect capacitances.

The interconnect capacitances can be extracted with EM solvers on 3D models, as done in the previous Chapter 3. Nevertheless, an abstract and yet universally valid model for the capacitances is still required for mainly two reasons. First, EM solvers have significant run times, and their results are not scaleable. If the line count of the interconnect structure changes, the capacitances have to be extracted again. Thus, in large systems with hundreds of global interconnect structures, possibly all having a different shape, the capacitances would have to be extracted for each bundle separately. As an illustration, for a single 8×8 TSV array, already 2,080 capacitance values have to be extracted for a full parasitic extraction based on an EM solver. The second reason why an abstract capacitance model is required is that it

is not possible to derive optimization techniques that are generally efficient for most 3D-interconnect structures without having an abstract knowledge about regularities in the capacitances. Moreover, the model should encapsulate complex physical phenomena (which affect the capacitance quantities) in an abstract way. Thereby, optimization techniques based on higher abstraction levels can effectively exploit these phenomena.

Previous works have proposed a set of equivalent circuits for TSV and metal-wire arrangements, which partially overcome the need for parasitic extractions by means of an EM solver [11,16,75–77]. However, all existing equivalent circuits for TSVs are too complex to work with the formulas derived in the previous chapter, as the equivalent circuits do not allow to obtain lumped coupling-capacitance and ground-capacitance quantities for TSV arrays. Hence, existing equivalent circuits for TSV arrays are only usable for a power and performance analysis in combination with computational complex circuit simulations.

In contrast, the commonly used metal-wire equivalent circuit exhibits such a low complexity that its parameters can be easily used to estimate the power consumption and performance of an arbitrary metal-wire bus through the derived formulas [39, 77]. Moreover, the capacitance model of the equivalent circuit has proven over many years to be very practical for the derivation of high-level techniques, which drastically optimize the power consumption and performance of metal wires [CA1, 39, 78–80].

Thus, a capacitance model for TSV arrays that is similar to the one for metal wires was used to systematically derive existing coding techniques that aim for an improvement in the TSV performance [21–23]. This capacitance model was derived by merely extending the model for metal wires into the third dimension (*i.e.*, increasing the number of adjacent neighbors per line from two to eight). However, with this approach, TSV-specific physical phenomena that affect the capacitances, such as the TSV MOS effect or the electric-field-sharing effect [11, 16, 81], are not captured in the capacitance model at all. To capture these effects in the capacitance model is of particular importance for two reasons. First, the degraded electrical-field-sharing effect at the edges of a TSV array typically results in an increase in the maximum capacitance value by over 45 %. Second, the MOS effect results in capacitance values that vary with the bit-level properties of the transmitted data by over 25 %.

Thus, there is a strong need to extend the previously used abstract TSV capacitance model by newly arising physical phenomena. Such an extended model is presented in this chapter of the present thesis. The proposed capacitance model exhibits a low complexity, is scaleable, and yet provides high accuracy.

An evaluation shows that using the proposed capacitance model for a high-level estimation of the power consumption of modern TSV arrangements for random transmitted bit patterns results in a precise estimation of the power consumption with an error as low as 2.6 %. In contrast, the previously used

model results in an underestimation of the power requirements by as much as 21.1 % for the same pattern set. Furthermore, the previous model results in an under-estimation of the propagation delay of edge TSVs by 25.1 % due to the neglected electrical-field sharing. In contrast, applying the proposed capacitance model instead results in a precise estimation of the edge-TSV delay as the error is only 4.2 %. Hence, the proposed capacitance model enables a precise TSV power-consumption and performance estimation without the need for computationally expensive full-chip parasitic extraction.

The rest of this chapter is structured as follows. First, the existing abstract capacitance models for metal wires and TSV arrays are reviewed in Section 4.1. In the following Section 4.2, TSV-specific physical phenomena and their impact on the capacitances are outlined. Afterward, the proposed capacitance model is presented in Section 4.3. The coefficients and the accuracy of the proposed model are quantified in the first part of the evaluation section, Subsection 4.4.1. Furthermore, a comparison of the accuracy of the proposed and the previously used capacitance model is drawn. In Subsection 4.4.2, the accuracy of a high-level TSV power-consumption and performance estimation based on the proposed and the previous TSV capacitance model is investigated through experimental results. Finally, the chapter is concluded.

Core parts of this chapter are published in the journal series "IEEE Transactions on Very Large Scale Integration (VLSI) Systems" and "Elsevier Integration" as regular articles (see [A1, A2]).

4.1. Related Work—Existing Capacitance Models

In this section, the previous capacitance models—used to estimate the interconnect power consumption and performance on higher abstraction levels—are reviewed. First, the model for metal wires is summarized. Subsequently, the model that was previously used for TSVs is discussed.

The scaleable capacitance model used for metal-wire power-performance estimation and optimization is illustrated in Figure 4.1. Algorithm 1 in Appendix B of this thesis is a pseudo code to generate the according capacitance matrix as it is required for the proposed high-level formulas. Between every adjacent metal-wire pair exists a coupling capacitance of size $C_{\mathrm{mw,c}}$, and every metal wire has a ground capacitance of size $C_{\mathrm{mw,g}}$ [77]. The overall ground capacitance of a metal wire is equal to the sum of $C_{\mathrm{mw,g}}$ and the input capacitance of the interconnect load. Capacitances between non-directly adjacent wires are negligible as interjacent conductors almost entirely shield the electric field (Faraday-cage effect) [77]. Both capacitance values, $C_{\mathrm{mw,c}}$ and $C_{\mathrm{mw,g}}$, increase linearly with the wire length. Thus, for a constant wire spacing and width, only two capacitance values, given per unit length, are needed to model all metal-wire buses of arbitrary length and line count. These capacitances

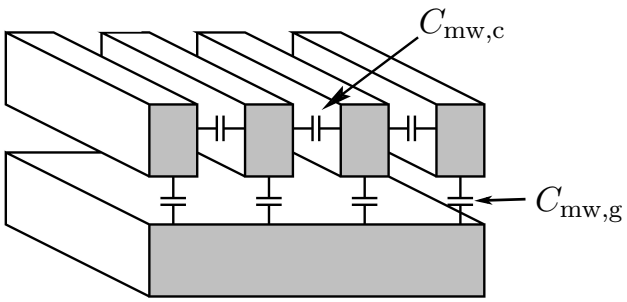


Fig. 4.1: Capacitances of metal wires.

can be obtained through EM solvers or analytical formulas such as the ones presented in [77]. With ongoing technology scaling—resulting in denser-spaced wires—the coupling capacitances in a metal-wire bus dominate more and more over the ground capacitances so that $C_{\mathrm{mw,g}}$ becomes negligible [39].

Despite its simplicity, the existing metal-wire capacitance model has proven over decades to be accurate for the estimation of the power consumption and performance of metal wires. Consequently, the accuracy of this model is not further investigated in this thesis. Furthermore, the simplicity and the scalability of the metal-wire capacitance model enabled the derivation of an extensive set of universally valid optimization techniques that effectively improve the power consumption and performance of metal-wire buses [CA1, 39].

This fact likely was the main driver for previous works to merely extend the existing capacitance model for metal wires into the third dimension for the derivation of high-level techniques aiming to optimize the TSV-array performance [21–23]. Furthermore, this enabled the reuse of an existing optimization approach for 2D ICs, namely crosstalk-avoidance coding, for TSV-based 3D integration.

The previously used TSV capacitance model is illustrated in Figure 4.2. Algorithm 2 in Appendix B of this thesis is a pseudo code for the generation of the according capacitance matrix. A TSV-based interconnect is surrounded by up to eight directly (*i.e.*, horizontally or vertically) and diagonally adjacent TSVs. Like in the model for metal wires, capacitances between non-adjacent interconnects are neglected in the existing TSV capacitance model. One parameter, $C_{\mathrm{n,prev}}$, is used to model all capacitances between directly adjacent TSVs. Another one, $C_{\mathrm{d,prev}}$, is used to model the capacitances between all diagonally adjacent TSVs. Due to the increased pitch, the coupling-capacitance value for diagonally adjacent TSVs, $C_{\mathrm{d,prev}}$, is smaller than $C_{\mathrm{n,prev}}$. In detail, the following

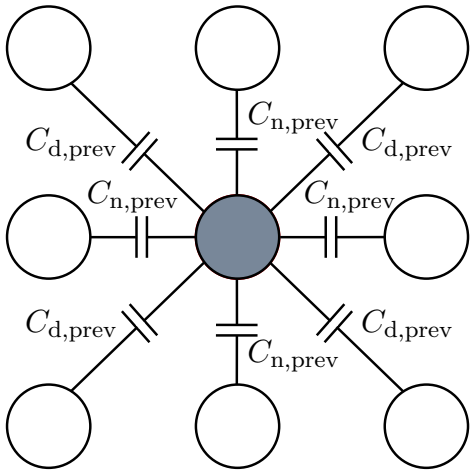


Fig. 4.2: Capacitances of a TSV in an array arrangement according to the previously used capacitance model.

relationship was assumed for the two capacitance values [22, 23]:

$$C_{\mathrm{d,prev}} = \frac{C_{\mathrm{n,prev}}}{4}. \tag{4.1}$$

This relationship was determined in previous work by extracting the two capacitance values for one exemplary TSV-array structure using an EM solver [22]. In the existing TSV capacitance model, all self/ground capacitances are neglected since eight TSVs surrounding a TSV form a strong Faraday cage. Thus, as the metal-wire capacitance model, the existing TSV capacitance model only consists of two parameters. These parameters are the coupling capacitance for directly adjacent TSV pairs, $C_{\mathrm{n,prev}}$, and the coupling capacitance for diagonally adjacent TSV pairs, $C_{\mathrm{d,prev}}$.

4.2. Edge and MOS Effects on the TSV Capacitances

The conductive substrate TSVs traverse, combined with their 3D arrangement, entails coupling effects that do not occur for planar metal wires traversing dielectric layers. These effects are categorized as TSV MOS and edge effects in this work. Both effect types and their general impact on the TSV capacitances are outlined in the remainder of this section.

4.2.1. MOS Effect

The MOS effect and its implications on the electrical characteristics of TSV structures have been investigated on the circuit level in a wide range of previous publications (*e.g.*, [11, 13–16, 82]). Thus, the TSV MOS effect is only briefly summarized in the following.

A TSV, its surrounding dielectric, and the conductive substrate form a metal-oxide-semiconductor (MOS) junction (see Figure 2.9 on Page 27). Consequently, TSVs are surrounded in the substrate by depletion regions [82]. The width of the depletion region surrounding a TSV generally depends on the voltage between the TSV conductor and the substrate. However, the width of the depletion region does not instantaneously change with the voltage on the TSV. Instead, it reaches a stable state depending on the mean TSV voltage over several hundred, or even thousands, of clock cycles. The reason for this behavior is that the substrate is relatively low doped, why the generation/recombination of inversion carriers, as well as the charging/discharging of interface states, cannot follow the fast voltage changes in digital signals [11].

Typical p-doped substrates are grounded and have a negative Flatband voltage [11].[1] In this case, a depletion region always exists around TSVs carrying digital signals (*i.e.*, mean voltage between 0 V and V_{dd}). The width of a depletion region increases with the mean voltage on the TSV, which further isolates the TSV conductor from the conductive substrate. Since the depletion region follows the "medium-frequency" curve for digital signals [11], at a specific mean TSV-to-substrate voltage, the depletion region reaches its maximum width (*i.e.*, no deep depletion or inversion). Thus, any further increase in the mean voltage beyond this point does not affect the depletion-region width.

The parameterisable 3D model of a TSV array, presented in Subsection 2.4.1, is used to outline the effect of the varying depletion-region widths on the capacitances of modern global TSVs (*i.e.*, $r_{\mathrm{tsv}} = 1\,\mu\mathrm{m}$, $d_{\mathrm{min}} = 4\,\mu\mathrm{m}$, $l_{\mathrm{tsv}} = 50\,\mu\mathrm{m}$, and $f_{\mathrm{s}} = 6\,\mathrm{GHz}$). In detail, the size of the coupling capacitance between the TSV in the middle of a 5×5 array and a directly adjacent neighbor TSV (modeled by $C_{\mathrm{n,prev}}$ in the previous capacitance model), is extracted with the *Q3D Extractor* for different mean voltages on all TSVs.[2]

In Figure 4.3, the results are illustrated. The analysis shows that the capacitance decreases with an ongoing increase in the mean voltages on the TSVs. However, from the point where the depletion-region widths do not further change with an increase in the mean voltages, also no further change in the capacitance occurs. However, the maximum possible TSV voltage is either-way bounded by the power-supply voltage, V_{dd}, which, in most modern technologies,

[1] If the TSV-to-substrate voltage exceeds the flatband voltage, a depletion region exists.

[2] The mathematical method to determine the varying depletion-region widths for the 3D model, as a function of the mean TSV voltages, is reviewed in Appendix C of this thesis.

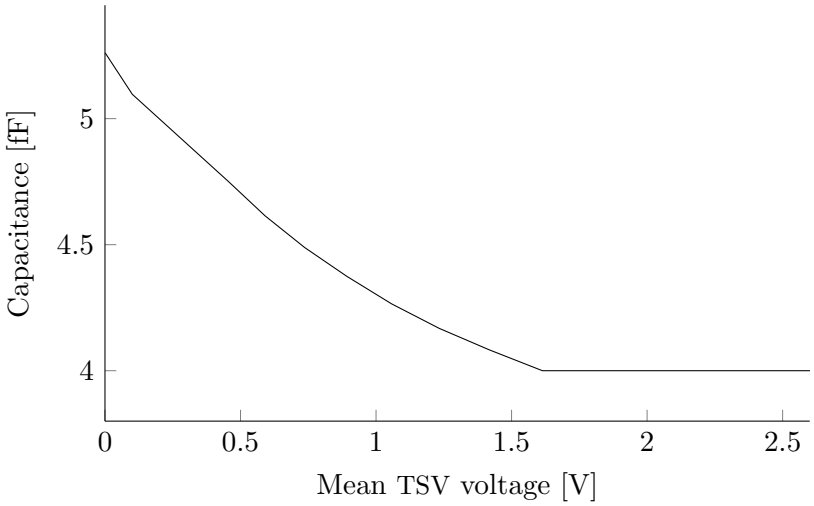


Fig. 4.3: Size of the coupling capacitance between two directly adjacent TSVs over the mean voltage on the TSVs for a p-doped substrate and a TSV radius, minimum pitch, and depth of $1\,\mu$m, $4\,\mu$m, and $50\,\mu$m, respectively.

is lower than the voltage at which a depletion region reaches its maximum width (in this example, about 1.6 V). For the analyzed modern TSV geometries and a power-supply voltage of 1 V, the MOS effect can affect the capacitance value by more than 25 %. This fact reveals that a high-level TSV capacitance model must consider dynamic instead of fixed capacitance quantities.

Finally, note that the width of a depletion region decreases, instead of increases, with an increasing mean voltage on the related TSV for n-doped substrates, which are biased at V_{dd} [11]. Thus, for n-doped substrates, the capacitances increase instead of decrease with increasing mean TSV voltages. However, since common substrates are p-doped, n-doped substrates are not investigated in-depth in this thesis. Nevertheless, the capacitance model presented in this chapter is also valid for n-doped substrates.

4.2.2. Edge Effects

Most previous works on TSV parasitic extraction focus on arrangements of only two TSVs. Hence, the already carried out research on effects that occur at the edges of TSV-array arrangements (referred to as edge effects throughout this thesis) is somewhat limited, even on the lower levels of abstraction. Therefore, the parameterisable TSV-array model from Subsection 2.4.1 is used to outline the TSV edge effects in the following. Electric-field vectors, $\vec{E}$, can visualize the capacitances. The *Ansys Electromagnetic Suite* is used to draw the electric-field vectors in the substrate cross-view of the 3D model for different TSV-conductor

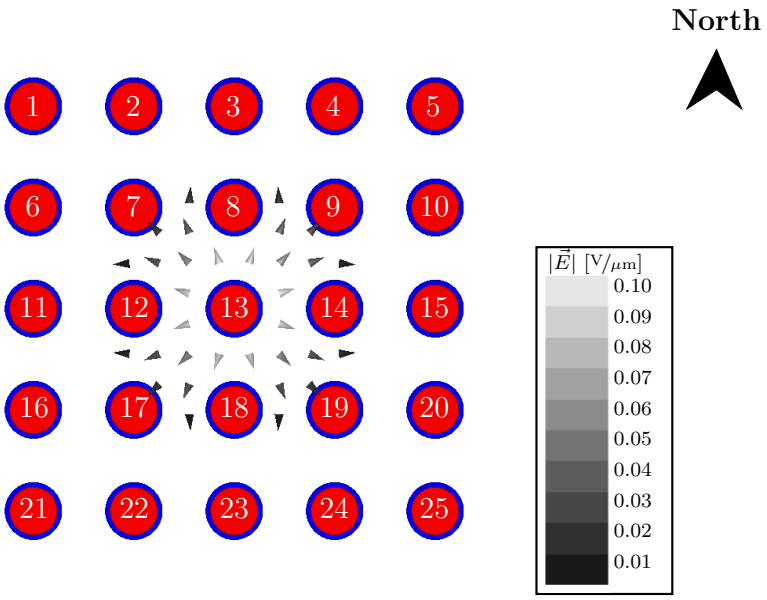


Fig. 4.4: Electric-field vectors for a potential of 1 V on a TSV located in the array middle, while all other TSVs are grounded.

potentials. In this section, the electric-field distribution for an exemplary 5×5 array with a TSV radius and minimum pitch of $2\,\mu$m and $8.5\,\mu$m, respectively, is investigated. Thereby, no further TSV is assumed to be located nearby the TSV array (other structures that are not implanted deep into the substrate, such as doping wells, have a negligible impact on the TSV parasitics).

In Figure 4.4, the electric-field vectors are illustrated for a potential of 1 V on a TSV located in the middle of the TSV array (referred to as a middle TSV) while all remaining TSVs are grounded. For the sake of clarity, only vectors with an absolute value bigger than 5 % of the maximum one are shown in the figures of this section. In accordance with the model used in previous works, only adjacent TSVs show a noticeable coupling in the array middle since the surrounding eight TSVs form a Faraday cage around a middle TSV, which terminates the electric-field vectors. Since the absolute value of $\vec{E}$ decreases with an increasing distance from the TSV, the coupling between diagonally adjacent TSVs is lower than the coupling between directly adjacent TSVs. This is also captured by the previous capacitance model.

Second, the edges of the array are analyzed. A TSV that is located at at-least one edge of the array is referred to in the remainder of this thesis as an edge TSV. For every TSV that is a not an edge TSV (*i.e.*, for all middle TSVs), the surrounding eight TSVs form a strong Faraday cage. This also entails that

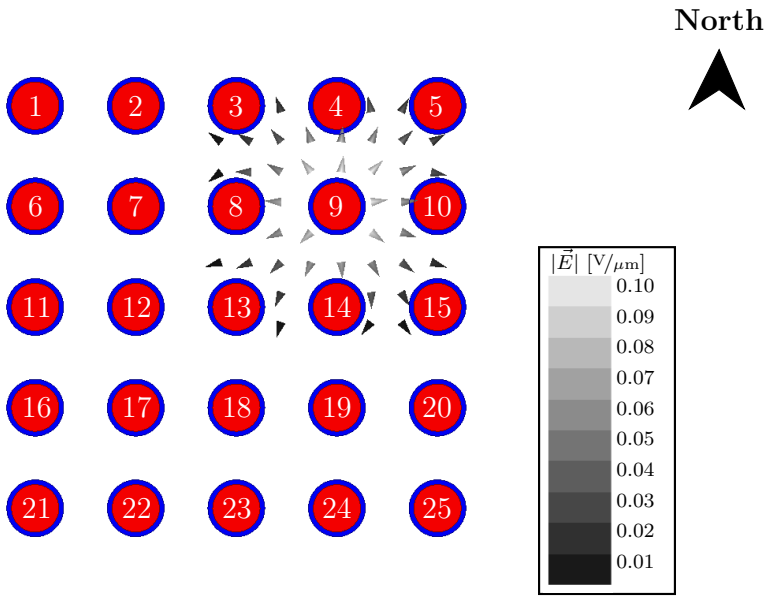


Fig. 4.5: Electric-field vectors for a potential of 1 V on a TSV located one row and one column away from two edges, while all other TSVs are grounded.

the coupling of a middle TSVs and any adjacent neighbor is mainly through the sides of the two TSVs that face each other. Consequently, an edge TSV couples with an adjacent middle TSV only over the side that is facing toward the middle. As a result, the edge effects can only noticeably influence the coupling between two edge TSVs. This is validated by Figure 4.5, which illustrates the electric-field vectors for a potential of 1 V on middle TSV_9, located one row and one column away from a corner. The figure shows that the coupling of TSV_9 with its adjacent edge TSVs does not noticeably differ from the coupling of the TSV with its adjacent middle TSVs.

To further investigate the edge effects, Figure 4.6 illustrates the electric-field distribution for a potential of 1 V on a TSV located at a single edge of the array. In Figure 4.7, the electric-field distribution for a potential of 1 V on a corner TSV, located at two-edges, is moreover illustrated. Between an edge-TSV pair, the electrical field is generally stronger (compared to a middle-TSV pair with the same distance between the TSVs), as edge TSVs are not enclosed by a solid Faraday cage. Consequently, the coupling between indirect/second-order adjacent TSV pairs (*e.g.*, TSV_2 and TSV_4) is no-longer negligible at the edges. Additionally, this implies that edge TSVs can have a non-negligible self-capacitance due to the substrate grounding.

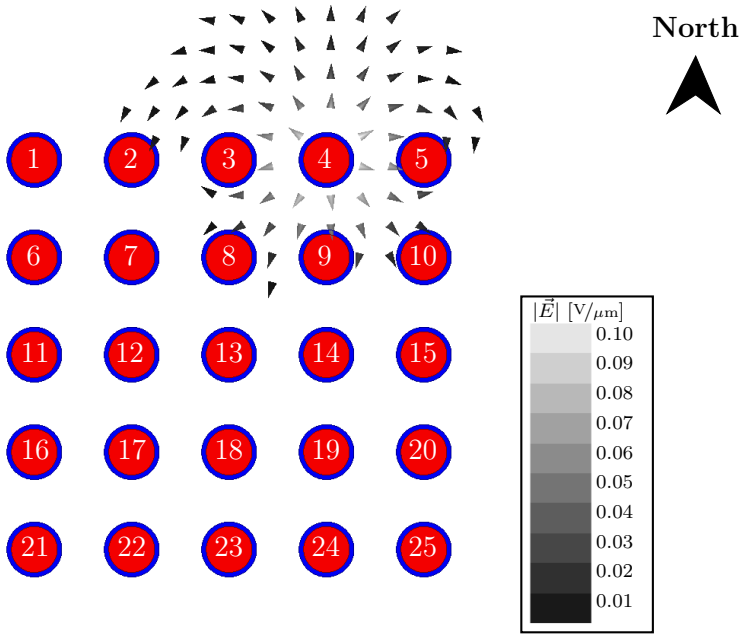


Fig. 4.6: Electric-field vectors for a potential of 1 V on a TSV located at a single edge of the array, while all other TSVs are grounded.

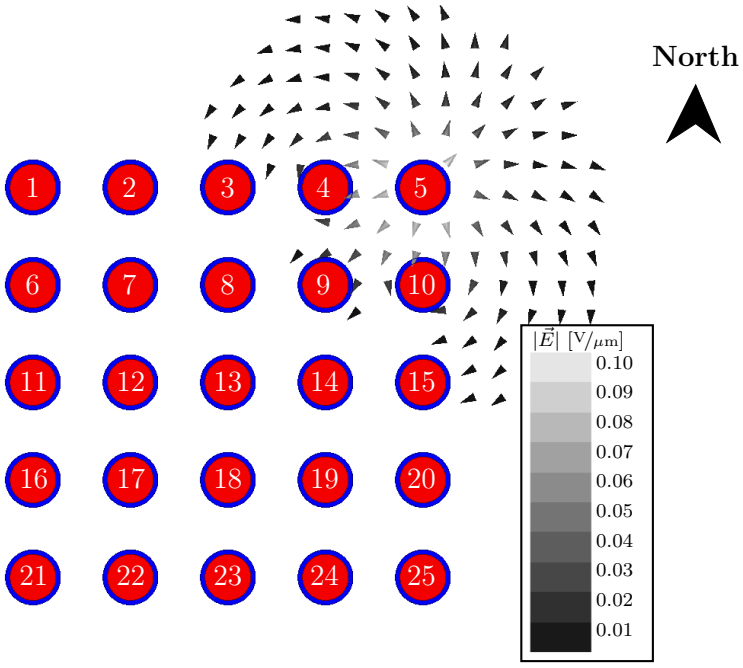


Fig. 4.7: Electric-field vectors for a potential of 1 V on a TSV located at a corner of the array, while all other TSVs are grounded.

Moreover, the free conductive substrate in at least one direction (*i.e.*, the reduced electric-field sharing with other TSVs) enables additional paths for the field lines at the array edges. This increases the coupling between two edge TSVs. At the array corners, the reduced electric-field sharing has an even stronger impact on the coupling, as a corner TSV is surrounded by only two directly adjacent TSVs. Consequently, the coupling of a corner TSV and a directly adjacent edge TSV is particularly strong.

The TSV edge effects are of particular importance since $2M + 2N - 4$ TSVs are located at the edges of an $M \times N$ array. In contrast, only two wires can be located at the edges of any metal-wire bus. For example, in the analyzed 5×5 array, as much as 64 % of the TSVs are located at one or more edges. Even for a large 10×10 array, this proportion is still 36 %.

Summarized, the edge effects result in increased capacitance quantities at the edges of an array, especially at the four corners, which cannot be neglected by a capacitance model. Thus, different parameters must be considered in the capacitance model for middle, single-edge, and corner TSVs.

4.3. TSV Capacitance Model

This section proposes a new TSV capacitance model, which considers the previously outlined MOS effect as well as the edge effects in an abstract way.

The MOS effect results in capacitance values which are a function of the mean TSV voltages, instead of being constant as in the previously used model. In principle, the average TSV voltages to determine the depletion-region widths have to be calculated over a sliding window, and the formulas to find the correct window duration exhibit a huge complexity. However, this work considers strongly stationary data streams. In this case, the average TSV voltages for the calculation of the depletion-region widths can be considered as constant for each individual data stream. Consequently, the mean TSV voltages, $\bar{\mathrm{V}}_i$, which determine the depletion-region widths can be expressed by means of the 1-bit probabilities on the TSVs, p_i:

$$\bar{\mathrm{V}}_i = \mathbb{E}\{b_i\} \cdot V_{\mathrm{dd}} = p_i \cdot V_{\mathrm{dd}}. \tag{4.2}$$

The resulting dependency between the size of the coupling capacitances and the bit probabilities is still quite complex. Nevertheless, from the previous section, it is known that the capacitance values can only increase or decrease (not both) with increasing mean voltages (*i.e.*, 1-bit probabilities) on the TSVs. As shown in Figure 4.3 on Page 63, the capacitance values decrease with an increase in the 1-bit probabilities for common p-doped substrates. In contrast, the capacitance values increase with the 1-bit probabilities for n-doped substrates. To achieve a low model complexity, the dependency between the bit probabilities and the capacitances is approximated in the proposed model by means of a linear

regression. The average 1-bit probabilities for the TSVs with the indices i and j, symbolized as $\bar{p}_{i,j}$, is used as the independent variable (*i.e.*, feature) to estimate/model the size of capacitance $C_{i,j}$. Please note that, for i equal to j (*i.e.*, the fit for the self-capacitance $C_{i,i}$), the feature $\bar{p}_{i,j}$ is equal to the logical 1-bit probability on the i^{th} TSV, p_i. Summarized, the following formula models the MOS effect on the TSV capacitances:

$$C_{i,j} = C_{\text{G},i,j} + \frac{\Delta C_{i,j}}{2}(p_i + p_j) = C_{\text{G},i,j} + \Delta C_{i,j} \cdot \bar{p}_{i,j}, \tag{4.3}$$

where $C_{\text{G},i,j}$ is the fitted capacitance value for a mean voltage of 0 V on all TSVs. $\Delta C_{i,j}$ is the fitted derivation of the capacitance value with increasing 1-bit probability on the TSVs. Hence, the full capacitance matrix is mathematically expressed as

$$\mathbf{C} = \mathbf{C}_{\text{G}} + \mathbf{\Delta C} \circ \mathbf{\bar{PR}}, \tag{4.4}$$

where $\mathbf{\bar{PR}}$ is an $n \times n$ probability matrix with $\bar{p}_{i,j}$ on entry (i, j). $\circ$ is the Hadamard-product operator, resulting in an element-wise multiplication of the entries of two matrices. $\mathbf{C}_{\text{G}}$ and $\mathbf{\Delta C}$ are $n \times n$ matrices with $C_{\text{G},i,j}$ and $\Delta C_{i,j}$ on the (i, j) entries, respectively. The entries of $\mathbf{\Delta C}$ are negative for typical p-doped substrates. In contrast, the entries are positive for n-doped substrates, indicating increasing capacitance values with increasing 1-bit probabilities.

If the power-supply voltage of the driver technology exceeds the absolute voltage at which the depletion region of a TSV reaches its maximum value, the capacitance values additionally must be clipped to a lower bound:

$$\mathbf{C} = \max\left(\mathbf{C}_{\text{G}} + \mathbf{\Delta C} \circ \mathbf{\bar{PR}},\ \mathbf{C}_{\text{max-dep}}\right), \tag{4.5}$$

where $\mathbf{C}_{\text{max-dep}}$ is the fitted capacitance matrix for all TSVs surrounded by a depletion region of maximum width, resulting in the lowest capacitance values. Function "max()" returns a matrix with the entry-wise maximum values of the two matrices. Important is in this particular case that the extracted capacitance values, used to fit a $\Delta C_{i,j}$ value and the respective $C_{\text{G},i,j}$ value, are obtained for mean voltages that do not result in a maximum depletion-region width for any TSV to avoid inaccurate $\mathbf{\Delta C}$ entries. However, for all TSV structures analyzed in this thesis, the power-supply voltages of the driver cells are low enough that the extended model (*i.e.*, Equation (4.5)) is not needed.

Finally, abstract, scaleable, and yet universally valid models to build $\mathbf{C}_{\text{G}}$ and $\mathbf{\Delta C}$ are required. The previously discussed TSV edge effects reveal that more than two different capacitance types, C_{n} and C_{d}, have to be considered in these models, as the coupling capacitance between two edge TSVs is generally higher compared to its counterpart in the middle of the array, especially if one of the two edge TSVs is even located at an array corner. Also, the ground capacitances

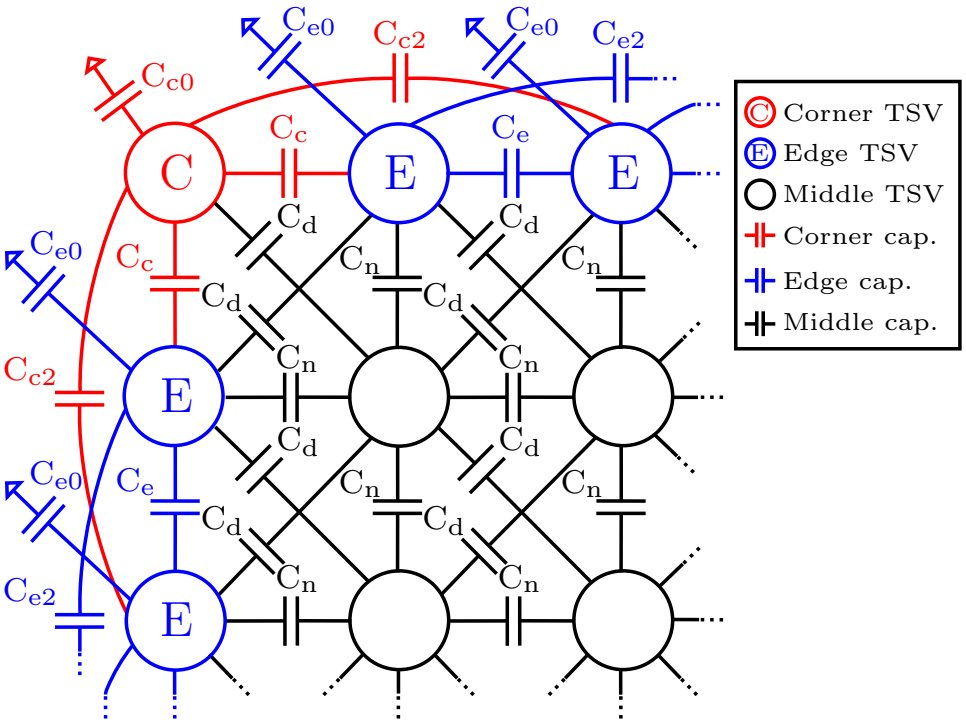


Fig. 4.8: Proposed edge-effect-aware TSV capacitance model.

of TSVs are not necessarily negligible at the edges. Thus, this thesis proposes the TSV capacitance model illustrated in Figure 4.8. The model distinguishes between capacitances connected to at least one middle TSV (marked black in Figure 4.8), and capacitances connected only to edge TSVs. For the second case, the capacitances also generally differ if an associated TSV is located at a corner of the array (marked red in Figure 4.8) or not (marked blue in Figure 4.8).

For the coupling capacitances between any two directly adjacent TSVs, out of which at least one TSV is located in the array middle, a single fit $\mathrm{C_n}(\bar{p}_{i,j})$ is used. This fit results in the two model coefficients $C_{\mathrm{G,n}}$ and ΔC_{n}. Although the edge effects sometimes have a slight impact on the capacitance value of two diagonally adjacent edge TSVs over a corner, a single linear fit, $\mathrm{C_d}(\bar{p}_{i,j})$, is used to model all coupling capacitances between diagonally adjacent TSVs.

The capacitances between the corner TSVs and their two directly adjacent edge TSVs are modeled using another fit, $\mathrm{C_c}(\bar{p}_{i,j})$. Also, the capacitance value between a directly adjacent edge-TSV pair that is not located at an array corner is modeled by an individual fit, $\mathrm{C_e}(\bar{p}_{i,j})$. At the edges, the capacitances between indirectly adjacent TSVs have to be included in the capacitance model. Thus, one linear fit, $\mathrm{C_{c2}}(\bar{p}_{i,j})$, is used to model the capacitances between a corner TSV and its two indirectly adjacent edge TSVs. Another fit, $\mathrm{C_{e2}}(\bar{p}_{i,j})$, is used to model the capacitances between indirectly adjacent TSVs pairs located only at

one edge of the array. Finally, the fits $C_{c0}(p_i)$ and $C_{e0}(p_i)$ model the ground capacitances of TSVs located at a corner and a single edge of the array.

Summarized, in the proposed capacitance model defined by Equation (4.4), $\mathbf{C}_G$ is constructed out of eight capacitance values: $C_{G,n}$, $C_{G,d}$, $C_{G,e0}$, $C_{G,e1}$, $C_{G,e2}$, $C_{G,c0}$, $C_{G,c1}$, and $C_{G,c2}$. Analogously, also $\mathbf{\Delta C}$ is constructed out of eight capacitance deviations: ΔC_n, ΔC_d, ΔC_{e0}, ΔC_{e1}, ΔC_{e2}, ΔC_{c0}, ΔC_{c1}, and ΔC_{c2}. Thus, the proposed model has in total 16 parameters. A pseudo-code to build the two matrices out of these 16 parameters is presented through Algorithm 3 in Appendix B on Page 259 of this thesis.

To fit the 16 capacitance coefficients of the model for a certain TSV technology, eight capacitance values have to be extracted for an exemplary TSV array bigger than 4×4 for varying (*i.e.*, at least two) depletion-region widths. The more depletion-region widths (and, thus, 1-bit probabilities) are considered for parasitic extraction, the better the resulting capacitance fits. On the downside, considering more depletion-region widths has the drawback of an increased computational complexity for the parasitic extraction. However, for each different TSV technology, the model coefficients only have to be fitted once, as the proposed model is universally valid and scaleable. Due to this strong reusability, the overhead of analyzing a more comprehensive set of depletion-region widths vanishes over time in most application scenarios. To further achieve a reusability of the fitted coefficients for different substrate thicknesses, all 16 values must be reported per unit TSV length. Moreover, the ΔC values must be additionally normalized by the investigated power-supply voltage to make the coefficients reusable for different driver technologies.

4.4. Evaluation

The evaluation section of this chapter consists of two subsections. Subsection 4.4.1 reports the coefficients of the proposed TSV capacitance model. Furthermore, the accuracy of the capacitance model is quantified and compared to the accuracy of the previously used capacitance model to put the accuracy into a better perspective. In detail, two analyses are conducted to quantify the accuracy of the TSV capacitance models. The first primarily quantifies how well the MOS effect is captured by applying linear fits and what are the errors of the previous model using constant capacitance values in comparison. In the second one, it is investigated how precise the abstract capacitance models construct the complete capacitance matrices of several TSV arrays with varying line count and bit-level properties. The following Subsection 4.4.2, investigates the accuracy of a high-level TSV power-consumption and performance estimation based on the proposed capacitance model combined with the formulas presented in Chapter 3. Again, a comparison is drawn with the results for the previous TSV capacitance model.

4.4.1. Model Coefficients and Accuracy

This section determines the coefficients and the accuracy of the presented as well as the previous TSV capacitance model.

Note that, in [A1], it is shown that the proposed MOS-effect-aware capacitance model will become even more essential for future submicrometric TSV dimensions. This is based on the fact that the relative impact of the depletion-region widths on the coupling-capacitance values increases with reduced TSV dimensions. For example, it is shown in [A1] that, for a TSV radius and pitch of $0.5\,\mu$m and $2\,\mu$m, respectively, the MOS effect can even impact the TSV capacitance sizes by over 40 %. Furthermore, [A1] includes a discussion on the necessity of more complex quadratic fits to capture the MOS effect for future TSV dimensions. However, these results are not included in this thesis since submicrometric global TSVs are not expected to become a reality in the near future.

Moreover, in the author's initial publication on edge effects in TSV-array arrangements (see [A2]), it is shown that the proposed edge-effect-aware capacitance model remains valid if structures that are deeply integrated into the substrate are located nearby the TSV arrays in order to suppress the substrate noise TSVs induce on active circuits. Examples for such structures are shielding lines or guard rings [20]. However, structures to reduce the TSV substrate noise are not investigated in this thesis as they are rather uncommonly implemented in modern 3D systems. Thus, these results are also not included in this thesis. The interested reader finds further information in the according publications of the author.

Experimental Setup

In the following, the experimental setup to obtain the model coefficients is outlined. To determine the model coefficients, parasitic extractions for a 5×5 instance of the scaleable TSV-array model are performed with the *Q3D Extractor*. All coefficients of the proposed capacitance model are reported per unit TSV length. Hence, the model coefficients only depend on the TSV radius, r_{tsv}, the minimum TSV pitch, d_{min}, and the significant signal frequency, f_{s}. Consequently, the TSV length is fixed to $50\,\mu$m for the determination of the model coefficients.

Two variants are analyzed for the TSV radius: $1\,\mu$m and $2\,\mu$m. In this thesis, primarily, densely spaced TSVs are considered as an enlarged TSV pitch increases the critical area occupation of TSV arrays even further. Moreover, previous research has shown that increasing the TSV spacing does not effectively reduce the TSV coupling capacitances, in contrast to metal-wire structures where spacing is very efficient [17]. The minimum predicted TSV pitches for the analyzed TSV radii of $1\,\mu$m and $2\,\mu$m are $4\,\mu$m and $8\,\mu$m, respectively. Besides these minimum values for the TSV pitches, also a reasonable increase

in the minimum pitches by 0.5 μm to 4.5 μm and 8.5 μm is considered. Two significant signal frequencies, f_s, are analyzed: 6 GHz and 11 GHz. The first one corresponds with a typical TSV-signal rise and fall time of about 80 ps, while the second one corresponds with a TSV-signal rise and fall time of less than 50 ps for strong drivers.

To obtain the model coefficients, different mean voltages on the TSVs against the ground are analyzed. Varying the voltages leads to varying depletion-region widths that differ for the two considered TSV radii (note that the depletion-region widths are independent of the TSV pitches). After the depletion-region widths for the investigated mean voltages are determined by means of the method presented in Appendix C, they are fed into the TSV-array model to extract the associated capacitance values. These extracted capacitance values are subsequently used to perform linear-regression analyses using the least-squares approach. Thereby, the feature values for the regression analyses are the mean voltages for the related TSVs—which can later be mapped to the actual 1-bit probabilities through Equation (4.2).

For each individual capacitance fit, extraction results for a single capacitance are considered. The extracted values for the coupling capacitances $C_{13,12}$ and $C_{13,7}$ of TSV_{13} in the middle of the array (see Figure 4.4 on Page 64), are analyzed to obtain the respective capacitance fits $\mathrm{C_n}(\bar{p}_{i,j})$ and $\mathrm{C_d}(\bar{p}_{i,j})$. To obtain the fits $\mathrm{C_{e0}}(p_i)$, $\mathrm{C_{e1}}(\bar{p}_{i,j})$, and $\mathrm{C_{e2}}(\bar{p}_{i,j})$, the extracted capacitance values for the capacitances $C_{2,2}$, $C_{2,3}$, and $C_{2,4}$ of TSV_2 located at a single array edge are considered, respectively. For the last three fits, $\mathrm{C_{c0}}(p_i)$, $\mathrm{C_{c1}}(\bar{p}_{i,j})$, and $\mathrm{C_{c2}}(\bar{p}_{i,j})$, the extracted capacitance values for $C_{1,1}$, $C_{1,2}$, and $C_{1,3}$ of TSV_1, located in a corner of the array, are analyzed.

Each analyzed capacitance is extracted for a wide set of different mean TSV voltages. First, the mean voltage on the related TSV is varied between 0 V and 1 V with a constant step-size. Furthermore, the mean voltage on all remaining TSVs is varied independently using the same voltage steps.

For example, consider an analysis where the step size between the analyzed TSV voltages is 0.5 V while the power supply voltage is 1 V. Thus, in the exemplary analysis, the 1-bit probabilities on the TSVs are either 0, 0.5, or 1. In this case, three different capacitance values would be extracted for an analyzed coupling capacitance $C_{i,j}$, and $\bar{p}_{i,j}$ equal to 0.5. One value for a 1-bit probability of 0.5 on all TSVs in the array; one value for a 1-bit probability of 0 on TSV_i, while the 1-bit probability is 1 for all other TSVs in the array; and one value for a 1-bit probability of 1 on TSV_i, while the 1-bit probability is 0 for all other TSVs in the array. Also, three different values would be extracted for an analyzed ground capacitance $C_{i,i}$ and p_i equal to 0.5: One value for a 1-bit probability of 0 on all other TSVs; one value for a 1-bit probability of 0.5 on all other TSVs; and one value for a 1-bit probability of 1 on all other TSVs.

This setup is chosen as it later allows to quantify the model-inaccuracy due to the averaging of the two bit probabilities p_i and p_j, as well as the neglected impact of the depletion regions of the remaining TSVs which are not directly associated with the analyzed capacitance. In the actual experiment, the step size in the considered voltages is 0.1V and not 0.5V to provide a higher model accuracy and to quantify the model errors better. Thus, 121 different values are extracted for each analyzed capacitance, and the smallest resolution of the feature variable for the linear regression, $\bar{p}_{i,j}$, is 0.05.

Model Coefficients

The extracted capacitance values are first used to perform linear-regression analyses to obtain the coefficients of the proposed model. Furthermore, the mean extracted value for $C_{13,12}$ is used as $C_{\mathrm{n,prev}}$ for the previous model. Afterward, the second parameter of the previously used model, $C_{\mathrm{d,prev}}$, is calculated as $C_{\mathrm{n,prev}}/4$.

The fittings of $C_{\mathrm{c1}}(\bar{p}_{i,j})$ and $C_{\mathrm{n}}(\bar{p}_{i,j})$ for the proposed model are exemplarily illustrated in Figure 4.9a for the capacitance values extracted for a TSV radius, minimum pitch, and significant frequency of $1\,\mu m$, $4.5\,\mu m$, and 6 GHz, respectively. Furthermore, the fitting of $C_{\mathrm{n,prev}}$ is illustrated in the figure, used in previous works to model the coupling capacitances of all directly adjacent TSV pairs. In Figure 4.9b, the relative errors of the two models to recreate the extracted capacitance values for $C_{13,12}$ and $C_{1,2}$ are plotted.

The resulting coefficients of the proposed capacitance model are reported in Table 4.1 for all analyzed geometrical TSV dimensions and significant frequencies. To obtain the actual model parameters for a given power-supply voltage, V_{dd}, and TSV length, l_{tsv}, out of the reported values, the normalized C' values have to be multiplied with l_{tsv} and the normalized $\Delta C'$ values with $l_{\mathrm{tsv}} \cdot V_{\mathrm{dd}}$. However, note that the reported $\Delta C'$ values are fitted for a voltage range of 0 V to 1 V. Hence, only for V_{dd} equal to 1 V the fits are least-square fits. For comparison purposes, the coefficients for the previously used TSV capacitance model are additionally reported in Table 4.2.

The results in Table 4.1 confirm that the edge effects lead to a drastic increase in the capacitance values. For example, the coupling capacitance between a corner TSV and a directly adjacent edge TSV, $C_{\mathrm{G,c1}}$, is 40 % to 46 % bigger than the coupling-capacitance between directly adjacent TSVs in the array middle, $C_{\mathrm{G,n}}$, for all analyzed TSV radii, pitches, and significant frequencies. Also, the fitted capacitance value for two directly adjacent single-edge TSVs, $C_{\mathrm{G,e1}}$, is always 30 % to 33 % bigger than $C_{\mathrm{G,n}}$. Moreover, the $C_{\mathrm{G,e2}}$ and $C_{\mathrm{G,c2}}$ values are in the range of the $C_{\mathrm{G,d}}$ values. This fact underlines the importance of considering the edge effects in the capacitance model.

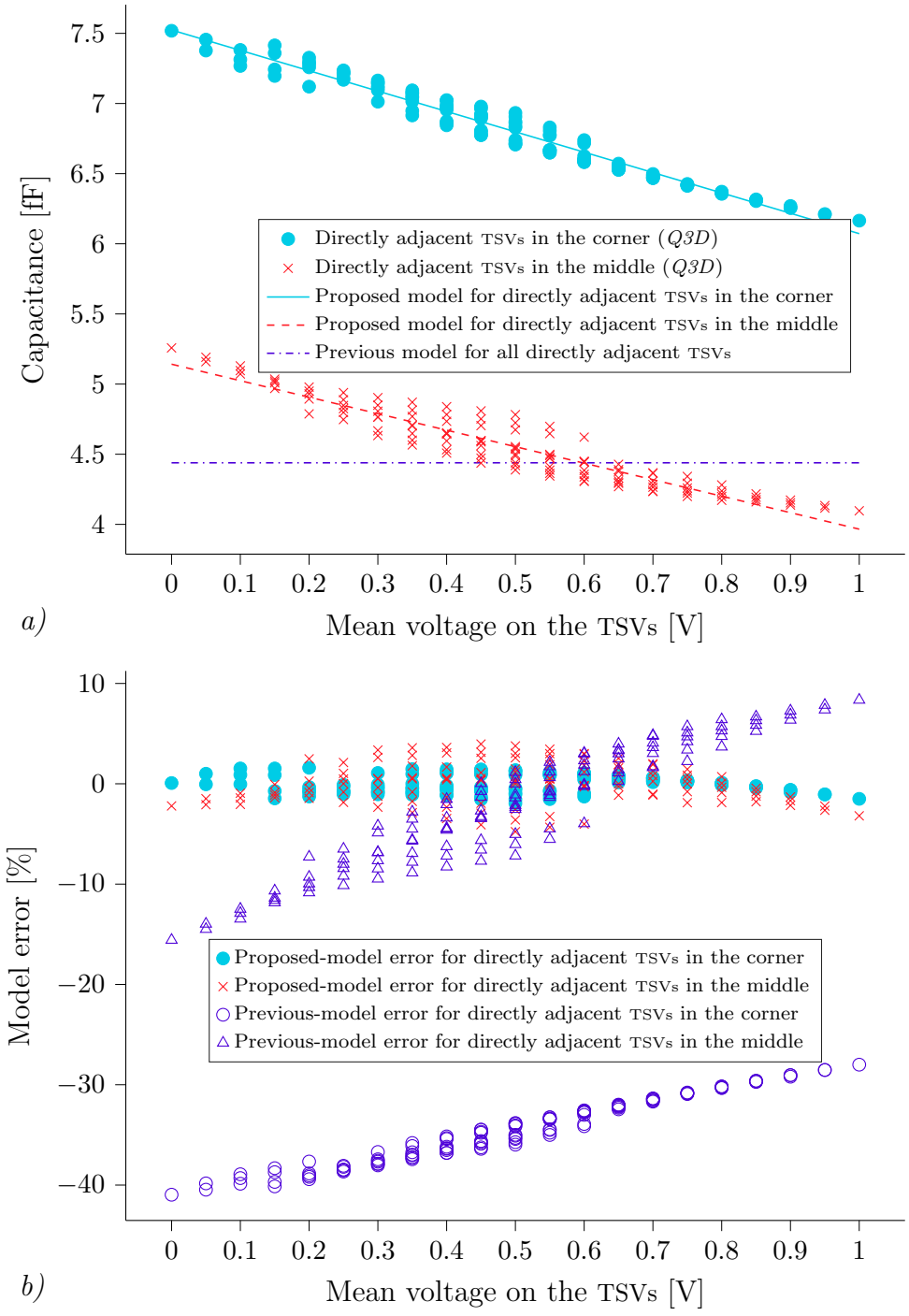


Fig. 4.9: Fitting of the capacitance-model coefficients: *a)* Extracted capacitance values and the fitted models over the mean voltage on the related TSVs; *b)* Relative errors of the models to reconstruct the extracted capacitance values.

Table 4.1.: Coefficients of the proposed TSV capacitance model for different TSV parameters.

TSV parameters			**Model coefficients**													
r_{tsv}	d_{min}	f_{s}	$C'_{\text{G,n}}$	$\Delta C'_{\text{n}}$	$C'_{\text{G,d}}$	$\Delta C'_{\text{d}}$	$C'_{\text{G,e0}}$	$\Delta C'_{\text{e0}}$	$C'_{\text{G,e1}}$	$\Delta C'_{\text{e1}}$	$C'_{\text{G,e2}}$	$\Delta C'_{\text{e2}}$	$C'_{\text{G,c0}}$	$\Delta C'_{\text{c0}}$	$C'_{\text{G,c1}}$	$\Delta C'_{\text{c1}}$	$C'_{\text{G,c2}}$	$\Delta C'_{\text{c2}}$
[μm]	[μm]	[GHz]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]	[pF/m]	[pF/m V]
1.0	4.0	6.0	104.0	-19.3	36.4	-4.2	8.3	-6.3	138.3	-24.6	21.8	0.0	8.6	-5.0	150.0	-23.6	28.8	0.0
1.0	4.0	11.0	91.4	-11.1	26.4	-1.6	2.7	-0.7	120.3	-13.0	14.7	0.0	5.6	-0.4	127.8	-14.2	19.5	0.0
1.0	4.5	6.0	102.8	-23.5	37.3	-5.4	3.4	-0.4	136.9	-29.9	22.8	-0.2	4.2	-1.2	150.5	-29.0	29.3	0.0
1.0	4.5	11.0	85.8	-10.3	26.3	-0.9	2.6	-0.1	112.8	-12.1	15.1	0.0	5.1	-2.1	121.8	-11.8	19.4	0.0
2.0	8.0	6.0	116.2	-10.2	38.4	-0.7	3.4	-0.9	154.1	-13.2	22.6	0.0	5.9	0.0	168.0	-13.9	29.5	0.0
2.0	8.0	11.0	95.7	-3.7	26.7	0.0	4.9	-0.9	125.2	-4.2	14.7	0.0	10.5	-0.1	134.2	-4.0	19.5	0.0
2.0	8.5	6.0	115.2	-9.9	38.5	-0.7	3.2	0.0	151.5	-12.3	22.2	0.0	9.6	-0.4	164.3	-12.3	29.0	0.0
2.0	8.5	11.0	92.7	-3.1	26.6	0.0	5.1	-0.6	120.8	-3.5	14.5	0.0	12.3	-0.6	129.6	-3.4	19.1	0.0

Table 4.2.: Coefficients of the previous TSV capacitance model for different TSV parameters.

TSV parameters			**Model coefficients**	
r_{tsv}	d_{min}	f_{s}	$C'_{\text{n,prev}}$	$C'_{\text{d,prev}}$
[μm]	[μm]	[GHz]	[pF/m]	[pF/m]
1.0	4.0	6.0	94.4	23.6
1.0	4.0	11.0	85.9	21.5
1.0	4.5	6.0	91.1	22.8
1.0	4.5	11.0	80.6	20.2
2.0	8.0	6.0	110.6	27.6
2.0	8.0	11.0	93.7	23.4
2.0	8.5	6.0	109.8	27.5
2.0	8.5	11.0	91.0	22.8

For the analyzed TSV structures, the self capacitances are relatively small. Nevertheless, the coefficients $C_{\mathrm{G,e0}}$, ΔC_{e0}, $C_{\mathrm{G,c0}}$, and ΔC_{c0} cannot be generally removed from the model to reduce its complexity. The reason is that these values drastically increase if the TSV arrays are enclosed by ground rings to reduce the substrate noise, as shown in the author's publication [A2].

Generally, the relative increase of the coupling-capacitance values toward the edges is nearly independent of the TSV radius and pitch but shows a slight dependency on the significant frequency. For example, the ratio $C'_{\mathrm{G,c1}}/C'_{\mathrm{G,n}}$ is always in the range from 1.44 to 1.46 for a significant frequency of 6 GHz, and 1.40 to 1.41 for a significant frequency of 11 GHz. In the following, the reason for this frequency dependency is outlined. Furthermore, the dependency on the substrate doping is discussed.

Through-silicon vias couple through their insulating silicon-oxide liners, the surrounding depletion regions, and the substrate—all connected in series. While the electrical behavior of TSV oxide liners and the depletion regions can be precisely modeled by capacitances, the substrate is modeled at lower abstraction levels as parallel capacitance-conductance components due to its electrical conductivity [11]. Thus, the admittance of a substrate unit element is electrically modeled as follows:

$$Y'_{\mathrm{subs}} = j2\pi f_{\mathrm{s}} C'_{\mathrm{subs}} + G'_{\mathrm{subs}}, \tag{4.6}$$

where C'_{subs} and G'_{subs} is the capacitance and the conductance of the unit element, respectively.

Here, the lossy (complex) equivalent capacitance is introduced to simplify the following discussion:

$$C'_{\mathrm{eq,subs}} \stackrel{\mathrm{def}}{=} \frac{Y'_{\mathrm{subs}}}{j2\pi f_{\mathrm{s}}} = C'_{\mathrm{subs}} + \frac{G'_{\mathrm{subs}}}{j2\pi f_{\mathrm{s}}}. \tag{4.7}$$

The absolute value of $C'_{\mathrm{eq,subs}}$ decreases with increasing f_{s} or decreasing G'_{subs} and asymptotically reaches C'_{subs}. Hence, the overall TSV capacitance values decrease with increasing significant frequency and decreasing substrate doping (reduces G'_{subs}). Furthermore, a decreasing $C'_{\mathrm{eq,subs}}$ value leads to slightly decreasing edge effects, since edge effects occur due to free substrate paths for the electric field. Thus, with higher frequencies or lower doping concentrations, the magnitude of the edge effects decreases. Asymptotically, for very high significant frequencies and low doping profiles, the magnitude of the edge effects is determined solely by the capacitive behavior of the substrate.

An analysis is conducted to quantify the magnitude of the edge effects for ultra-high-speed signaling ($f_{\mathrm{s}} \gg 10\,\mathrm{Ghz}$). For a mean voltage of 0.5 V on all TSVs, the coupling-capacitance values for directly adjacent edge TSVs (*i.e.*, C_{c1} and C_{e1}) and directly adjacent middle TSVs (*i.e.*, C_{n}) are extracted with the

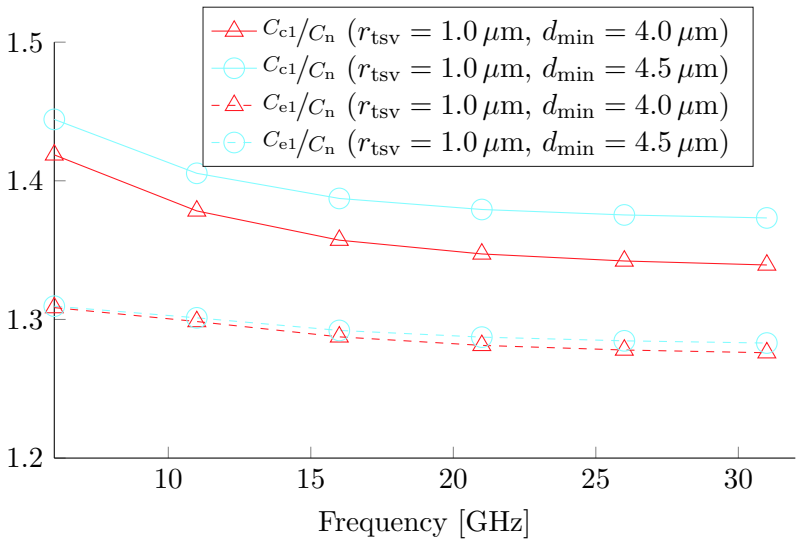


Fig. 4.10: Ratio of the coupling capacitance between two directly adjacent edge TSVs over the coupling capacitance between two directly adjacent middle TSVs for different significant signal frequencies and a mean voltage of 0.5 V on all TSVs.

Q3D Extractor over the significant signal frequency. Thereby, a TSV radius of $1\,\mu m$ is assumed exemplarily. The ratios C_{c1}/C_n and C_{e1}/C_n for the two different analyzed TSV pitches are plotted in Figure 4.10 over f_s. For all analyzed TSV dimensions and significant frequencies as high as 31 GHz, the C_{c1} and C_{e1} values are always at least 33 % and 27 % bigger than their counterpart in the array middle, C_n, respectively. With a further increase in the significant frequency, no significant change in the capacitance ratios is obtained. The same asymptotic C_{c1}/C_n and C_{e1}/C_n ratios are obtained if the substrate doping is decreased toward $0\,S/m$. Therefore, even for ultra-high frequencies and low doping profiles, the edge effects are still of great importance and consequently must be captured in the capacitance model.

So far, only the magnitude of the edge effects, but not the MOS effect, has been discussed. Also, the MOS effect has a significant impact on the capacitances, especially for moderate significant frequencies. However, the magnitude of the MOS effect decreases as well with increasing significant frequency or a decreasing substrate conductivity. The rationale for this behavior is again the capacitive-conductive duality of the substrate. Since a depletion region is an area in the substrate with a conductivity that is decreased to zero, the admittance of a depletion region unit element is

$$Y'_{subs,depleted} = j2\pi f_s C'_{subs}. \tag{4.8}$$

With increasing depletion-region widths, a larger area of the substrate has a smaller admittance described by Equation (4.8) instead of (4.6) and *vice versa*. However, with increasing f_s or decreasing G'_subs, the relative difference between Equation (4.8) and (4.6) decreases. Thus, the smaller the substrate conductivity, and the higher the significant frequency, the lower the impact of the depletion regions on the capacitances.

In contrast to the edge effects, the MOS effect even completely vanishes with increasing significant frequencies or decreasing substrate doping as

$$\lim_{f_\mathrm{s}\to\infty} Y'_\mathrm{subs} = \lim_{G'_\mathrm{subs}\to 0} Y'_\mathrm{subs} = Y'_\mathrm{subs,depleted}. \tag{4.9}$$

Consequently, for high frequencies or non-doped substrates, the capacitance model can be simplified by removing the MOS effect. In this case, all $\mathbf{\Delta C}$ entries are zero. Consequently, the capacitance matrix would be simply modeled by $\mathbf{C}_\mathrm{G}$, resulting in only eight capacitance values that must be extracted without the need for any linear-regression analysis. Furthermore, for indirectly adjacent TSV pairs, the magnitude of the MOS effect is generally negligible even for small significant frequencies and high doping profiles. Thus, the ΔC_{e2} and ΔC_{c2} values could be removed from the model in order to decrease the model complexity. Another fact worth mentioning is that the magnitude of the MOS effect increases with ongoing TSV scaling. The main reason for this behavior is that the relative dynamic width of the depletion region is higher for thinner TSV oxides.

Model Accuracy—Goodness of the Linear Fits

The following analysis quantifies the maximum absolute error (MAE) as well as the root-mean-square error (RMSE) values of the linear fits. All MAE and RMSE values are obtained by considering all 121 extracted values for each different capacitance, which are then compared with the respective model predictions. Furthermore, the error values for the previously used TSV capacitance model are determined.

The resulting error metrics for the proposed and the previous capacitance model are reported in Table 4.3 and Table 4.4, respectively. In the tables, all error values are reported in percentage numbers relative to the according $C_\mathrm{n,prev}$ value (*i.e.*, $C'_\mathrm{n,prev} \cdot l_\mathrm{tsv}$) in order to remove any scaling bias. This allows us to better relate the error values for different TSV parameters.

The results show that the proposed model enables us to accurately replicate the extracted capacitance values for all analyzed TSV parameters, as all normalized MAE values are below 10 %. In contrast, the previous model results in errors as high as 64.3 %. The normalized RMSE values of the linear fits of the proposed model do not exceed 4.3 %. This is a dramatic improvement compared to the previous model, which results in RMSE values of up to 46.3 %.

Table 4.3.: Normalized maximum-absolute and RMS errors of the linear fits for the extracted TSV capacitances.

TSV parameters			**Capacitance—linear model**															
r_{tsv}	d_{min}	f_s	$C_{13,12}$—C_n		$C_{13,7}$—C_d		$C_{2,2}$—C_{e0}		$C_{2,3}$—C_{e1}		$C_{2,4}$—C_{e2}		$C_{1,1}$—C_{c0}		$C_{1,2}$—C_{c1}		$C_{1,3}$—C_{c2}	
			MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
[μm]	[μm]	[GHz]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1.0	4.0	6.0	4.0	1.3	1.0	0.4	9.2	4.3	5.0	1.6	1.6	0.5	5.6	3.3	5.8	2.5	1.4	0.6
1.0	4.0	11.0	1.1	0.4	0.3	0.1	1.7	0.5	1.2	0.4	0.5	0.2	1.1	0.3	2.2	0.8	0.6	0.2
1.0	4.5	6.0	5.0	2.0	1.8	0.8	6.6	2.6	4.1	1.8	0.9	0.2	6.9	2.8	3.0	1.3	1.3	0.3
1.0	4.5	11.0	1.9	0.7	0.7	0.4	2.6	0.9	1.6	0.7	1.0	0.3	4.4	1.5	1.7	0.6	1.1	0.4
2.0	8.0	6.0	1.0	0.4	0.5	0.2	1.2	0.5	0.5	0.2	0.5	0.2	2.3	1.7	0.4	0.2	0.3	0.1
2.0	8.0	11.0	0.1	0.0	0.2	0.1	0.9	0.4	0.3	0.1	0.3	0.1	1.0	0.3	0.3	0.1	0.3	0.1
2.0	8.5	6.0	4.0	0.5	1.3	0.3	0.9	0.4	5.1	0.6	0.5	0.2	0.7	0.3	0.8	0.4	0.4	0.2
2.0	8.5	11.0	0.1	0.0	0.2	0.1	0.6	0.3	0.3	0.1	0.4	0.2	0.7	0.2	0.1	0.0	0.3	0.2

Table 4.4.: Normalized maximum-absolute and RMS errors of the previous TSV capacitances model for the nine capacitance values extracted for varying mean TSV voltages.

TSV parameters			**Capacitance—constant model value**															
r_{tsv}	d_{min}	f_s	$C_{13,12}$—$C_{n,prev}$		$C_{13,7}$—$C_{d,prev}$		$C_{2,2}$—0		$C_{2,3}$—$C_{n,prev}$		$C_{2,4}$—0		$C_{1,1}$—0		$C_{1,2}$—$C_{n,prev}$		$C_{1,3}$—0	
			MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
[μm]	[μm]	[GHz]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1.0	4.0	6.0	14.2	4.7	13.7	11.4	15.3	7.3	51.4	34.0	23.7	23.1	12.9	7.4	64.3	46.8	31.4	30.5
1.0	4.0	11.0	7.2	2.9	5.5	4.8	4.8	2.8	40.9	32.7	17.2	17.1	7.0	6.2	49.9	40.8	22.9	22.7
1.0	4.5	6.0	15.5	6.1	16.2	13.1	10.3	4.4	51.8	34.7	25.4	25.0	11.3	4.9	65.1	49.8	32.6	32.2
1.0	4.5	11.0	6.9	3.0	8.0	7.1	4.6	3.2	40.1	32.5	19.1	18.7	9.0	5.3	51.2	43.9	24.6	24.1
2.0	8.0	6.0	4.5	1.9	9.7	9.4	4.2	2.7	38.5	33.0	20.7	20.5	7.6	5.6	51.0	45.1	27.0	26.7
2.0	8.0	11.0	1.8	0.8	3.7	3.6	6.0	4.7	33.2	31.3	16.0	15.8	11.7	11.2	42.8	40.9	21.2	20.9
2.0	8.5	6.0	5.3	1.9	10.2	9.7	3.9	3.0	37.2	32.0	20.7	20.3	9.2	8.5	48.9	43.6	26.7	26.4
2.0	8.5	11.0	1.5	0.7	4.5	4.2	5.9	5.3	32.3	30.7	16.3	16.0	13.8	13.2	42.0	40.4	21.3	21.0

Generally, both models result in higher errors for aggressively scaled TSV dimensions and lower significant frequencies. For example, for a TSV radius of $2\,\mu$m, a minimum TSV pitch of $8\,\mu$m, and a significant frequency of 11 GHz, all reported MAE and RMSE values of the proposed model do not exceed 1 % and 0.4 %, respectively. For these TSV parameters, the previously used model still results in MAE and RMSE values of up to 42.8 % and 40.9 %, respectively.

The reason for the lower accuracy of the capacitance model for smaller TSV dimensions and lower frequencies has various reasons. First, the magnitude of the MOS effect increases with shrinking TSV geometries and a decreasing significant frequency. Thereby, also the error due to the applied simple linear fit increases. As shown in the author's publication [A1], using non-linear fits allows overcoming this issue for the proposed model in future technologies. Second, the impact of the depletion regions of the TSVs that are not directly connected to the modeled coupling capacitance increases with shrunk TSV geometries and lower significant frequencies. Last, non-idealities in the substrate become more dominant with lower frequencies.

Model Accuracy for Complete Capacitance Matrices

So far, it was shown that linear fits, using $\bar{p}_{i,j}$ as the feature variable, allow us to reproduce extracted capacitance values for varying depletion-region widths accurately. However, it is yet not proven that it is accurate to use only eight linear fits to model full capacitance matrices despite the strong heterogeneity in the capacitance values due to edge effects. Thus, the coefficients from Table 4.1 and 4.2 are used to construct the capacitance matrices of complete array arrangement. These modeled capacitance matrices are subsequently compared to the "true" capacitance matrices extracted with the *Q3D Extractor*.

In the first part of this analysis, the transmission of random data bits over the TSVs is considered, implying equal amounts of 1-bits and 0-bits transmitted over all TSVs (*i.e.*, $p_i = 0.5$ for all i). To show the reusability of the model, the coefficients—fitted by means of extracted capacitance values for 5×5 arrays and a TSV length of $50\,\mu$m—are used to: first, reproduce the capacitance matrix of the 5×5 arrays; and second, generate the capacitance matrices of 7×7 arrays with an increased TSV length of $70\,\mu$m.

The resulting overall RMSE and MAE values for the modeled capacitance matrices compared to the respective extracted matrices are reported in Table 4.5. Again all numbers are reported as a percentage of the according $C_{\mathrm{n,prev}}$ value to better relate the results for different TSV technologies. The table reveals that the errors of the presented model do not exceed 14.6 %, while the previous model shows errors as high as 52.9 % of $C_{\mathrm{n,prev}}$.

Only for some third-order-adjacent TSV pairs at the array edges, the proposed model results in errors higher than 10 %. Furthermore, using only a single parameter to model all coupling capacitances between diagonally adjacent

Table 4.5.: Normalized MAE and RMSE values of the proposed and the previous TSV capacitance models for full capacitance matrices and a transmission of random bit-patterns.

			Proposed model				*Previous model*			
TSV parameters			**5 × 5**		**7 × 7**		**5 × 5**		**7 × 7**	
r_{tsv}	d_{min}	f_{s}	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
[μm]	[μm]	[GHz]	[%]	[%]	[%]	[%]	[%]	[%]	[%]	[%]
1.0	4.0	6.0	14.5	3.5	13.2	2.2	47.9	12.1	52.9	7.7
1.0	4.0	11.0	11.6	2.5	10.1	1.7	41.1	9.9	46.4	6.4
1.0	4.5	6.0	14.6	3.6	13.4	2.2	50.0	12.8	51.4	8.1
1.0	4.5	11.0	11.6	2.7	10.4	1.7	43.8	10.5	46.4	6.6
2.0	8.0	6.0	12.5	2.6	10.4	1.6	44.3	10.9	49.3	7.1
2.0	8.0	11.0	10.6	2.1	8.9	1.3	39.5	9.2	43.2	5.8
2.0	8.5	6.0	12.7	3.0	10.3	1.7	44.2	10.8	45.8	6.8
2.0	8.5	11.0	10.8	2.3	9.0	1.3	41.0	9.5	43.2	5.9

TSVs—even if both TSVs are located at an edge—results in some noticeable model errors. Nevertheless, the RMSE value for the proposed model does not exceed 3.6 % in all cases. In contrast, the previously used capacitance model results in RMSE values that are by a factor of 3.5 × to 4.5 × higher than the ones of the proposed model.

For the 7×7 arrays, the presented approach models the capacitance matrices with RMSE values that are even smaller (always below 2.3 %), which proves the scalability of the model. The model errors for larger array dimensions are even lower due to the decreased ratio of edge TSVs over middle TSVs, resulting in a decreased impact of the neglected coupling effects at the edges. Also, the error values for the previously used model decrease with a decreased ratio of edge TSVs (maximum RMSE for the 7×7 arrays is equal to 8.1 %). Furthermore, as expected from the previous subsection, the accuracy of both models slightly increases for the less aggressively scaled TSV radius of 2 μm.

The increased errors of the traditional model in the previous analysis, compared to the proposed one, is only due to the non-considered edge effects in the previous analysis, as a transmission of random bits over the TSVs was considered. For scenarios in which the bit probabilities are not equally distributed, the MOS effect results in a further heterogeneity in the capacitance matrices, which is not captured by the previous capacitance model.

Hence, the accuracy of the models for three scenarios in which 1 and 0 bits are not equally distributed is quantified as the second analysis. Through the previous analysis, it was already proven that the relative model errors increase with a further scaled TSV radius and smaller array dimensions. Thus,

Table 4.6.: Normalized MAE and RMSE values of the TSV capacitance models for full capacitance matrices and the transmission of one-hot-encoded data.

Data	**TSV parameters**			*Proposed model* **5 × 5**		*Previous model* **5 × 5**	
	r_{tsv} [μm]	d_{min} [μm]	f_{s} [GHz]	MAE [%]	RMSE [%]	MAE [%]	RMSE [%]
One-hot encoded	1.0	4.0	6.0	13.7	4.2	67.9	16.6
	1.0	4.0	11.0	10.6	2.7	54.9	12.7
	1.0	4.5	6.0	13.8	3.3	69.7	17.2
	1.0	4.5	11.0	11.0	2.5	53.4	12.4
Inverted one-hot enhoced	1.0	4.0	6.0	14.6	4.0	37.2	10.3
	1.0	4.0	11.0	11.0	3.3	39.9	9.6
	1.0	4.5	6.0	17.0	5.0	41.8	11.4
	1.0	4.5	11.0	11.6	3.5	41.6	10.1
Half-inverted one-hot encoded	1.0	4.0	6.0	20.5	5.5	51.0	12.9
	1.0	4.0	11.0	14.9	3.9	47.5	11.0
	1.0	4.5	6.0	22.8	6.3	54.4	13.8
	1.0	4.5	11.0	13.0	3.8	47.7	11.2

in this analysis, only the 5×5 arrays with the smaller TSV radius of $1\,\mu$m are considered in order to report worst-case error values for the models. The first investigated scenario represents the transmission of random data words that are one-hot encoded.[3] One-hot-encoded data words result in a 1-bit probability of $1/25$ for all TSVs of a 5×5 array. The second analyzed scenario is equal to the first one but with inverting TSV drivers, resulting in swapped logical 1-bit and 0-bit probabilities for the TSVs. Hence, all 1-bit probabilities are equal to $24/25$. In the last analyzed scenario, only every second line has an inverting driver. Consequently, in the third scenario, the 1-bit probabilities for all TSVs with an odd-numbered index i are equal to $1/25$, while the 1-bit probabilities off all TSVs with an even-numbered index i are $24/25$. The last scenario is less common in practice than the other ones, but it quantifies the error of using the mean of only two 1-bit probabilities as the feature variable for fitting in case of huge variations in the 1-bit probabilities.

The resulting normalized RMSE and MAE values of the two capacitance models, compared to full parasitic extractions, are reported in Table 4.6. For the more conventional two scenarios, the proposed model results in error values that are in accordance with the ones previously reported for random data.

[3] A more in-depth explanation of the properties of one-hot encoded data can be found in the later Subsection 5.1.3 on Page 93 of this thesis.

In contrast, the error values of the previous model increase significantly due to the neglected MOS effect. The normalized MAE and RMSE of the previous model reach values as high as 69.7 % and 17.2 %, respectively. Consequently, the proposed model results here in MAE and RMSE values that are more than five times smaller than the ones for the previous model.

For the third less common scenario, the errors of the proposed model slightly increase due to the applied averaging of the two bit probabilities for the MOS-effect modeling. Here, the proposed model shows MAE and RMSE values of up to 22.8 % and 6.3 %, respectively. Thus, the RMSE value—being the more important metric for power and performance estimation—is still tolerable. Moreover, the proposed model still results in an improvement in the MAE and RMSE values by a factor of 2.2× to 3.7× compared to the previous model. This proves a substantial superiority of the proposed capacitance model for a wide range of scenarios.

4.4.2. Accuracy for the Estimation of the TSV Power Consumption and Performance

After the accuracy of the presented and the previous capacitance model has been quantified in the previous subsection, in the following, the accuracy of the capacitance models for the estimation of the TSV power consumption and performance is analyzed. For this purpose, a *Spectre* circuit simulation for the general setup, presented in the evaluation section of the previous chapter (Section 3.4), is performed.

A 4×5 array over which 10,000 random 20-bit patterns are transmitted is analyzed. This particular array shape is chosen to additionally prove that the proposed model remains valid for non-quadratic arrays, which have not been considered yet. The investigated transmission of random data/bit-patterns results in equally distributed 1 and 0 bits on the TSVs. This significantly reduces the errors of the previously used capacitance model, neglecting the MOS effect.

Edge effects—neglected as well by the previous capacitance model—decrease with an increasing significant frequency, as shown before. Thus, to report optimistic values for the previous model, the significant frequency is increased from 6 GHz to 10 GHz compared to the setup from Section 3.4. An increased significant frequency is a result of stronger TSV drivers. Thus, the strength of the 22-nm drivers in the circuit-simulation setup is increased by doubling the $W/L_{\min}$ of each transistor compared to the previous chapter. Again, the TSV array is modeled in the circuit simulations by a 3π-RLC equivalent circuit that was extracted with the *Q3D Extractor*. The same TSV radius, minimum pitch, and length as in Section 3.4 is considered (*i.e.*, $r_{\text{tsv}} = 1\,\mu\text{m}$, $d_{\min} = 4\,\mu\text{m}$, and $l_{\text{tsv}} = 50\,\mu\text{m}$). Thus, only the array dimensions and the significant frequency are changed in the 3D model used for parasitic extraction. The rest of the circuit-

simulation setup is kept the same as in the evaluation part of the previous chapter.

Three different capacitance matrices are investigated for the high-level estimation of the power consumption and performance employing the formulas presented in Chapter 3 of this thesis. The first one is the exact capacitance matrix from the *Q3D Extractor*; the second one the capacitance matrix according to the proposed model; and the third one the capacitance matrix according to the previously used model. An effective TSV load capacitance of about 2 fF is added to the diagonal/self-capacitance entries of each of the three capacitance matrices to account for the drivers. Furthermore, the driver-dependent parameters for the delay/performance estimation with Equation (3.49) (*i.e.*, R and $T_{\mathrm{D},0}$) have to be fitted again due to the modified drivers.

In contrast to the circuit simulations, which take several hours, the high-level estimations based on the derived formulas are performed within a few seconds. Nevertheless, using the more complex proposed capacitance model instead of the previous one does not significantly impact the overall run time (*i.e.*, the run time for creating the capacitance matrices, determining the bit-level properties, and finally applying the formulas). The bit-level analysis for the transmitted patterns either-way heavily determines the overall run time, not the generation of the capacitance matrix with the models.

Investigated is the power consumption and performance of a middle TSV, a single-edge TSV, and a corner TSV. Moreover, the overall power consumption and performance of the whole array is analyzed. For all cases, the high-level estimates are compared with the results from the circuit simulations.

In Table 4.7, the results are reported. The results show that using the previous capacitance model leads to an under-estimation of the power consumption of the TSV array by 21.1 %. In contrast, a high-level estimation employing the proposed capacitance model results in a precise power estimation as the error is as low as 2.6 %. This significant error for the previous model makes the usage of the proposed model inevitable for a precise power estimation.

While the proposed, as well as the previous, capacitance model, allow for a precise estimation of the power consumption of the middle TSVs (error below 3.5 %), the previous model results in a large underestimation of the power consumption of the TSVs located at the edges (25.5 % to 42.0 %). According to the estimates obtained for the previous model, a TSV located at a single edge of the array results in a 29.6 % lower power consumption than a middle TSV. For a corner TSV, the previous capacitance model even indicates a 51.7 % lower power consumption. However, in reality, a TSV at a single edge and a TSV at a corner only result in power consumptions that are by 8.6 % and 19.5 % lower than for a middle TSV, respectively.

The proposed model accurately estimates the power consumption of the middle and the edge TSVs. Only for the corner TSVs, the proposed model

Table 4.7.: Power consumption, P, and maximum propagation delay, $\hat{T}_{\mathrm{pd}}$, of the TSVs in a 4×5 array for the transmission of random data according to circuit simulations and high-level estimates. The percentage errors relative to the circuit simulation results are reported in bold for the high-level estimates.

			High-level estimation		
		Circuit sim.	**Exact capacitances**	**Proposed model**	**Previous model**
Middle TSV	P [μW]	6.40	6.39 (**-0.2 %**)	6.52 (**+1.8 %**)	6.19 (**-3.4 %**)
	$\hat{T}_{\mathrm{pd}}$ [ps]	117.0	116.1 (**-0.8 %**)	115.9 (**-0.9 %**)	113.7 (**-2.8 %**)
Edge TSV	P [μW]	5.85	5.83 (**-0.3 %**)	5.68 (**-2.9 %**)	4.36 (**-25.5 %**)
	$\hat{T}_{\mathrm{pd}}$ [ps]	109.5	108.7 (**-0.07 %**)	104.8 (**-4.2 %**)	82.04 (**-25.1 %**)
Corner TSV	P [μW]	5.15	5.04 (**-2.1 %**)	4.65 (**-9.7 %**)	2.99 (**-42.0 %**)
	$\hat{T}_{\mathrm{pd}}$ [ps]	98.8	87.5 (**-11.5 %**)	79.8 (**-19.3 %**)	53.2 (**-46.1 %**)
Complete array	P [μW]	117.53	116.82 (**-0.5 %**)	114.51 (**-2.6 %**)	92.7 (**-21.1 %**)
	$\hat{T}_{\mathrm{pd}}$ [ps]	117.0	116.1 (**-0.8 %**)	115.9 (**-0.9 %**)	113.7 (**-2.8 %**)

results in a non-negligible miss-prediction by 9.7 %. This relatively large error is mainly due to the neglected third-order coupling capacitances. However, since always only four corner TSVs exist in an array arrangement, the power consumption of the whole array is still estimated precisely with the proposed model.

The maximum propagation delay occurs for the middle TSVs in an array arrangement as they have a larger accumulated capacitance. For middle TSVs, the previous and the proposed capacitance model only differ slightly in this analysis due to the balanced bit probabilities on each TSV. Consequently, both capacitance models result in a relatively accurate estimation of the maximum performance of the TSV array (error below 3 %). However, note that the previous capacitance model results in an underestimation of the maximum propagation delay of an edge TSV by 25.1 %. In contrast, the proposed model results in an underestimation of this quantity by only 4.2 %. According to the estimates obtained for the previous capacitance model, the maximum delay of an edge TSV is 27.8 % lower compared to a middle TSV. However, the maximum delay value of an edge TSV is actually only 6.4 % lower. This fact is not essential for the performance estimation in case of a transmission of random data as the performance is defined as the reciprocal of the overall maximum propagation delay of any TSV in the array (*i.e.*, $\hat{T}_{\mathrm{pd}}$). However, the fact becomes essential for the derivation of optimizing techniques. The reason is that the proposed model reveals that one already has to optimize the middle-TSV as well as the edge-TSV performance to increase the overall performance by over 10 %.

For corner TSVs, the proposed and the previously used capacitance model results in an underestimation of the maximum propagation delay by 19.3 % and 46.1 %, respectively. These more significant errors for both models are also because the driver-related coefficients in the formula for the propagation-delay estimation have been fitted to be most accurate for larger effective capacitance values. However, since the measured maximum propagation delay of a corner TSV is significantly lower compared to the value for a middle or an edge TSV, this limitation of the proposed approach can be typically tolerated for performance estimation.

4.5. Conclusion

In this chapter, an abstract, universally valid, and yet scalable model for the capacitances of TSV-array arrangements was presented. The model extends the previously used model such that the edge effects, as well as the MOS effect, are considered in an abstract way.

An in-depth evaluation showed that the model is highly accurate compared with precise, but computationally expensive, full parasitic extractions (overall NRMSE below 5 %). Furthermore, the proposed model allows us to precisely estimate the power consumption and performance of TSVs on high abstraction levels (error below 3 %). In contrast, a high-level estimation employing the previously used capacitance model showed an underestimation of the TSV power consumption by over 21 %.

The proposed capacitance model enables the systematic derivation of universally valid optimization approaches in Part III of this thesis. Furthermore, the strong heterogeneity in the capacitances of edge and middle TSVs, revealed by the proposed model, can be exploited to improve the TSV power consumption and performance effectively. Another important finding of this chapter is that the capacitance values can be optimized through the 1-bit probabilities, which can be affected by data-encoding techniques. Thus, exploiting the TSV edge effects and the TSV MOS effect has the potential to improve the efficiency of optimization techniques remarkably.

CHAPTER 5

Estimation of the Bit-level Statistics

In Chapter 3, formulas were presented that allow for a fast and yet precise estimation of the power consumption and the performance of 3D interconnects on high abstraction levels. These formulas require a knowledge of the interconnect capacitances and the bit-level switching. Thus, an abstract and yet precise model to estimate the TSV capacitances—complementary to the well-known capacitance model for metal wires—was presented in the subsequent Chapter 4.

The switching characteristics can already be determined on higher levels of abstraction with general-purpose programming languages, used to perform bit-level analyses, as done in the previous two chapters of this thesis. However, to obtain the required statistical parameters (*e.g.*, the switching activities, α_i) for the estimation of the power consumption, requires detailed knowledge (*i.e.*, a representative amount of samples) of the transmitted data. Unfortunately, representative data samples are sometimes hard to get, especially at the early

design stages of a system. Furthermore, bit-level simulations can be even to slow for the simulation of the application-dependent power consumption of complex systems with a data transfer of several GB/s. Thus, generally valid models to estimate the bit-level statistics out of higher-level parameters (*e.g.*, word-level statistics) are required.

Moreover, such models are needed to evaluate competing architectures in terms of power consumption at early design stages [83, 84], as well as to derive low-power techniques for 3D interconnects such as the ones presented in Part III of this thesis. Not only do the models allow us to find the right low-power technique despite not having access to a representative number of samples of the transmitted data, but they also enable to derive techniques that are generally efficient for a broader set of data streams without any further modification. With the help of abstract models for the bit-level statistics, an extensive set of low-power coding techniques for metal wires have been derived in the past, which are efficient, for example, for most address or DSP signals [85–98]. Such generally usable techniques are also desired for 3D interconnects.

While the new transmission medium (*i.e.*, TSVs) demands a new capacitance model, existing models for the bit-level statistics remain valid for 3D integration since the abstract bit-level characteristics are independent of the transmission channel. Thus, existing models for the data-stream statistics, such as the ones presented in [83, 84, 99–105], can be fully reused for 3D integration.

However, all existing models only allow to precisely estimate the bit-level statistics in case of a sequential transmission of a single data stream and not if multiple data streams are transmitted in parallel, using a time-multiplexed transmission scheme. Time multiplexing implies here that a single physical-transmission medium is used to transmit multiple data streams in parallel by granting the different sources access to the medium in a time-multiplexed manner.

The general advantage of data-stream multiplexing is that a lower total amount of interconnects is required. However, a high interconnect performance is required in order to not have any data jams as a result of the multiplexing. However, due to their low resistance, TSVs can provide a relatively high data-throughput compared to long global metal wires but have the drawback of a large area occupation, making them a great candidate to apply data-stream multiplexing. Moreover, the TSV performance can be even further improved by the optimization techniques presented in Part III of this thesis—if this is needed to apply data-stream multiplexing. Another advantage of data-stream multiplexing for 3D integration is that it helps to overcome TSV-related yield issues due to the resulting reduced overall TSV count. Thus, data-stream multiplexing is often applied for 3D interconnects why even high-level-synthesis approaches for an automatic implementation of the technique for TSVs exist [63, 64, 106].

Additionally, one of the recently most popular interconnect architectures, a virtual-channel-based network on chip (NoC), results in a multiplexed access to the transmission medium in order to transmit multiple data packets in parallel over a single physical channel [48, 98]. This fact even further increases the probability that data-stream multiplexing is applied for 3D interconnects.

On the downside, data-stream multiplexing has a strong negative impact on the interconnect power consumption for the transmission of correlated data streams (*e.g.*, from DSP applications) and furthermore destroys the efficiency of existing low-power codes (LPCs) as shown in this chapter. Thus, there is a strong need for a technique to estimate the bit-level statistics in the presence of data-stream multiplexing. Garcia *et al.* have presented a model to estimate the switching activities, α_i, in case of a continuous pattern-by-pattern multiplexing of two data streams [104]. However, conventional multiplexing approaches generally result in a complex multiplexing of more than two data streams. Furthermore, due to the large coupling capacitances in modern metal-wire and TSV arrangements, it is of significant importance to also be capable of estimating the switching correlations of the bits, $\gamma_{i,j}$.

This chapter presents the first accurate high-level method to estimate the bit-level statistics in the presence of any data-stream multiplexing. In detail, the existing data models are extended by a technique that enables the estimation of the bit-level statistics also if the individual patterns of multiple streams are multiplexed in a complex manner. An in-depth evaluation shows that the proposed technique enables to predict the bit-level statistics with maximum absolute errors (MAEs) and normalized root-mean-square errors (NRMSEs) that are always below 3 % and 1 %, respectively. Without the proposed technique, the error values are almost by a factor of 10 × larger. Furthermore, the evaluation shows that not using the proposed model to estimate the power savings of competing LPCs can result in the implementation of coding architectures that even increase the power consumption. The reason is that data-stream multiplexing can destroy the efficiency of coding techniques completely. In contrast, using the proposed technique enables us to always pick the best suitable coding architecture, as it always predicts the coding gains precisely.

The remainder of this chapter is organized as follows. First, existing models to estimate the bit-level statistics for non-multiplexed data streams are reviewed in Section 5.1. Afterward, the concept of data-stream multiplexing and its impact on the power consumption are outlined in Section 5.2. The proposed model to estimate the bit-level statistics for multiplexed data streams is presented in Section 5.3 and evaluated in Section 5.4. In the following Section 5.5, a case study is presented in which the model is integrated into the 3D-NoC simulator from [CP2] to show the usability of the proposed model beyond the scope of this thesis. Finally, the chapter is concluded.

Core parts of this chapter are published as a proceeding of the "29th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)" (see [P2]). The paper was selected as one of the best papers of the symposium for an invited, peer-reviewed, extension in "Elsevier Integration" (see [A6]). This extension also covers the detailed integration of the proposed technique into the NoC simulator, which was mainly carried out by Jan Moritz Joseph.[1] Consequently, this extended content is not part of this thesis. The interested reader is here referred to the according publication.

5.1. Related Work—Bit-Level Statistics for Single Data Streams

In this section, formulas to estimate the bit-level statistics for a sequentially (non-multiplexed) transmission of data streams are reviewed. As shown in the previous two chapters of this thesis, the three bit-level statistics that are required to estimate the interconnect power consumption are the switching activities, α_i, the switching correlations, $\gamma_{i,j}$, and the bit probabilities, p_i. Since the switching correlations are always zero in the case of temporally misaligned signal edges, irrespective of the data stream (see Equation (3.27) on Page 45), temporally aligned edges are considered throughout this chapter. Due to the wide range of different types and models to estimate their bit-level statistics (*e.g.*, [83,84,99–105]), this section only covers the data types that are considered throughout this thesis.

5.1.1. Random Data

An m-bit random data stream implies in this thesis a sequence of integer data words in the range from 0 to $2^m - 1$ (unsigned base-2 representation), which are completely uncorrelated and uniformly distributed.

Hence, for the transmission of samples of a random data stream, each possible bit pattern, as well as every possible pattern transition, occurs with the same probability. Consequently, each bit of the binary words has a 1-bit probability of 50 %, the switching probability of the bits is 50 %, and all bit pairs switch completely uncorrelated. Hence, the required bit-level statistics are

$$\alpha_i = \mathbb{E}\{\Delta b_i^2\} = 0.5; \tag{5.1}$$

$$\gamma_{i,j} = \mathbb{E}\{\Delta b_i \Delta b_j\} = 0; \tag{5.2}$$

$$p_i = \mathbb{E}\{\Delta b_i\} = 0.5. \tag{5.3}$$

To validate these formulas, a 16-bit random data stream containing 100,000 samples is generated synthetically, which is subsequently analyzed. Exact

[1] The extended work is published with both, Jan Moritz Joseph and Lennart Bamberg, denoted as lead authors with equal contribution.

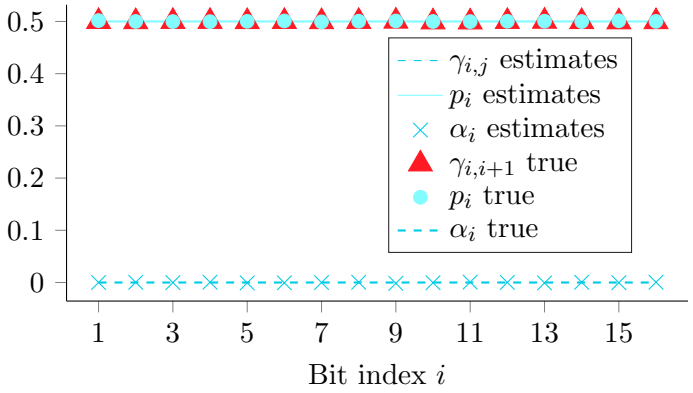


Fig. 5.1: Bit-level statistics of a stream of purely random data words (*i.e.*, uniform-distributed and uncorrelated) according to bit-level simulations and higher-level estimates.

bit-level statistics, from bit-level simulations that were performed with *Python*, alongside the values from Equation (5.1) to (5.3), are plotted in Figure 5.1. The results show the correctness of the formulas.

5.1.2. Normally Distributed Data

In many scenarios (*e.g.*, DSP applications), the data words tend to be Gaussian/normally distributed. The rationale is the central limit theorem, which states that the normalized sum of mutually independent, zero-mean random variables tends to follow a Gaussian distribution, even if the original variables themselves are not normally distributed [108].

The joint probability density function of the discrete-time mean-free Gaussian processes X_k and X_{k-1} (X_k delayed by one sample instance) is mathematically expressed as

$$f_{PD}(x,y) = \frac{1}{2\pi\sigma^2\sqrt{1-\rho^2}}\, e^{-\frac{x^2+y^2-2\rho xy}{2\sigma^2(1-\rho^2)}}, \tag{5.4}$$

where X_k and X_{k-1} are substituted for x and y, respectively [105]. In this equation, σ is the standard deviation of the mean-free data words:

$$\sigma = \sqrt{\mathbb{E}\{X_k^2\}}. \tag{5.5}$$

ρ is the word-level correlation of the data words:

$$\rho = \frac{\mathbb{E}\{X_k \cdot X_{k-1}\}}{\sigma^2}. \tag{5.6}$$

Throughout this work, an m-bit Gaussian/normally distributed data stream with a given standard deviation, σ, and correlation, ρ, is a sequence of digitalized

data words (signed integer, base-2 representation) that were generated according to the probability density function described by Equation (5.4).

In the following, the model to estimate the required bit-level statistics of such normally distributed data streams, as a function of the word-level statistics σ and ρ, is briefly reviewed. For a more in-depth analysis of the bit-level properties of normally distributed data, please refer to [83, 84, 101, 103, 105].

After analyzing the switching of the bits in a set of samples from DSP signals, Landman and Rabaey concluded that the bit-switching statistics can be represented by a piece-wise-linear model composed of three regions [83, 84]. For each of the three regions, different bit-level properties are assumed in the model. The first region ranges from the least significant bit (LSB) of the data words to the so-called "LSB break-point", BP_{lsb}, and the second region ranges from the "MSB break-point", BP_{msb}, to the most significant bit (MSB) of the data words. All bits in between BP_{lsb} and BP_{msb} compose the third intermediate region. In [84], the following formulas to calculate the breakpoints were derived:

$$BP_{\text{lsb}} = \log_2 \sigma + \log_2 \left(\sqrt{1-\rho^2} + \frac{|\rho|}{8} \right); \tag{5.7}$$

$$BP_{\text{msb}} = \log_2 (3\,\sigma). \tag{5.8}$$

The bits in the LSB region up to BP_{lsb} tend to be completely random. Hence, the toggle activity and the 1-bit probability of a bit belonging to this first region are approximately 0.5. Furthermore, the switching correlation of a bit belonging to this region with any other bit (from the same or another region) is 0 in the model.

The bits in the MSB region show a very strong spatial correlation, which implies that if one of the bits in the MSB region toggles, in nearly all cases, all other bits in the region toggle as well and in the same direction. According to [101], the toggle activity of the bits in the MSB region can be calculated from the world-level correlation via

$$\alpha_{\text{msb-reg}} = \frac{\cos^{-1}(\rho)}{\pi}. \tag{5.9}$$

For uncorrelated data words (*i.e.*, $\rho = 0$), $\alpha_{\text{msb-reg}}$ is equal to 0.5. With an increasing (positive) pattern correlation, the switching activity of the bits in the MSB-region decreases toward 0. For a decreasing (negative) pattern correlation, the switching activity of the MSBs decreases toward 1. The switching correlation, $\gamma_{i,j}$, between two bits that both belong to the MSB region, is also equal to $\alpha_{\text{msb-reg}}$ due to their strong spatial correlation. For all bits in the MSB region, the 1-bit probability is 0.5.

In the model, a continuous linear model between the bit-level statistics in the LSB and the MSB region is assumed for the bit-level statistics in the intermediate region. Thus, the model to estimate the bit-level statistics of zero-mean, normally distributed data by means of the word-level statistics is expressed through the following formulas:

$$\alpha_i = \begin{cases} 0.5 & \text{for } i \leq BP_{\text{lsb}} \\ 0.5 + \frac{(i-BP_{\text{msb}})(\alpha_{\text{msb-reg}}-0.5)}{BP_{\text{msb}}-BP_{\text{lsb}}} & \text{for } BP_{\text{lsb}} < i < BP_{\text{msb}} \\ \alpha_{\text{msb-reg}} & \text{for } i \geq BP_{\text{msb}}; \end{cases} \tag{5.10}$$

$$\gamma_{i,j} = \begin{cases} 0 & \text{for } i \text{ or } j \leq BP_{\text{lsb}} \\ \frac{(i-BP_{\text{msb}})(\alpha_{\text{msb-reg}})}{BP_{\text{msb}}-BP_{\text{lsb}}} & \text{for } BP_{\text{lsb}} < i < BP_{\text{msb}} \text{ and } j > i \\ \frac{(j-BP_{\text{msb}})(\alpha_{\text{msb-reg}})}{BP_{\text{msb}}-BP_{\text{lsb}}} & \text{for } BP_{\text{lsb}} < j < BP_{\text{msb}} \text{ and } j < i \\ \alpha_{\text{msb-reg}} & \text{for } i \text{ and } j \geq BP_{\text{msb}}; \end{cases} \tag{5.11}$$

$$p_i = 0.5. \tag{5.12}$$

To validate these formulas, five Gaussian-distributed 16-bit data streams are generated, with each containing 100,000 samples. The individual data streams are generated for different word-level correlations and standard deviations in the range from -0.9 to 0.9 and 2^7 to 2^{10}, respectively. Resulting exact bit-level statistics from bit-level simulations, alongside the estimates from Equation (5.10) to (5.12), are plotted in Figure 5.2. The results show a good agreement between the bit-level simulations and the model estimates.

Finally, please note that although the previously reviewed formulas were derived by Landman *et al.* for primary Gaussian-distributed data, they are usable for other uni-modal distributions (*e.g.*, Laplace) as well [83].

5.1.3. One-Hot-Encoded Data

An m-bit one-hot-encoded data stream implies throughout this thesis a sequence of integer numbers in the range from 0 to $m-1$, which are completely uncorrelated and uniformly distributed. The representation of the data is one-hot, meaning that in each of the m possible binary data words (all occurring with the same probability), only a single bit is 1 while all other bits are 0 [109]. The pattern with a 1 on its i^{th} bit represents the integer value $i-1$. For example, the four possible data words of a 4-bit one-hot-encoded data stream "0001", "0010", "0100" and "1000" represent the integer numbers 0, 1, 2 and 3, respectively. A one-hot encoding is often used for states due to its robustness against single flipped bits (Hamming distance between two valid codewords/states is always two). Furthermore, a one-hot encoding is often used in machine-learning applications to encode categorical data.

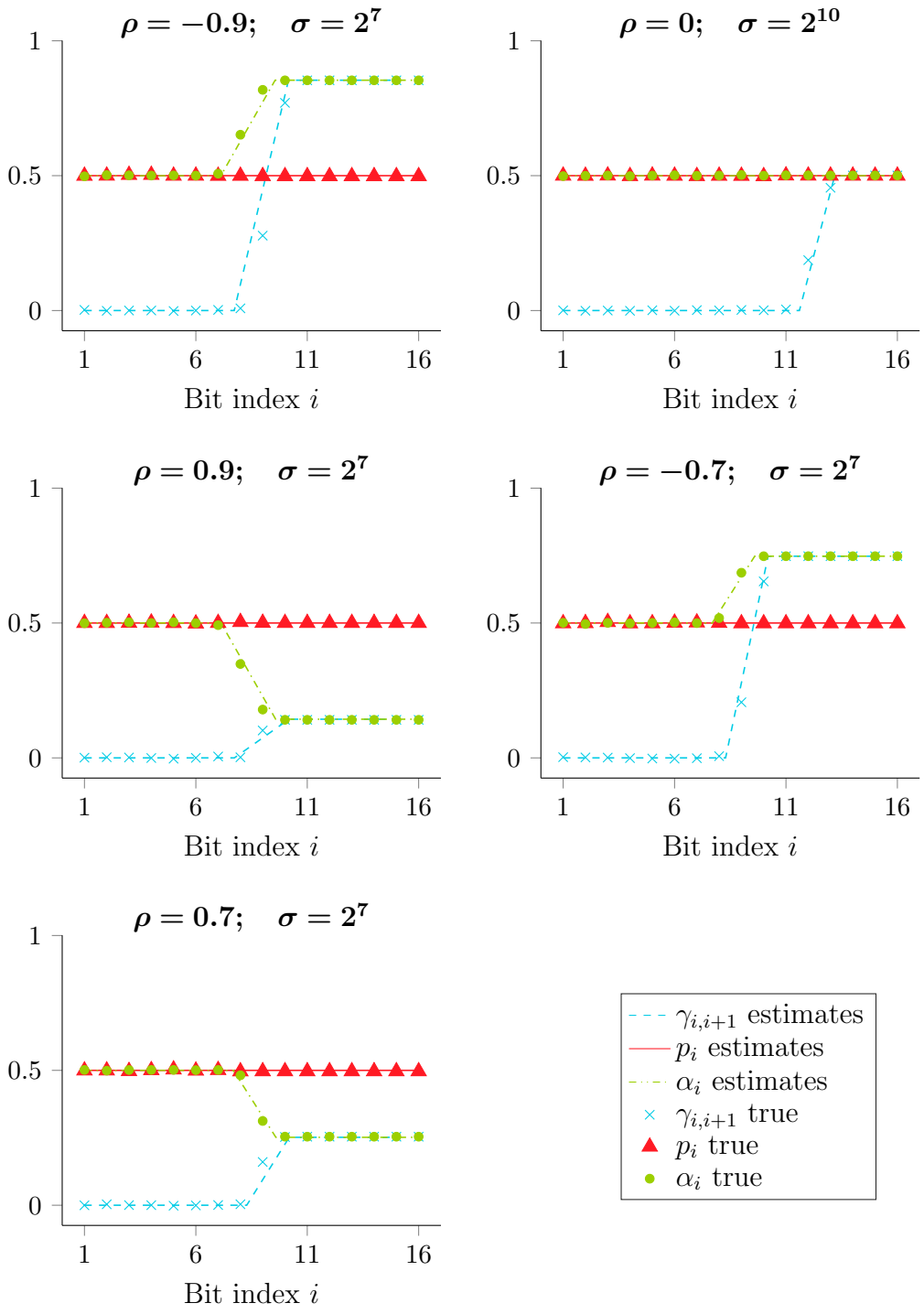


Fig. 5.2: Bit-level statistics of normally distributed data according to bit-level simulations and higher-level estimates.

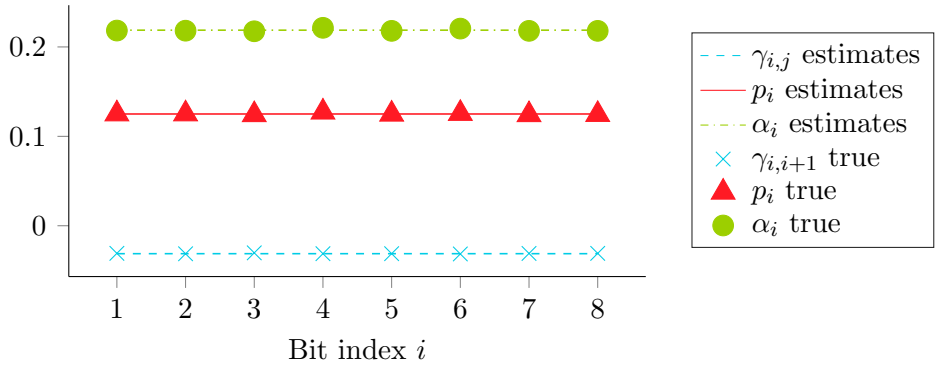


Fig. 5.3: Bit-level statistics of one-hot-encoded random data streams according to bit-level simulations and higher-level estimates.

To derive the bit-level statistics of one-hot-encoded random data is a straightforward task. Since each of the m patterns occurs with the same probability $1/m$, while each bit is only 1 in a single pattern, the 1-bit probability of each bit is

$$p_i = \frac{1}{m}. \tag{5.13}$$

Thus, the probability of a bit being 0 is $(m-1)/m$. Hence, the probability of a switching from logical 1 to logical 0 in two respective clock-cycles is $(m-1)/m^2$ for all bits. Since the probability of a 0 to 1 transition obviously is the same as of a 1 to 0 transition, the switching activity of each bit is

$$\alpha_i = 2\frac{m-1}{m^2}. \tag{5.14}$$

In a one-hot-encoded data stream, two or more bits can never switch in the same direction, only in the opposite direction. Hence, the $\Delta b_i \Delta b_j$ values can only be 0 or -1, which implies negative switching correlations for the bit pairs. A switching in the opposite direction for two bit values b_i and b_j (*i.e.*, $\Delta b_i \Delta b_j = -1$) occurs only for two pattern combinations. For an exemplary 4-bit one-hot-encoded data stream, opposite transitions in the first two bits only occur for the two pattern sequences "0001" $\rightarrow$ "0010" and "0010" $\rightarrow$ "0001". Thus, the switching correlation of the bit-pairs are

$$\gamma_{i,j} = -\frac{2}{m^2}. \tag{5.15}$$

The derived formulas are validated through an exemplary 8-bit one-hot-encoded data stream made up of 100,000 samples. Resulting exact bit-level statistics from bit-level simulations, alongside the estimates from the derived formulas, are plotted in Figure 5.3. The results prove the correctness of the formulas.

5.1.4. Sequential Data

Sequential data streams build a specific data type for which subsequent data words are strongly correlated. This data type is one of the most prominent ones since sequential data can be found in most real systems in the form of address signals or program counters [93].

A sequential m-bit data stream is a stream of integer words (unsigned base-2 representation) in which the value of a data word always is the increment (+1) of the value of the previous data word, except in case of a branch where the value changes randomly. Sequential signals are strongly correlated but tend to be uniformly distributed. Hence, the 1-bit probability of each bit is

$$p_i = 0.5. \tag{5.16}$$

For an ideal sequential data stream without any branches, the toggle activity of the bits decreases exponentially:

$$\alpha_i = 2^{1-i}. \tag{5.17}$$

Considering that a branch occurs with a probability of p_{branch}, the switching activities of the bits are precisely estimated by

$$\alpha_i = (1 - p_{\text{branch}}) \cdot 2^{1-i} + p_{\text{branch}} \cdot 0.5. \tag{5.18}$$

Due to the uniform distribution, the individual bits toggle uncorrelated for sequential data:

$$\gamma_{i,j} = 0. \tag{5.19}$$

The derived formulas for the bit-level statistics of sequential data streams are validated through two synthetically generated sequential data streams with different branch probabilities of 5 % and 20 %. Each of the two synthetically generated data streams is made up of 100,000 8-bit words. Exact bit-level statistics from bit-level simulations, alongside the estimates of the formulas, are plotted in Figure 5.4. The results prove the correctness of the derived formulas.

5.2. Data-Stream Multiplexing

In this section, the two main use-cases of data-stream multiplexing are outlined: TSV-count reduction to reduce TSV-related area and yield issues, and virtual-channel usage to improve the NoC performance. Furthermore, it is shown that data-stream multiplexing can have a dramatic impact on the interconnect power consumption and the efficiency of a wide set of low-power techniques.

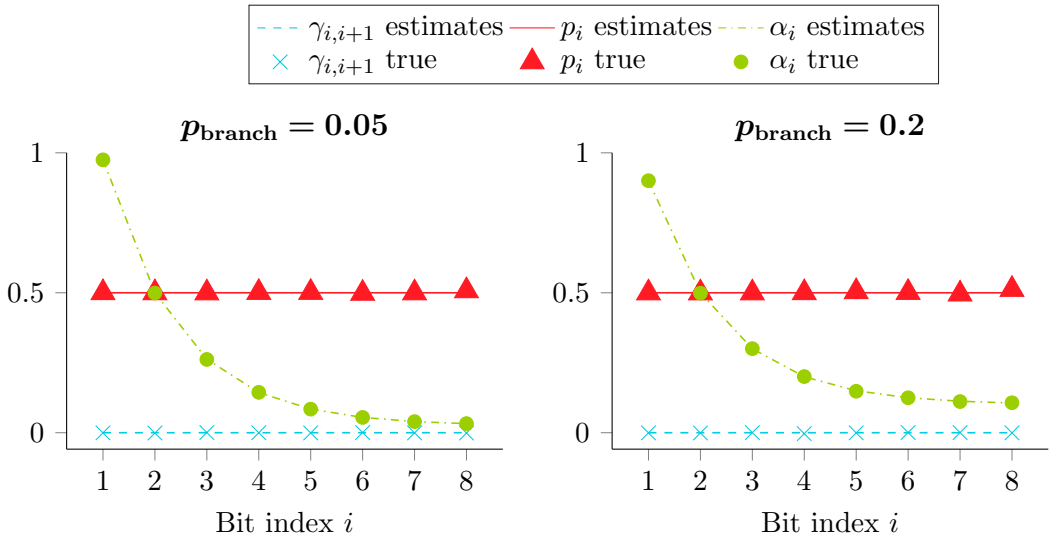


Fig. 5.4: Bit-level statistics of sequential data according to bit-level simulations and higher-level estimates.

5.2.1. Data-Stream Multiplexing to Reduce the TSV Count

Consider three different data streams A, B, C with the same bit-width m that have to be transmitted from one die of a 3D IC to an adjacent one. One possibility would be to integrate an array containing $3m$ data TSVs so that the three data streams can be transmitted in parallel. Following this simple approach can sometimes result in a too-large TSV area occupation (due to the large TSV radius) and a low overall yield (due to the high TSV-defect probability). To tackle this issue, the concept of TSV multiplexing has been introduced [64].

Consider the simple example where the source of data stream A only produces new data samples with a rate that is two times lower than the maximum frequency at which we can transmit bits over a TSV. Furthermore, we assume that for the data streams B and C, the rate of new samples is even four times lower than the maximum TSV transmission rate. In this scenario, it is possible to integrate only an array of size m over which the samples are transmitted in a time-multiplexed manner as $A_1\, B_1\, A_2\, C_1\, A_3\, B_2\, A_4\, C_3 \ldots$ (here, the indices are the sample numbers of the individual data streams). In this example, the number of required TSV can be drastically reduced by using a multiplexed data transmission to one third for the cost of an extra circuit that manages the multiplexed transmission.

5.2.2. Data-Stream Multiplexing in NoCs

The main functionality of an NoC is to forward data in the form of packets from source processing elements to destination processing elements [98]. Thereby, packets pass multiple routers (multi-hop transmission). Each packet is made up of several link-level atomic pieces called flits (flow control units). The first flit, referred to as head flit or short header, contains important packet information such as the source address, the destination address, and the packet length. This header information is typically used in each router to set the routing for all flits of the according packet. In some NoCs, packets contain more than one leading head flit. The remainder of a packet is made up of data flits, containing the actual information transmitted from the source to the destination processing element.

A flit is the minimum amount of data that can be exchanged between two routers. In most NoC architectures, the size of a flit is equal to the size of a phit, which is the amount of data that can be transmitted per clock cycle over a single NoC link. Hence, this particular scenario is considered in the following. Thus, a link is an m_{flit}-bit interconnect structure, where m_{flit} is the bit width of a flit. If the two routers that are connected by the link are located in different dies of the 3D stack, the interconnect structure is (partially) made up of TSVs, else the interconnect structure only consists of metal wires (and the related drivers).

In an NoC, several packets (*i.e.*, data streams) typically traverse the network simultaneously (parallelism). Hence, multiple packets may compete for the usage of a single link. An exemplary scenario, where three packets, *P1*, *P2*, and *P3*, compete for a single link is considered in the following. Without virtual channels, one packet (*e.g.*, *P1*) will be granted to use the link, while the other two packets, *P2* and *P3*, are blocked until *P1* is completely transmitted.

This has two main disadvantages. First, due to the commonly applied flit-level flow control [107], the next flit of the granted packet *P1* might be blocked due to a contention elsewhere in the network. In this scenario, the link is idle even though another packet, *P2* or *P3*, could make effective use of the channel. The second problem is that blocked packets result in heavily unequally balanced transmission times for different messages.

To mitigate this degradation and to provide better quality-of-service guaranties, the bandwidth of a link is divided among different packets using virtual channels. More than one logical buffer is integrated into each input port when virtual channels are implemented. Thereby, different packets can be buffered simultaneously and thus transmitted interleaved. While the assignment of virtual channels (input buffers) is packet-based, the arbitration for physical-channel bandwidth is on a flit-by-flit basis [107].

Different techniques exist to arbitrate the usage of a physical channel (*i.e.*, the interconnect structure). The most common one is a balanced time-multiplexing [110]. Thereby, the available bandwidth of the link is equally partitioned on each virtual channel (strong fairness). Recalling the example of three packets *P1*, *P2*, and *P3*, requesting the usage of one link at the same time, the available link bandwidth is shared by transmitting the flit sequence $P1_1\, P2_1\, P3_1\, P1_2\, P2_2\, P3_2 \ldots P1_{NF}\, P2_{NF}\, P3_{NF}$, assuming that no congestion occurs in the preceding paths and that the amount of virtual channels is greater than two. Here, the indices are the flit numbers of the packets containing *NF* flits. Another common technique uses a priority-based multiplexing [111]. Thereby, each virtual channel is associated with a different priority, depending on the service class of the according message. The transmission of packets with a lower priority is deferred if higher priority packets make use of the link. This guarantees a good quality of service for high priority traffic at the cost of a fairness reduction.

Summarized, virtual channels result in not only sequential package transmissions but also in multiplexed/interleaved transmissions of multiple packages. The probability of a data-stream multiplexing in two subsequent cycles is slightly lower for a priority-based virtual-channel arbitration.

5.2.3. Impact on the Power Consumption

In this subsection, the impact of data-stream multiplexing on the interconnect power consumption and coding-based low-power techniques is outlined. For the sequential transmission of highly correlated data (*e.g.*, DSP signals or address signals), the switching activities of the bits are small compared to the transmission of uncorrelated data, as outlined in the previous section. Hence, the interconnect power consumption is relatively low for the sequential transmission of such data. However, when the data streams are transmitted in a multiplexed manner, this beneficial behavior is lost as the individual data streams are uncorrelated among each other. Consequently, the interconnect power consumption can increase drastically when data-stream multiplexing is applied.

This is validated by the following analysis. The power consumption for the transmission of two data sets is investigated. Each set contains 100,000 Gaussian-distributed 16-bit data words with a word-level correlation ρ of 0.99 and a standard deviation σ of 2^8. A clock frequency of 1 GHz is considered for the pattern transmission. Two interconnect structures are investigated. A TSV array and a metal-wire bus, both driven by commercial 40-nm drivers. To obtain the metal-wire capacitances, a commercial wire tool, provided by the vendor of the 40-nm technology, is used. The metal-wire width and pitch are set to 0.3 μm and 0.6 μm, respectively. Through-silicon-via parasitics are generated with the capacitance model presented in the previous Chapter 4.

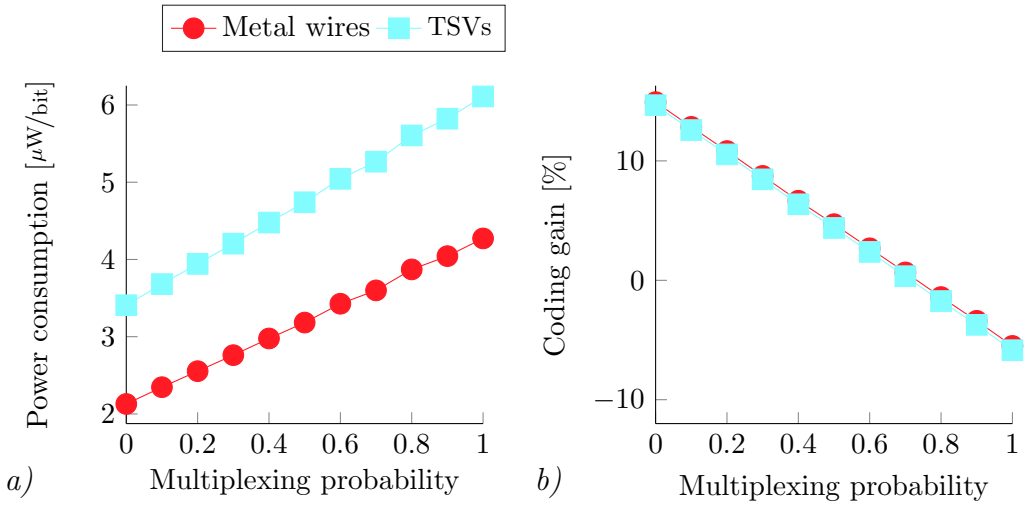


Fig. 5.5: Effect of data-stream multiplexing on: *a)* the interconnect power consumption for the transmission of correlated data words; *b)* the gain of CBI coding for the transmission of two random data streams.

For the quadratic 4×4 TSV array, a typical TSV radius of $2\,\mu$m and the corresponding minimum TSV pitch of $8\,\mu$m are analyzed. Again, a common TSV depth of $50\,\mu$m is considered, while the metal wires have a length of $100\,\mu$m to obtain a somewhat comparable power consumption for both structures. The parasitic capacitances of the drivers, required for the power estimation, are extracted from the process design kit of the 40-nm standard-cell library.

Employing exact bit-level simulations and the precise high-level formula from Chapter 3, the power consumption for various data-stream-multiplexing probabilities is quantified. The multiplexing probability is defined as the probability of a change in the transmitted data stream so that the next transmitted bit-pattern belongs to another data stream than the previous one.

In Figure 5.5a, the resulting interconnect power consumption, normalized by the number of transmitted bits per cycle, is plotted over the data-stream-multiplexing probability.[2] For no data-stream multiplexing, the power consumption of the metal wires and the TSVs is $2.10\,\mu\text{W/b}$ and $3.40\,\mu\text{W/b}$, respectively. In the case of a continuous cycle-by-cycle data-stream multiplexing (*i.e.*, multiplexing probability equal to 1), the power consumption of the links is almost doubled (metal wires: $4.27\,\mu\text{W/b}$; TSVs: $6.11\,\mu\text{W/b}$), compared to the scenario without data-stream multiplexing. This increase shows the possible dramatic effect of data-stream multiplexing on the power consumption.

[2]The power values are plotted normalized to evaluate an LPC technique in the following, which adds redundancy to the data words, reducing the number of effectively transmitted bits per cycle.

To illustrate the impact of data-stream multiplexing on the efficiency of existing low-power coding techniques, the transmission of two random data streams, encoded individually with the classical bus invert (CBI) technique [112], is analyzed. The CBI technique is one of the most well-known low-power techniques based on higher abstraction levels and belongs to the class of low-power codes (LPCs). The conceptual idea of low-power coding is to integrate an encoder and a decoder architecture at the beginning and the termination of a communication path, respectively, which adapt the bit-level characteristics of the signals which are transmitted over the physical medium such that the power consumption is reduced.

For example, the CBI technique aims for a reduction in the switching activities of the bits. For this purpose, one redundant bit is added to the transmitted codewords. Depending on the logical value of this additional so-called invert bit, the remaining bits are either equal to the initial/unencoded data word (if invert bit is 0) or equal to its bit-wise negation (if invert bit is 1). A memory-based block is used to set the invert bit if this reduces the Hamming distance between the previously encoded codeword and the currently encoded codeword. Thereby, CBI coding reduces the switching activity for random data by up to 25 % [112].

The impact of CBI coding on the power consumption of the interconnect structures, over the data-stream-multiplexing probability, is investigated for an end-to-end encoding scheme. End-to-end encoding implies here that the data is encoded and decoded at its source and sink, respectively, and not on a link-level. Such an end-to-end encoding is, for example, widely used in NoCs, which employ a multi-hop/link transmission of the data. Here, with a link-level coding, the data must be encoded and decoded once per link it traverses [48,80]. In contrast, with an end-to-end encoding, each data stream only undergoes one encoding and decoding in total, which maximizes the coding gain as it minimizes the power consumption induced by the coding architectures.

Figure 5.5b includes the results of the analysis. If the two data streams are transmitted without any data-stream multiplexing, the CBI technique leads to a reduction in the TSV and metal-wire power consumption by approximately 14 %. However, with increasing data-stream multiplexing, the coding efficiency vanishes. Due to the redundancy the technique adds to the codewords (invert bit), the encoding approach even increases the power consumption by 6 % for a continuous data-stream multiplexing. One can show in the same manner that the effectiveness of other existing LPCs vanishes as well if they are used on an end-to-end basis in the presence of data-stream multiplexing.

The previous analyses underline the strong need to consider the effect of data-stream multiplexing on the interconnect power consumption in high-level models. Not only to precisely estimate the power requirements but also to systematically derive efficient LPCs whose power gains do not vanish for multiplexed data streams.

5.3. Estimation of the Bit-Level Statistics in the Presence of Data-Stream Multiplexing

In this section, a model to estimate the bit-level statistic in the presence of data-stream multiplexing is presented. Let us assume that up to n_{ds} different data streams, represented by D^1 to $D^{n_{\mathrm{ds}}}$, are transmitted over an interconnect structure.

One can use one of the various existing models to obtain the bit-level statistics for the case of a sequential transmission of the individual data streams. For the approach presented in this chapter, not only the 1-bit probabilities for the individual data streams (*i.e.*, p_i^1 to $p_i^{n_{\mathrm{ds}}}$) have to be determined through the existing methods, but joint bit-probability matrices, symbolized as $\mathbf{JP^1}$ to $\mathbf{JP^n}$, with

$$JP^x_{i,j} = \mathbf{E}\{b_i^x b_j^x\}. \tag{5.20}$$

The i^{th} diagonal entry of a joint bit-probability matrix, $JP^x_{i,i}$, is equal to the 1-bit probability of b_i in the data stream D^x:

$$\mathbf{E}\{b_i^x b_i^x\} = \mathbf{E}\{b_i^x\} = p_i. \tag{5.21}$$

A non-diagonal entry, ${JP^x}_{i,j}$, is equal to the probability that both bits b_i and b_j of a data word of D^x are 1.

First, the data-flow-independent **JP** matrices, are used to estimate the switching activities (${\alpha_i}^{x\to y} = \mathbb{E}\{\Delta b_i^2\}^{x\to y}$) and the switching correlations (${\gamma_i}^{x\to y} = \mathbb{E}\{\Delta b_i \Delta b_j\}^{x\to y}$ with $i \neq j$) for the case that two data streams D^x and D^y are transmitted in a continuous/cycle-by-cycle multiplexed manner:

$$\begin{aligned}\mathbb{E}\{\Delta b_i \Delta b_j\}^{x\to y} &= \mathbb{E}\{(b_i^y - b_i^x)(b_j^y - b_j^x)\} \\ &= \mathbb{E}\{b_i^y b_j^y + b_i^x b_j^x - b_i^y b_j^x - b_i^x b_j^y\} \\ &= JP^y_{i,j} + JP^x_{i,j} - JP^y_{i,i} JP^x_{j,j} - JP^x_{i,i} JP^y_{j,j}.\end{aligned} \tag{5.22}$$

For i equal to j, the self-switching probability ${\alpha_i}^{x\to y}$ is obtained and, for i unequal to j, the switching correlation ${\gamma_{i,j}}^{x\to y}$. Note that in the derived equation, it is exploited that the cross-correlation of two different data streams is zero, resulting in

$$\begin{aligned}\mathbb{E}\{b_i^y b_j^x\} &= \mathbb{E}\{b_i^y\} \cdot \mathbb{E}\{b_j^x\} \\ &= JP^y_{i,i} \cdot JP^x_{j,j}.\end{aligned} \tag{5.23}$$

Employing the resulting switching statistics for a continuous multiplexing as well as the switching statistics for no multiplexing (*i.e.*, $\alpha_i^x = \alpha_i^{x\to x}$ and $\gamma_{i,j}^x = \gamma_{i,j}^{x\to x}$), the overall switching statistics for an interconnect structure with

given multiplexing probabilities, $M_{x,y}$, can be determined through

$$\alpha_i = \sum_{x=1}^{n_{\mathrm{ds}}} \sum_{y=1}^{n_{\mathrm{ds}}} (M_{x,y} + M_{x+n_{\mathrm{ds}},y}) \cdot \alpha_i^{x \to y}; \tag{5.24}$$

$$\gamma_{i,j} = \sum_{x=1}^{n_{\mathrm{ds}}} \sum_{y=1}^{n_{\mathrm{ds}}} (M_{x,y} + M_{x+n_{\mathrm{ds}},y}) \cdot \gamma_{i,j}^{x \to y}. \tag{5.25}$$

In these equations, α_i represents the overall switching activity of the i^{th} line of the interconnect structure, while $\gamma_{i,j}$ is the overall switching correlation of the i^{th} and the j^{th} line. $M_{x,y}$ is equal to the probability of transmitting a pattern of data stream D^y after transmitting a pattern of data stream D^x (*i.e.*, a switching in the active data stream from D^x to D^y). Thus, $M_{x,x}$ is equal to the probability of two subsequently transmitted patterns belonging to the same data stream D^x (*i.e.*, no multiplexing event). $M_{x,x+n_{\mathrm{ds}}}$ is equal to the probability that the link is idle, holding a pattern of stream D^x. Therefore, $M_{x+n_{\mathrm{ds}},y}$ is the probability of transmitting a pattern of D^y after being idle, holding a pattern of D^x. The $2n_{\mathrm{ds}} \times 2n_{\mathrm{ds}}$ matrix $\mathbf{M}$, containing the $M_{x,y}$ values, is referred to as the data-flow matrix since it contains the abstract information about the word-level data flow.

Note that in Equation (5.24) and (5.25) the switching characteristics for a sequential/non-multiplexed transmission of a data stream D^x (*i.e.*, $\alpha_i^{x \to x}$ and $\gamma_{i,j}^{x \to x}$) must be determined with the traditional methods. The reason is that the derived Equation (5.22) only holds if the pattern of D^x and D^y are uncorrelated among each other.

Besides the switching activities and the switching correlations, the 1-bit probabilities are furthermore required for interconnect structures that contain TSV segments in order to estimate the TSV capacitances. The 1-bit probabilities do not depend on the bit switching. Thus, the overall 1-bit probabilities are independent of the multiplexing; it is only relevant how many samples of each data stream are transmitted on average. Consequently, previous methods generally can estimate the 1-bit probabilities correctly. In the proposed model, the overall 1-probabilities are calculated via

$$p_i = \sum_{x=1}^{n_{\mathrm{ds}}} \sum_{y=1}^{n_{\mathrm{ds}}} (M_{x,y} + M_{x,y+n} + M_{x+n_{\mathrm{ds}},y}) \cdot p_i^y. \tag{5.26}$$

Summarized, Equation (5.22) to (5.26) extend existing methods such that the bit-level statistics can also be estimated in the case of a complex time-multiplexed transmission of several data streams over arbitrary 2D and 3D interconnect structures.

5.4. Evaluation

The proposed method is evaluated in this section. First, the accuracy of the technique to predict the bit-level statistics for an extensive set of data-flow scenarios is quantified in Subsection 5.4.1. Afterward, the implications of the model for low-power coding techniques are outlined.

5.4.1. Model Accuracy

The accuracy of the presented model to estimate the bit-level statistics is investigated in this section. For this purpose, the parallel transmission of two to five different data streams is analyzed for different multiplexing probabilities. Furthermore, each simulation is executed 1,000 times for 16-bit data words and 1,000 times for 32-bit data words.

To cover the large space of data-type combinations in the best possible way, in each run, the synthetically generated data streams vary randomly. The pattern distribution of each single data stream is either uniform, Gaussian, or log-normal. For the last two distributions, the standard deviation of the n-bit data words varies in the range from $2^{n/10}$ to 2^{n-1}. Furthermore, the world-level pattern correlation varies between 0 to 1 for the individual data streams. Exact knowledge of the bit-level statistics of the individual data streams is assumed throughout the evaluation section for an isolated analysis of the accuracy of the proposed method.

Analyzed are the switching statistics as they are required to estimate the interconnect power consumption for aligned signal edges. The matrix $\mathbf{S}_{\mathbb{E}}$, which contains the required switching statistics in the matrix formulation of the power consumption (Equation (3.46) on Page 51), is constructed twice through Equation (3.45). Once employing estimates of the α_i and $\gamma_{i,j}$ values obtained with the newly proposed model, and once employing the exact switching characteristics determined by means of complex bit-level simulations. Afterward, the RMSE and MAE values of the proposed model for all $\mathbf{S}_{\mathbb{E}}$ entries are reported. Since the switching properties are independent of the interconnect type (*i.e.*, TSV or metal wire), this particular setup enables us to quantify the general accuracy of the proposed method for a possibly large set of scenarios.

In Figure 5.6, the results, normalized by the mean of the according $\mathbf{S}_{\mathbb{E}}$ entries, are reported. The resulting error values show that the presented method enables a precise estimation of the switching statistics independent of the multiplexing probabilities, the number of multiplexed data streams, or the bit width of the data words. For all analyzed scenarios, the NRMSE of the estimates is in the range of 0.6 % to 0.8 %. The normalized maximum absolute error (NMAE), for all 5,120,000 estimated $\mathbf{S}_{\mathbb{E}}$ entries, does not exceed 2.8 %. In contrast, if multiplexing effects are neglected, normalized errors in the estimates bigger than 50 %, with NRMSEs of up to over 25 %, are obtained for the same analysis.

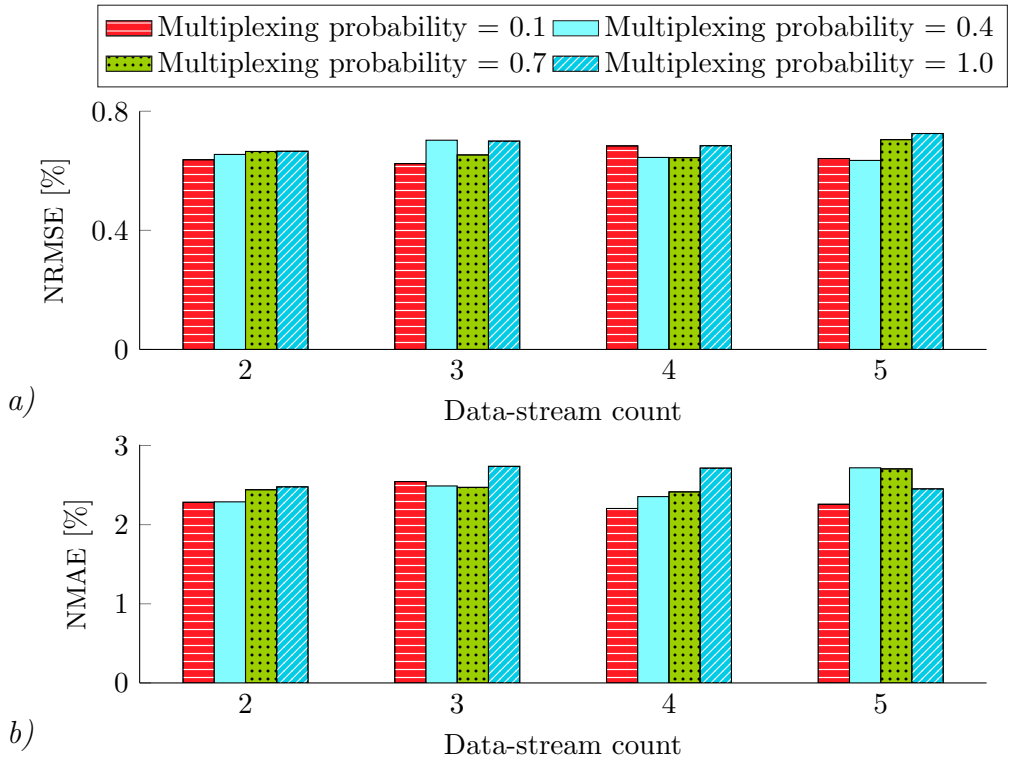


Fig. 5.6: Errors of the proposed model to estimate the switching statistics for various data-stream counts and multiplexing probabilities: *a)* NRMSE; *b)* NMAE.

Furthermore, the experiment proves that, as expected, the switching activities increase linearly with an increasing multiplexing probability (see also Figure 5.5a), while the total number of multiplexed data streams generally does not affect the switching properties. Since switching is only determined by the bit-level differences between directly consecutively transmitted data words, a multiplexing of two data streams, on average, results in the same interconnect power consumption as a multiplexing of three or more data streams. Thus, without loss of generality, in the remainder of this section, the analysis is restricted to scenarios where only two data streams are multiplexed.

5.4.2. Low-Power Coding

As outlined in Section 5.2, there is a strong need for coding techniques which reduce the interconnect power consumption in the presence of data-stream multiplexing. The modeling technique presented in this chapter enables a fast estimation of the efficiency of such techniques. Furthermore, the proposed method enables the design of effective end-to-end data-encoding techniques for time-multiplexed data streams.

The simultaneous transmission of 2 MB of data, in the form of 16-bit patterns, over a 4×4 TSV array and a metal-wire bus, is analyzed. Thereby, the same interconnect structures as in Section 5.2 are considered. The analyzed two (unencoded) data streams are made up of either data words that tend to be uniformly distributed with a word-level correlation of 0.99, or data words that are completely random. All three possible data-stream combinations are analyzed: Two data streams that are each made up of different correlated data words; two data streams that contain different random data words; and one data stream that is made up of correlated data words while the other one contains random data words.

Here, CBI coding is investigated exemplary as the low-power-coding technique for random data. A correlator coding is analyzed as the LPC for correlated data. Correlator coding performs a bit-wise XOR operation for every data word with the previous data word of the stream [104]. For the analyzed strongly correlated data words, unencoded, the bits in the MSB-region rarely change/toggle between two subsequent bit patterns (see Subsection 5.1.2). However, over a large number of patterns, every bit is with a probability of 50 % logical 1 (equally distributed bit values). Thus, the XORing of subsequent data words, performed by the correlator encoder, leads to codeword MSBs that are nearly stable on logical 0 [104]. Therefore, due to the considered inverting interconnect drivers, the MSBs of the actually transmitted correlator-encoded data words are almost stable on logical 1.

Investigated is the power consumption, normalized to the number of bits that are effectively transmitted per codeword over the multiplexing probability. The according power quantities are determined twice: Once using the high-level model presented in this chapter to estimate the bit-level statistics, and once using the exact values from bit-level simulations.

The results are presented in Figure 5.7. Markers indicate the power quantities obtained with bit-level simulations, which are in perfect accordance with the estimates obtained with the proposed formulas, illustrated by the solid and dashed lines. Thus, the method proposed in this chapter enables a fast and precise estimation of the efficiencies of LPCs for multiplexed data streams. For example, the model precisely predicts the decreasing coding efficiency of the CBI technique with an increasing multiplexing probability as well as the simultaneously increasing efficiency of the correlator coding for the transmission of two correlated data streams. Thus, it allows identifying that for a high data-stream-multiplexing probability, a different low-power technique than CBI coding is required. In contrast, correlator coding performs particularly well in this scenario.

Without the proposed high-level model, it is not trivial to explain this observation, which complicates the design of new coding techniques for time-multiplexed data streams. However, the derived formulas provide a clear

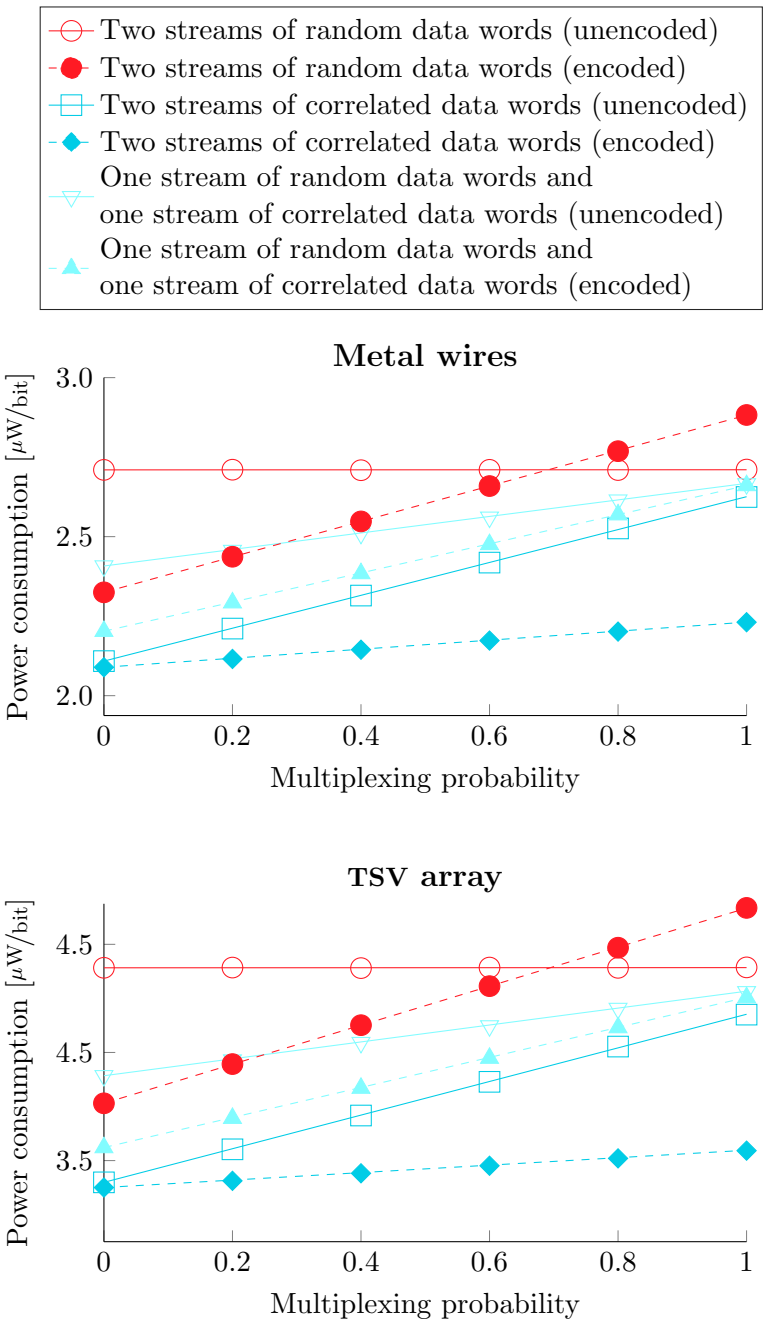


Fig. 5.7: Effect of existing low-power coding techniques on the 3D-interconnect power consumption for the transmission of correlated and completely random data words in the presence of data stream multiplexing. Marks indicate results for exact bit-level simulations; lines indicate estimates of the proposed method.

answer. A CBI encoding only affects the switching probabilities for the sequential (non-multiplexed) transmission of data streams. However, it does not affect the joint bit probabilities, $JP_{i,j}$. Thus, as revealed by the proposed model, it cannot decrease the power consumption for a continuous data-stream multiplexing. Moreover, due to the induced redundancy, it even increases the power consumption per effectively transmitted bit. In contrast, the correlator coding leads to MSBs nearly stable on logical 1, resulting in increased $JP_{i,j}$ values. Consequently, as revealed by Equation (5.22), the switching activities are reduced, and the switching correlations increased, for the multiplexed transmission of the data streams. Both effects have a positive impact on the power consumption. Hence, correlator coding can significantly improve the interconnect power consumption in the presence of data stream multiplexing.

Summarized, the method proposed in this chapter reveals an important message for the design of end-to-end LPCs: In order to obtain efficient coding approaches for interconnect architectures which use a data-stream multiplexing, the technique must affect not only the switching for the individual data streams but also the bit probabilities.

5.5. Case Study

This section presents a case study to prove that the proposed method is suitable for usage in combination with system simulators. In detail, the usage of the proposed model to perform an estimation of the interconnect energy consumption in a heterogeneous 3D NoC with a *SystemC* simulator is analyzed. Furthermore, an exploration of the efficiency of LPC techniques with the NoC simulator is performed.

A heterogeneous 3D vision system on chip (VSoC) is considered in this case study. In contrast to mere CMOS image sensors, VSoCs are used to capture and process the images in a single chip. This overcomes the limitations of traditional systems due to expensive image transmissions between sensors and processors, especially for high frame rates and resolutions [113].

The NoC architecture of the analyzed VSoC has a flit width of 16 bit with one head and 31 body flits (payload) per packet, supports up to four virtual channels per port, and has an input-buffer depth of four. The full 3D stack consists of three layers: One mixed-signal layer, which contains six CMOS image sensors (S1–S6) at the top, one underlying memory layer, and at the bottom one digital layer containing the actual processors. In this case study, the transmission of 8-bit gray-scale-image pixels (two per flit) from the sensors to the memory is analyzed and optimized toward a low interconnect energy consumption. Since the analyzed NoC uses a dimension-order (XYZ) routing [48], while images are read by the cores from memory, the traffic from the memory to the sensors can be analyzed without considering the traffic of the cores.

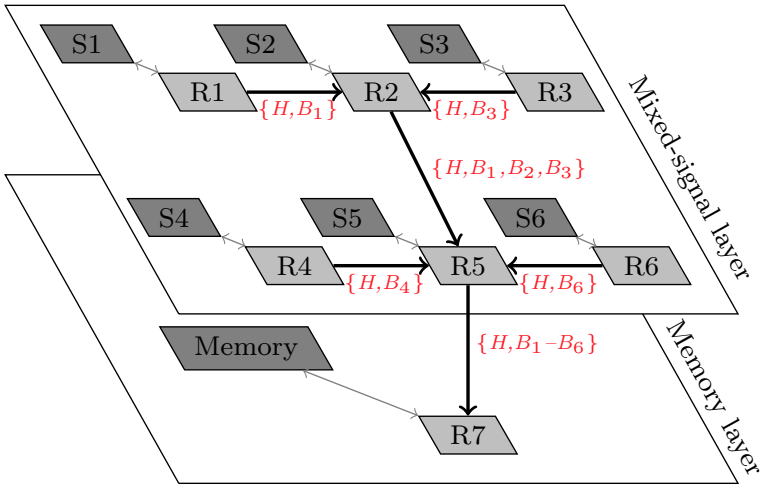


Fig. 5.8: Part of the 3D VSoC analyzed in the NoC case study.

Thus, the structure sketched in Figure 5.8 is analyzed. In total, seven different data/flit types are transmitted over the links: One for head flits, H, and six for body flits, B_1–B_6 (one per source). Each of the six image sensors in the mixed-signal layer is linked to one router, R1–R6, which are connected by 2D/metal-wire links. Router R5 is connected via a 3D/TSV link with router R7 in the memory layer. Connected to R7 is a memory block to store the sensor data. Thus, over the links connecting R1→R2, R3→R2, R4→R5, and R6→R5, only data stemming from one sensor and head flits are transmitted, resulting in unused virtual channels. Usage of virtual channels occurs in the links R2→R5 and R5→R7 as they are part of the transmission paths from multiple sensors to the memory. In this section, the same metal-wire and TSV structures, as in the previous sections of this chapter, are considered.

To obtain the data-flow matrices $\mathbf{M}$ (one per link)—containing the abstract information about the data flow—the 3D-NoC simulator RATATOSKR [CP2] is used. It is publicly available on *GitHub* at `https://github.com/jmjos/ratatoskr`. The simulator targets heterogeneous 3D NoCs and is implemented in *C++* using *SystemC*. The simulator is chosen as it allows to arbitrary configure the router architecture (*e.g.*, virtual-channel count, buffer depth), the network topology, and the data flow.

The simulator is extended in cooperation with Jan Moritz Joseph to have the capabilities to provide detailed statistics on the dynamic power consumption of the NoC links through the high-level models derived and validated in the present and the previous two chapters of this thesis. For this purpose, the task model of the simulator is extended to support colored tokens in the Petri nets (colors represent different data types). Additionally, a unidirectional link model is implemented, which monitors the data flow between routers. Finally, the $\mathbf{M}$

Table 5.1.: Results of the NoC case study.

Data	**Energy per transmitted packet [pJ]**		
	Bit-level sim.	*Proposed model*	*Standard model [80]*
Unencoded	4.18	4.15	2.40
Gray encoded	4.11 (**-1.67 %**)	4.09 (**-1.44 %**)	2.23 (**-7.08 %**)
Correlator encoded	2.69 (**-35.64 %**)	2.68 (**-35.42 %**)	2.71 (**+12.92 %**)

matrices are built by evaluating the color (*i.e.*, data type) of flits traversing the links. This method is superior to saving a whole protocol of the transmitted flit types since this would require memory, which linearly increases with the number of simulated clock cycles. In contrast, the effort to save the data-flow matrices is constant over the simulated clock cycles because the matrices are of size $2n_{ds} \times 2n_{ds}$ for $n_{ds} - 1$ transmitted data streams.[3]

A simulation for the analyzed 3D-NoC architecture with the extended NoC simulator is executed for a mean traffic injection rate of 20 % per sensor element. To allow for subsequent bit-level simulations (to obtain reference values), the simulator is temporarily modified in a way that it saves the whole protocol of the transmitted flits. After the simulation, the energy quantities per transmitted packet are determined based on: first, the proposed method; second, the standard method used in [72, 79, 80] (neglects effects of data-stream multiplexing); and, third, exact bit-level simulations.[4] Again, thereby, a perfect knowledge of the bit-level statistics of the individual data streams is assumed.

Two different traffic scenarios are analyzed, representing realistic use cases of a VSoC. In the first scenario, all six sensors capture road images with a resolution of 512 × 512 pixels during daylight and, in the second scenario, during the night. These particular traffic scenarios are chosen as they result in relatively high errors for the proposed technique, which requires that the cross-correlation between the individual data streams is zero to be exact. This assumption is not necessarily correct if all sensors capture pictures of the same environment from different perspectives.

In the first row of Table 5.1, the results are presented. They show that, for realistic NoC-traffic scenarios, the proposed model can precisely predict the energy consumption (error below 1 %). In contrast, a high-level model that neglects the effect of data stream multiplexing, leads to an error of almost 50 %, even though, in the analyzed scenario, less than 50 % of the links make use of virtual channels. However, the two links which use virtual channels show the highest energy consumption and, for these links, the traditional model leads to

[3] One additionally data type is used for head flits, which are assumed to be random.

[4] Circuit simulations to obtain the energy quantities on the lowest levels of abstraction are here not possible due to the large complexity of the system.

an underestimation of the energy consumption by more than a factor of 4 ×.

Additionally, the integration of the two low-power techniques correlator-coding and Gray-coding [114] for the body flits is investigated. These bit-overhead-free LPCs are analyzed as a wider bit width for the codewords would increase the buffer cost of a router, which is a crucial concern in NoCs. Gray encoding reduces the switching activities for a sequential transmission of the highly correlated pixels, while a correlator mainly affects the bit probabilities. As outlined in the previous section of this chapter, correlator coding shows an excellent coding efficiency for multiplexed correlated data streams, while the efficiency for no multiplexing is very limited.

The energy quantities for the transmission of the encoded data are shown in the second and third row of Table 5.1. Furthermore, the (estimated) percentage reductions in the energy consumption of the links due to the LPC techniques are shown in bold. The presented method—in accordance with the results from the bit-level simulations—indicates that the correlator leads to a drastically better coding gain (36 % reduction in the link energy requirements compared to 1 %). In contrast, the standard model to estimate the bit-level statistics falsely predicts a much higher coding efficiency for Gray coding and even a significant increase in the energy consumption for the correlator coding. This underlines once again that using any other than the model presented in this chapter can result in the implementation of inefficient coding techniques and to a dramatic underestimation of the power requirements.

5.6. Conclusion

In this chapter, the first method to precisely estimate the bit-level statistics in the case of a complex data-stream multiplexing was presented. The method enables the precise estimation of the 3D-interconnect power consumption without the need for complex bit-level simulations for systems that use TSV-sharing approaches or virtual-channel NoCs.

Furthermore, the proposed technique enables the systematic derivation of techniques that effectively reduce the interconnect power consumption of such systems. This is of particular importance since the interconnect power consumption can dramatically increase if data streams are transmitted in a time-multiplexed manner. Furthermore, an important finding of this chapter for the design of end-to-end LPCs is that the joint bit probabilities of the transmitted data have to be maximized in order to effectively improve the 3D-interconnect power consumption in the presence of extensive data-stream multiplexing.

An evaluation of the proposed technique, in the form of a case study for a 3D-NoC simulator, showed that using standard models results in an underestimation of the interconnect power consumption by up to a factor of 4 ×. Moreover, the standard models showed to miscalculate the gains of LPCs dramatically. In

contrast, the power consumption of the NoC links, as well as the coding gains, are estimated precisely when existing models are extended by the proposed method (all errors below 1 %).

Part III: Optimization

CHAPTER 6

Low-Power Technique for 3D Interconnects

The edge effects, as well as the MOS effect, outlined in Chapter 4 of this work, can significantly impact the TSV power consumption. Nevertheless, existing low-power techniques do not consider nor exploit these effects, as they have been designed for traditional metal-wire interconnects which do not show these effects. Low-power techniques designed explicitly for TSV-based interconnects are generally still lacking, mainly due to the previous absence of a pattern-dependent model for the TSV power consumption. Such a model was presented in the previous Part II of this thesis. Consequently, efficient low-power techniques for TSV-based 3D interconnects can now be derived.

In this chapter, the first-ever coding-based low-power technique for 3D interconnects is presented. This technique is based on the observation that the varying bit-level statistics of typical data streams in modern SoCs can be exploited to effectively reduce the TSV power consumption by an intelligent net-to-TSV assignment. The reason for this is the strong heterogeneity in the TSV capacitances due to the edge effects. Furthermore, an assignment of some

bits as inverted (*i.e.*, logically negated) can further decrease the power consumption, mainly due to the MOS effect. Moreover, the proposed power-optimal assignment can boost the efficiency of traditional low-power codes (LPCs) for TSV-based 3D interconnects.

The presented approach only affects the local net-to-TSV assignments within the individual TSV arrays, while the global net-to-TSV-bundle assignment remains routing-optimal. A modification in the local net-to-TSV assignment, even in the worst-case, has a negligible impact on the parasitics of the full 3D-interconnect paths for reasonable array sizes, as shown in this chapter. Thus, the implementation costs of the proposed technique are negligibly low.

A key contribution of this chapter is a formal method to find the power-optimal net-to-TSV assignment (including inversions) for any given TSV arrangement by means of the formulas for the estimation of the TSV power consumption and capacitances derived in Part II of this thesis. The method considers the bit-level statistics of the data that is transmitted over the nets and the capacitances of the TSV array to find the power-optimal assignment.

To overcome the necessity to have in-depth knowledge about the transmitted data and the TSV capacitances, systematic net-to-TSV assignments, which are generally valid for the transmission of correlated and normally distributed data words, are contributed as well. These systematic assignments are derived by means of the insights provided by Part II of this thesis. The proposed technique is evaluated for a broad set of data streams and modern TSV arrangements of different sizes. These analyses show that the proposed technique can reduce the TSV power consumption by over 45 %, despite its negligible implementation costs.

The remainder of this chapter is structured as follows. First, the fundamental idea of the proposed technique is outlined in Section 6.1. Subsequently, a formal method to determine the power-optimal TSV assignment is derived in Section 6.2. Systematic assignments for correlated or normally distributed patterns are presented in Section 6.3. Afterward, the combination of the proposed technique with traditional LPC techniques is discussed in Section 6.4. In Section 6.5, the proposed method is evaluated in-depth. Finally, the chapter is concluded.

The core idea of this chapter is published as a proceeding of the "55$^{\text{th}}$ Design Automation Conference (DAC)" (see [P2]). Parts considering the costs of a non-routing-minimal local net-to-TSV assignment are published in "Elsevier Integration" as an invited, peer-reviewed article (see [A4]).

6.1. Fundamental Ideal

In this section, the fundamental idea to minimize the TSV power consumption through a power-optimal, fixed, net-to-TSV assignment is presented. The TSV-assignment technique is embedded at the end of the detailed routing step of the

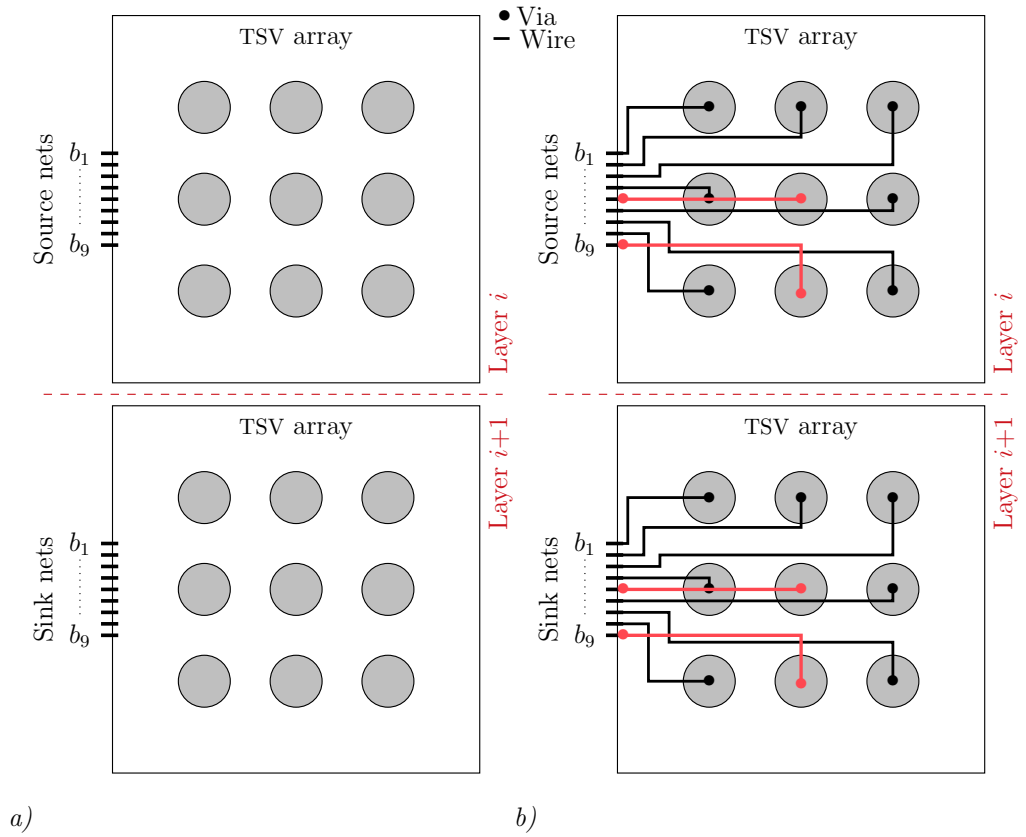


Fig. 6.1: Simplified TSV-array layout: *a)* before local/detailed TSV routing; *b)* after local TSV routing.

physical design and does not result in a TSV overhead. In detail, the technique only affects the layout-generation step at which a set of source nets, located at an edge of a TSV array in the i^{th} layer, have to be electrically connected with the sink nets in the adjacent $(i+1)^{\text{th}}/(i-1)^{\text{th}}$ layer through a predetermined TSV array. The layout for an exemplary 3×3 array before and after this step is illustrated in Figure 6.1a and Figure 6.1b, respectively. Each source net must be connected to a TSV through a metal wire. Additionally, the associated sink net in the adjacent layer must be connected to the other end of the same TSV. The standard approach will perform this task with the aim of minimizing the wiring length (routing-minimal assignment).

Instead, the proposed technique performs this task intending to minimize the TSV power consumption. For this purpose, two possible variation possibilities in the net-to-TSV assignment are taken into account simultaneously. First, to interleave the net-to-TSV assignment; and, second, to assign negated nets by swapping inverting drivers with non-inverting drivers in the source and the sink layer. Interleaving has promising potential to reduce the TSV power

consumption due to the heterogeneity in the TSV capacitances, arising from the edge effects, outlined in Subsection 4.2.2. Inversions swap the logical 1-bit and 0-bit probabilities for the according nets, which can reduce the power consumption through the TSV MOS effect outlined in Subsection 4.2.1.

Furthermore, if the logical bit values b_i and b_j on two nets with temporally aligned signal edges have a negative switching correlation (*i.e.*, $\gamma_{i,j} = \mathbb{E}\{\Delta b_i \Delta b_j\} < 0$), an inversion can improve the power consumption. As shown in Section 3.1, a negative switching correlation increases the power consumption due to the coupling capacitance between the two TSVs over which b_i and b_j are transmitted, compared to the case of an uncorrelated switching on the two lines (coupling effect). However, after negating one of the two lines, the sign of the switching correlation is changed since $\Delta(\neg b_i)\Delta b_j = -\Delta b_i \Delta b_j$, resulting in a positive switching correlation (*i.e.*, $\gamma_{i,j} > 0$) and consequently to a reduced power consumption. An example of bit-pairs with a negative switching correlation are bits that belong to the same one-hot encoded signal, as shown in Subsection 5.1.3.

6.2. Power-Optimal TSV assignment

In this section, a formal method to determine the power-optimal net-to-TSV assignment is derived. For this purpose, the formulas that were derived in Chapter 3 and Chapter 4 of this thesis are used.

If the logical value on the i^{th} net in the k^{th} cycle is symbolized as $b_i[k]$, the matrix notation of the power consumption (Equation (3.46) on Page 51) expresses the power consumption for the initial assignment[1]:

$$P_{\text{init}} = \frac{V_{\text{dd}}^2 f}{2} \sum_{i=1}^{n} \bar{C}_{\text{eff},i} = \frac{V_{\text{dd}}^2 f}{2} \langle \mathbf{S}_{\mathbb{E}}, \mathbf{C} \rangle. \tag{6.1}$$

In this equation, the first term, $V_{dd}^2 f/2$, depends on the power-supply voltage and the clock frequency. This term can only be affected at the circuit level, not at higher abstraction levels. The metric that can be optimized at higher abstraction levels, and thus through changing the net-to-TSV assignment, is the sum of the mean effective capacitances:

$$\sum_{i=1}^{n} \bar{C}_{\text{eff},i} = \langle \mathbf{S}_{\mathbb{E}}, \mathbf{C} \rangle. \tag{6.2}$$

To systematically determine the power-optimal assignment, the effect of a certain assignment on the switching matrix, $\mathbf{S}_{\mathbb{E}}$, as well as on the capacitance matrix, $\mathbf{C}$, must be expressed mathematically.

[1] An initial assignment implies here that the i^{th} net is assigned to the i^{th} TSV.

First, a method to formally express the assignment is required. A mere reordering of the net-to-TSV assignment could be mathematically expressed using the concept of the permutation matrix [115]. A permutation matrix is a matrix with exactly one 1-entry in each row and column while all other entries are 0. If an assignment of the i^{th} net to the j^{th} TSV is expressed by a 1 on matrix entry (j, i), each permutation matrix represents a valid assignment (*i.e.*, each net is assigned to exactly one TSV and *vice versa*). Furthermore, in that case, the set of all permutation matrices represents the set of all possible assignments that do not include inversions of nets.

However, inversions are an additional assignment variant in the proposed technique. Thus, a signed form of the permutation matrix, represented by $\mathbf{I}_\sigma$, is used here. Such a signed permutation matrix has one 1 or -1 in each column or row while all other entries are 0. If the i^{th} net is assigned negated to the j^{th} TSV, the (j, i) entry of the signed permutation matrix is set to -1. Still, the (j, i) entry is set to 1 if the net is assigned as non-negated. Thus, for an exemplary 2×2 TSV array, if net 1 is assigned to TSV 2, net 2 to TSV 3, net 3 negated to TSV 1, and net 4 to TSV 4, the assignment matrix is

$$\mathbf{I}_\sigma = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.3}$$

In the following, formulas to express the switching matrix and the capacitance matrix as a function of the assignment, $\mathbf{I}_\sigma$, are derived. A Boolean inversion of a net does not affect the switching activity of the logical value transmitted over a TSV, as 0-to-1 transitions are transformed in 1-to-0 transitions and *vice versa*. However, the inversion negates/sign-changes the switching correlations with all other TSV signals.

Thus, the switching matrix, defined by Equation (3.45) on Page 51, is here divided into two sub-matrices to separate the switching activities (*i.e.*, the α_i values), and the switching correlations (*i.e.*, the $\gamma_{i,j}$ values) of the nets that have to be assigned to the TSVs:

$$\mathbf{S}_\mathbb{E} = \mathbf{S}_\alpha \mathbf{1}_{n \times n} - \mathbf{S}_\gamma, \tag{6.4}$$

where $\mathbf{S}_\alpha$ is a matrix with the α_i values on the diagonal entries and zeros on the non-diagonal entries:

$$S_{\alpha,i,j} = \begin{cases} \alpha_i & \text{for } i = j \\ 0 & \text{else.} \end{cases} \tag{6.5}$$

$\mathbf{S}_\gamma$ contains the $\gamma_{i,j}$ values on the non-diagonal entries and zeros on the diagonal entries:

$$S_{\gamma,i,j} = \begin{cases} 0 & \text{for } i = j \\ \gamma_{i,j} & \text{else.} \end{cases} \tag{6.6}$$

$\mathbf{1}_{n\times n}$ is an $n \times n$ matrix of ones. Afterward, the effect of a reassignment on $\mathbf{S}_\mathbb{E}$ is expressed as

$$\begin{aligned} \mathbf{S}_{\mathbb{E},\text{assign}} &= \mathbf{S}_{\alpha,\text{assign}}\mathbf{1}_{n\times n} - \mathbf{S}_{\gamma,\text{assign}} \\ &= \mathbf{I}_\sigma \mathbf{S}_\alpha \mathbf{I}_\sigma^\mathrm{T} \mathbf{1}_{n\times n} - \mathbf{I}_\sigma \mathbf{S}_\gamma \mathbf{I}_\sigma^\mathrm{T}. \end{aligned} \tag{6.7}$$

As desired, the switching activities are merely reordered since the minuses in $\mathbf{I}_\sigma$ cancel each other out on the diagonal entries for a left-hand-side multiplication with $\mathbf{I}_\sigma$ and a right-hand-side multiplication with $\mathbf{I}_\sigma^\mathrm{T}$. In contrast, the signs of the switching correlations are changed if some nets are assigned negated. If all signal edges on the different nets are completely temporal misaligned, all switching correlations, and thus all entries of $\mathbf{S}_\gamma$, are 0. Consequently, Equation (6.7) simplifies in this scenario as

$$\begin{aligned} \mathbf{S}_{\mathbb{E},\text{assign}} &= \mathbf{S}_{\alpha,\text{assign}}\mathbf{1}_{n\times n} \\ &= \mathbf{I}_\sigma \mathbf{S}_\alpha \mathbf{I}_\sigma^\mathrm{T} \mathbf{1}_{n\times n}. \end{aligned} \tag{6.8}$$

Second, a formula to express the TSV capacitances as a function of the net-to-TSV assignment is derived. As shown in Chapter 4, the following equation can be used to estimate the size of a capacitance for an assignment of the nets i and j to the TSVs i and j:

$$C_{i,j} = C_{\mathrm{G},i,j} + \frac{\Delta C_{i,j}}{2}(p_i + p_j). \tag{6.9}$$

Here, the requirement is a formula where an inversion of the bits/nets leads to simple negations in the formula to use the signed permutation matrix. Thus, a shifted form of Equation (6.9) is used:

$$C_{i,j} = C_{\mathrm{R},i,j} + \frac{\Delta C_{i,j}}{2}(\epsilon_i + \epsilon_j), \tag{6.10}$$

where $C_{\mathrm{R},i,j}$ is the capacitance value for all 1-bit probabilities, p_i equal to $1/2$ (*i.e.*, $C_{\mathrm{R},i,j} = C_{0,i,j} + {}^{\Delta C_{i,j}}/_2$). ϵ_i is mathematically expressed as:

$$\epsilon_i = p_i - \frac{1}{2} = \mathbb{E}\{b_i\} - \frac{1}{2}. \tag{6.11}$$

Since $\mathbb{E}\{\neg b_i\} = 1 - \mathbb{E}\{b_i\}$, an inversion of b_i negates the related ϵ_i value. Thus,

the capacitance matrix as a function of $\mathbf{I}_\sigma$ is formulated as follows:

$$\mathbf{C}_{\text{assign}} = \mathbf{C}_\text{R} + \frac{\mathbf{\Delta C}}{2} \circ (\mathbf{I}_\sigma \vec{\epsilon} \vec{1}_n^{\,\text{T}} + \vec{1}_n \vec{\epsilon}^{\,\text{T}} \mathbf{I}_\sigma^{\,\text{T}}), \tag{6.12}$$

where $\mathbf{C}_\text{R}$ and $\mathbf{\Delta C}$ are matrices containing the $C_{\text{R},i,j}$ and $\Delta C_{i,j}$ values, respectively. $\vec{\epsilon}$ is the vector containing the ϵ_i values. $\vec{1}_n$ is a vector of n ones.

Summarized, the assignment dependent power consumption is

$$\begin{aligned} P_{\text{assign}} &= \frac{V_\text{dd}^2 f}{2} \langle \mathbf{S}_{\mathbb{E},\text{assign}}, \mathbf{C}_{\text{assign}} \rangle \\ &= \frac{V_\text{dd}^2 f}{2} \langle \mathbf{I}_\sigma \mathbf{S}_\alpha \mathbf{I}_\sigma^\text{T} \mathbf{1}_{n\times n} - \mathbf{I}_\sigma \mathbf{S}_\gamma \mathbf{I}_\sigma^\text{T}, \mathbf{C}_\text{R} + \frac{\mathbf{\Delta C}}{2} \circ (\mathbf{I}_\sigma \vec{\epsilon} \vec{1}_n^{\,\text{T}} + \vec{1}_n \vec{\epsilon}^{\,\text{T}} \mathbf{I}_\sigma^{\,\text{T}}) \rangle. \end{aligned} \tag{6.13}$$

Thus, the power-optimal bit assignment, $\mathbf{I}_{\sigma,\text{power-opt}}$, full-fills

$$\mathbf{I}_{\sigma,\text{power-opt}} = \underset{\mathbf{I}_\sigma \in \mathbb{S}_{\mathbf{I}_\sigma,n}}{\arg\min} \left(\langle \mathbf{S}_{\mathbb{E},\text{assign}}, \mathbf{C}_{\text{assign}} \rangle \right), \tag{6.14}$$

where $\mathbb{S}_{\mathbf{I}_\sigma,n}$ is the set of valid signed permutation matrices of shape $n \times n$.

In practice, $\mathbf{I}_{\sigma,\text{power-opt}}$ can be determined with any of the several optimization tools available to reduce the computational complexity compared to iterating over the full search space. Although up to several hundreds of TSVs exist in modern 3D ICs, the run-time to perform the proposed optimization in the TSV assignment is relatively low as it is applied for each TSV bundle individually whose sizes are relatively small. In this thesis, simulated annealing [116] is exemplarily used to determine the "power-optimal" assignment for a TSV array.

6.3. Systematic Net-to-TSV Assignments

In some scenarios, a representative number of samples of the transmitted data, required to precisely determine the exact bit-level statistics of the nets, may not be available at design time. Alternatively, the exact capacitance values may not be known. In such cases, the power-optimal assignment cannot be determined by means of Equation (6.14). However, the essential characteristics of the data and the capacitances can be used to obtain systematic assignments. Furthermore, this approach results in regular assignments that are generally valid for a broad set of data streams and TSV technologies.

Exemplary, this section proposes systematic assignments that are generally applicable for normally distributed and correlated data streams, as they build two of the most common data types in SoCs.[2] In the following, the term "assigning bit i to TSV j" is used as an expression for "assigning the net over

[2]The bit-level statistics for correlated and normally distributed data streams have been outlined in Section 5.1

which the i^{th} bit of the data words is transmitted to the j^{th} TSV" to increase the readability.

As shown in Subsection 5.1, for mean-free normally distributed as well as correlated data words, 1 and 0 bits are equally distributed on all lines (*i.e.*, $\epsilon_i = 0$ for all i). Hence, the TSV capacitance matrix is simply $\mathbf{C}_R$ for the transmission of such data streams why assigning nets negated cannot optimize the capacitances through the MOS effect. In fact, inversions can only result in an increased power consumption, as they potentially destroy the positive switching correlations of the MSBs of normally distributed data streams, outlined in Subsection 5.1.2. Consequently, the systematic assignments presented in the following do not include inversions.

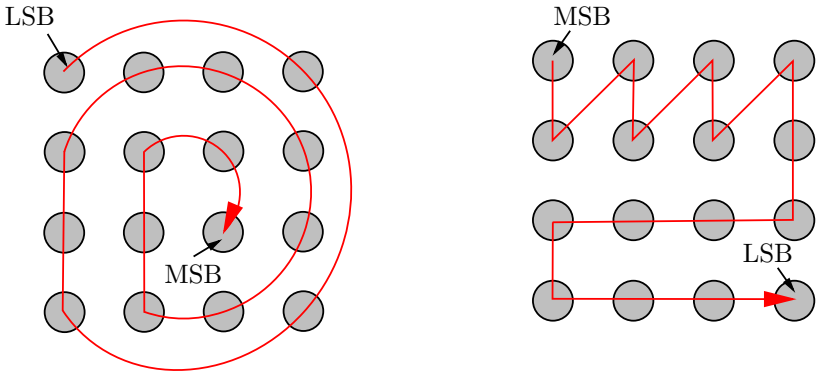
First, positively correlated and uniformly distributed data is considered (*e.g.*, sequential data streams). The bit-level statistics of such data streams have been outlined in Subsection 5.1.4 of this thesis. All logical bits values switch completely uncorrelated (*i.e.*, all $\gamma_{i,j}$ values are equal to 0). Thus, $S_{\mathbb{E},i,j}$ is simply equal to switching activity α_i. Hence, the power consumption for the initial assignment simplifies in that case as

$$P_{\text{corr,init}} = \frac{V_{\text{dd}}^2 f}{2} \langle \mathbf{S}_{\mathbb{E}}, \mathbf{C}_{\text{R}} \rangle = \frac{V_{\text{dd}}^2 f}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} C_{\text{R},i,j} \alpha_i. \tag{6.15}$$

Therefore, bits with the highest switching activity, α_i, should optimally be reassigned to TSVs with the lowest overall capacitance, $\sum_{j=1}^{n} C_{\text{R},i,j}$, and *vice versa*, as it minimizes the power consumption. The TSVs in the array corners have the lowest overall capacitance, and edge TSVs have a lower overall capacitance than TSVs in the middle of an array, as shown in Chapter 4. Thus, optimally, the bits with the highest self-switching are assigned to the TSVs at the array corners. The bits with the next highest switching activities are assigned to the array edges. Remaining bits are assigned to the array middle. The switching activity decreases with an increase in the significance of the bit for correlated data streams, as shown in Chapter 5. Hence, the proposed systematic assignment forms a spiral starting at the LSB and ending at the MSB, illustrated in Figure 6.2a.

The proposed spiral mapping is validated for various sequential data streams with varying branch probability, each containing 1×10^5 samples.[3] With the branch probability, the correlation of subsequent data words varies. The resulting power-consumption reductions due to the spiral mapping and due to an optimal assignment are shown in Figure 6.3 for two different TSV arrays: First, a 4×4 array with a TSV radius of $2\,\mu$m and a minimum pitch of $8\,\mu$m

[3] In the evaluation section of this chapter, the efficiency of the mapping for real correlated image data streams is analyzed.



a) Spiral *b)* Sawtooth

Fig. 6.2: Systematic net-to-TSV assignments: *a)* Spiral mapping for correlated data words; *b)* Sawtooth mapping for normally distributed data words.

is considered. Moreover, a 5×5 array with a TSV radius of $1\,\mu$m and a minimum pitch of $4\,\mu$m is investigated. All power-consumption values reported in this section are calculated with the proposed high-level formula, employing exact capacitance quantities obtained by parasitic extractions with the *Q3D Extractor* for the TSV-array model from Subsection 2.4.1. Thereby, a significant signal frequency of 6 GHz is considered for parasitic extraction. The results in Figure 6.3 show that the power savings for both assignment techniques, optimal and systematic, are almost equal. Thus, the analysis proves the optimal nature of the systematic spiral mapping for uniformly distributed and correlated data words.

As a second scenario, a systematic assignment for normally distributed, but uncorrelated, data is derived. Such data implies that the switching probability of each bit is $1/2$. Thus, the power consumption for an initial assignment can be expressed as:

$$\begin{aligned} P_{\text{normal-dist,init}} &= \frac{V_{\text{dd}}^2 f}{2} \langle \mathbf{S}_{\mathbb{E}}, \mathbf{C}_{\text{R}} \rangle \\ &= \frac{V_{\text{dd}}^2 f}{2} \sum_{i=1}^{n} \left(\frac{1}{2} C_{\text{R},i,i} - \sum_{\substack{j=1 \\ i \neq j}}^{n} \left(\frac{1}{2} - C_{\text{R},i,j} \right) \gamma_{i,j} \right) . \end{aligned} \tag{6.16}$$

Thus, to minimize the power consumption, bit pairs with a strong switching correlation, $\gamma_{i,j}$, have to be assigned to TSV pairs connected by a large coupling capacitance, $C_{\text{R},i,j}$, in order to reduce the contribution of the capacitance to the power consumption as much as possible. The biggest coupling capacitances in an array are located between the four corner TSVs and their two directly adjacent edge TSVs due to the edge effects, outlined in Chapter 4.

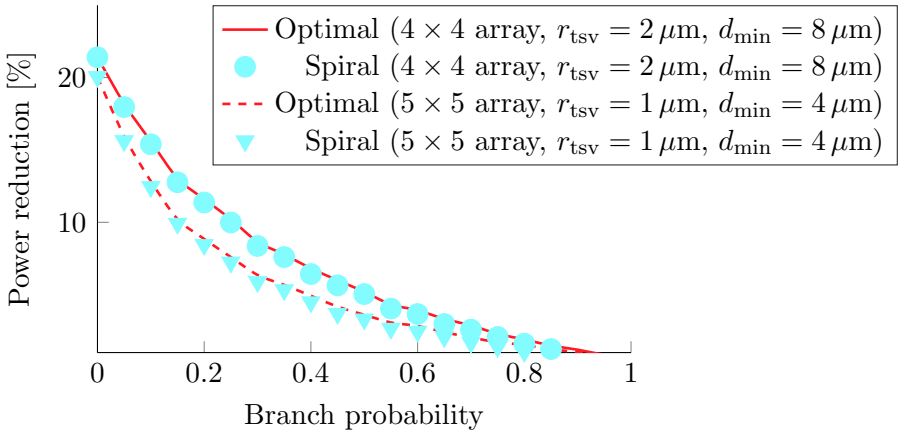


Fig. 6.3: Reduction in the TSV power consumption due to the proposed power-optimal net-to-TSV assignment and the systematic spiral mapping for sequential data streams.

In Subsection 5.1.2, it was shown that the MSBs of normally distributed signals have the highest switching correlation. Thus, the second systematic mapping assigns the MSB onto a corner TSV and the next lower significant bit onto one of its directly adjacent edge TSVs. The following bits, recursively, are assigned by finding the TSV in the array that has the biggest accumulated coupling capacitance with all previously assigned TSVs. Figure 6.2b illustrates the resulting systematic assignment. Over the first two rows, the bits, from the MSB downwards, are mapped in a sawtooth manner. From the third row on, a simple row-by-row mapping is used. This second systematic assignment is referred to as sawtooth mapping in the following. Figure 6.4a shows the reduction in the power consumption due to the proposed optimal assignment technique and the systematic sawtooth mapping for the transmission of Gaussian distributed 16-bit data streams with varying standard deviation, σ, over a 4×4 TSV array ($r_{tsv} = 2\,\mu$m and $d_{min} = 8\,\mu$m). The results underline the optimal nature of the sawtooth mapping for normally distributed, but temporally uncorrelated, data streams.

In some applications, normally distributed signals occur, which are also temporally correlated. In these cases, the optimal net-to-TSV assignment is not as trivial and dependent on the correlation quantities. As shown in Figure 6.4, the sawtooth mapping leads to the lowest power consumption (reduction up to 40 %) for negatively correlated (*i.e.*, $\rho < 0$) and normally distributed data words. However, neither the sawtooth nor the spiral mapping results in the lowest possible power consumption for positively correlated and normally distributed data words. However, compared to a random assignment, both approaches still lead to a significant improvement in the power consumption.

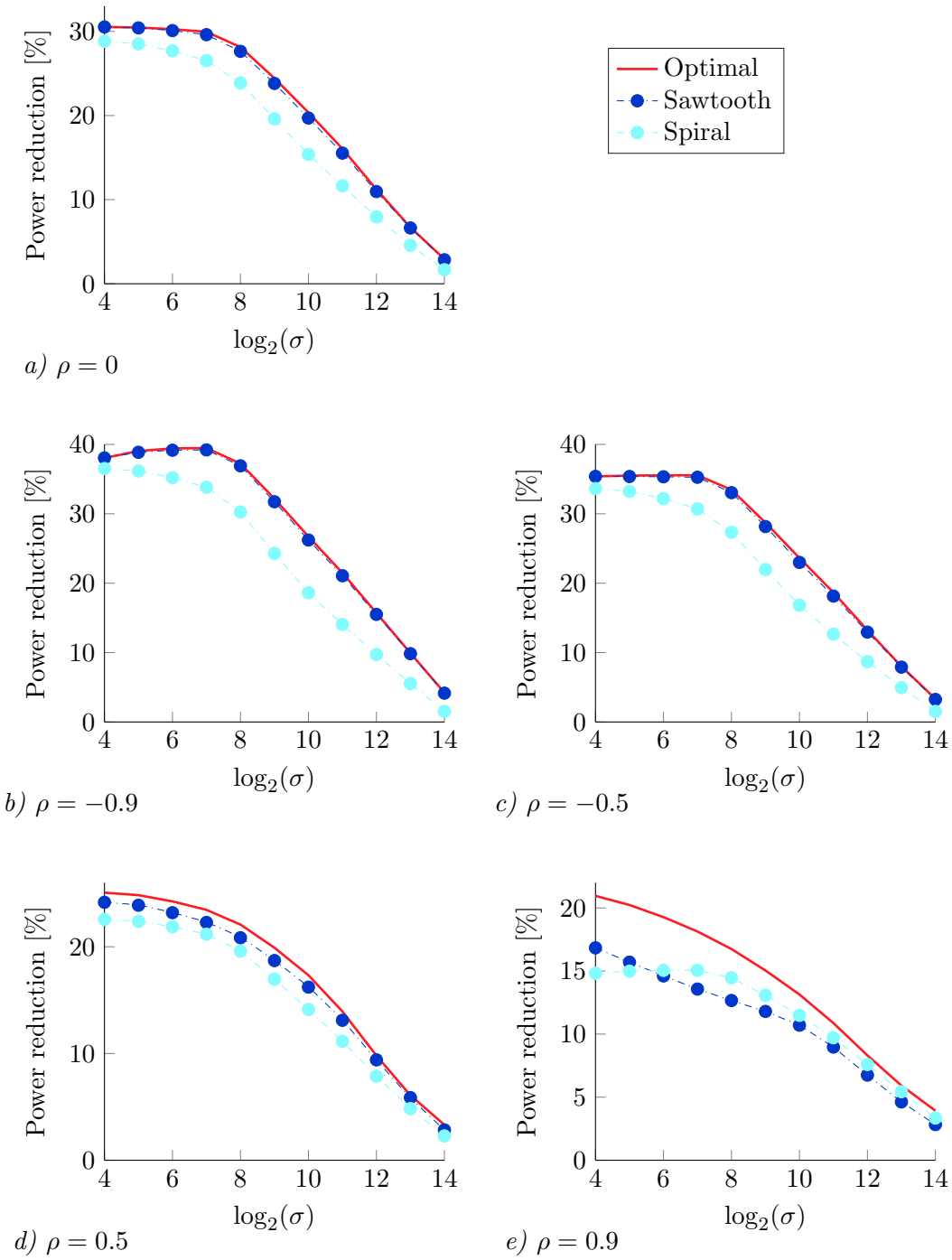


Fig. 6.4: Reduction in the TSV power-consumption due to the proposed assignment techniques for uncorrelated ($\rho = 0$), negatively correlated ($\rho < 0$), and correlated ($\rho > 0$), normally distributed data words.

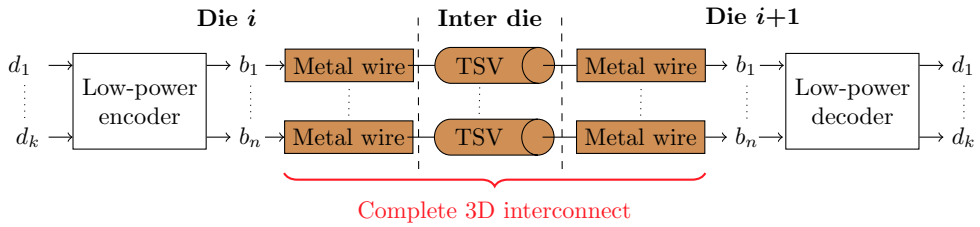


Fig. 6.5: Proposed low-power coding approach for 3D interconnects.

Summarized, if it is not possible to determine the optimal assignment by means of Equation (6.14), the proposed sawtooth mapping should be applied for normally distributed signals and the spiral mapping for primarily temporally correlated signals. Following the same approach, systematic assignments can be derived for other scenarios as well.

6.4. Combination with Traditional Low-Power Codes

Long 3D interconnects are commonly made up of multiple metal-wire and TSV segments. Since both structures typically contribute significantly to the overall power consumption of a 3D system, a simultaneous reduction in the power consumption of TSVs and metal wires is desirable. Using one of the existing low-power coding techniques to reduce the power consumption of the metal wires and a different, completely new designed coding technique for the vertical TSVs is impractical due to the overhead cost of the encoder-decoder pairs. In that scenario, $2 \cdot n_{\text{die}} - 1$ encoder-decoder pairs would be required for a 3D interconnect spanning over n_{l} dies. $n_{\text{die}} - 1$ for the TSV segments, and n_{die} for the metal-wire segments. For example, three encoder-decoder pairs would be already required for a 3D interconnect that only spans over two dies. Two encoder-decoder pairs for the metal wires in the source and the destination layer, and one encoder-decoder pair for the TSVs in between.

Thus, there is a high demand for low-power coding techniques, which simultaneously optimize the power consumption of metal wires and TSVs. Such techniques—referred to in this thesis as 3D low-power codes (LPCs)—only require a single low-power encoder-decoder pair enclosing the full 3D interconnect as illustrated in Figure 6.5. Nevertheless, 3D LPCs still effectively optimize the power consumption of all TSV and metal-wire segments. The technique proposed in the present chapter enables us to use traditional LPCs as 3D LPCs in the best possible way, without impairing the encoding overhead by finding the optimal codeword-bit-to-TSV assignment.

Unencoded, most data streams generally have a balanced number of 0 and 1 bits. However, data-encoding techniques often lead to a larger fraction of 0

bits, which affects the power consumption of TSVs traversing a typical p-doped substrate negatively due to the MOS effect. Here, the proposed assignment technique further improves the efficiency of the data encoding by transmitting inverted bits. As mentioned before, an inversion can be realized by using inverting drivers instead of non-inverting ones (or *vice versa*) on both sides of the according TSV.

However, inversions can also be hidden in the encoder and the decoder architecture. For example, Gray encoding is a popular approach to reduce the power consumption of gates and metal wires. Gray encoding does not only reduce the power consumption for sequential but also for normally distributed data. This is due to the strong spatial correlation of the bits in the MSB-region of normally distributed data words, outlined in Subsection 5.1.2. The i^{th} output of a Gray encoder is equal to the i^{th} input XORed with the $(i{+}1)^{\text{th}}$ input (*i.e.*, $b_i = d_i \oplus d_{i+1}$), while the MSB of the encoder output is equal to the MSB of unencoded data word (*i.e.*, $b_n = d_n$). This XORing in the Gray encoder for two neighbored spatial-correlated MSBs results in output bits that are almost stable on logical 0, independent of the pattern correlation. Hence, Gray encoding reduces the switching activities of the MSBs for normally distributed signals to nearly zero, which is beneficial for the power consumption of TSVs and metal wires. However, the encoding also decreases the 1-bit probabilities, resulting in a higher TSV power consumption for typical p-doped substrates. Thus, inverting the codeword bits has good potential to further increase the power savings of the coding technique for TSV-based interconnects. For many data-encoding techniques, inversions of the codeword bits can be realized inside the encoder-decoder architectures, instead of swapping inverting and non-inverting drivers. For example, for a Gray coding, the XOR operations are swapped with XNOR operations in the encoder and the decoder to obtain negated codeword bits. Since XOR and XNOR operations typically have the same costs, this optimization of the data encoding technique is again overhead-free.

6.5. Evaluation

In this section, the proposed low-power technique is evaluated in depth. The only implementation cost of the proposed technique is a non-routing-minimal local net-to-TSV assignment. Thus, in Subsection 6.5.1, the impact of the local TSV assignment on the 3D-interconnect parasitics is quantified. In Subsection 6.5.2, the power savings of the proposed systematic assignments are investigated for real data streams and compared to the power savings for the respective power-optimal assignments. Afterward, the TSV power-consumption reductions of the proposed technique, used in combination with traditional LPCs, are analyzed.

6.5.1. Worst-Case Impact on the 3D-Interconnect Parasitics

In this subsection, the worst-case impact of a non-routing-minimal local net-to-TSV wiring on the parasitics is quantified. To keep the TSV delays as small as possible, the TSV drivers have to be placed as close as possible to the arrays [17]. Thus, the sources and the sinks of the nets in Figure 6.1 on Page 117 are driver outputs and inputs, respectively. The minimum distance between a driver and a TSV is determined by the TSV keep-out zone (KOZ) constraint, which, in this analysis, is equal to two times the minimum TSV pitch, representing a realistic value. Since longer metal wires reduce the relative impact of the local routing within the TSV array on the overall parasitics, the full 3D-interconnect paths in this analysis only range between these nearby TSV drivers. Thus, in the following, worst-case values for the impact of the assignment are reported.

The effect of one million random assignments of the driver inputs and outputs to the TSVs is analyzed. Thereby, the driver inputs and outputs are assumed to be aligned centrally at two edges of the TSV-array KOZ, as illustrated in Figure 6.1. Typical global TSV dimensions (*i.e.*, $r_{\mathrm{tsv}} = 2\,\mu\mathrm{m}$, $d_{\mathrm{min}} = 8\,\mu\mathrm{m}$, $l_{\mathrm{tsv}} = 50\,\mu\mathrm{m}$, and $f_{\mathrm{s}} = 6\,\mathrm{GHz}$) are considered and the array size is varied to quantify how the results scale with the number of TSVs/nets. Two scenarios for the local routing of the nets to the TSVs are considered. Scenario one uses the lower metal layers for routing. In this scenario, the nets are initially spaced with the minimum M1 spacing reported in [65] for a 22-nm technology. The second scenario considers intermediate metal layers for routing, why the input nets are spaced with the minimum M4 spacing reported in [65]. The higher metal layers—reserved for long global metal wires—are not required for the relatively short connections between the nearby drivers and the TSVs, especially since the lower metal layers have sufficient capacity in that area due to the large TSV dimensions and the absence of active elements caused by the KOZ constraints. Thus, a routing in the global metal layers is not analyzed.

The Manhattan distances between the driver inputs and outputs and the assigned TSVs are calculated to estimate the lengths of the metal wires for all assignments. Afterward, the two assignments which result in the lowest and the highest maximum length of a single wire, are taken as the routing-minimal (*i.e.*, best-case parasitics) and the routing-maximal (*i.e.*, worst-case parasitics) assignment, respectively. For both assignments, the maximum wire length is multiplied with the wire resistance per unit-length reported in Table 2.3 on Page 30 to obtain best-case and worst-case values for the maximum wire resistances. Added to both values are the TSV resistance and the driver's channel-resistance $R_{\mathrm{eff,d}}$ of $3.02\,\mathrm{k}\Omega$ (see Section 2.4). Furthermore, a 100-Ω resistances are added to model the connections between the metal wires and the TSVs [74]. Thereby, values for the maximum resistance of an interconnect path, $\hat{R}$, for a routing-minimal and a routing-maximal assignment are obtained.

Table 6.1.: Worst-case parasitic increases due to a non-routing-minimal local TSV assignment for a routing in the local or the intermediate metal layers.

Metal layer	**3 × 3 array**		**5 × 5 array**		**7 × 7 array**		**10 × 10 array**	
	$\hat{R}$ [%]	$\hat{C}_{\text{eff}}$ [%]	$\hat{R}$ [%]	$\hat{C}_{\text{eff}}$ [%]	$\hat{R}$ [%]	$\hat{C}_{\text{eff}}$ [%]	$\hat{R}$ [%]	$\hat{C}_{\text{eff}}$ [%]
M1	0.24	0.12	0.68	0.35	1.22	0.65	2.37	1.34
M4	0.17	0.24	0.53	0.75	0.99	1.38	1.92	2.63

The metal-wire capacitances per unit-length in Table 2.3 are only valid for minimum-spaced wires, which result in the highest coupling capacitances. However, the analyzed minimum TSV pitch is more than 60 and 100 times larger than the minimum pitch of the local and intermediate metal wires, respectively. Thus, the wire pitches are typically way higher for the local net-to-TSV routing within an array. Here, a mean metal-wire pitch equal to half of the minimum TSV pitch is considered. This is still a pessimistic estimation, as the low utilization of the metal layers allows for an even higher spacing for critical/worst-case paths. The normalized wire capacitances for the increased spacing are extracted by means of the PTM interconnect tool [69]. Compared to the values reported in Table 2.3, the capacitances of local and intermediate metal wires decrease to $0.09\,^{\text{fF}}/_{\mu\text{m}}$ and $0.10\,^{\text{fF}}/_{\mu\text{m}}$, respectively, due to the increased spacing. Moreover, the coupling capacitances become negligible compared to the ground capacitances.

These normalized wire capacitances, multiplied with the maximum wire lengths for the routing-minimal and the routing-maximal assignments, are used to obtain worst-case and best-case values for the maximum wire capacitances. Added are the maximum effective TSV capacitance—extracted with *Q3D Extractor*—and a load capacitance of 0.22 fF to model a small inverter as the load [65].

In Table 6.1, the resulting worst-case percentage increases in the maximum-path resistance and effective capacitance due to a non-routing-minimal TSV assignment are reported. The results show that—in contrast to the global net-to-array routing—the effect of the local routing within the individual arrays is, as expected, negligible for reasonable array sizes. Considering local metal wires, the maximum possible increase in the parasitics is below 0.24 % for a 3×3 array. The maximum increase for a 7×7 array is still as low as 1.22 %. Thus, with an increase in the TSV count by a factor of 5.44 ×, the maximum parasitic increase goes up by a factor of 5.08 ×. Generally, the values increase sub-linearly with the TSV count, as the maximal relative increase is 2.37 % for 100 TSVs arranged as a 10×10 array.

Using higher metal layers decreases the relative impact of the assignment on the resistances at the cost of a higher increase in the capacitances. For the 7×7 array, the maximum possible increase in the interconnect capacitance is 1.38 % for M4 wires and only 0.75 % for M1 wires. Also for intermediate metal layers, the relative increases in the parasitics scale almost linearly with the TSV count.

In summary, one can in fact neglect the added parasitics due to a non-minimal local TSV assignment for arrays with less than 100 TSVs. Hence, in such cases, the power reduction of the proposed technique comes at negligible costs and performance degradation. For larger arrays, the added parasitics should be considered, which requires a small extension of the proposed technique. For example, one could only include assignments in $\mathbb{S}_{\mathbf{I}_\sigma,n}$ in Equation (6.14) that do not increase the maximum Manhattan distance between a sink/source net and the assigned TSV beyond a threshold value. However, arrays that are large enough that this extension is required are rather uncommon, and thus not further considered in this work.

6.5.2. Systematic Versus Optimal Assignment for Real Data

In Section 6.3, the efficiency of the proposed systematic assignments has been only investigated for synthetically generated data streams to show the optimal nature of the systematic assignments. In the following, the efficiencies of the systematic and optimal TSV assignments are compared for real data streams. Thereby, the focus lies on an important class of systems: Heterogeneous 3D SoCs. Two commercially relevant examples are VSoCs [113], including dies for image sensing and dies for digital image processing; and SoCs with one or more mixed-signal dies for sensing and sampling of signals, bonded to one or more digital dies for computation [117].

Image-Sensor Data

In a 3D VSoC, some dies (typically integrated into a rather conservative technology node) are dedicated to image sensing, while others are used for image processing (*e.g.*, filtering, compression). In this subsection, the proposed assignment technique is analyzed for the transmission of digitized image pixels from a mixed-signal die to a processor die in such a VSoC.

The first three analyses are performed for data stemming from a 0–255 RGB image sensor using a standard Bayer filter [118]. First, the parallel transmission of all four RGB colors (1 red, 2 green, and 1 blue) of each Bayer-pattern pixel over one 4×8 TSV array is analyzed. For the second analysis, four additional TSVs are considered in the array, which then has a 6×6 shape. Two of the additional TSVs are used to transmit enable-signals with a set probability of 1×10^{-5}. The other two additional TSVs are a power/ground (P/G) signal pair to supply the sensor.

In the third analysis, the four colors of each pixel are transmitted one after another (*i.e.*, one-by-one time-multiplexed) together with an enable signal (set probability 1×10^{-5}) over a 3×3 array. The multiplexing allows decreasing the number of required TSVs for the image transmission to eight. The fourth analysis is performed for a data stream stemming from a 0–255 grayscale image sensor. Here, the transmission of one pixel per cycle over a 3×3 array, including the same enable-signal as in the third analysis, is investigated. All analyzed data streams are composed of an extensive set of pictures of cars, people, and landscapes, to obtain representative results.

For all analyzed scenarios, the reduction in the TSV power consumption, compared to worst-case random assignments, is investigated for the optimal assignment and the systematic spiral mapping since the strong correlation of adjacent pixels results in transmitted data words that are strongly correlated but tend to be uniformly distributed. The enable and P/G signals are (almost) stable. An enable signal is here assumed as set to logical 0 when unused, which can be exploited by an inversion. Power or ground lines are always on logical 1 and logical 0, respectively. However, an inversion of the logical values for P/G lines is not possible and consequently forbidden for the assignment. For the simultaneous transmission of a complete RGB pixel, the bits of the four color-components are spatially interleaved one-by-one for the proposed systematic spiral mapping. Stable lines are added as the MSBs for the spiral assignment, as they have the lowest switching activities of all nets.

Aggressively scaled global TSV dimensions are considered in this analysis (*i.e.*, $r_{\text{tsv}} = 1\,\mu\text{m}$, $d_{\text{min}} = 4\,\mu\text{m}$, and $l_{\text{tsv}} = 50\,\mu\text{m}$). The power consumption for the 3×3 and the 6×6 array is also investigated for a TSV radius, r_{tsv}, and minimum pitch, d_{min}, of $2\,\mu\text{m}$ and $8\,\mu\text{m}$, respectively, to show the effect of varying TSV geometries. Again, the power-consumption quantities are obtained through the high-level formula, employing capacitance matrices that were extracted with the *Q3D Extractor* for a significant frequency of 6 GHz.

In Figure 6.6, the resulting TSV power-consumption reductions due to the proposed technique are reported. The results show that the spiral mapping is nearly optimal for the transmission of the RGB data without additional stable lines in the array and leads to a power-consumption reduction of more than 11 %. For the multiplexed colors, the lowest power improvement of only about 6 % is achieved, as the correlation between subsequently transmitted bit-patterns is lost by the multiplexing. Thus, only a wise assignment of the almost stable enable signal can be applied here to reduce the power consumption.

With (almost) stable lines in the TSV array, the power-consumption reduction due to an optimal assignment is generally higher than for the systematic assignment (up to 2.5 percentage points (pp)). The reason for this is that only the optimal assignment considers possible inversions for the enable signals in order to increase the 1-bit probabilities. Moreover, additional (almost) stable

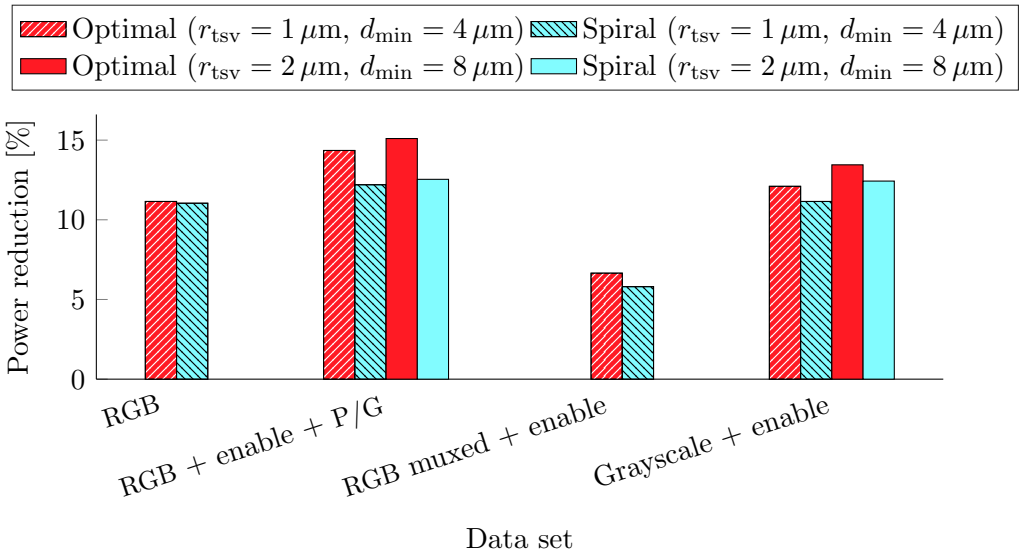


Fig. 6.6: Reduction in the TSV power consumption due to the proposed power-optimal assignment and the systematic spiral mapping for Bayer-pattern (RGB) and grayscale image-sensor data.

enable or P/G nets in the array, generally, increase the power-consumption improvement of the proposed technique. The reason is that the bit-level statistics of the nets become more heterogeneous due to the added lines, which increases the optimization potential.

In conclusion, the results show that both net-to-TSV assignment approaches, optimal and systematic, can effectively improve the TSV power consumption for the transmission of real image data. However, in the presence of additional control-signal and P/G nets, the optimal approach can result in noticeably higher power-consumption reductions than the systematic one.

Smartphone Sensor Data

In the following, the efficiency of the proposed technique for real smartphone sensor data, transmitted from a sensing die to a processing die, is investigated. For this purpose, sensor signals from a modern smartphone in various daily-use scenarios are captured. Analyzed are the magnetometer, the accelerometer, and the gyroscope sensor, all sensing on three axes x, y, and z. Considered is a transmission of one 16-bit sample per cycle over a 4×4 array with a TSV radius and minimum pitch of $2\,\mu m$ and $8\,\mu m$, respectively. The transmission of the individual data streams is analyzed for two scenarios. In the first one, the root mean square (RMS) values—calculated from the respective three axes values—are transmitted. Scenario two represents the transmission of the x-axis, y-axis, and z-axis values in a one-by-one time-multiplexed manner (XYZ multiplexed). Since the sample time can be way lower than the maximum

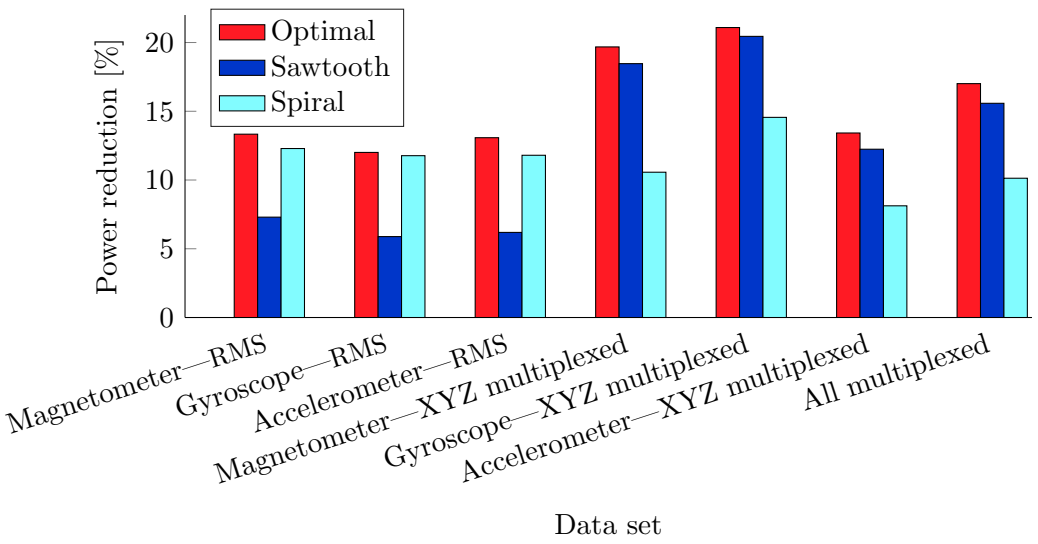


Fig. 6.7: Reduction in the TSV power consumption due to the proposed optimal and systematic net-to-TSV assignment approach for real smartphone-sensor data.

propagation delay of a TSV, the transmission of all three data streams over a single 4×4 array is analyzed as well. Thereby, a regular pattern-by-pattern multiplexing of the three individual XYZ-multiplexed data streams is considered. In this analysis, both systematic bit-to-TSV assignments are investigated since normally distributed as well as temporally correlated data streams are amongst the analyzed set.

Figure 6.7 shows the mean power-consumption reductions compared to worst-case assignments. The figure reveals that, for the multiplexed data streams, the proposed sawtooth mapping is only slightly worse than the optimal assignment, which reduces the power consumption by up to 21.1 %. Generally, the single-axis values are normally distributed and temporally correlated. However, for multiplexed data streams, the correlation of subsequently transmitted patterns is lost, as outlined in Chapter 5. Thus, these scenarios build examples for temporally uncorrelated and normally distributed data, as multiplexing does not affect the pattern distribution. The small gain for the optimal net-to-TSV assignment over the spiral assignment is because not all sensor signals are perfectly mean-free, resulting in slightly unequal bit probabilities [105].

In contrast, for the RMS data, the spiral mapping significantly outperforms the sawtooth mapping as the RMS patterns are unsigned (*i.e.*, no mean-free normal distribution) and correlated. However, for the RMS data, the maximum possible power reduction due to a reassignment of the nets is 13.3 %, which is noticeably lower than the maximum power reduction for the interleaved data streams.

In conclusion, to exploit a normal distribution in the data words shows to be more efficient than to exploit a strong correlation of consecutive data words. Furthermore, due to non-idealities in real signals compared to the abstract data model, the optimal assignment approach has a slightly higher gain than the systematic one. However, both assignment approaches, systematic and optimal, generally lead to a significant improvement in the TSV power consumption.

6.5.3. Combination with Traditional Coding Techniques

In this subsection, the combination of the proposed net-to-TSV assignment technique with traditional LPCs is investigated for real data streams by means of *Spectre* circuit simulations for aggressively scaled TSVs (*i.e.*, $r_{\text{tsv}} = 1\,\mu\text{m}$, $d_{\min} = 4\,\mu\text{m}$, and $l_{\text{tsv}} = 50\,\mu\text{m}$) and 22-nm drivers. The setup with the moderate-sized 22-nm drivers from Section 3.4 is used here for the circuit simulations in order to obtain a realistic driver power consumption. Compared to Section 3.4, the duration of the transmitted bit-patterns is reduced to 333 ps, resulting in a throughput of about $3\,{}^{\text{Gb}}/_{\text{s}}$ per data TSV. To report values independent of the TSV count in the array, and redundant bits in the transmitted patterns, the power consumption quantities in Figure 6.8 are scaled to values for a transmission of one 32-bit word per cycle.

The power consumption is investigated twice for the transmission of four different data streams. Once for the power-optimal net-to-TSV assignment, and once for a random assignment. The first data stream contains the sensor data from Subsection 6.5.2, where, for 3,900 cycles, patterns of a single axis of one sensor are transmitted. Subsequently, patterns stemming from the next sensor are transmitted for 3,900 cycles and so on, until data for all axes and sensors has been transmitted. This data stream is referred to in the following as "sequential sensor data". For the second data stream, labeled as "multiplexed sensor data", the patterns belonging to the individual axes are multiplexed one-by-one. The pattern width is 16 bit for the first two data streams, and a 4×4 array is chosen as the transmission medium. As expected, the results in Figure 6.8 show that the multiplexed sensor data leads to a dramatically higher power consumption since the correlation between consecutively transmitted patterns and its beneficial impact on the bit switching is lost. Nevertheless, because of limited buffer capabilities in the mixed-signal sensing layer, a multiplexed transmission is more likely than a sequential transmission. However, in Subsection 5.4.2, it was shown that the interconnect power consumption, for the transmission of multiplexed data streams, can be effectively reduced on an end-to-end basis by increasing or decreasing the joint 1-bit probabilities (*i.e.*, the probabilities that two bits are on logical 1 in a single cycle).

Gray encoding can be used to decrease the joint bit probabilities for normally distributed data as it results in codeword MSBs that are almost stable on logical 0, as outlined previously. Furthermore, Gray encoding has the advantage here

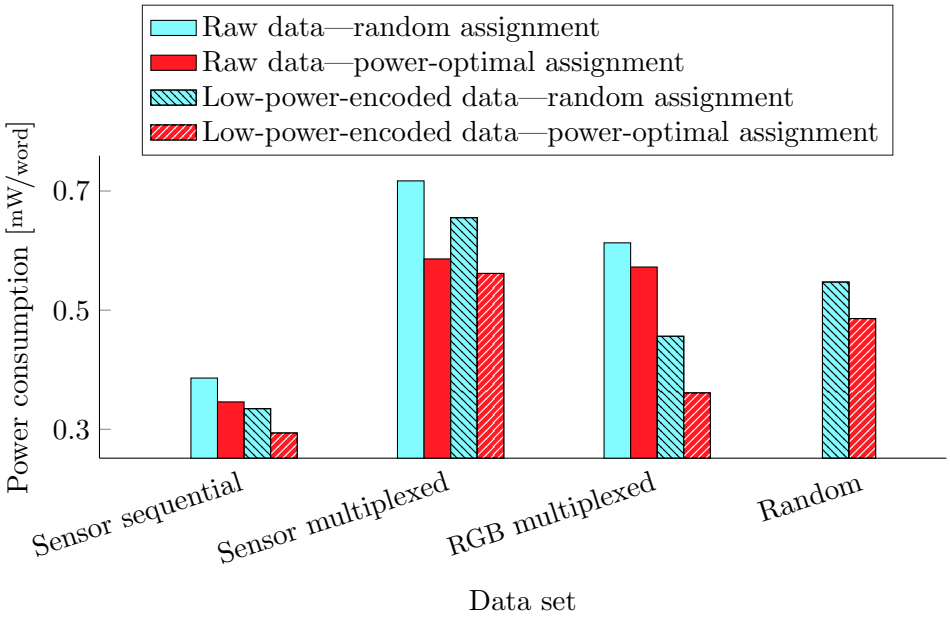


Fig. 6.8: Power consumption of TSVs (including drivers and leakage) in case of a transmission of raw and low-power-encoded data for the proposed power-optimal as well as a random net-to-TSV assignment.

that it can be realized on an end-to-end basis in the analog-to-digital converters (ADCs) of the sensors to cut the implementation cost for the coding technique. Thus, Gray coding is analyzed twice for the sensor data. Once in the traditional way, and once in combination with the proposed optimal net-to-TSV assignment technique.

For the raw multiplexed sensor data, the proposed assignment technique leads to a reduction in the power consumption by 18.3 %, compared to a random assignment. This is even more than two times higher than the power-consumption reduction of a mere Gray encoding, which is only 8.6 %. Thus, the proposed low-cost net-to-TSV assignment technique can even significantly outperform well-established data-encoding techniques. A combination of Gray coding with the proposed technique leads to the lowest power consumption and more than doubles the Gray-coding efficiency (power reduction 21.7 %).

The third analyzed data stream contains multiplexed RGB pixels for different pictures. A transmission of the 8-bit data stream together with an enable signal (set probability 1×10^{-5}) over a 3×3 array is investigated. Again, the beneficial correlation between consecutively transmitted patterns is lost if the RGB colors are multiplexed. This leads to a dramatic increase in the interconnect power consumption and to no power reduction for a Gray encoding. A correlator is used to reduce the switching activity for the transmission of the multiplexed color values on an end-to-end basis, hidden in the ADCs of

the image sensor. For a new red, green, or blue value, the pattern is first bitwise XORed with the previous value of the same color and subsequently transmitted. Since consecutive red, green, and blue values are highly correlated, this again leads to MSBs nearly stable on logical 0. Thereby, the correlator coding drastically reduces the switching activities on the TSVs, without inducing a bit overhead. Furthermore, the proposed assignment technique exploits the resulting increased number of 0 bits through inversions. Thus, the transmission of the correlator encoded RGB data over a 3×3 array, including the enable signal, is analyzed in addition to the unencoded data.

The results show that, combined with the correlator, the proposed optimal assignment technique results in a drastic improvement in the power consumption from 0.61 mW to 0.36 mW (−41.0 %). In contrast, the correlator itself only reduces the power consumption by 25.2 %. Thus, the technique proposed throughout this chapter again significantly enhances the gain of the coding technique. For the mere assignment approach without the correlator encoding, the proposed technique only results in a power-consumption improvement by 6.8 %. Thus, without a strong heterogeneity in the bit-level statistics, the proposed technique should be supplemented by a traditional low-power coding technique to be most effective.

To show the general usability of the proposed technique for all data types, the last analyzed data stream is a random 7-bit data stream, encoded to an 8-bit data stream using the coupling-invert LPC presented in [79]. The encoded data words are transmitted in this analysis together with a flag with a set probability of 1×10^{-4} over a 3×3 array. The analyzed coding technique is tailored for the specific capacitance structure of a metal-wire bus and is thus intrinsically not suitable for TSV arrays. Here, a 3D NoC is considered as an exemplary use case in which the data is mainly transmitted over 2D links made up of metal wires, while a dedicated encoding for each TSV-link is too cost-intensive. However, the coding approach leads to a positive switching correlation in between some bit pairs. These correlations and the set probability of the flag are exploited by the low-power approach proposed in this chapter. Thereby, a further reduction in the TSV power consumption by 11.2 % is achieved. This proves the efficiency of the technique for a broad set of applications.

Please note that the TSV dimensions that are analyzed in this subsection are equal to the minimum ones predicted by the ITRS. For thicker TSVs and larger TSV pitches, which is the more common case today, the proposed technique results in an even higher reduction in the TSV power consumption. This increase in the efficiency is mainly due to the more dominant edge effects for larger TSV dimensions. For example, the power-consumption improvement of the proposed technique reaches values as high as 48 % if the experiment presented in the current subsection is repeated for a TSV radius, minimum pitch, and length of 2 μm, 8 μm, and 50 μm, respectively.

6.6. Conclusion

In this chapter, a technique was presented, which effectively reduces the TSV power consumption at negligible costs. The fundamental idea of the technique is based on a physical-effect-aware net-to-TSV assignment, which exploits the bit-level properties of the transmitted patterns and the heterogeneity in the TSV-array capacitances, outlined in Chapter 4. Analyses for an extensive set of real and synthetic data streams have proven the efficiency of the proposed low-power technique, which is able to reduce the power consumption of modern TSVs by over 45 %, without inducing noticeable overhead costs. Furthermore, the proposed technique is the key enabler for efficient low-power coding for 3D interconnects as it allows the reuse of traditional low-power codes for TSVs in the most effective way. Thereby, such traditional techniques can be used to efficiently improve the power consumption of the horizontal (*i.e.*, metal-wire) and vertical (*i.e.*, TSV) interconnects in 3D SoCs simultaneously.

CHAPTER 7

Low-Power Technique for High-Performance 3D Interconnects

It was shown in the previous chapter that the TSV edge and MOS effects, outlined in Chapter 4, can be exploited to improve the TSV power consumption effectively. In this chapter, a coding technique is presented that aims for an increase in the performance of TSVs and metal wires. The method takes the edge effects into account, which enables a much higher improvement in the TSV performance than previous techniques and, for the first time, in a simultaneous improvement in the 3D-interconnect power consumption.

High-level approaches that improve the performance of traditional VLSI interconnects are well established today and commonly referred to as crosstalk-avoidance coding techniques [39].[1] For metal wires, this data-encoding approach can improve the performance by over 90 % while it additionally reduces their power consumption by over 20 % [39].

[1] The term crosstalk is used to indicate that the propagation delay of a VLSI interconnect heavily depends on the simultaneous switching on the other interconnects.

In contrast to TSV low-power codes (LPCs), several TSV crosstalk-avoidance codes (CACs) have been proposed prior to this doctoral research project [21–23]. These 3D CACs aim to improve the TSV performance by keeping the effective capacitance of each TSV in the middle of an array always below a certain level. However, none of the existing methods ever analyzed the performance of the TSVs at the edges of an array arrangement. The edge TSVs are completely neglected in previous works—claiming that the maximum propagation delay of a TSV at an array edge is either-way significantly lower than for a middle TSV due to their reduced number of directly adjacent neighbors.

However, this assumption does not hold because of the edge effects. The coupling capacitance between two adjacent edge TSVs is significantly larger than its counterpart in the middle of the array, as shown in Chapter 4. Consequently, the maximum propagation delay for the edge TSVs is only slightly lower compared to the middle TSVs. This fact drastically reduces the coding efficiency of all existing 3D CACs. For modern TSV arrays, the actual performance improvements of the CACs are less than 50 % of the previously reported values, as shown in this chapter. The lower coding gains in combination with their high overhead costs make existing 3D CACs impractical for most real applications. Thus, an efficient coding method needs to be aware of the edge effects, which demands a simultaneous improvement in the performance of the middle and the edge TSVs.

Another limitation of existing 3D CACs is that they only aim to improve the TSV performance. However, metal wires are not absent in 3D ICs, and the impact of long metal wires on the system's performance is often not negligible. Hence, an efficient technique should improve the performance of metal wires and TSVs simultaneously. Furthermore, the high bit overheads of existing 3D CACs lead to a drastic increase in the TSV power consumption by up to 50 %. This often strictly forbids the usage of existing techniques due to the importance of a low TSV power consumption in 3D ICs.

Since existing 2D CACs can effectively improve the performance and power consumption of metal wires, the technique proposed in this chapter focuses on 2D CACs and provides a methodology to make them efficient for arbitrary TSV arrangements without inducing noticeable costs. Thereby, the proposed technique fully overcomes the severe limitations of previous techniques. The main idea is to use a performance-optimal, edge-effect-aware, bit-to-TSV assignment, similar to the low-power technique presented in the previous Chapter 6. Thereby, the CACs retain their gains for metal wires, but they additionally improve the TSV performance.

A formal method to determine the performance-optimal bit-to-TSV assignment for a given array is derived in this chapter. Moreover, a systematic assignment that is generally valid for arbitrary TSV arrays is contributed. For the optimal/formal method, the optimization constraints can furthermore be

adapted to solely optimize toward a high TSV performance, or additionally toward a low TSV power consumption. In the second scenario, the weight of the power consumption as an optimization objective can be defined arbitrarily.

An in-depth evaluation considering modern TSV arrangements shows that, for all analyzed underlying 2D CACs, the proposed technique outperforms all existing 3D CACs drastically. For example, considering an underlying forbidden-transition-fee (FTF) encoding—one of the most promising 2D CACs [39]—the evaluation shows a more than five times higher TSV performance improvement, compared to the most recently proposed 3D CAC [23]. Moreover, while the recently most promising technique increases the TSV power consumption by 50.0 %, the proposed FTF approach decreases the power consumption by 5.3 %. Despite this drastic improvement in the coding gain, the proposed technique requires a 12.0 % lower bit overhead and a 42.8 % lower coder-decoder circuit (CODEC) area. Furthermore, the technique results in an improvement in the power consumption and performance of global metal wires by 21.9 % and 90.8 %, respectively.

The remainder of this chapter is organized as follows. In Section 7.1, the recently used model to classify the TSV crosstalk is extended to consider the edge effects. Afterward, the limitations of previous 3D CACs are precisely outlined in Section 7.2 with the help of the extended crosstalk classification. The proposed 3D-CAC technique is presented in Section 7.3 and extended toward a low-power 3D-CAC technique in Section 7.4. An in-depth evaluation of the proposed technique, alongside a comparison with previous approaches, is presented in Section 7.5. Finally, the chapter is concluded in Section 7.6.

The idea of a 3D-CAC technique that takes the edge effects into account is published as a proceeding of the "28$^{\text{th}}$ International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)" (see [P1]). The paper was awarded as the best paper of the symposium. As a consequence, an invited peer-reviewed extension is published in "Elsevier Integration" (see [A3]). The extended article mainly covers the enhancement of the proposed technique toward an efficient low-power technique. Furthermore, the extended publication proves that the proposed technique—like all optimization techniques presented in this thesis—is also effective for hexagonal TSV arrangements, proposed in [119], as they still show edge and MOS effects. However, the present thesis does not investigate hexagonal arrangements due to the predominance of mesh arrangements, which are manufactured with lower costs. The interested reader is here referred to [A3].

7.1. Edge-Effect-Aware Crosstalk Classification

The metric that allows for an optimization of the performance at higher abstraction levels is the maximum effective capacitance due to its strong bit-pattern

dependent nature, as the other two parameters in the formula for the maximum propagation delay (Equation (3.43) on Page 50) are purely technology dependent. Thus, for the derivation of existing high-level optimization techniques, the pattern-dependent performance of the i^{th} interconnect was classified by its effective-capacitance value, mathematically expressed in previous works as follows (*e.g.*, [22, 39]):

$$C_{\text{eff},i}[k] = C_{i,i}\Delta b_i^2[k] + \sum_{\substack{j=1 \\ i \neq j}}^{n} C_{i,j}\delta_{i,j}[k]. \tag{7.1}$$

In this equation, $\Delta b_i^2[k]$, multiplied with the interconnect ground capacitance, $C_{i,i}$, is equal to 1 if the interconnect toggles in the respective k^{th} clock cycle, else it is 0. $\delta_{i,j}$ depends on the transitions in the logical values on the i^{th} and the j^{th} interconnect and quantifies the impact of the coupling capacitance between the two lines, $C_{i,j}$, on the performance of the i^{th} interconnect. The impact of the switching on a remote (j^{th}) interconnect on the signal integrity (*e.g.*, propagation delay, switching noise) of the interconnect is commonly referred to as the crosstalk between the two lines [39]. Hence, $\delta_{i,j}$, is referred to as the crosstalk factor throughout this thesis.

Existing CAC techniques assume perfectly temporal aligned signal edges on all interconnects. Consequently, the performance gains of all existing CAC techniques vanish completely for strongly temporal misaligned signal edges. Thus, temporally aligned signal edges are considered throughout this chapter.[2] Hence, the interconnect performance is classified in this chapter by the effective capacitance in the case of temporally aligned edges. Comparing the derived formula for this quantity (*i.e.*, Equation (3.18) on Page 44) with Equation (7.1) reveals that, for perfectly temporal aligned edges, the crosstalk factor can be expressed as follows:

$$\delta_{i,j}[k] = \Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k]. \tag{7.2}$$

For temporally aligned edges, the crosstalk factor can be only 0, 1, or 2. It is equal to 2 if reverse signal transitions occur on the two interconnects (*e.g.*, b_i switches from logical 0 to 1, while b_j switches from logical 1 to 0). $\delta_{i,j}$ is equal to 1 if only interconnect i switches. Otherwise, it is 0. Note that this systematically derived expression for $\delta_{i,j}$ slightly differs from the one used in previous works:

$$\delta_{\text{prev},i,j}[k] = \left|\Delta b_i[k] - \Delta b_j[k]\right|. \tag{7.3}$$

The expression derived in this work differs in so far from the traditional one that

[2] In the following Chapter 8, a technique to improve the performance in the presence of an arbitrary temporal misalignment between the signal edges is presented.

it always results in an effective capacitance of zero for interconnects that are stable. This has the advantage that it will always consider a propagation delay of zero for stable interconnects. However, for the estimation of the performance (*i.e.*, the maximum propagation delay for all possible switching scenarios), it is irrelevant which expression is used.

Combining Equation (7.1) with an abstract capacitance model for the considered interconnect structure, results in a discrete crosstalk classification. Figure 4.1 on Page 60 illustrates the capacitance model for metal wires. Between every adjacent metal-wire pair exists a coupling capacitance of size $C_{\mathrm{mw,c}}$, and every metal wire has a ground capacitance of size $C_{\mathrm{mw,g}}$. Combining this metal-wire capacitance model with Equation (7.1) results in effective capacitance values that range from 0 to $C_{\mathrm{mw,g}}{+}4C_{\mathrm{mw,c}}$, where the worst case occurs if both adjacent aggressor lines of a wire switch in the opposite direction. Previous works neglect the ground capacitances due to the dominance of $C_{\mathrm{mw,c}}$ over $C_{\mathrm{mw,g}}$. Consequently, the metal-wire crosstalk is traditionally classified using five classes: $0C_{\mathrm{mw,c}}$, $1C_{\mathrm{mw,c}}$, $2C_{\mathrm{mw,c}}$, $3C_{\mathrm{mw,c}}$, and $4C_{\mathrm{mw,c}}$ [39].

In the traditional TSV capacitance model, each directly adjacent conductor pair is connected by a coupling capacitance of size $C_{\mathrm{n,prev}}$, and each diagonally adjacent pair by a coupling capacitance of size $C_{\mathrm{d,prev}}$. Thus, the crosstalk of a TSV was previously classified in 81 classes in the range from 0 to $(8C_{\mathrm{n,prev}} + 8C_{\mathrm{d,prev}})$, where the maximum crosstalk class occurs if all eight adjacent neighbors of a middle TSV switch in the opposite direction. To simplify the crosstalk classification, the capacitance value between two directly adjacent middle TSVs (*i.e.*, $C_{\mathrm{n,prev}}$) was used in previous works to normalize the crosstalk classes. Moreover, the normalization value was referred to as C_{3D}. Hence, the TSV crosstalk was classified in the range from $0\,\mathrm{C_{3D}}$ to $(8{+}8\lambda_{\mathrm{d,prev}})C_{\mathrm{3D}}$. Furthermore, in some works, the maximum crosstalk class was denoted as $10\,\mathrm{C_{3D}}$ as $\lambda_{\mathrm{d,prev}}$ (*i.e.*, ${}^{C_{\mathrm{d,prev}}}\!/_{C_{\mathrm{n,prev}}}$) is 0.25 in the traditional capacitance model.

In the following, an edge-effect-aware crosstalk classification for TSV structures, based on the TSV capacitance model from Chapter 4, is presented. The MOS effect is not considered explicitly for TSV crosstalk classification due to two main reasons. First, all existing 3D CAC techniques, as well as the one presented in this chapter, result in equally distributed 0-bit and 1-bit probabilities for all data TSVs. Second, exploiting the MOS effect is not well suited for performance improvement. In contrast to power optimization—where the mean values of the capacitances over time are essential—for performance improvement, the maximum capacitance values over all cycles count. Thus, one must ensure a much stronger stationary of the transmitted data in order to exploit the MOS effect for performance improvement safely. This typically demands additional coding complexity. Consequently, for performance optimization and estimation, capacitance matrices for equally balanced bit probabilities on all TSV (*i.e.*, $p_i = 0.5$ for all i) are considered throughout this thesis.

Table 7.1.: C_{3D} and λ values for the proposed edge-effect-aware crosstalk classification based on the TSV capacitance model proposed in Chapter 4.

TSV technology			**Coefficients**							
r_{tsv} [μm]	d_{min} [μm]	f_s [GHz]	C'_{3D} [pF/m]	λ_d	λ_{e0}	λ_{e1}	λ_{e2}	λ_{c0}	λ_{c1}	λ_{c2}
1.0	4.0	6	94.4	0.36	0.05	1.33	0.23	0.06	1.46	0.31
1.0	4.0	11	85.9	0.30	0.03	1.33	0.17	0.06	1.41	0.23
1.0	4.5	6	91.1	0.38	0.04	1.34	0.25	0.04	1.49	0.32
1.0	4.5	11	80.6	0.32	0.03	1.32	0.19	0.05	1.44	0.24
2.0	8.0	6	111.1	0.34	0.03	1.33	0.20	0.05	1.45	0.27
2.0	8.0	11	93.8	0.29	0.05	1.31	0.16	0.11	1.41	0.21
2.0	8.5	6	110.3	0.35	0.03	1.32	0.20	0.09	1.43	0.26
2.0	8.5	11	91.2	0.29	0.05	1.31	0.16	0.13	1.40	0.21

The same straightforward approach as in previous works is used to obtain an edge-effect-aware TSV crosstalk classification based on the proposed regular capacitance structure (see Figure 4.8 on Page 69). Again, all coefficients in the crosstalk model are capacitance values that are normalized by C_{3D}, which is here equal to C_n. These unit-less coefficients are represented by λ. Thus, λ_x is equal to C_x/C_n. Employing the edge-effect-aware TSV capacitance model, the crosstalk of a TSV located at a single edge of an exemplary 5×5 array is classified in the range from $0C_{3D}$ to $(2{+}4\lambda_d{+}\lambda_{e0}{+}4\lambda_{e1}{+}4\lambda_{c2})C_{3D}$, while the crosstalk classes of a corner TSV range from $0C_{3D}$ to $(2\lambda_d + \lambda_{c0}{+}4\lambda_{c1}{+}4\lambda_{c2})C_{3D}$.

Since dynamic capacitance values due to the MOS effect are not considered here, there is no significant difference between the proposed and the traditional capacitance model for middle TSVs. In both models, a middle TSV has four coupling capacitances with its directly adjacent neighbors and four with its diagonally adjacent neighbors. Thus, the edge-effect-aware crosstalk model still classifies the middle-TSV crosstalk in 81 classes, ranging from $0\,C_{3D}$ to $(8{+}8\lambda_{d,prev})C_{3D}$.

To map the crosstalk classes to actual effective-capacitance values, and thereby to comparable performance quantities, the capacitance-model coefficients reported in Table 4.1 on Page 75 are used to obtain values for C_{3D} and the λ coefficients for various TSV parameters. Table 7.1 reports the resulting values for a mean voltage of 0.5 V on all TSVs (*i.e.*, $V_{dd} = 1\,V$). The table reveals that the relative λ coefficients only slightly differ for the different analyzed TSV geometries. Hence, the proposed crosstalk classification still allows us to derive and rapidly evaluate the general gains of different crosstalk-avoidance strategies without an in-depth analysis of an extensive set of different TSV technologies—despite the added complexity due to the considered edge effects.

7.2. Related Work and its Limitations

This section precisely outlines the limitations of existing CAC techniques for 3D ICs. Crosstalk-avoidance coding was initially proposed for metal wires [39]. The idea of such traditional 2D CACs is to strictly avoid all pattern transitions that result in a crosstalk class for a metal wire that exceeds a specific threshold value. For example, in contrast to an unencoded metal-wire bus, in which the effective capacitance of a line can be as big as $4C_{\mathrm{mw,c}}+C_{\mathrm{mw,g}}$, in a *3C*-encoded and a *2C*-encoded encoded bus the effective-capacitance of a metal wire never exceeds $3C_{\mathrm{mw,c}}+C_{\mathrm{mw,g}}$ and $2C_{\mathrm{mw,c}}+C_{\mathrm{mw,g}}$, respectively. For aggressively scaled metal-wire pitches with $C_{\mathrm{mw,c}} \gg C_{\mathrm{mw,g}}$, *3C* and *2C* CACs improve the maximum propagation delay by asymptotically 25 % and 50 %, respectively. Since the performance is defined as the reciprocal of the maximum propagation delay, $1/\hat{\tau}_{\mathrm{pd}}$, this implies a performance increase by up to 33.3 %, for the *3C* encoding, and, 100 %, for the *2C* encoding. Besides the performance improvement, some 2D CACs moreover improve the power consumption by reducing the sum of the mean effective-capacitance values [39]. This makes these techniques particularly efficient.

Since 2D CACs itself cannot optimize the TSV performance, dedicated crosstalk-avoidance techniques have been proposed recently for TSV arrays [21–23]. For the derivation of all existing TSV CACs, the edge effects are not considered as the traditional capacitance model, used for the underlying crosstalk classification, does not capture them. Furthermore, the efficiency of previous CAC approaches was only evaluated for TSVs located in the middle of an array. The implicit claim of the authors of the related publications was that the edge TSVs either-way have a much lower maximum effective capacitance due to the reduced number of adjacent TSVs and thus do not have to be optimized nor investigated.

However, the proposed edge-effect-aware TSV crosstalk classification shows that this assumption does not hold. Table 7.2 reports the maximum effective capacitance of a middle, an edge, and a corner TSV according to the edge-effect-aware crosstalk model. In accordance with the previously used crosstalk model, the reported maximum normalized effective-capacitance value of a middle TSV is about $10\,\mathrm{C_{3D}}$ to $11\,\mathrm{C_{3D}}$. According to the previously used crosstalk classification, the maximum effective capacitance of an edge TSV is equal to $7\,\mathrm{C_{3D}}$ ($(6 + 4\lambda_{\mathrm{d,prev}})C_{\mathrm{3D}}$), and thus much lower. However, taking the edge effects into account reveals that the maximum effective capacitance of an edge TSV can actually also exceed $10\,\mathrm{C_{3D}}$. Consequently, the maximum effective capacitance of an edge TSV is only 8 % to 10 % (instead of the previously reported 30 %) lower than for a middle TSV. This disproves the fundamental assumption existing 3D CACs rely on.

Existing 3D CACs have in common that they limit the maximum possible amount of opposite transitions on adjacent neighbors for each TSV. For example,

Table 7.2.: Maximum effective-capacitance values of a middle, an edge, and a corner TSV according to the edge-effect-aware crosstalk classification.

TSV technology			**Maximum crosstalk class**		
r_{tsv} [μm]	$d_{\min}$ [μm]	f_{s} [GHz]	*Middle TSV* $(8+8\lambda_{\text{d}})$	*Edge TSV* $(2+4\lambda_{\text{d}}+\lambda_{\text{e0}}+4\lambda_{\text{e1}}+4\lambda_{\text{c2}})$	*Corner TSV* $(2\lambda_{\text{d}}+\lambda_{\text{c0}}+4\lambda_{\text{c1}}+4\lambda_{\text{c2}})$
1.0	4.0	6	$10.90\,C_{\text{3D}}$	$10.06\,C_{\text{3D}}$	$7.87\,C_{\text{3D}}$
1.0	4.0	11	$10.38\,C_{\text{3D}}$	$9.43\,C_{\text{3D}}$	$7.19\,C_{\text{3D}}$
1.0	4.5	6	$11.04\,C_{\text{3D}}$	$10.20\,C_{\text{3D}}$	$8.06\,C_{\text{3D}}$
1.0	4.5	11	$10.57\,C_{\text{3D}}$	$9.57\,C_{\text{3D}}$	$7.41\,C_{\text{3D}}$
2.0	8.0	6	$10.74\,C_{\text{3D}}$	$9.77\,C_{\text{3D}}$	$7.60\,C_{\text{3D}}$
2.0	8.0	11	$10.28\,C_{\text{3D}}$	$9.27\,C_{\text{3D}}$	$7.15\,C_{\text{3D}}$
2.0	8.5	6	$10.77\,C_{\text{3D}}$	$9.74\,C_{\text{3D}}$	$7.56\,C_{\text{3D}}$
2.0	8.5	11	$10.33\,C_{\text{3D}}$	$9.28\,C_{\text{3D}}$	$7.17\,C_{\text{3D}}$

the *6C* CAC ensures that for each TSV, at most three directly adjacent (aggressor) TSVs switch in the opposite direction; and if three directly adjacent neighbors switch in the opposite direction, the fourth one always switches in the same direction [21]. Consequently, for a middle TSV, the maximum crosstalk class is reduced to $(6+8\lambda_d)C_{\text{3D}}$. As an example, for aggressively scaled global TSV dimensions (*i.e.*, $r_{\text{tsv}} = 1\,\mu$m and $d_{\min} = 4\,\mu$m) and a significant frequency of 6 GHz, $(6+8\lambda_d)C_{\text{3D}}$ is equal to $8.88\,\text{C}_{\text{3D}}$.

Edge TSVs have a maximum of three directly adjacent neighbors. Thus, the *6C* CAC does not provide any optimization of the maximum crosstalk class of an edge TSV. Encoded and unencoded, the maximum effective capacitance of an edge TSV is the same ($10.06\,\text{C}_{\text{3D}}$ for the considered example). Thus, the worst-case delay for the encoded patterns occurs at the array edges and no longer in the middle. Consequently, the edge effects reduce the true coding efficiency. Instead of the previously reported theoretical performance increase by 25.0 % ($\hat{C}_{\text{eff}}$ decreases from $10\,C_{\text{3D}}$ to $8\,C_{\text{3D}}$), the improvement is only 7.8 % ($10.90\,C_{\text{3D}}$ to $10.06\,C_{\text{3D}}$) for the analyzed TSV technology. The same analysis for other TSV technologies reveals the same drastic degradation in the coding efficiency by at least 50 % due to the neglected edge effects. Furthermore, one can show in the same way that also all other existing 3D CACs have an over 50 % lower performance gain when the edge effects are considered.

To precisely quantify the real efficiencies of existing 3D CACs, the reduction in the maximum propagation delay of the previously most promising approaches is reanalyzed by means of circuit simulation. Besides the *6C* CAC, the *4LAT* coding [22] and the *6C-FNS* coding [23] are investigated. The *4LAT* coding limits, for each TSV, the maximum number of switching neighbor-TSVs to four. A *6C-FNS* encoding limits the maximum crosstalk class of each middle TSV to $6.5\,\text{C}_{\text{3D}}$—according to the previous crosstalk classification. The *6C* and *4LAT*

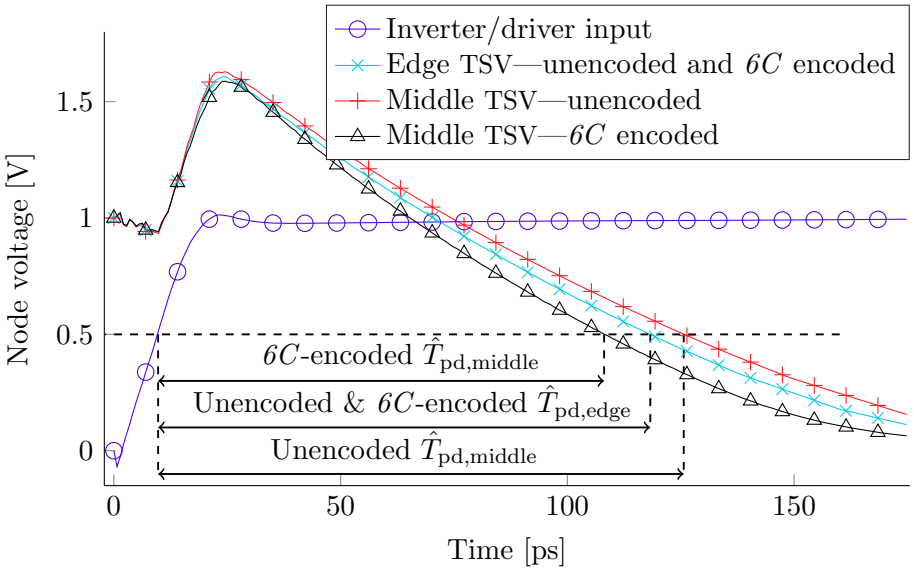


Fig. 7.1: Driver-input and TSV-output voltage waveforms for the worst-case TSV propagation delay, $\hat{T}_{\mathrm{pd}}$, for unencoded data and the *6C* CAC technique.

coding technique are evaluated for a quadratic 5×5 array with r_{tsv}, d_{min}, and l_{tsv} equal to 1 μm, 4 μm, and 50 μm, respectively. A drawback of the *6C-FNS* coding is that it only works for $3\times x$ arrays. Thus, for the analysis of this 3D CAC, the array dimensions are changed to 3×8.

In contrast to the analyses in the respective papers, where only the delay of a middle TSV is investigated, here, the delay of an edge TSV is analyzed as well. To determine the pattern-dependent TSV delays with the *Spectre* circuit simulator, the setup from Section 3.4 is used. Thereby, realistic non-linear driver effects are considered. In the simulations, the strong 22-nm drivers from Subsection 4.4.2 are used. Due to these high-performance drivers, TSV equivalent circuits that were extracted for a significant signal frequency of 10 GHz are integrated into the circuit-simulation setup.

Figure 7.1 illustrates the simulation results for the *6C*-encoded and the unencoded data. Table 7.3 reports the measured maximum propagation delays for all pattern sets. As expected, the worst-case propagation delay always occurs at the array edges for the 3D CACs, while it occurs in the array middle only for the unencoded patterns. Table 7.3 also reports the relative improvements in the worst-case delay for middle TSVs, $\Delta\hat{T}_{\mathrm{pd,middle}}$, and all TSVs, $\Delta\hat{T}_{\mathrm{pd}}$. The reduction in the overall worst-case delay determines the actual performance improvement of the respective CAC technique. All measured delay improvements for the middle TSVs are in accordance with the ones reported in [21–23] for the overall delay improvements. However, the results prove that the edge effects

Table 7.3.: Maximum propagation delay of a middle TSV, $\hat{T}_{\mathrm{pd,middle}}$, and an edge TSV, $\hat{T}_{\mathrm{pd,edge}}$, for unencoded and CAC-encoded data besides the relative coding gains. Here, large-sized drivers are considered.

Data	$\hat{T}_{\mathrm{pd,middle}}$ [ps]	$\hat{T}_{\mathrm{pd,edge}}$ [ps]	$\Delta\hat{T}_{\mathrm{pd,middle}}$ [%]	$\Delta\hat{T}_{\mathrm{pd}}$ [%]
Unencoded	117.0	109.6	—	—
6C [21]	100.6	109.6	−14.1	−6.3
4LAT [22]	93.3	104.5	−20.3	−10.7
6C-FNS [23]	82.9	110.1	−29.2	−5.9

drastically decrease the improvements in the maximum delay, and consequently the efficiencies of the CACs. For example, the delay reduction of the *6C* and the *4LAT* encoding are only about 50 % of their previously expected values.

The delay improvement is only about one-fifth of the previously expected value (5.9 % instead of 29.2 %) for the recently most promising *6C-FNS* coding [23]. This implies that the performance improvement is only 6.2 % instead of 41.2 %, which is a dramatic loss. One reason for this drastic degradation in the coding efficiency is the neglected edge effects. Another one is the generally unconsidered edge TSVs. The *6C-FNS* CAC simply limits the maximum crosstalk class of the middle TSVs to $6.5C_{\mathrm{3D}}$, according to the previous crosstalk classification, while the maximum crosstalk class of the edge TSVs remains unaffected. If all neighbors of a TSV located at a single edge switch in the opposite direction, its crosstalk class is equal to $6C_{\mathrm{n}} + 4C_{\mathrm{d}}$ ($7C_{\mathrm{3D}}$), according to the previously used crosstalk classification. Therefore, the *6C-FNS* encoding is actually a *7C-FNS* encoding, even when the traditional crosstalk classification is considered.

In contrast to the power consumption, the delay has a significant dependency on the drivers. Thus, to prove that the previously outlined degradation in the gain of existing 3D CACs is independent of the drivers, the previous analysis is repeated for the more moderate-sized drivers from Section 3.4. Hence, the width of each transistor is halved compared to the previous analysis. Thereby, the strength of the drivers is reduced. Besides the driver sizing, the rest of the experimental setup is the same as before. Thus, although the resulting weaker drivers actually lead to a reduction in the significant signal frequency, the TSV equivalent circuits are not changed since the edge effects increase for lower frequencies, as shown in Chapter 4. Increasing edge effects are undesired here, as it would further decrease the gain of existing CACs, which prohibits to quantify the driver impact isolated.

Table 7.4 includes the results for the repeated experiment with the more moderate-sized drivers. As expected, the individual delay values are about two

Table 7.4.: Maximum propagation delay of a middle TSV, $\hat{T}_{\mathrm{pd,middle}}$, and an edge TSV, $\hat{T}_{\mathrm{pd,edge}}$, for unencoded and CAC-encoded data besides the relative coding gains. Here, moderate-sized drivers are considered.

Data	$\hat{T}_{\mathrm{pd,middle}}$ [ps]	$\hat{T}_{\mathrm{pd,edge}}$ [ps]	$\Delta\hat{T}_{\mathrm{pd,middle}}$ [%]	$\Delta\hat{T}_{\mathrm{pd}}$ [%]
Unencoded	212.1	197.4	—	—
6C [21]	179.6	197.5	−15.3	−6.9
4LAT [22]	171.1	192.1	−19.3	−9.4
6C-FNS [23]	148.0	197.6	−30.0	−6.8

times higher compared to the previous experiment. Nevertheless, the percentage delay reductions of the 3D CACs do not change significantly compared to the results for the stronger drivers. Still, the edge effects result in the same dramatic decrease in the delay reductions by about 50 % to 75 %. Hence, the outlined dramatic degradation in the gains of the previous 3D CACs is independent of the TSV drivers.

A general drawback of CACs is that they add several redundant bits to the transmitted codewords, compared to the unencoded data words. This cost is quantified by the (bit) overhead of a coding technique, which is mathematically defined as follows:

$$OH(m) = \frac{n-m}{m}, \tag{7.4}$$

where m and n are the bit widths of the unencoded data words and the codewords, respectively [39]. Existing 3D-CAC techniques require asymptotic overheads (*i.e.*, $\lim_{m\to\infty} OH(m)$) of 44 % to 80 %. These high bit overheads of existing 3D CACs lead to a drastically increased overall TSV power consumption as they surpass the power savings per TSV reported in previous works. In the later evaluation section of this chapter, it is shown that existing 3D CACs increase the overall TSV power consumption by 8 % to 50 %.

Two severe limitations of previous 3D CACs were outlined so far. First, the limited performance improvements and, second, the increased TSV power consumption due to the large bit overheads. However, there even is a third drawback of existing approaches: No CAC technique can optimize the performance and the power consumption of metal wires and TSVs simultaneously. This drawback results in high coding-hardware costs for an effective CAC coding of 3D interconnects that are made up of several metal-wire and TSV segments. In such cases, one CAC encoding and decoding is needed per segment. Thus, a 3D CAC should be capable of optimizing the performance of metal wires and TSVs at the same time. Thereby, only one encoder-decoder pair would be required to optimize the performance of arbitrary 3D-interconnect structures.

7.3. Proposed Technique

A 3D-CAC technique that overcomes all outlined limitations of previous techniques is derived throughout this section. First, a TSV-CAC approach that overcomes the limitations due to the edge effects is presented in Subsection 7.3.1. Afterward, it is shown in Subsection 7.3.2 how the proposed TSV-CAC technique is implemented such that a simultaneous improvement in the metal-wire and the TSV performance is achieved. Finally, this 3D CAC is extended to a low-power 3D CAC, which increases the performance of TSVs and metal wires while it simultaneously decreases their power consumption.

7.3.1. General TSV-CAC Approach

In this subsection, a novel CAC approach for TSV arrays, called $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ TSV CAC, is proposed. The presented coding approach overcomes the limitations of previous 3D CACs, which arise due to the edge effects. In detail, the presented coding approach does not only improve the performance of the middle TSVs, but it also improves the performance of TSVs located at the array edges.

The general idea is to reduce the maximum possible effective capacitance of each middle and each single-edge TSV by at least $\omega_\mathrm{m} C_\mathrm{3D}$ and $\omega_\mathrm{e} C_\mathrm{3D}$, respectively. Consequently, the maximum crosstalk class of a TSV in the middle of an array is reduced by an $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ encoding to

$$(8 + 8\lambda_\mathrm{d} - \omega_\mathrm{m}) C_\mathrm{3D}, \tag{7.5}$$

while the maximum crosstalk class of TSVs at an array edge is reduced to

$$(2 + 4\lambda_\mathrm{d} + \lambda_\mathrm{e0} + 4\lambda_\mathrm{e1} + 4\lambda_\mathrm{c2} - \omega_\mathrm{e}) C_\mathrm{3D}. \tag{7.6}$$

As shown in Table 7.2, the highest possible crosstalk class of a middle TSV results in an effective-capacitance value that is about $0.8\,C_\mathrm{3D}$ to $1.0\,C_\mathrm{3D}$ bigger than the counterpart for the highest crosstalk class of an edge TSV. Hence, the most effective $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ encoding has an ω_m that is in that range bigger than ω_e.

7.3.2. 3D-CAC Technique

In this subsection, a coding technique is presented, which implements the previously introduced $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$-CAC approach by means of a traditional 2D CAC. Thereby, a 3D CAC is obtained, which simultaneously improves the TSV and the metal-wire performance.

The fundamental idea is to exploit the bit-level properties of a pattern set that is encoded with a 2D CAC, by means of an assignment of the codeword bits to the TSVs that results in an $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ TSV CAC. In practice, this approach is implemented through the local net-to-TSV assignment at negligible extra costs compared to the traditional 2D CAC, as shown in Subsection 6.5.1.

Besides this $^{\omega_m}/_{\omega_e}$-CAC assignment, the proposed technique is completely constructed of coder-decoder circuits (CODECs), which are well known from 2D crosstalk-avoidance coding. An in-depth explanation of the implementation of these well-known circuits can be found in [39] and is thus not repeated in this thesis. However, the fundamental ideas of the used 2D CACs are briefly reviewed in the following.

Memory-less *2C* CACs are the most popular 2D CACs [78, 120–122]. One major advantage of memory-less over memory-based CACs is their significantly lower hardware requirements [39]. Furthermore, memory-based CACs also work for multiplexed data streams when the encoding and decoding are applied on an end-to-end basis since the coding is based solely on a pattern level. Thus, only memory-less CACs are considered in this work.

Mainly, two different data-encoding approaches exist for memory-less *2C*-CAC encoding: Forbidden-pattern free (FPF) [78] and forbidden-transition free (FTF) [122] encoding. The FPF-CAC approach forbids bit patterns that contain a "010" or a "101" bit sequence as codewords. For example, "111000110" is a valid 9-bit FPF codeword, while "1101100011" is a forbidden pattern and thus not included in the set of codewords. In [39], the authors prove that an FPF metal-wire bus is a *2C*-encoded bus (*i.e.*, maximum possible crosstalk class is $2C_{\mathrm{mw,c}}$), since, for all i, if the crosstalk factor $\delta_{i,i-1}$ is equal to 2 (*i.e.*, bits switch in the opposite direction), the crosstalk factor $\delta_{i,i+1}$ is always 0 (*i.e.*, bits switch in the same direction) and *vice versa*. For the FTF *2C* CAC, all $\delta_{i,i-1}$ and $\delta_{i,i+1}$ values are limited to a maximum of 1 by prohibiting adjacent bits from switching in the opposite direction. Hence, the forbidden transitions are "01" $\rightarrow$ "10" and "10" $\rightarrow$ "01". It is proven in [122], that the largest set of FTF codewords is generated by eliminating the "01" pattern from the odd-even (*i.e.*, $b_{2i+1}b_{2i}$) bit boundaries, and the "10" pattern from the even-odd (*i.e.*, $b_{2i}b_{2i-1}$) boundaries. The asymptotic bit overheads of both CAC techniques, FPF and FTF, are equal (about 44 %) [39].

For the proposed technique, only FTF encoding is investigated, since it has several advantages over FPF encoding. One advantage is that the CODEC of an FTF CAC requires an about 17 % lower gate count and an almost 50 % lower circuit delay [39]. Nevertheless, a traditional FTF CODEC still exhibits a quadratic growth in complexity with an increasing bit width [39]. To overcome this limitation, the bus can be partitioned into smaller groups that are encoded individually. In this case, a difficulty arises due to undesired transitions between adjacent lines of different groups. Two techniques exist which address this issue: Group complement and bit overlapping [39]. Both cause a significant bit and CODEC overhead, which make them suboptimal.

In this chapter, a more effective technique—applicable only for FTF encoding—is presented. To build a 3D power-distribution network requires several power and ground (P/G) TSVs. Power and ground lines are stable, and stable lines

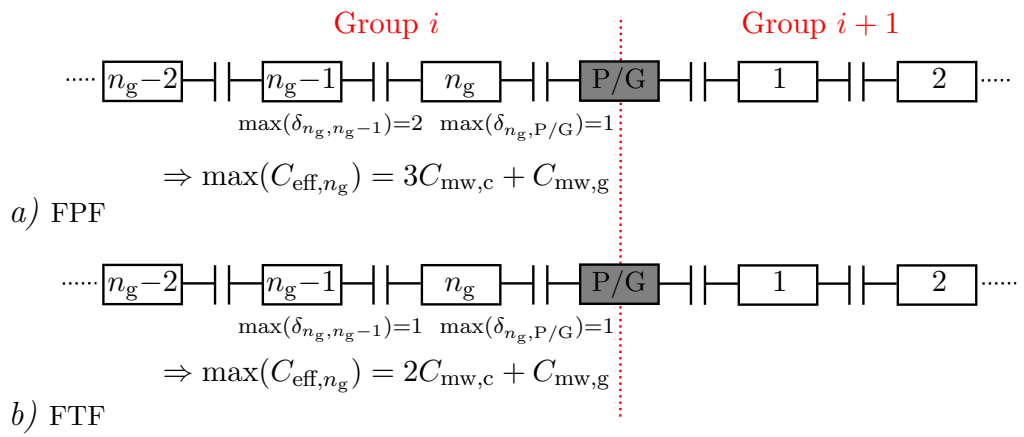


Fig. 7.2: Power or ground lines used for 2D bus partitioning and its influence on the maximum effective capacitance of a metal wire for: *a)* FPF encoding; *b)* FTF encoding.

can be used for FTF-encoding partitioning, as illustrated in Figure 7.2. The metal-wire bus containing n data lines is divided into multiple groups, which are encoded individually using an m_g-to-n_g-bit FTF CAC, where n_g is the number of data lines in a group. Between each first line of a group and the last (*i.e.*, n_g^{th}) line of the previous group, a stable P/G line is added. This bus partitioning generally causes no overhead if dynamic data lines, as well as stable P/G lines, are transmitted over the same 3D-interconnect cluster, which is the typical case [123].

The crosstalk factor $\delta_{i,s}$ of a data line and a stable line is $\Delta b_i^2[k]$ (Equation (7.2) with $\Delta b_j[k]$ equal to 0), and thus limited to a maximum of 1. Consequently, stable lines cannot violate the FTF criterion. In contrast, for an FPF encoding, additional stable lines only lead to a *3C* encoding for the metal wires instead of a *2C* encoding, as illustrated in Figure 7.2b. For an exemplary group size of 5, "00001" → "11110" is a valid FPF-CAC sequence. If the group is terminated by a power line (constant logical 1), the effective pattern sequence, including the stable line, is "00001**1**" → "111<u>10**1**</u>". The second pattern is a forbidden pattern (includes "101" sequence). Thus, a stable line violates the FPF condition, and the n_g^{th} line experiences a crosstalk of $3\,C_{mw,c}$ in the respective cycle. Therefore, the metal-wire crosstalk would no longer be bounded to $2\,C_{mw,c}$.

Another advantage of FTF encoding is that it can be implemented for free for one-hot-encoded data lines. One-hot encoded bit pairs can only take the values "00", "01" or "10". Thus, inverting every second one-hot encoded line by swapping inverting with non-inverting drivers, similar to the approach presented in Chapter 6, already results in an FTF encoding for the lines. After

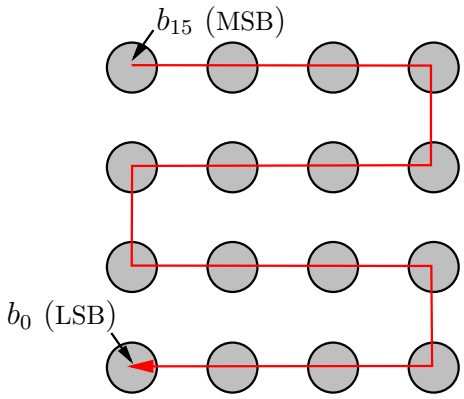


Fig. 7.3: Snake mapping of the bits of a 2D-CAC-encoded data stream onto a TSV array.

the fixed inversions, the one-hot-encoded bit pair $b_{2i+1}b_{2i}$ can only be "10", "11" or "00", while $b_{2i}b_{2i-1}$ can only be "01", "00" or "11". Hence, the "01" pattern is eliminated from the $b_{2i+1}b_{2i}$ boundaries and the "10" pattern from the $b_{2i}b_{2i-1}$ boundaries, resulting in an FTF encoding without inducing any additional bit overhead.

The second analyzed 2D-CAC technique is shielding, which adds stable P/G lines between the data lines to avoid transitions in the opposite direction for adjacent wires. For *2C* shielding, data lines, D, are regularly interleaved with stable shield lines, S, resulting in a $DSDSDS\ldots D$ 2D layout. In this case, each data line is shielded by two adjacent stable lines, resulting in a metal-wire crosstalk of $2C_{\mathrm{mw,c}}\Delta b_i^2[k] \leq 2C_{\mathrm{mw,c}}$ for each data line i. For *3C* shielding, data-line pairs are shielded by stable lines, resulting in a $DDSDDS\ldots D$ 2D layout, which limits the maximum occurring crosstalk class to $3C_{\mathrm{mw,g}}$.

An assignment of all consecutive bit pairs of any 2D-CAC-encoded data stream onto directly adjacent TSV pairs leads to an $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ crosstalk-avoidance encoding for the TSVs. One possible systematic assignment, referred to as snake mapping, is illustrated in Figure 7.3. The snake mapping for a traditional *3C* CAC results in a guaranteed $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ TSV CAC with ω_m and ω_e at least equal to 1 (*i.e.*, a $^1/_1$ CAC). A traditional *2C* CAC results in a guaranteed $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ TSV CAC with ω_m and ω_e at least equal to 2. Thus, one can use any 2D CAC as an $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ CAC by simply applying the snake mapping.

Performance-Optimal 2D-CAC-to-3D-CAC Assignment

Although the previously introduced snake mapping for a 2D CAC results in an $^{\omega_\mathrm{m}}/_{\omega_\mathrm{e}}$ CAC for the TSVs, another assignment can still result in an even higher TSV performance. This has mainly two reasons. First, ω_m should ideally be slightly bigger than ω_e, which is not satisfied by the systematic snake mapping.

Second, the snake mapping does not fully exploit the properties of stable lines. As outlined previously, the crosstalk factor between any data line and a stable line is limited to a maximum of 1. Thus, stable lines should be mapped to the middle of a TSV array in order to reduce the crosstalk of as many data lines as possible. The systematic snake mapping does not meet this requirement, as it only exploits a reduced maximum crosstalk between directly neighbored bit pairs.

Hence, in the following, a formal method to find the performance-optimal assignment of the bits of a given pattern set onto an interconnect structure is derived. By means of this mathematical method, the optimal use of any 2D CAC for a specific TSV array can be determined. Equation (3.47) on Page 51 can be used to express the effective capacitances for an initial mapping, assigning codeword bit b_i to the i^{th} interconnect:

$$\vec{C}_{\text{eff,init}}[k] = \text{diag}(\mathbf{S}[k] \cdot \mathbf{C}). \tag{7.7}$$

As outlined before, the capacitance matrix $\mathbf{C}$ is simply equal to $\mathbf{C}_\text{R}$ for performance optimization.

The maximum values of the crosstalk factors $\delta_{i,j}$ and $\delta_{i,l}$ (for all $j \neq l$) are assumed to be independent in the following. Please note that this assumption holds for random data words and the 2D CACs discussed in this chapter but not for an FPF CAC, where: $\max_k(\delta_{i,i-1})$ and $\max_k(\delta_{i,i+1})$ are equal to 2, but $\max_k(\delta_{i,i-1} + \delta_{i,i+1})$ is also equal to 2 and not 4. For independent crosstalk factors, the maximum effective-capacitance value, for any possible pattern transition and any line (defining the interconnect performance), can be mathematically expressed as

$$\hat{C}_{\text{eff,init}} = \max \text{diag}(\mathbf{S}_\text{wc} \cdot \mathbf{C}_\text{R}). \tag{7.8}$$

where "max diag()" returns the maximum diagonal entry of a matrix. $\mathbf{S}_\text{wc}$ is a matrix containing the worst-case switching/crosstalk with

$$S_{\text{wc},i,j} = \begin{cases} \max_k(\Delta b_i^2[k]) & \text{for } i = j \\ \max_k(\delta_{i,j}[k]) = \max_k(\Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k]) & \text{else.} \end{cases} \tag{7.9}$$

The characteristics of a specific pattern set are captured by $\mathbf{S}_\text{wc}$. For a random/unencoded bit patterns, $S_{\text{wc},i,j}$ is equal to 2, except for the diagonal entries, which are equal to 1. For an FTF-encoded pattern set, $S_{\text{wc},i,j}$ is equal to 2 except for the diagonal elements and its adjacent entries (*i.e.*, $j = i + 1$ or $j = i - 1$) which are equal to 1. An additional stable line at position s (*i.e.*, $\Delta b_s = 0$) leads to $S_{\text{wc},s,j}$ equal to 0 for all j, and $S_{\text{wc},i,s}$ equal to 1 for all $i \neq s$.

In contrast to the technique presented in Chapter 6, here, only a mere reordering of the bit-to-TSV assignment is considered. This is formally expressed by swapping rows and the according columns of the switching matrix, mathematically expressed as follows:

$$\mathbf{S}_{\text{wc,assign}} = \mathbf{I}_\pi \mathbf{S}_{\text{wc}} \mathbf{I}_\pi^{\text{T}}, \tag{7.10}$$

where $\mathbf{I}_\pi$ is a valid $n \times n$ permutation matrix defining the bit-to-TSV assignment. A valid $\mathbf{I}_\pi$ has exactly one 1 in each row and column while all other entries are 0. To assign the i^{th} bit to TSV j, the i^{th} entry of the j^{th} row of the matrix is set to 1. Thus, for an exemplary 2×2 TSV array, to assign bit 1 to TSV 2, bit 2 to TSV 3, bit 3 to TSV 1, and bit 4 to TSV 4,

$$\mathbf{I}_\pi = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{7.11}$$

Thus, the maximum effective capacitance as a function of the bit-to-TSV assignment is expressed as:

$$\hat{C}_{\text{eff,assign}} = \max \operatorname{diag}(\mathbf{I}_\pi \mathbf{S}_{\text{wc}} \mathbf{I}_\pi^{\text{T}} \cdot \mathbf{C}_{\text{R}}). \tag{7.12}$$

Finally, the performance-optimal assignment $\mathbf{I}_{\pi,\text{perf-opt}}$, which minimizes $\hat{C}_{\text{eff}}$, has to full-fill

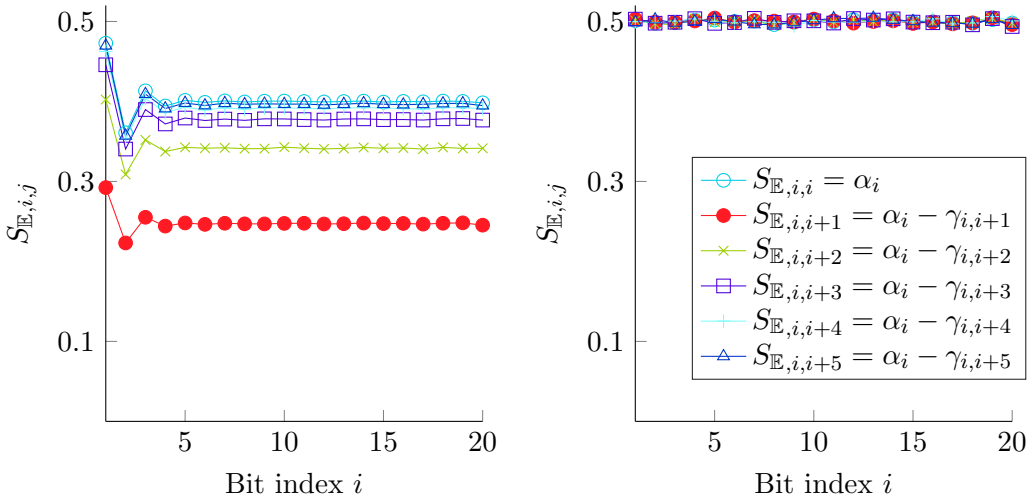
$$\mathbf{I}_{\pi,\text{perf-opt}} = \underset{\mathbf{I}_\pi \in \mathbb{S}_{\mathbf{I}_\pi,n}}{\arg\min} \left(\max \operatorname{diag}(\mathbf{I}_\pi \mathbf{S}_{\text{wc}} \mathbf{I}_\pi^{\text{T}} \cdot \mathbf{C}_{\text{R}}) \right), \tag{7.13}$$

where $\mathbb{S}_{\mathbf{I}_\pi,n}$ is the set of all valid $n \times n$ permutation matrices. Instead of solving Equation (7.13) exactly, $\mathbf{I}_{\pi,\text{perf-opt}}$ is determined in this work with simulated annealing to reduce the computational complexity.

7.4. Extension to a Low-Power 3D CAC

In this section, the proposed 3D-CAC technique is extended to a low-power 3D CAC technique, improving the performance and the power consumption of metal wires and TSVs, simultaneously. From the underlying 2D CACs in the proposed 3D-CAC approach, only FTF encoding effectively reduces the metal-wire power consumption [39]. Therefore, here only an $\omega_{\text{m}}/\omega_{\text{e}}$ TSV CAC based on an FTF encoding is analyzed (with and without bus partitioning).

For the low-power extension, the formal method to express the interconnect power consumption as a function of the TSV assignment, presented in the previous chapter, is reused. However, in the previous method, the possibility



a) FTF *b)* Unencoded

Fig. 7.4: $S_{\mathbb{E},i,j}$ values for an FTF-encoded and an unencoded random data stream.

of assigning negated nets and the MOS effect are considered. Since neither is required here, the formula for the assignment dependent power consumption (Equation (6.13) on Page 121) simplifies to

$$\begin{aligned} P_{\text{assign}} &= \frac{V_{\text{dd}}^2 f}{2} \langle \mathbf{S}_{\mathbb{E},\text{assign}}, \mathbf{C}_{\text{assign}} \rangle \\ &= \frac{V_{\text{dd}}^2 f}{2} \langle \mathbf{I}_\pi \mathbf{S}_\mathbb{E} {\mathbf{I}_\pi}^\mathsf{T}, \mathbf{C}_\text{R} \rangle. \end{aligned} \tag{7.14}$$

Matrix $\mathbf{S}_\mathbb{E}$ includes the switching statistics of the bit values (see Equation (3.45) on Page 51). These switching statistics for unencoded and FTF-encoded data streams are briefly discussed in the following. Thereby, the transmission of uniform-random data (before the CAC encoding) is considered. $S_{\mathbb{E},i,j}$ values for the FTF-encoded and the unencoded data are plotted in Figure 7.4. For the unencoded data, all $\mathbf{S}_\mathbb{E}$ entries are about 0.5 (see Figure 7.4b), as the toggle probability of each bit is 50 % (*i.e.*, $\alpha_i = 0.5$) while the bit pairs switch completely uncorrelated (*i.e.*, $\gamma_{i,j} = 0$). The FTF encoding reduces the switching activities to about 40 % as it results in a positive temporal correlation between consecutively transmitted bit patterns. Furthermore, FTF encoding induces positive switching correlations between bit pairs (*i.e.*, $\gamma_{i,j} > 0$), which further decreases some $\mathbf{S}_\mathbb{E}$ entries. $S_{\mathbb{E},i,j}$ for directly adjacent neighbors is even reduced by about 50 % compared to its value for unencoded data. The further j is apart from i, the more the bit pairs b_i and b_j tend to switch uncorrelated, resulting in increased $S_{\mathbb{E},i,j}$ values which asymptotically reach $\alpha_i \approx 0.4$.

These bit-level statistics reveal that neighboring bit pairs have to be assigned to TSVs connected by a large coupling capacitance to effectively reduce the TSV power consumption for the transmission of FTF encoded patterns. Consequently, neighboring bit pairs have to be mapped onto directly adjacent TSVs. This criterion is satisfied by the proposed snake mapping, presented in the previous section as a high-performance assignment. Thus, the snake mapping of FTF-encoded patterns even results in a low-power 3D CAC.

However, the snake mapping will likely not lead to the lowest power consumption possible, as it neither considers the heterogeneity in the $\mathbf{S}_\mathbb{E}$ entries (*e.g.*, Figure 7.4 shows a $S_{\mathbb{E},i,j}$ increase for i equal to 1 and a decrease for i equal to 2), nor the properties of eventual stable lines. Thus, a formal method to obtain the optimal low-power and high-performance assignment is derived in the following. For this purpose, two metrics must be minimized through the assignment: The power consumption and the maximum propagation delay. Again, the delay is estimated through $\hat{C}_\text{eff,assign}$ to obtain solutions that do not depend on the driver technology. To combine the two metrics into a single total-cost function, TC, the individual costs are normalized by the values for the initial assignment. Hence, the assignment has to minimize

$$\begin{aligned} TC &= \kappa_\text{p} \frac{P_\text{assign}}{P_\text{init}} + (1-\kappa_\text{p}) \frac{\hat{C}_\text{eff,assign}}{\hat{C}_\text{eff,init}} \\ &= \kappa_\text{p} \frac{\langle \mathbf{I}_\pi \mathbf{S}_\mathbb{E} {\mathbf{I}_\pi}^\text{T}, \mathbf{C}_\text{R} \rangle}{\langle \mathbf{S}_\mathbb{E}, \mathbf{C}_\text{R} \rangle} + (1-\kappa_\text{p}) \frac{\max \operatorname{diag}(\mathbf{I}_\pi \mathbf{S}_\text{wc} \mathbf{I}_\pi^\text{T} \mathbf{C}_\text{R})}{\max \operatorname{diag}(\mathbf{S}_\text{wc} \mathbf{C}_\text{R})}, \end{aligned} \tag{7.15}$$

where κ_p—which can take any real value between 0 and 1—defines the weighting of the two optimization objectives. To achieve a low power consumption is more and more prioritized with increasing κ_p, and *vice versa*.

Finally, the optimal bit-to-TSV assignment, for a given κ_p, can be determined by the following equation:

$$\mathbf{I}_{\pi,TC\text{-opt}} = \underset{\mathbf{I}_\pi \in \mathbb{S}_{\mathbf{I}_\pi,n}}{\arg\min} \left(\kappa_\text{p} \frac{P_\text{assign}}{P_\text{init}} + (1-\kappa_\text{p}) \frac{\hat{C}_\text{eff,assign}}{\hat{C}_\text{eff,init}} \right). \tag{7.16}$$

Instead of solving Equation (7.16) exactly, again, simulated annealing is used to reduce the computational complexity.

7.5. Evaluation

Throughout this section, the proposed low-power technique for high-performance 3D interconnects is evaluated in-depth. The expected improvement in the TSV performance due to the proposed 3D-CAC technique is analyzed in Subsection 7.5.1 for various TSV arrays, assuming that the power consumption is

not considered as an optimization objective. Afterward, the trade-off, if the power consumption and the performance are both considered as optimization objectives, is investigated in Subsection 7.5.2. Finally, the proposed technique is compared to previous 3D-CAC techniques.

7.5.1. TSV-Performance Improvement

The effect of the proposed CAC technique on the interconnect performance for various TSV arrangements is quantified in the following for the case that the TSV power consumption is not an optimization objective.

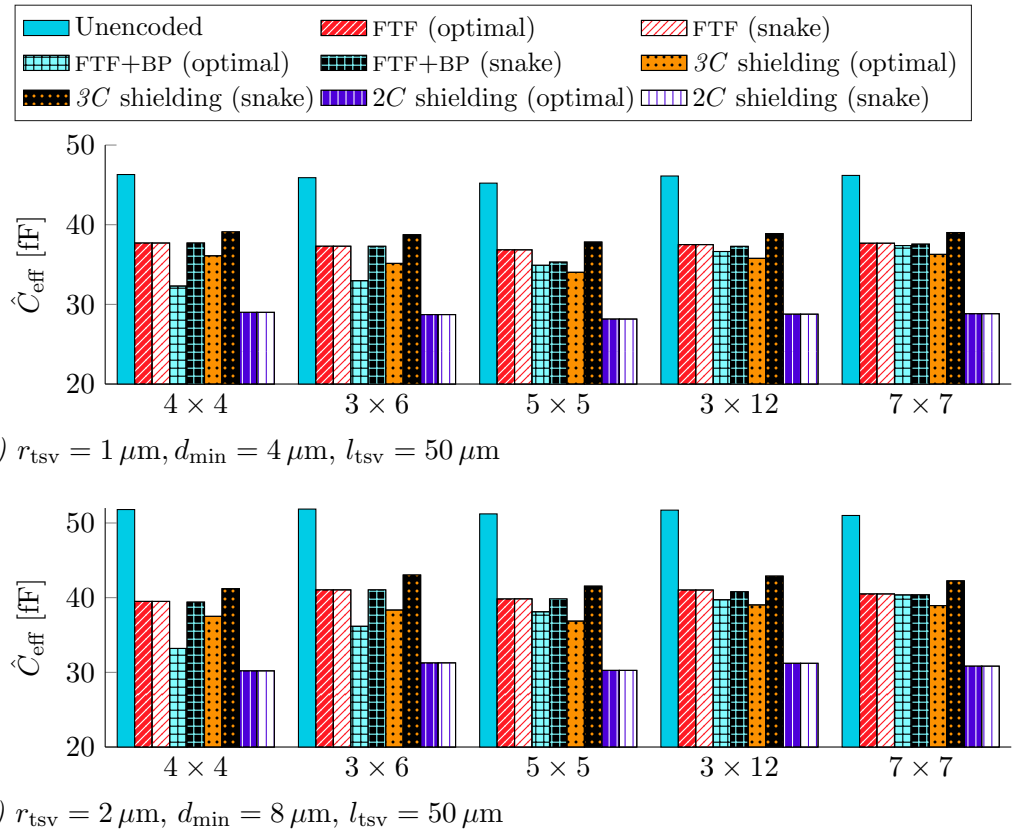
For the analyses, TSV capacitance matrices, extracted with the *Q3D Extractor* for a significant frequency of 10 GHz, are employed in the formulas. The capacitance matrices are extracted for a TSV radius and minimum pitch of 1 μm and 4 μm, respectively. To show the gain for larger TSV dimensions, capacitance matrices are also extracted for a TSV radius and minimum pitch of 2 μm and 8 μm, respectively. The TSV length is 50 μm in both cases. Capacitance matrices for a wide range of TSV-array dimensions $M \times N$ are investigated. More precisely, quadratic arrays from 3×3 to 7×7 and non-quadratic arrays, with M equal to 3 and N equal to 6, 9, and 12, are analyzed.

As the underlying 2D CACs for the proposed technique, *2C* and *3C* shielding, as well as FTF encoding, are analyzed. Besides the traditional FTF encoding, the CAC is additionally investigated for the scenario where approximately 10 % stable TSVs are present in each array, which are exploited by the proposed bus partitioning (BP) technique to reduce the CODEC complexity. For example, two stable lines are assumed in a 4×4 array, which partition the 14 remaining data bits into three groups (b_1 to b_5, b'_1 to b'_5, and b^*_1 to b^*_4).

The overall-maximum effective capacitance, $\hat{C}_{\text{eff}}$, is directly proportional to the maximum delay for ideal drivers, modeled by a simple pull-up/pull-down resistance, as shown in Section 3.2. Thus, $\hat{C}_{\text{eff}}$ is used throughout this thesis to estimate the driver-independent/theoretical delay or performance improvement of optimization techniques.

For the analyzed variants of the proposed technique, as well as for random/unencoded pattern sets, the maximum effective-capacitance values for the TSVs are reported in Figure 7.5. The figure reveals that unencoded the performance is almost independent of the TSV-array shape. For all TSV arrays, the snake mapping leads to a $^2/_{1+\lambda_{e1}}$ ($^{\omega_m}/_{\omega_e}$) CAC for FTF pattern sets and, if no stable lines are present, the expected delay improvement due to the snake mapping and an optimal assignment do not differ noticeably. Compared to an unencoded pattern set, the FTF encoding without BP always leads to an improvement in the maximum effective capacitance by 18 % to 21 %.

As expected, when stable lines are used for shielding or BP, an optimal TSV assignment generally results in a noticeably higher TSV performance improvement than the systematic snake mapping. The only exception is the *2C*-shielding



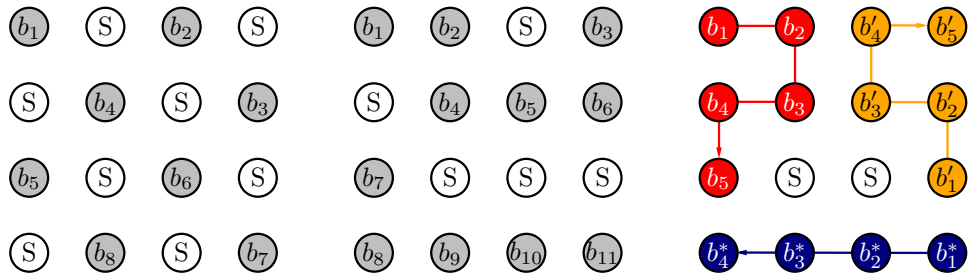
a) $r_{\text{tsv}} = 1\,\mu\text{m}, d_{\text{min}} = 4\,\mu\text{m}, l_{\text{tsv}} = 50\,\mu\text{m}$

b) $r_{\text{tsv}} = 2\,\mu\text{m}, d_{\text{min}} = 8\,\mu\text{m}, l_{\text{tsv}} = 50\,\mu\text{m}$

Fig. 7.5: Effect of the proposed technique on the TSV delay/performance (*i.e.*, $\hat{C}_{\text{eff}}$), for different underlying 2D crosstalk-avoidance techniques. Compared are the two assignment techniques, optimal and snake mapping, for two different TSV dimensions and various array shapes.

technique, where the snake mapping is equivalent to the optimal assignment. Here, the snake mapping results in a $^{4}/_{(2+2\lambda_{e1})}$ CAC and thus completely avoids opposite transitions between directly adjacent TSVs, as illustrated in Figure 7.6a. Thus, the maximum possible effective-capacitance value is drastically improved by about 40 %—implying a performance improvement by over 60 %.

For all other analyzed scenarios, a reassignment of the stable lines, performed by the optimal assignment technique, increases the efficiency of the proposed 3D-CAC technique, as outlined for an exemplary 4×4 array. The snake mapping of *3C*-shielded patterns leads to a $^{(1+2\lambda_{\text{d}})}/_{(1+\lambda_{\text{d}})}$ CAC ($\hat{C}_{\text{eff}}$ reduction by about 15 %). Here, a different assignment can lead to a further increased ω_{m} value, resulting in a higher performance. As illustrated in Figure 7.6b, the stable shields (S) are optimally placed in a way that ω_{m} is increased to $2 + 2\lambda_{\text{d}}$. Consequently, the *3C* shielding ideally results in an improvement in $\hat{C}_{\text{eff}}$ by 22 % instead of the 15 % that are obtained for the snake mapping.



a) 2C shielding *b) 3C* shielding c) FTF+BP

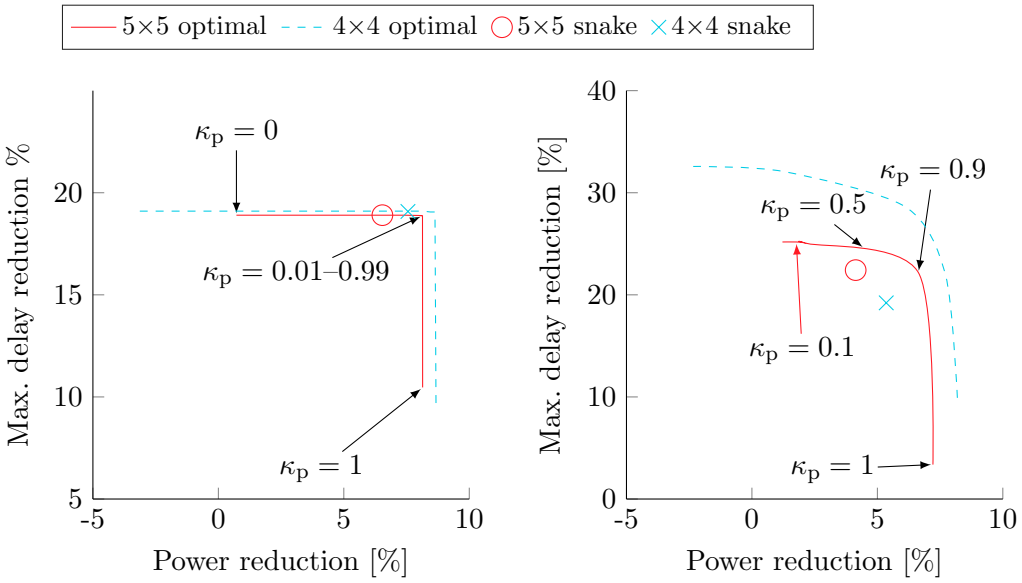
Fig. 7.6: Performance-optimal bit-to-TSV assignments for traditional crosstalk-avoidance techniques and a 4×4 array.

Also, for the combination of FTF encoding and the proposed BP technique, the performance improvement of the optimal assignment technique is significantly higher than for the snake mapping, which results in a $^2/_{(1+\lambda_{e1})}$ TSV CAC. For the exemplary 4×4 array, the optimal assignment—illustrated in Figure 7.6c—increases the ω_m value to $3 + \lambda_d$ by mapping the two stable lines to the array middle. This boosts the theoretical performance improvement from about 23 % to above 40 %. However, a significant increase in ω_m is here only possible for small TSV bundles due to the small fraction of stable TSVs considered. For bigger bundles, the ratio of middle TSVs over edge TSVs increases. Thus, for the FTF encoding with BP, the gain of the optimal assignment technique over the snake mapping decreases toward zero with an increasing TSV count.

In summary, the snake mapping is equivalent to the performance-optimal assignment if no stable lines are present. Stable lines, located at the array edges for the snake mapping, are optimally reassigned to TSVs located in the middle of the arrays in order to shield the maximum amount of TSVs, resulting in a further performance improvement over the snake mapping.

7.5.2. Simultaneous TSV Delay and Power-Consumption Reduction

In the following, the simultaneous reduction in the TSV performance and power consumption of the proposed low-power 3D-CAC technique is investigated. Besides the trade-off between performance and power-consumption improvement, the gain of the optimal assignment over the systematic snake mapping is analyzed. For this purpose, the optimal bit-to-TSV assignments are determined by means of Equation (7.16) for different κ_p values between 0 and 1. Two TSV-array structures are analyzed in this subsection: A 4×4 and a 5×5 array. In both arrays, the TSVs have a radius of 2 μm, a minimum pitch of 8 μm, and a length of 50 μm. The TSV capacitance matrices are extracted with



a) FTF *b)* FTF+BP

Fig. 7.7: Effect of the proposed low-power 3D-CAC technique on the maximum delay and the power consumption of TSVs, compared to unencoded/random data, for an underlying: *a)* FTF encoding; *b)* FTF encoding with two stable lines used for BP. Arrows indicate the locations of κ_p values for the 4×4 array. Negative values indicate a degradation in the power consumption or performance due to the applied technique.

the *Q3D Extractor* for a significant frequency of 10 GHz. Investigated are the transmission of random data, FTF-encoded random data, and FTF-encoded random data with two stable lines exploited for bus partitioning (FTF+BP). To take the induced overhead of an FTF encoding as well as the different numbers of data TSVs into account, the power consumptions normalized by the number of transmitted bits per cycle are compared.

The resulting theoretical delay and power-consumption reductions are shown in Figure 7.7. For the FTF encoding without BP, the results show a reduction in the maximum delay by about 19 % (*i.e.*, 23 % performance increase) for both arrays if the power-consumption is not considered as an optimization objective (*i.e.*, $\kappa_p = 0$). However, no reduction or even a slight increase in the power consumption is obtained in this case. For all κ_p values between 0.01 and 0.99, still, a delay reduction of about 19 % is achieved, but also a power-consumption reduction by about 8 % to 9 %. Therefore, the low-power extension enables a significant reduction in the power consumption while providing the same performance improvement as the mere high-performance approach. If only the

power consumption is considered (*i.e.*, $\kappa_\mathrm{p} = 1$), the reduction in the delay drops to about 10 %, without a noticeable further decrease in the power consumption.

As already discussed in the previous subsection, the snake mapping also leads to the best TSV performance for FTF-encoded data without bus partitioning. The analysis in this subsection shows that the systematic assignment furthermore improves the power consumption by 6.5 % to 7.6 % for FTF-encoded data and thus also results in an effective low-power 3D CAC. However, due to some unexploited heterogeneity in the $\mathbf{S}_\mathbb{E}$ entries, the snake mapping does not result in the minimum possible power consumption and is therefore not of optimal nature.

If stable lines are exploited to partition the FTF encoding (FTF+BP), there is a much stronger trade-off between performance and power-consumption optimization. When the power consumption is not considered during optimization (*i.e.*, $\kappa_\mathrm{p} = 0$), the delay reduction is 25.2 % and 32.6 % for the 5×5 and the 4×4 array, respectively. However, here, the power consumption can be even increased. If both quantities are weighted equally (*i.e.*, $\kappa_\mathrm{p} = 0.5$), the delay reduction is 24.5 % and 28.8 %, while the power consumption is decreased by 4.5 % and 6.7 %, for the 5×5 array and the 4×4 array, respectively. κ_p equal to 0.5 offers a good trade-off for larger TSV counts. While for the smaller 4×4 array, the degradation in the delay reduction compared to the performance-optimal case is about four percentage points (pp), the degradation is below one percentage point for the 5×5 array. For a power weight of 10 % (*i.e.*, $\kappa_\mathrm{p} = 0.1$), the delay reduction is 25.2 % to 32.1 % and the power-consumption reduction is 1.1 % to 1.8 %. Hence, a small penalty in the delay reduction by maximum 0.5 pp, already results in a reduction, instead of an increase, in the power consumption. If the power weight is nine times higher than the performance weight (*i.e.*, $\kappa_\mathrm{p} = 0.9$), the power consumption can be reduced by 6.6 % and 7.7 % while the delay reduction is degraded to 22.2 % to 28.6 %.

The significantly higher trade-off for the FTF encoding with the BP is caused by the stable lines, which can either be mapped onto the TSVs with the overall highest accumulated capacitance, in order to reduce the power consumption effectively, or between the highest amount of dynamic data lines, where they are the most effective shields. Furthermore, since stable lines are not exploited by the systematic snake mapping, it neither results in the lowest possible power consumption nor the highest performance if BP is applied.

In conclusion, the theoretical evaluation reveals that the low-power extension presented in this chapter allows for a decrease in the power consumption while still providing nearly the same performance improvement, but only if no stable lines are present in the array. If stable lines are present, there often is a clear trade-off between power and performance improvement. However, the trade-off vanishes with a decreasing fraction of stable lines. Thus, for big TSV arrays with only a few stable TSVs, a power and performance aware assignment results

in (almost) the same performance improvement as an assignment, which only aims to optimize the TSV performance.

7.5.3. Comparison with existing 3D CACs

In the following, the proposed technique is compared to the previous 3D CACs from [21–23]. Thereby, not only the power consumption and the performance of TSVs are investigated, but also the impact of the techniques on the metal-wire performance and power consumption, as well as the coding overhead.

To measure the power and performance metrics, the transmission of 16-bit data words over a metal-wire bus and a TSV array is simulated with the *Spectre* circuit simulator. For this purpose, the driver and measurement setup from Section 3.4 is used. The investigated stream of data words that have to be transmitted is made up of 1×10^5 uniform-random bit patterns.

In this experiment, the dimensions of the global TSVs are equal to the minimum predicted ones (*i.e.*, $r_{\text{tsv}} = 1\,\mu\text{m}$, $d_{\text{min}} = 4\,\mu\text{m}$, and $l_{\text{tsv}} = 50\,\mu\text{m}$). To obtain the capacitance matrices of the TSV arrays, as well as 3π-RLC equivalent circuits for the *Spectre* simulations, parasitic extractions for the parameterisable TSV-array model are performed. The capacitances matrices and the equivalent circuits of the metal-wire structures are generated with the 40-nm *TSMC* wire tool, which is based on parasitic extractions with *Synopsys Raphael*. Here, metal wires in the fourth layer (M4) with a width/spacing of $0.15\,\mu\text{m}$ and a segment length of $100\,\mu\text{m}$ are considered. The chosen value for the wire width and spacing corresponds with the minimum possible one for M4.

As the underlying 2D CAC in the proposed technique, FTF encoding, *2C* shielding, and *3C* shielding are analyzed. Thereby, two P/G lines are considered for the FTF encoding, which are exploited for BP. This represents a typical case as at least one power and one ground TSV are required to set a power network, spanning over both dies. Thus, the FTF encoding is partitioned into three groups employing two 5-bit-to-7-bit FTF encoders and one 6-bit-to-9-bit FTF encoder.

Thus, arrays containing 25 TSVs are required for the 23 FTF-encoded data lines and the two P/G lines. A 5×5 array is considered here as TSV arrays as square as possible are considered in this subsection. Consequently, a 4×4 TSV array is considered for the unencoded 16-bit data. For the *2C* and the *3C* shielded data, TSV arrays containing 32 and 24 TSVs are required, respectively. Thereby, the two required P/G TSVs are simply embedded into the TSV arrays as shields to minimize the TSV overhead. Thus, a 4×8 and a 4×6 TSV array are used for the analysis of the shielding techniques.

For all previous 3D CACs except the *4LAT*, the encoded data words cannot be transmitted together with P/G lines over one array without violating the CAC-pattern conditions of the encoding. Thus, the existence of a separated bundle for P/G TSVs is assumed for the analysis of these previous 3D CACs.

To obtain the minimum overhead for the previous *6C*, *4LAT*, and *6C-FNS* encoding, a 5×4, a 3×9, and a 3×8 TSV array are required, respectively.

Reported in this subsection are the power-consumption and delay/performance improvements according to circuit simulations for the moderate-sized drivers from Section 7.2.[3] To determine the CODEC complexities, all encoder-decoder pairs are synthesized in a commercial 40-nm technology, and the resulting areas in gate equivalents (GE) are reported. Here, CODEC delays are not reported since they can be hidden in a pipeline [21]. Furthermore, the growths in the coding complexities with the input bit width, m, and the asymptotic bit overheads are reported. Thereby, a minimum of 5 % of stable lines is assumed in the interconnect structures.

The results, reported in Table 7.5, show that previous techniques reduce the maximum TSV delay by at most 9.4 % (*4LAT* encoding). This means a performance increase of only 10.4 %. For the presented technique, the TSV-performance improvement can be more than five times larger (*2C* shielding: 60.8 %). Moreover, the *2C* shielding improves the metal-wire performance by 90.8 % and does not require any CODEC. In comparison, the *4LAT* approach does not optimize the metal-wire performance and requires a large CODEC of 1,915 GE. Moreover, a *4LAT* encoding dramatically increases the TSV and the metal-wire power consumption by 50.1 % and 46.0 %, respectively, while the proposed shielding techniques do not affect the power consumption noticeably. However, the proposed *2C*-shielding has the drawback that it result in a drastically increased number of TSVs (95 %), quantified by the bit overhead, which typically is a citrical metric.

Considering all metrics, the best previous technique is the *6C* coding. It is the only one that does not drastically increase the interconnect power consumption, results in the lowest bit overhead, and needs the smallest CODEC. However, it only achieves a TSV performance improvement by 7.4 % (6.9 % delay reduction).

The best-proposed $\omega_\mathrm{m}/\omega_\mathrm{e}$-CAC variant—if all metrics are taken into account—is the FTF encoding with BP based on a power and performance-aware TSV assignment (highlighted in Table 7.5). With the same minimal bit-overhead as the best-previous *6C* technique, it achieves a more than three times higher TSV-performance increase, and furthermore an increase in the metal-wire performance by 90.8 %. At the same time, it can decrease the power consumption of TSVs and metal wires by 5.3 % and 20.4 %, respectively. Thus, at the same increase in the line count, the proposed technique results in a drastically higher power and performance gains than all previous methods.

The only drawback of the proposed FTF technique over the *6C* CAC is a slight increase in the CODEC size. However, the CODEC is of secondary importance, as both techniques generally have relatively low hardware requirements.

[3] For the larger-sized drivers from Section 7.2, the results show no remarkable differences.

Table 7.5.: Effect of the proposed and existing 3D-CAC techniques on the maximum delay of TSVs ($\hat{T}_{\mathrm{pd,tsv}}$) and metal wires ($\hat{T}_{\mathrm{pd,mw}}$), as well as the power consumption of TSVs (P_{tsv}) and metal wires (P_{mw}). Moreover, the induced bit overhead (OH), the required CODEC area (A_{codec}), the increase in the CODEC complexity with increasing input width, and the asymptotic bit overhead are reported.

Method	**Pattern set**	**Random 16-bit input data**						**General**	
		$\Delta\hat{T}_{\mathrm{pd,tsv}}$ [%]	$\Delta\hat{T}_{\mathrm{pd,mw}}$ [%]	OH [%]	A_{codec} [GE]	ΔP_{tsv} [%]	ΔP_{mw} [%]	$\lim_{m\to\infty} OH(m)$ [%]	*CODEC complexity*
Proposed 3D-CAC technique	FTF BP (κ_{p}=0)	-20.4	-47.6	43.8	411	1.2	-21.9	44	$O(m)$
	FTF BP (κ_{p}=0.5)	**-20.4**	**-47.6**	**43.8**	**411**	**-5.3**	**-21.9**	**44**	$\boldsymbol{O(m)}$
	2C shielding	-37.8	-47.6	87.5	0	1.8	0.0	95	0
	3C shielding	-20.6	-21.3	37.5	0	-1.4	0.0	45	0
Previous 3D-CAC techniques	*6C* [21]	-6.9	0.0	25.0	181	8.1	-7.4	44	$O(m^{1.5})$
	4LAT [22]	-9.4	0.0	68.8	1915	50.1	46.0	80	$O(e^m)$
	6C-FNS [23]	-6.8	0.0	50.0	718	25.7	19.1	50	$O(m)$

Furthermore, due to the superior scaling of the proposed technique due to the applied BP technique, this minor drawback will vanish with an increasing bit width of the transmitted patterns. Hence, the proposed technique significantly outperforms existing 3D CACs in all relevant metrics.

7.6. Conclusion

A low-power technique for high-performance 3D interconnects was presented in this chapter. Previous 3D-CAC techniques only allow for a rather limited improvement in the TSV performance due to the neglected edge effects. Moreover, previous 3D CACs increase the already critical TSV power consumption significantly.

By taking the edge effects into account, the proposed technique overcomes both limitations. Furthermore, the presented technique is designed in a way that it allows for a simultaneous improvement in the power consumption and performance of the metal wires in 3D ICs. The fundamental idea of the proposed method is to exploit the characteristics of pattern sets that are encoded with a traditional 2D CAC by means of a performance and power-aware net-to-TSV assignment that takes the edge effects into account.

Various underlying 2D CACs have been analyzed for the proposed technique, which shows a maximum TSV and metal-wire performance improvement of 60.8 % and 90.8 %, respectively. Despite this vast performance gain, the technique can improve the TSV and metal-wire power consumption by 5.3 % and 21.9 %, respectively. In comparison, the best-previous approach improves the TSV performance by at most 12.0 %, while it provides no performance increase for metal wires. Additionally, the previous technique results in significantly higher hardware costs and a dramatic increase in the interconnect power consumption by up to 50 %. Hence, the proposed technique significantly outperforms previous methods in all important metrics.

CHAPTER 8

Low-Power Technique for High-Performance 3D Interconnects in the Presence of Temporal Misalignment

In the previous chapter, a technique was presented that increases the performance and yet reduces the power consumption of 3D interconnects. However, due to the high coupling/crosstalk complexity in 3D, the technique still leads to an asymptotic bit overhead of 44 %, which is unacceptable in cases where the large TSV dimensions or their low manufacturing yield are serious concerns. Additionally, the technique is only applicable if the signal edges on the interconnects are perfectly temporal aligned.

Thus, a technique to increase the TSV performance in the presence of temporal misalignment (skews) between the signal edges, without inducing a bit overhead at all, is presented in this chapter. The proposed technique is again based on an intelligent local net-to-TSV assignment that exploits temporal misalignment in order to improve the TSV performance.

In detail, the switching times on the signal nets that have to be electrically connected through a given TSV array are analyzed. Afterward, the proposed technique interleaves the net-to-TSV assignment such that the performance is optimized. The proposed technique only requires an interleaving in the local net-to-TSV assignment within the individual arrays, and thus does not result in noticeable implementation costs, as shown in Subsection 6.5.1.

Despite the negligible implementation costs, experimental results show that the approach can improve the TSV performance by 65 %. Moreover, the proposed assignment technique increases the efficiency of traditional light-weight low-power codes (LPCs), which aim to optimize the switching activities of the individual lines [CA1]. In detail, it is shown how the proposed technique is extended by low-complex LPCs such that the TSV power consumption is reduced by over 15 %.

Thus, the technique can improve the TSV performance more significantly than all 3D-CAC techniques, and this while still allowing for a more than two times higher power-consumption reduction at a bit overhead that is lower by a factor of 5×. Hence, the proposed technique can outperform all other techniques significantly.

The remainder of this chapter is structured as follows. Section 8.1 extends the crosstalk model from the previous chapter in a way that it is also applicable in the case of an arbitrary partial temporal misalignment between the signal edges.[1] The proposed optimization technique is presented in Section 8.2. Afterward, it is outlined in Section 8.3 how the assignment method can be used to reduce the 3D-interconnect power consumption effectively. In Section 8.4, the technique is evaluated. Finally, a conclusion is drawn.

The fundamental idea to exploit temporal misalignment to improve the TSV performance at nearly no costs is published as a proceeding of the "29$^{\text{th}}$ International Symposium on Power and Timing Modeling, Optimization, and Simulation (PATMOS)" (see [P3]). The paper was awarded as the best paper of the symposium. As a consequence, an invited peer-reviewed extension is published in "Elsevier Integration" [A4].

8.1. Temporal-Misalignment Effect on the Crosstalk

For the high-level optimization techniques presented in this thesis, the interconnect performance is mathematically quantified through the pattern-dependent maximum effective capacitance, which is proportional to the maximum signal propagation delay (for ideal drivers) as shown in Section 3.2. For all CAC techniques, the effective capacitance is expressed by Equation (7.1) on Page 142.

[1] A partial temporal misalignment between two signals implies here that one signal switches after the other one, but still before the switching of the other line is completed.

In this equation, the crosstalk factor, $\delta_{i,j}$, determines the increase in the delay of the i^{th} interconnect in the respective clock cycle due to the coupling capacitance between the i^{th} and the j^{th} interconnect. Generally, $\delta_{i,j}$ does not only depend on the switching of the binary value transmitted over the i^{th} interconnect in the current cycle, $\Delta b_i[k]$, but also on the switching on the remote j^{th} interconnect, $\Delta b_j[k]$. Hence, the term crosstalk. Perfectly temporal aligned signal edges were assumed for the derivation of all existing CACs. In this case, the crosstalk factor can be mathematically expressed as

$$\delta_{i,j}[k] = \Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k], \tag{8.1}$$

as shown in the previous chapter. According to Equation (8.1), a crosstalk factor can only take one of the three values 0, 1, or 2. The highest/worst-case crosstalk factor of 2 occurs for transitions in the opposite direction on the two interconnects i and j (*i.e.*, $\Delta b_i[k]\Delta b_j[k] = -1$).

However, the formula for the effective capacitance changes if the signals on the interconnects switch completely temporal misaligned (*e.g.*, the input of the i^{th} line b_i switches after the switching of b_j is already propagated over the j^{th} line). In this case, the effective capacitance has to be calculated by means of Equation (3.19) (see Page 44) instead of (3.18). Comparing Equation (3.19) with the general formula Equation (7.1) reveals that the crosstalk factor for completely temporal misaligned signal edges is equal to

$$\delta_{i,j}[k] = \Delta b_i^2[k]. \tag{8.2}$$

Hence, the delay/performance of interconnect i no longer depends on the switching on interconnect j when the events on both nets occur with a large enough temporal misalignment (skew). The expected values of Equation (8.1) and Equation (8.2) are generally equal. With misalignment, $\delta_{i,j}$ is increased from 0 to 1 for transitions in the same direction (*i.e.*, $\Delta b_i[k]\Delta b_j[k] = 1$). In contrast, $\delta_{i,j}$ is reduced from 2 to 1 for transitions in the opposite direction (*i.e.*, $\Delta b_i[k]\Delta b_j[k] = -1$). Thus, misalignment itself cannot improve the power consumption, determined by the mean effective capacitance values—at least not for random bit patterns with $\mathbb{E}\{\Delta b_i \Delta b_j\}$ equal to 0 for $i \neq j$.

However, the performance can be improved, as the misalignment reduces the maximum value of a crosstalk factor from 2 to 1, which implies an improvement in the maximum propagation delay. Consequently, a circuit designer could force a sufficient misalignment on some input lines artificially, in order to improve the interconnect performance without increasing the power requirements. For TSVs, a small temporal misalignment is often already enough to effectively improve the performance. The reason is the relatively fast TSV switching when crosstalk effects are reduced due to the low TSV resistances and ground-capacitances.

One possibility to force misalignment is to use the rising clock edges for the data transmission over one half of the TSVs in the array, and the falling clock edges for the other half. In this case, 50 % of the signal-edge pairs are temporally misaligned. However, intrinsic small temporal skews between the input-signal transitions are typically present either-way, if the inputs of the TSVs are not direct flip-flop outputs. In that case, some signals switch always temporally misaligned as a result of different timing paths in the preceding circuit. Generally, such an intrinsic misalignment tends to be normally distributed with zero-mean. Hence, the assumption that only completely misaligned or perfectly aligned edges occur does not hold in this case.

To estimate the interconnect crosstalk in the presence of an arbitrary/partial misalignment using Equation (7.1), $\delta_{i,j}$ must be a function of the temporal alignment of the signal edges on the interconnects i and j:

$$\delta_{i,j}[k] = \Delta b_i^2[k] - A_{i,j}\Delta b_i[k]\Delta b_j[k]. \tag{8.3}$$

Here, $A_{i,j}$ is the alignment factor which depends on the time skew between the edges of the binary signals transmitted over interconnect i and j. The skew is mathematically expressed as

$$T_{\text{skew},i,j} = T_{\text{edge},i} - T_{\text{edge},j}, \tag{8.4}$$

where $T_{\text{edge},i}$ and $T_{\text{edge},j}$ are the switching times of the binary values on the interconnects i and j (if both signals toggle in the cycle) relative to the rising clock edge. It is already known from the previous analysis that $A_{i,j}$ is 0 if the signals switch completely temporal misaligned, and 1 if they switch perfectly temporal aligned (*i.e.*, for $T_{\text{edge},i} = T_{\text{edge},j}$).

To obtain a good estimate for $A_{i,j}$ as a function of $T_{\text{skew},i,j}$, *Spectre* circuit simulations for the setup illustrated in Figure 3.3 on Page 52 are carried out. A 4×4 array with typical global TSV dimensions (*i.e.*, $r_{\text{tsv}} = 2\,\mu\text{m}$, $d_{\text{min}} = 8\,\mu\text{m}$, and $l_{\text{tsv}} = 50\,\mu\text{m}$) is analyzed. All TSV parasitics used throughout this chapter are obtained by parasitic extractions with the *Q3D Extractor* for the TSV-array model presented in Subsection 2.4.1. In the simulations, each TSV (modeled by means of an extracted of 3π-RLC circuits) is driven by an inverter made up of 22-nm PTM transistors. The $^{W}/_{L_{\text{min}}}$ ratios of the n-MOS and the p-MOS transistor of these drivers are 24 and 48, respectively. This sizing is chosen for two reasons. First, a transistor-sizing ratio of 2$\times$ between the p-MOS and the n-MOS results in almost equal rise and fall times for the driver outputs. This ensures that the drivers do not increase misalignment effects artificially. Second, the crosstalk noise that is induced on the TSVs can exceed $^{V_{\text{dd}}}/_{2}$ (*i.e.*, 0.5 V) for smaller transistor sizes, resulting in undesired signal glitches. Such glitches are particularly critical for misaligned signal edges where they do not necessarily

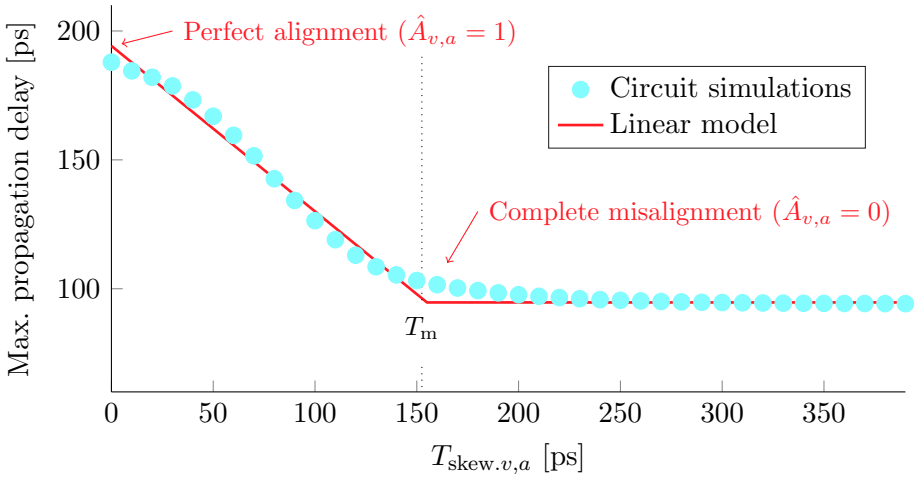


Fig. 8.1: Maximum propagation delay of a victim TSV, over the delay of its input-signal edges relative to the signal edges on all remaining aggressor TSVs.

occur at the beginning of the cycles. At the far end, each TSV is terminated in a 22-nm PTM inverter (W/L_{min} equal to 4 for the n-MOS and 8 for the p-MOS) charging a 1 fF load.

Bit sources ($V_{\text{dd}} = 1\,\text{V}$) with a transition time of 10 ps and a bit duration of 1 ns are used to generate the stimuli for the circuit simulations. For each line, the transmitted bit sequence and the delay of the signal edges, $T_{\text{edge},i}$, relative to the rising clock edges (at $k \cdot 1\,\text{ns}$ with $k \in \mathbb{N}$), can be varied to model different misalignment scenarios. Again, the inverter circuits to create a realistic shape of the input-voltage waveforms of the TSVs, are sized as the load drivers at the far end. The worst-case propagation delay, $\hat{T}_{\text{pd}}$, from the TSV input, v_{in}, to the TSV output, v_{tsv}, is measured for each analyzed scenario.

The measured propagation delays are interpreted to build models for $A_{i,j}$ for the worst-case switching (*i.e.*, transitions in the opposite direction on TSV_i and TSV_j) as a function of the switching times of the input signals. The alignment factor for the worst-case switching is symbolized as $\hat{A}_{i,j}$ in the following.

8.1.1. Linear Model

In the author's initial publication about exploiting misalignment [P3], the worst-case propagation delay of a crosstalk victim, TSV_v, located in the middle of the TSV array, was analyzed as a function of the misalignment/time-skew of its signal edges against the signal edges for all other TSVs, which served in the analysis as aggressors. In other words, in the analysis, all TSV-input signals switched at the exact same time, except the input signal of the crosstalk victim TSV_v which switched by $T_{\text{skew},v,a}$ later, where the $T_{\text{skew},v,a}$ value is varied.

The resulting maximum propagation delay of the victim TSV over the time skew, $T_{\text{skew},v,a}$, is shown by the blue dots in Figure 8.1. Without misalignment, the worst-case TSV delay is 187.89 ps. However, the delay decreases significantly with an increasing misalignment until a certain point from which the curve flattens out rapidly. This behavior is captured quite well by a simple linear model for $\hat{A}_{v,a}$. From $\hat{A}_{v,a}$ equal to 1 for no misalignment (*i.e.*, $T_{\text{skew},v,a} = 0$), up to a threshold value, T_{m}, $\hat{A}_{v,a}$ decreases linearly to 0. Afterward, $\hat{A}_{v,a}$ is clipped to 0. The fitted T_{m} value is in the range of the mean propagation delay of the victim for no signal misalignment.

The red lines in Figure 8.1 illustrate the estimated propagation delays according to this linear regression. In the initial publication [P3], the same behavior was assumed for a negative temporal misalignment. Negative misalignment implies here that the input of an aggressor TSV switches later than the input of the victim TSV. Thus, the following formula was proposed to estimate the $\hat{A}_{i,j}$ values:

$$\hat{A}_{i,j} \approx \max\left(1 - \frac{|T_{\text{skew},i,j}|}{T_{\text{m}}},\ 0\right). \tag{8.5}$$

This equation, combined with Equation (8.3), builds a simple model to estimate the maximum interconnect crosstalk in the presence of arbitrary signal misalignment. In the remainder of this chapter, this simple model is referred to as linear crosstalk model.

8.1.2. Look-up-Table Model

An in-depth analysis of the crosstalk as a function of the misalignment shows that a crosstalk optimization by means of the linear model might even result in a lower interconnect performance than without any optimization. In the following, the reason for this is outlined. Furthermore, a new model to estimate the alignment factors, $\hat{A}_{i,j}$ is presented, which resolves this issue.

First, the circuit simulations from Subsection 8.1.1 are extended such that the effect of negative temporal misalignment between the edges on the victim line and the aggressor lines is analyzed as well. As illustrated in Figure 8.2 for a set of representative examples, this analysis shows that, in contradiction to the linear model, temporal misalignment does not only affect the crosstalk/performance positively. When aggressor signals switch their level shortly after a victim, they induce crosstalk noise on the victim line during its transition phase. In the worst-case crosstalk scenarios, where multiple aggressors switch in the opposite direction of the victim, the aggressors induce a huge noise peak on the victim line, which counteracts its active switching. Thus, if this noise is induced before the victim net exceeds the threshold value of the load driver (typically $V_{\text{dd}}/2$), or if the noise raises the signal temporarily over the threshold again (glitch), the misalignment increases the propagation delay compared to

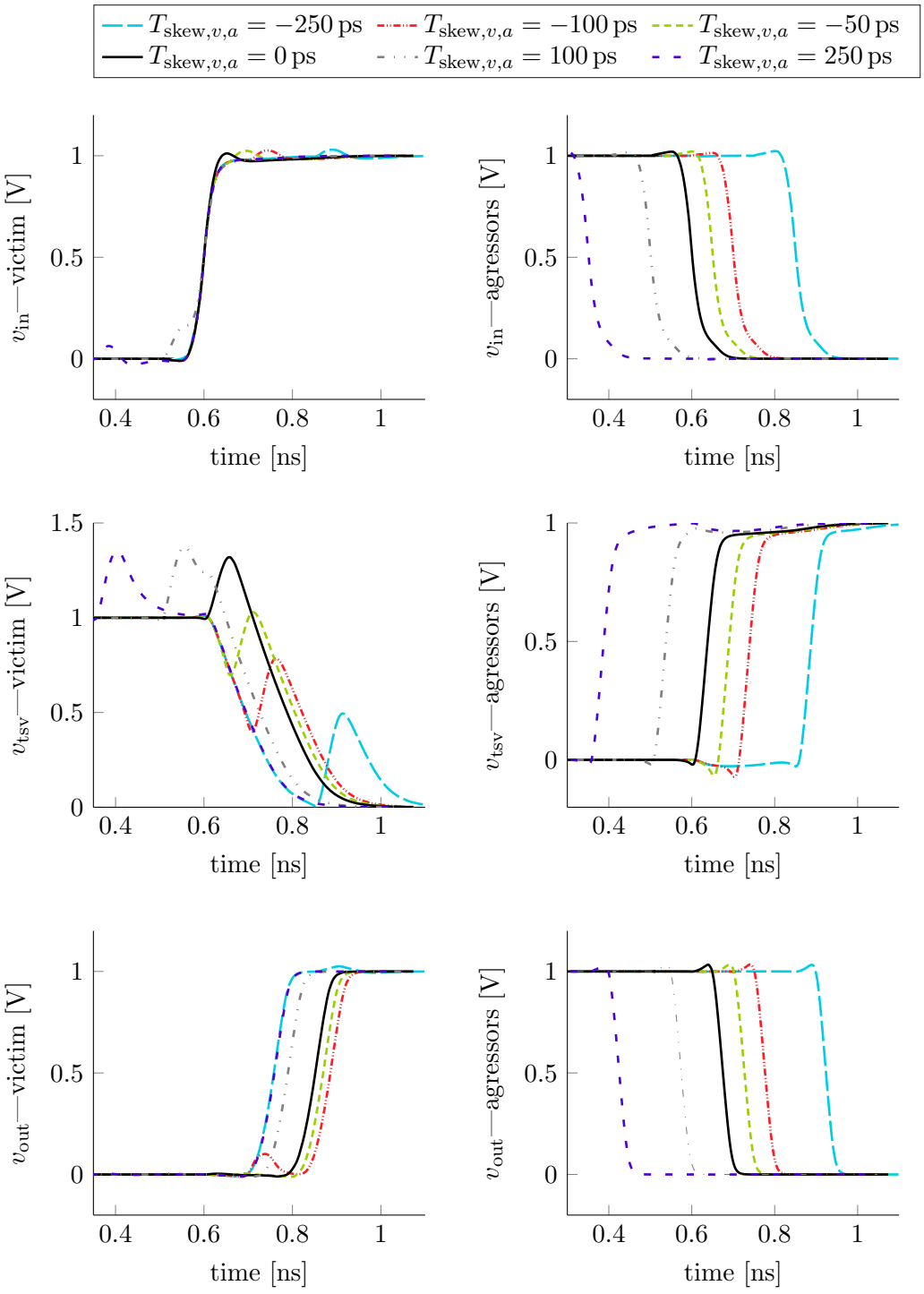


Fig. 8.2: Effect of positive and negative time skews between the signal edges of a victim TSV relative to the edges on the remaining aggressor TSVs on the signal integrity.

the scenario of no misalignment. These scenarios are illustrated in Figure 8.2 by the green and the red curves, which represent a negative victim-to-aggressor misalignment of −50 ps and −100 ps, respectively. In both scenarios, the victim output toggles/switches later than for the case of no misalignment (black/solid curves), although the victim input switches at the same time in all cases. The green curves ($T_{\text{skew},v,a} = -50\,\text{ps}$) represent the scenario where the noise is induced before the signal crosses the threshold value; while the red curve ($T_{\text{skew},v,a} = -100\,\text{ps}$) represents the scenario where the aggressor noise leads to a glitch on the signal. Note that this glitch is also noticeable in the output signal, v_{out}, of the aggressor line. For large negative skews, represented by the blue curves ($T_{\text{skew},v,a} = -250\,\text{ps}$), the misalignment again has a strongly positive impact on the performance. In this case, the crosstalk noise is induced after the switching of the logical value on the TSV is already completed. Thus, the crosstalk does not affect the output signal negatively here, resulting in the lowest propagation delay (*i.e.*, highest performance).

Another phenomenon, not captured by the linear crosstalk model, is that the effect of the skew between two signal edges also depends on the overall standard deviation of the switching times. In those scenarios where most signal edges are strongly misaligned, the maximum height of the induced noise is significantly lower, compared to the scenario where most aggressor edges occur temporally aligned. Additionally, when the mean misalignment is strong, the transition times of the signals generally tend to be shorter. Hence, with an increasing standard deviation of the switching times, the time range where misalignment has a negative impact is reduced.

Thus, to obtain a more precise crosstalk model, the propagation delay of a victim TSV_v is analyzed as a function of the temporal misalignment between its input signal and the input signal of one directly adjacent aggressor TSV_a. Furthermore, this simulation is executed for various values for the standard deviation in the switching times of the remaining input signals, σ_{skew}. The time skew $T_{\text{skew},v,a}$ is swept from −500 ps to 500 ps with a step width of 2 ps to cover a wide range of positive and negative misalignment scenarios. Additionally, the standard deviation, σ_{skew}, is swept from 0 ps to 500 ps. Out of the maximum signal propagation delays, the $\hat{A}_{v,a}$ values are determined as follows:

$$\hat{A}_{v,a}(\sigma_{\text{skew}}, T_{\text{skew},v,a}) = \frac{\hat{T}_{\text{pd}}(\sigma_{\text{skew}}, T_{\text{skew},v,a}) - \hat{T}_{\text{pd}}(\sigma_{\text{skew}}, 500\,\text{ps})}{\hat{T}_{\text{pd}}(\sigma_{\text{skew}}, 0\,\text{ps}) - \hat{T}_{\text{pd}}(\sigma_{\text{skew}}, 500\,\text{ps})}. \tag{8.6}$$

where $\hat{T}_{\text{pd}}(\sigma_{\text{skew}}, 500\,\text{ps})$ and $\hat{T}_{\text{pd}}(\sigma_{\text{skew}}, 0\,\text{ps})$ are equal to the delays for a complete misalignment and a perfect alignment between the switching times on TSV_v and TSV_a for a standard deviation in the misalignment of σ_{skew}.

The results—illustrated in Figure 8.3—show that misalignment can even increase some $\hat{A}_{i,j}$ values to over 1.2, implying an increase in the worst-case

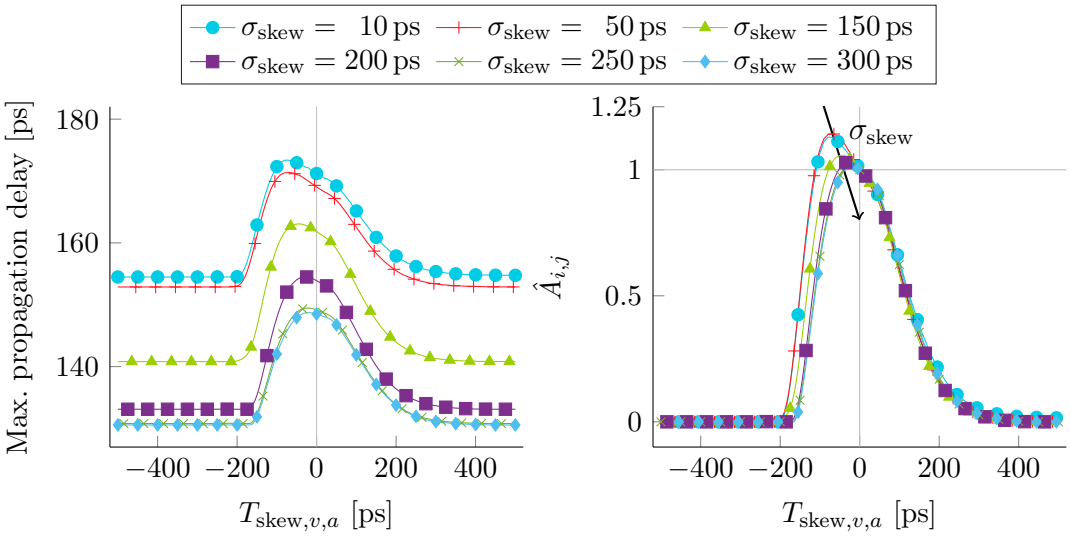


Fig. 8.3: Worst-case TSV propagation delay and resulting alignment coefficients, $\hat{A}_{v,a}$, over the misalignment, $T_{\text{skew},v,a}$, between the victim TSV_v and one adjacent aggressor TSV_a.

crosstalk between the lines by more than 20 %, compared to the scenario of no signal misalignment. As expected, the maximum possible $\hat{A}_{i,j}$ value decreases with an increasing standard deviation in the switching times. Furthermore, an increased standard deviation leads to a narrower $\hat{A}_{i,j}$ curve, which results in generally lower $\hat{A}_{i,j}$ values for a given time skew. However, the $\hat{A}_{i,j}$ curves remain unchanged from a certain standard deviation $\sigma_{\text{skew,max}}$ onward.

The observed behavior is hard to capture in a single closed-form expression. Thus, this work proposes to create a 2D lookup table (LUT) out of the circuit simulation results, which can be used at design time to estimate the crosstalk peak and thereby the required timing budget for the TSVs.[2] The output of the LUT is the $\hat{A}_{i,j}$ value, the first input is the σ_{skew} value, and the second input the $T_{\text{skew},v,a}$ value. In between the discreet analyzed points, a linear interpolation is used to determine the $\hat{A}_{i,j}$ values for arbitrary misalignment scenarios. The values are clipped outside the analyzed range, as an $\hat{A}_{i,j}$ value does not change with any further change in σ_{skew} or $T_{\text{skew},v,a}$. After using the LUT to determine the $\hat{A}_{i,j}$ values, the maximum crosstalk of the lines is again estimated through Equation (8.3). This enhanced model for the crosstalk in the presence of arbitrary misalignment scenarios is referred to as the LUT-based crosstalk model in the remainder of this thesis.

[2]Since this approach requires extensive circuit simulations for each new driver sizing/technology, the resulting high-level model for the TSV performance is not universally valid. For this reason, it is not included in Part II of this thesis.

8.2. Exploiting Misalignment to Improve the Performance

In the previous chapter, a technique was presented, which effectively improves the TSV performance by means of a crosstalk-aware net-to-TSV assignment, exploiting the specific bit-level characteristic of 2D-CAC-encoded data. The idea of using the crosstalk-aware assignment is reused for the technique presented in this chapter. However, here, the effect of misalignment between the edges of the input signals is exploited.

The following formula for the performance-optimal assignment was derived in Subsection 7.3.2:

$$\mathbf{I}_{\pi,\text{perf-opt}} = \underset{\mathbf{I}_\pi \in \mathbb{S}_{\mathbf{I}_\pi,n}}{\arg\min} \left(\max \operatorname{diag}(\mathbf{I}_\pi \mathbf{S}_\text{wc} \mathbf{I}_\pi^\text{T} \cdot \mathbf{C}_\text{R}) \right). \tag{8.7}$$

In this equation, the entries of the worst-case switching matrix are equal to

$$S_{\text{wc},i,j} = \begin{cases} \max_k(\Delta b_i^2[k]) & \text{for } i = j \\ \max_k(\delta_{i,j}[k]) & \text{else.} \end{cases} \tag{8.8}$$

The technique presented in the previous chapter is based on the crosstalk model for perfectly aligned edges. Thus, $\Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k]$ was recently used for $\delta_{i,j}[k]$. In this section, the misalignment-aware crosstalk model (*i.e.*, Equation (8.3)) has to be used instead for $\delta_{i,j}[k]$, resulting in

$$S_{\text{wc},i,j} = \begin{cases} \max_k(\Delta b_i^2[k]) & \text{for } i = j \\ \max_k(\Delta b_i^2[k] - A_{i,j}\Delta b_i[k]\Delta b_j[k]) & \text{else.} \end{cases} \tag{8.9}$$

Without any further extension, the approach from the previous chapter can be used to obtain the performance optimal TSV assignment for arbitrary data streams (*e.g.*, encoded or random) in the presence of temporal misalignment.

For perfectly temporal aligned signal edges, a data encoding is needed to improve the TSV performance effectively. In contrast, an intelligent assignment can already significantly improve the TSV performance for any random/unencoded data stream when the signal edges are temporally misaligned, as shown in the following. This implies that no bit or circuit overhead due to an encoding is required for performance improvement.

For random data, $\max_k(\Delta b_i^2[k])$ is 1 for a signal line i which eventually toggles its logical value. Again, the properties of stable P/G lines are exploited to obtain an even better TSV performance. For a stable line, $\max_k(\Delta b_i^2[k])$ is clearly 0. The worst-case crosstalk (*i.e.*, $\max_k(\Delta b_i^2[k] - A_{i,j}\Delta b_i[k]\Delta b_j[k])$) for two random data lines i and j is equal to $1 + \hat{A}_{i,j}$, occurring for opposite

transitions in the bit values on the two lines. However, if the i^{th} line is a dynamic data line while the j^{th} line is stable, $\max_k(\delta_{i,j}[k])$ is equal to 1, as $\Delta b_j[k]$ is always 0. If i is a stable line, $\max_k(\delta_{i,j}[k])$ is 0 in any case.

Thus, the technique proposed throughout this chapter consists of the following three steps. First, either the linear crosstalk model or the LUT-based crosstalk model is used to determine the $\hat{A}_{i,j}$ values, out of which $\mathbf{S}_{\text{wc}}$ is generated using

$$S_{\text{wc},i,j} = \begin{cases} 0 & \text{if } i \text{ is a stable line} \\ 1 & \text{else if } i = j, \text{ or if } j \text{ is a stable line} \\ 1 + \hat{A}_{i,j} & \text{else.} \end{cases} \tag{8.10}$$

Subsequently, the heterogeneity in the $\mathbf{S}_{\text{wc}}$ entries due to the varying $\hat{A}_{i,j}$ values is exploited to improve the performance through an optimal net-to-TSV assignment, determined by means of Equation (8.7).

After the optimal assignment is determined, a final sanity check should be performed if the LUT-based crosstalk model was used in the first step. Initially, the standard deviation of the switching times on all nets is considered to obtain the $\hat{A}_{i,j}$ values in the LUT-based model. However, only adjacent TSVs show significant crosstalk due to their large coupling capacitances. The $\hat{A}_{i,j}$ values, and thereby the crosstalk, increases with a decreased σ_{skew}. Therefore, one has to check if, for any TSV, the standard deviation of the switching times for (only) the adjacent TSVs is smaller than the used σ_{skew} to determine the $\hat{A}_{i,j}$ values. If so, the related $\hat{A}_{i,j}$ values have to be adjusted.

However, it is improbable that an adjustment is needed, since the technique proposed in this chapter generally maximizes the standard deviation of the switching times for adjacent TSVs, as it always assigns heavily misaligned nets to strongly coupled lines to reduce their crosstalk effectively. In fact, none of the cases analyzed in the remainder of the chapter needed an adjustment of an $\hat{A}_{i,j}$ value during the sanity check.

8.3. Effect on the TSV Power Consumption

The effect of the proposed technique on the TSV power consumption is investigated in this section. Thereby, it is shown that the proposed method drastically improves the effectiveness of traditional hardware-efficient low-power codes.

In Chapter 3, an equation for the mean power consumption due to each individual interconnect was derived, which helped to outline the magnitude of the edge effects, and the resulting optimization potential in the following chapters. In contrast, the power consumption due to the individual capacitances is analyzed to outline the effect of the proposed misalignment-aware assignment on the TSV power consumption and its implications for the integration of low-power techniques.

A formula for the power consumption due to an individual interconnect ground/self-capacitance, $C_{i,i}$, can be derived from the formula for the respective cycle-based energy dissipation (Equation (3.11) on Page 41):

$$\begin{aligned} P_{C,i,i} &= \mathbb{E}\{E_{i,i}\} \cdot f \\ &= \frac{V_{\mathrm{dd}}^2 f}{2} C_{i,i} \cdot \mathbb{E}\{\Delta b_i^2\}. \end{aligned} \tag{8.11}$$

In the same way, a formula for the mean power consumption due to a coupling capacitance, $C_{i,j}$, for perfectly temporal aligned edges on the two interconnects i and j, can be derived from Equation (3.9) on Page 41:

$$\begin{aligned} P_{C,i,j} &= \mathbb{E}\{E_{i,j}\} \cdot f \\ &= \frac{V_{\mathrm{dd}}^2 f}{2} C_{i,j} \cdot \mathbb{E}\{(\Delta b_i - \Delta b_j)^2\}. \end{aligned} \tag{8.12}$$

These two formulas were used for the derivation of most existing LPCs [CA1]. Hence, two main low-power coding approaches exist. The first one aims to minimize the power consumption due to the coupling capacitances, by minimizing the coupling-switching activities defined as

$$\begin{aligned} \alpha_{\mathrm{coup},i,j} &\stackrel{\mathrm{def}}{=} \mathbb{E}\{(\Delta b_i - \Delta b_j)^2\} \\ &= \mathbb{E}\{\Delta b_i^2 + \Delta b_j^2 - 2\Delta b_i \Delta b_j\}. \end{aligned} \tag{8.13}$$

Note that $\Delta b_i^2[k] + \Delta b_j^2[k] - 2\Delta b_i[k]\Delta b_j[k]$ is equal to the the sum of the two crosstalk factors for the i^{th} and the j^{th} line (*i.e.*, $\delta_{i,j}[k] + \delta_{j,i}[k]$) for aligned signals edges. Thus, the first approach aims to minimize the crosstalk between the lines. The second approach reduces the self-switching activities, α_i, equal to the expected Δb_i^2 values. This approach traditionally aims at primary reducing the power consumption due to the ground/self-capacitances.

A metric for the choice of the right coding approach is the coupling-switching factor, ζ. It defines the ratio of the power consumption due to coupling switching over the power consumption due to self-switching for random/unencoded data (*i.e.*, $\alpha_{\mathrm{coup},i,j} = 1$, and $\alpha_i = 0.5$). Without temporal misalignment, the coupling-switching factor is mathematically expressed as

$$\zeta = \frac{2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} C_{i,j}}{\sum_{i=1}^{n} C_{i,i}}. \tag{8.14}$$

The coupling-switching factor is much greater than 1 for modern 2D and 3D interconnects. Thus, recent LPCs for metal wires focus on primary reducing the coupling switching, instead of the self-switching (*e.g.*, [79, 80, 124]). However, these codes are typically not effective for TSV arrays as the coupling changes due

to the different capacitance structures of metal wires and TSV arrays, outlined in Chapter 4. Another drawback of codes reducing the $\alpha_{\text{coup},i,j}$ values is that they show a significantly higher complexity since the probability of opposite transitions on adjacent lines has to be minimized.

Low-power codes focusing on the α_i values need to minimize the activities of the individual bits, which typically can be achieved with a much simpler encoder-decoder architecture and thus lower costs. Furthermore, note that encoding techniques which primary improve the self-switching, typically also reduce the coupling switching, and *vice versa*, since

$$\alpha_{\text{coup},i,j} = \mathbb{E}\{\Delta b_i^2 + \Delta b_j^2 - 2\Delta b_i \Delta b_j\} = \alpha_i + \alpha_j - 2\gamma_{i,j}. \tag{8.15}$$

The formula for the power consumption due to a coupling capacitance differs for completely misaligned signal edges. From Equation (3.10) on Page 41, the following formula can be derived for this case:

$$\begin{aligned} P_{C,i,j} &= \frac{V_{\text{dd}}^2 f}{2} C_{i,j} \cdot \mathbb{E}\{(\Delta b_i^2 + \Delta {b_j}^2)\} \\ &= \frac{V_{\text{dd}}^2 C_{i,i} f}{2} (\alpha_i + \alpha_j). \end{aligned} \tag{8.16}$$

For random data, which implies that $\mathbb{E}\{\Delta b_i \Delta b_j\}$ is equal to 0, the expected values for Equation (8.12) and Equation (8.16) are equal. Hence, solely a temporal misalignment does not affect the overall power consumption for the transmission of random data. However, Equation (8.16) reveals that the mean power consumption due to a coupling capacitance only depends on the self-switching and no longer on the coupling switching for completely misaligned signal edges. Thus, the coupling capacitances behave as self-capacitances for temporally misaligned signal edges. This implies a theoretical coupling-switching factor of 0. Hence, for a complete temporal misalignment between the signal edges, simple codes that focus on reducing the self-switching have the highest gains, independent of the sizes of the coupling capacitances.

The effect of partial misalignment on the power consumption is discussed in the following. For this purpose, the clock-cycle-average power consumption ($T_{\text{clk}} = 1\,\text{ns}$) of two lines connected by a 20 fF coupling capacitance is analyzed with *Spectre* over the time skew between the edges of the input signals of the 22-nm TSV drivers. The results are plotted in Figure 8.4 for all switching patterns that are affected by temporal misalignment (*i.e.*, all patterns with a switching on both lines). Note that, due to the symmetry, the concept of "negative" time skews does not hold in this power analysis. The figure reveals that a linear fit between the power estimates for full and no misalignment, employing an alignment factor $\bar{A}_{i,j}$, enables to model cases of partial misalignment accurately. Hence, the power consumption due to a coupling capacitance for arbitrary

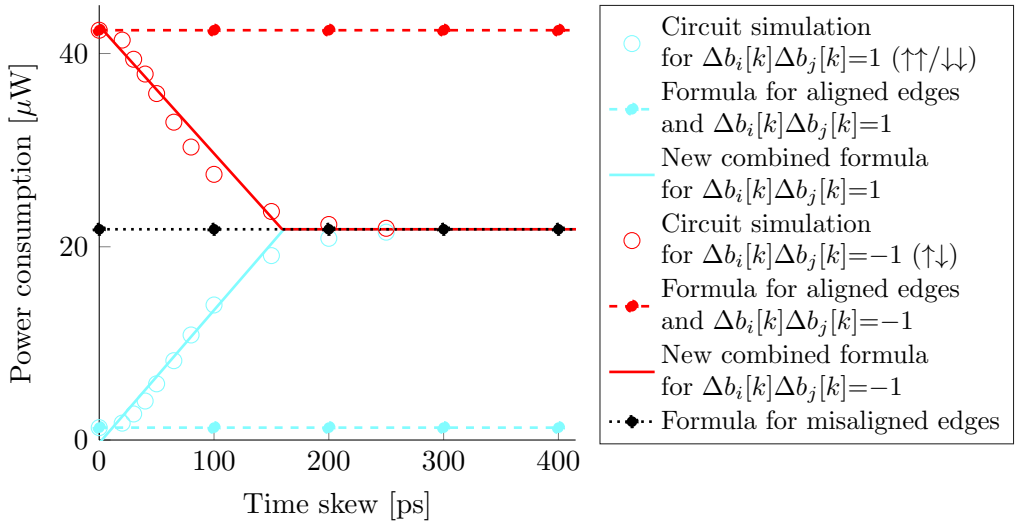


Fig. 8.4: Clock-cycle-average power consumption of two lines, connected by a 20 fF coupling capacitance, over the skew between the signal edges.

misalignment scenarios is estimated as

$$\begin{aligned} P_{C,i,j} &= \frac{V_{\mathrm{dd}}^2 f}{2} C_{i,j} \cdot \mathbb{E}\{\Delta b_i^2 + \Delta b_j^2 - 2\bar{A}_{i,j}\Delta b_i \Delta b_j\} \\ &= \frac{V_{\mathrm{dd}}^2 f}{2} C_{i,j} \cdot \big((1-\bar{A}_{i,j})(\alpha_i + \alpha_j) + \bar{A}_{i,j}\alpha_{\mathrm{coup},i,j}\big). \end{aligned} \tag{8.17}$$

This equation reveals that the power consumption due to a coupling capacitance depends on the coupling switching as well as on the self-switching for partially misaligned signal edges. Consequently, the effective coupling-switching factor for the case of arbitrary temporal misalignment is mathematically expressed as

$$\zeta_e = \frac{2\sum_{i=1}^{n}\sum_{j=i+1}^{n}\bar{A}_{i,j}C_{i,j}}{\sum_{i=1}^{n}C_{i,i} + 2\sum_{i=1}^{n}\sum_{j=i+1}^{n}(1-\bar{A}_{i,j})C_{i,j}}. \tag{8.18}$$

Hence, ζ_e is effectively minimized if strongly misaligned signals (*i.e.*, small $\bar{A}_{i,j}$) are transmitted over interconnects connected by a large coupling capacitance, $C_{i,j}$, implying an increased effectiveness of low-cost LPCs. This requirement is satisfied by the proposed assignment technique, as a significant temporal misalignment on strongly coupled lines improves the TSV performance effectively.

Consider the example of artificial misalignment employing both clock edges. In this case, $\bar{A}_{i,j}$ is 0 if one of the two nets i or j is an output of a falling-edge triggered flip-flop while the other one belongs to a rising edge triggered flip-flop, and 1 if both nets are outputs of the same kind of flip-flop. Without the

artificial misalignment (*i.e.*, for all $\bar{A}_{i,j}$ equal to 1), ζ_e is above 60 for typical global TSV dimensions. Thus, to use coding techniques that minimize the coupling switching is reasonable here, despite the resulting increased coding complexity. However, after the proposed assignment technique is applied for the artificial misalignment, ζ_e is drastically reduced to about 0.2. Hence, after applying the proposed technique, existing light-weight LPCs are even several times more effective than complex ones. Consequently, the proposed technique maximizes the efficiency of many traditional LPCs for TSV-based interconnects.

8.4. Evaluation

This section evaluates the proposed technique in depth. First, the expected reduction in the maximum propagation delay due to the proposed technique is investigated in Subsection 8.4.1. Furthermore, the gain of exploiting an artificially generated misalignment compared to an intrinsic misalignment is analyzed. To report representative numbers for the expected/mean gain of the proposed technique, several thousand different misalignment scenarios are considered in this first subsection. This makes an evaluation by means of circuit simulations impossible. Hence, delay reductions according to the high-level formulas (ideal drivers) are presented in Subsection 8.4.1.

In the following two subsections, the impact of the technique on the TSV performance for the 22-nm drivers is investigated. For this purpose, extensive *Spectre* circuit simulations are carried out for a 4×4 array. In Subsection 8.4.2, the performance improvement of the proposed technique is investigated for different misalignment scenarios. Thereby, the focus lies on the difference between the application of the simple linear and the more complex LUT-based misalignment-aware crosstalk model for the proposed technique.

Afterward, an in-depth comparison of the presented technique and existing 3D-CAC techniques is drawn in Subsection 8.4.3. The common objective of all analyzed techniques is an improved TSV performance, while the techniques proposed in this thesis furthermore aim for an improved TSV power consumption. Nevertheless, the proposed and the previous techniques are analyzed based on a large number of criteria in Subsection 8.4.3. In addition to the power consumption and performance, the switching noise, the bit overhead, the CODEC area, and the maximum throughput of the 4×4 array are analyzed.

8.4.1. Expected Delay Reduction

In this subsection, the expected reduction in the maximum propagation delay of the newly proposed technique is investigated for a large set of misalignment scenarios. Thereby, the advanced LUT-based crosstalk model is applied to determine the performance-optimal local TSV assignment for various misalignment scenarios in the switching times of the nets.

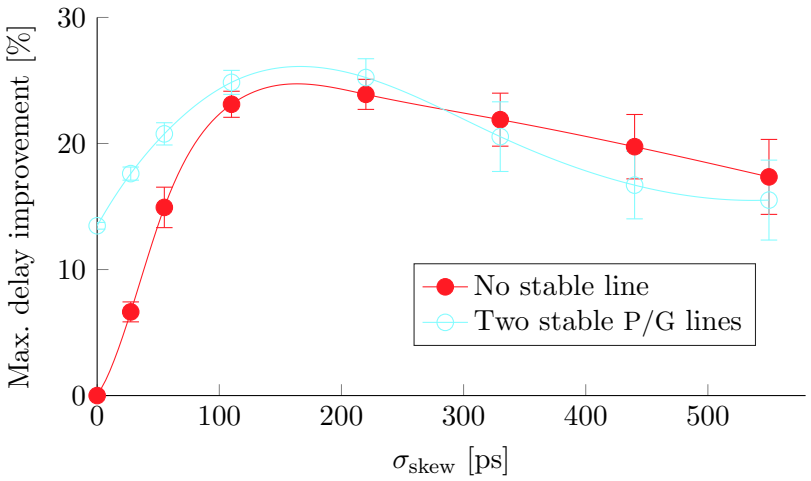


Fig. 8.5: Expected TSV-delay reduction, with 68 % confidence intervals, over the standard deviation of the input-signal switching times, σ_{skew}. For each analyzed σ_{skew} value, 1,000 misalignment scenarios are analyzed.

A zero-mean, normally distributed time skew in the switching times of the signals transmitted over a typical 4×4 TSV array (*i.e.*, $r_{\text{tsv}} = 2\,\mu\text{m}$, $d_{\text{min}} = 8\,\mu\text{m}$, and $l_{\text{tsv}} = 50\,\mu\text{m}$) is considered in the first analysis. The standard deviation of the time skews, σ_{skew}, is varied to quantify the expected performance gains of the proposed technique for various misalignment scenarios. In order to obtain representative results, all simulations are executed 1,000 times for no stable lines, and 1,000 times for one power and one ground line in the array (new time skews are drawn randomly for each run).

The resulting mean improvements in the maximum TSV delay, compared to a random assignment, as well as the 68 % confidence intervals, are plotted in Figure 8.5 over σ_{skew}. For no stable line and no signal misalignment (*i.e.*, for $\sigma_{\text{skew}} = 0$), the performance/delay improvement is obviously zero. Thus, a routing-minimal assignment should be applied in this case. However, for a standard deviation in the switching times of 55 ps, the mean reduction in the TSV delay is already above 15 %, with a confidence interval of ± 1.6 %. The highest delay reduction of 24.7 % (*i.e.*, 32.9 % performance increase) is obtained for a σ_{skew} of about 160 ps. For a huge standard deviation in the switching times, a significant misalignment between most adjacent lines is generally present before the proposed optimization in the TSV assignment is applied. Thus, with an ongoing increase in σ_{skew}, the performance/delay improvement of the proposed technique decreases again. However, even for a standard deviation of 500 ps, the expected delay reduction is still above 17 %.

With stable lines, the curve has a similar shape. However, in this case, the delay reduction is already 13.5 % for no temporal misalignment (*i.e.*, $\sigma_{\text{skew}} = 0$),

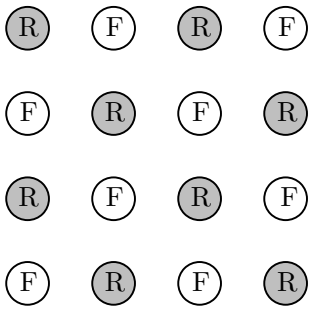


Fig. 8.6: Assignment for artificially temporal misaligned signals: Regular interleaving of the nets of rising-edge-triggered (R) and falling-edge-triggered (F) flip-flops.

as the proposed technique exploits the stable P/G lines. Here, the technique finds the position in the array where the stable lines serve as the most effective shields for the dynamic data lines. Exploiting stable lines also leads to a potentially higher delay reduction by approximately two percentage points (pp). However, for a large standard deviation in the switching times, the delay reduction for an array with stable lines is lower than for one without stable lines (*ca.* 2.5 pp for σ_{skew} equal to 500 ps). With stable lines, fewer adjacent lines potentially toggle in the opposite direction. Consequently, the optimization potential is reduced for higher σ_{skew} values, compared to the scenario where no stable lines are present.

Compared to the expected delay improvements according to the linear crosstalk model—presented in [P3], but not included in this thesis—the reported reductions are lower for small σ_{skew} values and higher for a large σ_{skew} values. The reason is that the advanced LUT-based model takes into account that misalignment is not always beneficial for the crosstalk/delay.

In the following, a second analysis is presented, whose first goal is to compare the optimization potential of the proposed technique for artificially generated and intrinsic signal misalignment. For the analyzed scenario of artificial misalignment, the TSVs are driven by flip-flops where one-half of the flip-flops are rising-edge triggered, and the other half falling-edge triggered. The clock frequency is set to 1 GHz. Thereby, 50 % of the signal pairs switch perfectly temporal aligned and 50 % completely temporal misaligned.

Applying the proposed assignment technique results in a regular interleaving of nets belonging to rising-edge (R) and falling-edge-triggered (F) flip-flops for the artificial misalignment, illustrated in Figure 8.6 for a 4×4 array. This regular interleaving leads to the highest performance as it completely avoids temporally aligned transitions on directly adjacent TSVs, which are connected by the largest coupling capacitances.

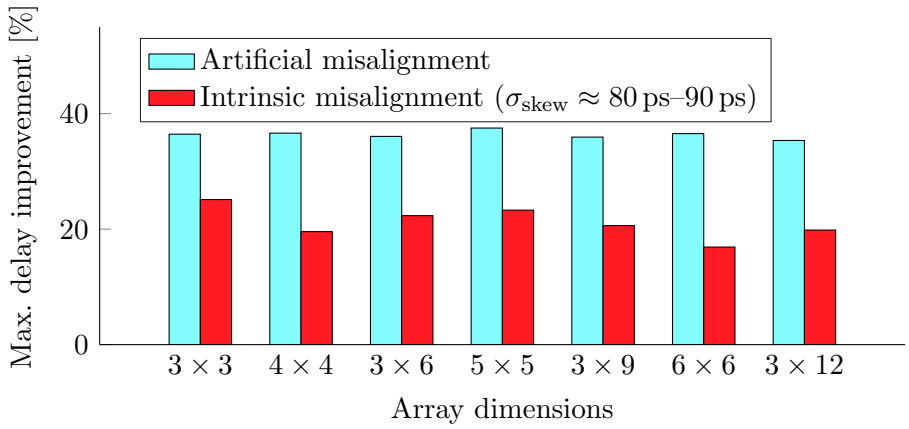


Fig. 8.7: Reduction in the maximum TSV delay, $\hat{T}_{\text{pd}}$, over the array shape for artificial and intrinsic temporal signal misalignment. Thereby, aggressively scaled TSV dimensions are considered (*i.e.*, $r_{\text{tsv}} = 1\,\mu\text{m}$ and $d_{\text{min}} = 4\,\mu\text{m}$).

A zero-mean Gaussian-distributed skew in the switching times with a standard deviation of around 80 ps to 90 ps is analyzed for the intrinsic temporal misalignment. In this analysis, the array dimensions M and N are varied to determine if the array size affects the optimization potential. In the second analysis, the TSV radius and minimum pitch are scaled down by a factor of $2\times$ compared to the first one to $1\,\mu\text{m}$ and $4\,\mu\text{m}$, respectively. This aims at quantifying the effectiveness of the proposed technique for more advanced TSV manufacturing processes.

The resulting reduction in the maximum TSV delay, compared to no temporal misalignment for artificially generated misalignment and a random assignment for intrinsic misalignment, are reported in Figure 8.7. The delay reduction for the artificial misalignment is almost stable over the array dimensions (*ca.* 36 % to 37 %). These values are significantly bigger compared to the scenarios where an intrinsic temporal misalignment is exploited. For intrinsic signal misalignment, even a non-optimal assignment typically results in a higher TSV performance, compared to the scenario of no signal misalignment. In these scenarios, the optimization potential is consequently lower. Furthermore, the variance in the switching times is way bigger for artificial misalignment. The amplitude of the misalignment between the outputs of falling-edge and rising-edge triggered flip-flops is big enough to consider the transitions as completely independent (*i.e.*, $\hat{A}_{i,j} = 0$). This leads to a particularly small crosstalk.

Moreover, the delay improvement shows significantly higher fluctuations for intrinsic misalignment, as the gain depends on the quality of the random assignment. However, a general dependency between the performance gains and

the array dimensions, $M \times N$, cannot be identified. Furthermore, the average reduction in the maximum delay for the more aggressively scaled TSV dimensions is about 21 %, which is in accordance with the expected delay reduction for the larger TSV dimensions, reported in Figure 8.5. Thus, the potential gain of the approach tends to be independent of the TSV-manufacturing maturity.

Summarized, the analyses show that even exploiting a small intrinsic signal misalignment has the potential to improve the TSV performance effectively—especially if stable lines are present in the array. For a guaranteed performance improvement, an artificially generated signal misalignment is required, which has the drawback of additional costs, for example, due to the integration of falling-edge-triggered flip-flops. Furthermore, the analyses indicate that the theoretical performance improvement is more or less unaffected by the array size and the geometrical TSV dimensions.

8.4.2. Delay Reduction for Various Misalignment Scenarios

The maximum TSV propagation delay is analyzed in this subsection with *Spectre* for no stable line in the array and five different scenarios of intrinsic temporal misalignment in the input signals. Here, an analysis of a wide range of misalignment scenarios, as done in the theoretical analysis in the previous subsection, is not possible due to the high computational complexity of the circuit simulations. In all scenarios, the mean switching time of the signals is 390 ps after the rising clock edge in the respective cycle. The analyzed standard deviations of the switching times are 0 ps (no misalignment), 15 ps, 25 ps, 90 ps, and 180 ps. All switching times are truncated to a minimum of 0 ps and a maximum of 700 ps in order to present realistic misalignment scenarios. For all analyzed scenarios, the maximum propagation delay is measured for:

1. a random net-to-TSV assignment;
2. an assignment determined by means of the proposed technique employing the linear crosstalk model;
3. an assignment determined by means of the proposed technique employing the LUT-based crosstalk model.

These different settings allow for quantifying the effect of the proposed technique for rather small, as well as relatively large, intrinsic misalignment quantities. Artificial misalignment is analyzed in the following Subsection 8.4.3.

Table 8.1 reports the resulting maximum propagation delays. The relative percentage deviations, compared to the scenario of no signal misalignment, are also included. This comparison is used to prevent an influence of the varying quality of the random assignment on the reported performance improvements. The results show that the maximum possible propagation delay in the presence of signal misalignment is even higher than for no misalignment (reported up to 11.8 %). Thus, the results validate again that input-signal misalignment does not necessarily have a positive effect on the TSV performance. Since

Table 8.1.: Maximum propagation delay of a TSV, for five scenarios of temporal misalignment, if the proposed TSV assignment technique is applied and if a random assignment is used.

Assignment	**Maximum propagation delay [ps]**				
	σ_{skew} $= 0\,\text{ps}$	σ_{skew} $\approx 15\,\text{ps}$	σ_{skew} $\approx 25\,\text{ps}$	σ_{skew} $\approx 90\,\text{ps}$	σ_{skew} $\approx 180\,\text{ps}$
Random	185.3	186.9 (**+0.8 %**)	187.9 (**+1.4 %**)	207.2 (**+11.8 %**)	155.0 (**-16.4 %**)
Linear model	185.3	186.6 (**+0.7 %**)	195.0 (**+5.2 %**)	155.9 (**-15.9 %**)	110.4 (**-40.4 %**)
LUT-based model	185.3	169.9 (**-8.3 %**)	175.5 (**-5.3 %**)	150.2 (**-19.0 %**)	103.4 (**-44.2 %**)

the linear model does not consider such adverse effects of misalignment, the usage of this model occasionally results in assignments that actually have poor performance properties. Small negative time skews between the signal edges, with an adverse impact on the crosstalk, are treated as positive. Thus, for smaller standard deviations in the switching times, the TSV performance for a random assignment can be even better than for an assignment determined by means of the linear crosstalk model.

In contrast, the proposed optimization technique based on the LUT-based crosstalk model considers positive and negative aspects of temporal misalignment. Consequently, the usage of the LUT-based crosstalk model always results in the best TSV performance. Furthermore, in this case, the performance is always significantly improved, compared to the case of no signal misalignment.

Positive effects of misalignment dominate more and more with an increasing standard deviation in the switching times. Therefore, asymptotically, the proposed assignment technique employing the linear crosstalk model shows the same improvement as the technique employing the complex LUT-based model. For example, the usage of the linear model for the proposed technique results in a delay increase (*i.e.*, reduced performance), compared to the scenario of no misalignment, by 5.2 % for a standard deviation in the switching times of 25 ps. Contrary to this, the usage of the LUT-based crosstalk model results in a delay improvement by 5.3 %. However, for a standard deviation of about 180 ps, the usage of both models results in almost the same significant delay improvement of more than 40 %.

In conclusion, the usage of the more complex crosstalk model employing a LUT is indispensable for smaller misalignment quantities in order to ensure that the assignment results in a good TSV performance. For larger misalignment quantities, the linear model likely also results in a high TSV performance.

8.4.3. Comparison with 3D-CAC Techniques

As a final analysis, the effect of the proposed optimization technique on various metrics is compared to the values obtained for CAC techniques. Investigated is the transmission of 8 kB of random data over a TSV array of fixed size (4×4). Thereby, two P/G lines are considered in the array. Furthermore, typical TSV dimensions are investigated (*i.e.*, $r_{\mathrm{tsv}} = 2\,\mu\mathrm{m}$, $d_{\mathrm{min}} = 8\,\mu\mathrm{m}$, and $l_{\mathrm{tsv}} = 50\,\mu\mathrm{m}$).

The transmission of the data is analyzed under nine different scenarios. In scenario 1, the data is transmitted without any optimization. Scenario 2 represents the transmission of the data optimized with the $\omega_{\mathrm{m}}/\omega_{\mathrm{e}}$ CAC (based on an FTF encoding), presented in the previous Chapter 7 of this thesis. As shown in Chapter 7, the $\omega_{\mathrm{m}}/\omega_{\mathrm{e}}$-CAC technique leads to the highest performance improvement of all existing 3D CACs. Additionally, based on an FTF encoding, it is the only technique that can reduce the TSV power consumption noticeably. Nevertheless, previous 3D CACs, which neglect the edge effects, are also analyzed for the sake of completeness. Scenario 3 represents the data transmission for the best previous technique, the *6C* CAC.

Aligned signal edges are assumed for the first three scenarios since existing CAC techniques are only effective in this particular case. In the scenarios 4 and 5, the net-to-TSV assignment approach presented in this chapter is investigated for an intrinsic misalignment between the switching times. Thereby, the simple linear model is used in scenario 4 for the crosstalk estimation, and in scenario 5, the LUT-based model. For the intrinsic temporal misalignment, the switching times of the input signals (with respect to the rising clock edges at $k \cdot 1\,\mathrm{ns}$ with $k \in \mathbb{N}$) are normally distributed with a mean of 500 ps and a standard deviation, σ_{skew}, of 100 ps. Furthermore, the switching times are truncated to a minimum of 0 ps and a maximum of 700 ps.

Scenario 6 represents the case where rising-edge and falling-edge triggered flip-flops are used to generate an artificial signal misalignment, exploited by the technique proposed in this chapter. For artificial temporal misalignment, there is no need to differentiate whether the linear or the LUT-based crosstalk model is applied. All input signal pairs are either-way perfectly aligned or completely misaligned for the artificial misalignment, resulting in the same $\hat{A}_{i,j}$ values (only ones and zeros) for both crosstalk models. Thus, both models result in the same crosstalk estimation and consequently into the same assignment.

The scenarios 7 to 9 represent the scenarios 4 to 6, respectively, with an additional classical bus invert (CBI) coding [112]. Classical bus invert is a well-known LPC, which effectively reduces the power consumption of metal wires through a reduction in the switching activities by up to 25 %, despite inducing a relatively low bit overhead. The CBI is analyzed to validate that, after exploiting misalignment to improve the TSV performance, low-power techniques that aim to reduce the switching activities reduce the TSV power

consumption effectively. A CBI encoding requires only one invert bit as the overhead per codeword (here, equivalent to a bit overhead of 7.7 %). In contrast, the 3D CACs result in a bit overhead of 40 %, due to the coupling complexity in 3D. Thus, in the scenarios 1, 4, 5, and 6, 14 data bits are effectively transmitted per bit-pattern (no overhead); in the scenarios 7 to *9*, 13 bits; and in the scenarios 2 to 3, 10. This results in 9,363, 10,083, and 13,108 bit-patterns to transmit the 8 kB of data, respectively.

The analyzed power, performance, area, and noise metrics, relative to the values for the transmission of the raw data (scenario 1), are reported in Figure 8.8. In accordance with the results presented in the previous chapter, the power-consumption reduction is *ca.* 5 % for the $^{\omega_m}/_{\omega_e}$ CAC, while the *6C* CAC even slightly increases the TSV power consumption. As expected, the proposed overhead-free assignment technique leaves the power consumption more or less unaffected. However, a combination of the proposed method with the CBI coding leads to a reduction in the TSV power consumption by about 17 %, independent of the applied crosstalk model. Thus, CBI coding combined with the exploitation of misalignment outperforms the best CAC technique in terms of power-savings by a factor of 3×, despite a more than five times lower induced bit overhead.

Another investigated metric is the maximum switching noise that is induced on an interconnect due to coupling effects, reported in Figure 8.8c. Previous techniques lead to a noise-peak reduction of about 12 %. Switching on adjacent lines induces the noise, and the more these aggressors switch at the same time, the higher the noise peak. Consequently, the more adjacent lines switch misaligned, the more the noise peak will be reduced. Thus, the proposed assignment approach also leads to much better noise properties. Compared to the raw data, the noise peak can be reduced by up to 50 % for artificial misalignment. For the intrinsic misalignment, the application of the linear crosstalk model leads to lower noise values than the LUT-based model. The reason is that misalignment always has a positive effect on the height of the noise peak, which is captured by the linear, but not the LUT-based, crosstalk model. However, for all analyzed misalignment scenarios, the noise reduction of the proposed technique employing a LUT for the crosstalk estimation is still 2.8 × higher than the noise reduction of the CAC approaches.

The reductions in the maximum propagation delay are shown in Figure 8.8d. In accordance with the experimental results from the previous chapter, the maximum delay reduction of the best/proposed 3D-CAC technique is about 20 %. The proposed assignment method leads to significantly higher performance improvements for all investigated misalignment scenarios. As expected, the highest delay reduction of approximately 40 % (*i.e.*, 65 % performance increase) is obtained for artificial misalignment. Due to the relatively large standard deviation in the switching times for the intrinsic misalignment, the application

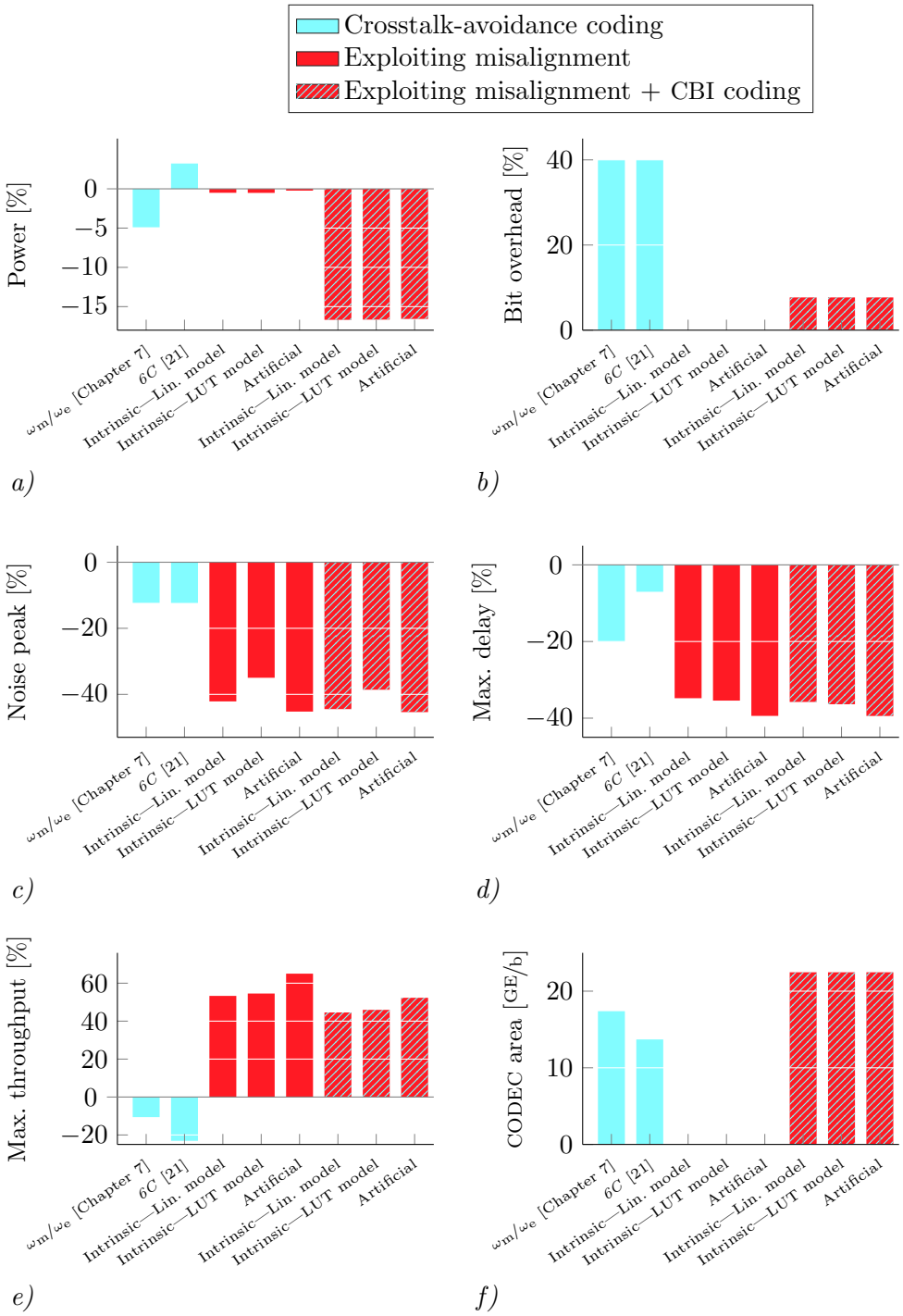


Fig. 8.8: Effect of the proposed technique exploiting misalignment and best-practice 3D CACs on the: *a)* power consumption per transmitted bit; *b)* bit overhead; *c)* noise peak; *d)* maximum propagation delay; *e)* maximum throughput; *f)* coder-decoder circuit (CODEC) area.

of the more complex LUT-based crosstalk model only leads to an increase in the performance improvement by about 1 pp, compared to the application of the linear model. Generally, both newly proposed assignment techniques lead to delay reductions by about 34 % to 35 % for the intrinsic misalignment—implying a performance increase of 53 % to 55 %.

An alternative performance metric—in the case that the number of TSVs is fixed as in this analysis—is the maximum possible data throughput reported in Figure 8.8e. This metric takes the propagation delay as well as the induced bit overhead into account. The bit overheads of existing 3D CACs are much higher than their delay reductions. Thus, when the number of TSVs in the array is fixed (to keep the TSV area occupation and yield unaffected), existing CACs even decrease the maximum data throughput.

In contrast, the technique presented in this chapter allows for a significant throughput increase. Since the proposed technique without CBI coding induces no overhead, it results in throughput increases that are the same as the previously reported performance gains (*i.e.*, 65 % for artificial misalignment, and 53 % to 55 % for intrinsic misalignment). Applying CBI encoding reduces the power consumption for the cost of a decrease in the throughput due to the induced bit overhead. Thus, the method, combined with the CBI encoding, leads to a slightly lower throughput increase. Exploiting artificial misalignment, in combination with a CBI encoding, increases the maximum throughput by 52 %. For the intrinsic misalignment and the CBI coding, the throughput can be increased by about 45 % for both applied crosstalk models, linear and LUT-based.

While the proposed mere assignment approach does not require any active circuit components, coding techniques require coder-decoder circuits (CODECs). Thus, the CODEC areas in GE, resulting from gate-level syntheses with *Synopsys* tools, are included in Figure 8.8f for the sake of completeness.

Summarized, the in-depth analysis shows that the proposed technique can significantly outperform CAC techniques in all performance metrics if a sufficient temporal signal misalignment is present. In combination with traditional LPCs, the proposed assignment technique furthermore effectively reduces the 3D-interconnect power consumption.

8.5. Conclusion

In this chapter, the effects of temporal misalignment between the switching times of the input signals of TSV-based interconnect structures have been investigated and subsequently exploited in order to improve the TSV performance without noticeable costs. Therefore, the proposed technique exploits the heterogeneity in the TSV capacitances, which arise from the edge effects, outlined in Chapter 4 of this thesis.

The proposed technique drastically improves the TSV performance and switching noise. Moreover, the approach mitigates the TSV coupling problem such that low-complex LPCs—aiming for a reduction in the switching activities of the transmitted bits—can effectively improve the TSV power consumption. For a 22-nm technology and modern TSVs, the CAC technique presented in Chapter 7 (which already outperforms all previous techniques drastically) shows to improve the power consumption, the performance, and the maximum switching noise of TSVs by a maximum of 5 %, 25 %, and 12 %, respectively. For a sufficient temporal misalignment, the presented approach can even improve the performance by up to 65 %, while it simultaneously decreases the noise and the power consumption by about 45 % and 17 %, respectively, despite showing a more than five times lower bit overhead than the best CAC technique. This underlines the massive potential of the proposed technique in scenarios where temporal misalignment between the signals is naturally present.

CHAPTER 9

Low-Power Technique for Yield-Enhanced 3D Interconnects

In the previous chapters, techniques that significantly improve the 3D-interconnect power consumption and performance at relatively low cost were presented. However, issues due to the relatively low TSV manufacturing yield have not been adequately addressed up to this point.

Previous techniques add redundant TSVs to address the yield issues. However, the redundant TSVs of such techniques are only required in the presence of manufacturing defects, while they are a waste of resources in the most likely case of a defect-free manufacturing. Furthermore, due to their implementation costs, existing redundancy techniques lead to an increase in the already critical TSV power consumption.

In this chapter, the first technique ever is presented, which addresses TSV yield issues, while it moreover significantly reduces the power consumption of arbitrary 3D interconnects. For this purpose, novel low-power encoder-decoder architectures are contributed, which are capable of improving the TSV manufacturing yield. The encoder-decoder structures increase the manufacturing yield by the same amount as existing redundancy techniques, while they additionally provide drastic power-consumption improvements. This duality of the coding gain is achieved by ensuring that the redundancy in the low-power codewords, compared to the unencoded data words, can be exploited for yield enhancement. Such a hybrid coding approach allows us to drastically reduce the implementation costs compared to a separate integration of a low-power coding (LPC) technique and a redundancy scheme.

To reduce the implementation costs even further, the technique is designed in a way that technological heterogeneity between the individual dies of a 3D IC is exploited. Technological heterogeneity implies that the logic costs differ for the individual dies of a 3D IC. Hence, the proposed LPC-based redundancy technique is constructed such that it is of minimum possible complexity in costly mixed-signal or radio frequency (RF) dies, in exchange for a small increase in the hardware overhead in dies implemented in an aggressively scaled technology node.

An evaluation of the proposed technique shows that it results in a reduction in the relative number of defective chips by a factor of 17 ×, while it is additionally capable of reducing the power consumption by over 40 %. Additionally, a case study for the commercially available heterogeneous SoC from [125] and real RF data streams shows that the proposed technique improves the logic-area requirements and power consumption, compared to the best previous redundancy technique, by 69 % and 33%, respectively, while still providing the same TSV yield enhancement. This underlines the substantial superiority of the proposed approach for real systems.

The remainder of this chapter is organized as follows. In Section 9.1, the related work is reviewed. The logical impacts of the common TSV defects on the transmitted signals are outlined in Section 9.2. Limitations of existing yield-enhancement approaches, as well as the fundamental idea of the approach proposed in this chapter, are outlined in Section 9.3. A formal problem description, regarding defect fixing (decodability) and minimum possible circuit complexity, is derived in Section 9.4. Afterward, in Section 9.5, the proposed technique is presented, which is subsequently evaluated in depth. The aforementioned case study is presented in Section 9.7. Finally, the chapter is concluded in Section 9.8. The technique presented in the remainder of this chapter is published in the journal series "IEEE Transactions on Very Large Scale Integration (VLSI) Systems" as a regular paper (see [A5]).

9.1. Related Work—Existing TSV Yield-Enhancement Techniques

The related work on TSV yield enhancement is presented in this section. In order to cope with the relatively low TSV manufacturing yield, a dedicated test method in combination with a TSV redundancy scheme is typically used [9]. Initially, the TSVs are grouped into sets of size m, and to each set, one or more redundant TSVs are added. A dedicated methodology tests the TSVs on manufacturing defects. The test results are subsequently interpreted to identify faulty TSVs. If the maximum number of faulty TSVs in a set does not exceed the number of redundant TSVs per set, the redundancy scheme is used to repair the TSV-based links. Otherwise, the die is not repairable and must be discarded.

In the following, existing TSV test approaches and redundancy schemes are reviewed. Thereby, the focus lies more on redundancy schemes, since this chapter contributes a novel redundancy approach, which is generally independent of the testing methodology. A wide range of research has been conducted on TSV testing methods which are classified into two major categories: Pre-bond testing, using dedicated test instruments (*e.g.*, [25, 27–31]), and post-bond testing, which operates similar to traditional logic testing known from 2D ICs (*e.g.*, [32–35]).

The main advantage of the second approach is that it also allows for detecting bonding failures. Furthermore, it potentially reduces the test time as it enables to combine logic and TSV testing. However, post-bond TSV testing has the drawback that the testing is executed after the manufacturing of the complete stack is completed. Thus, whenever a single die has an irreparable TSV defect, the full stack has to be discarded. In contrast, with pre-bond testing, only the single faulty die needs to be replaced. Thus, post-bond testing can significantly increase manufacturing costs. Consequently, a combination of pre-bond and post-bond testing is often applied [36].

Also existing TSV redundancy schemes are classified into two main categories, illustrated in Figure 9.1: Multiplexer based and coding based. The first approach uses multiplexers to reroute the signals around faulty TSVs. Two representative concepts exist for this approach: *Signal-Reroute* [37] and *Signal-Shift* [9, 37]. *Signal-Reroute*, illustrated in Figure 9.2a, assigns every signal bit to a TSV and, if a signal TSV is faulty, only the associated net is rerouted over a redundant TSV. In the second concept, illustrated in Figure 9.2b, beginning at the position of the fault, all signal nets are shifted toward the redundant TSV. Signal shifting increases the number of rerouted signals, but it decreases the maximum delay of the redundancy scheme [38].

Configuration bit-vectors at the transmitter side, $\vec{c}_{\mathrm{tx}}$, and the receiver side, $\vec{c}_{\mathrm{rx}}$, of a TSV group control the multiplexers. These configuration bits are

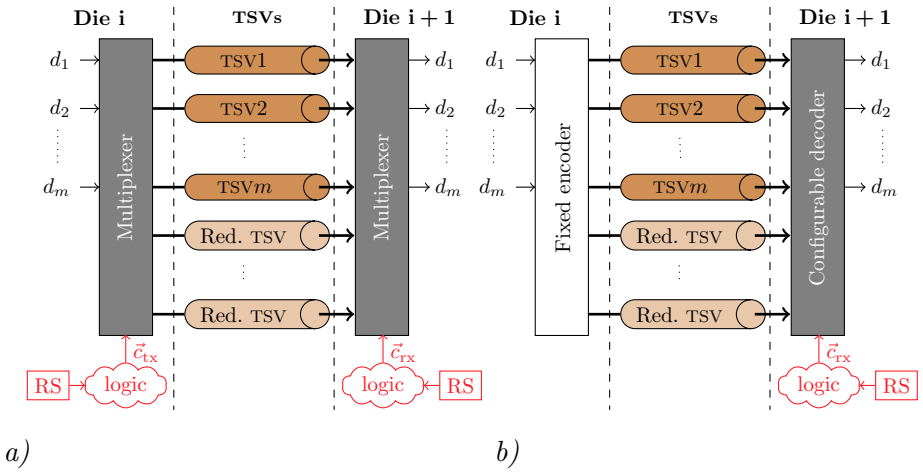


Fig. 9.1: Types of TSV redundancy schemes: *a)* Multiplexer based; *b)* Coding based.

determined through the repair signature (RS) for the set, defining the bypassed TSV. To repair one fault, the compressed repair signature of a set containing n lines is $\lceil \log_2(n) \rceil$ bits wide. Since the configuration is fixed after identifying the faulty TSVs, redundancy schemes generally require one-time-programmable non-volatile memory (NVM) cells to store the RSs. In standard technologies, E-fuses are typically used to implement one-time-programmable NVM cells [126]. A high-voltage circuit is required to program such NVM cells, which results in high area costs. Furthermore, a high-voltage power grid that spans over all dies of a 3D IC drastically increases the layout/design complexity [126].

Therefore, the advantages of having a single/global NVM macro in one die, from which the repair signatures are shifted/loaded serially into distributed repair registers (RRs) during start-up, are extensively discussed in [127]. In summary, besides overcoming the need for NVM cells in each die, using a global macro also reduces the required NVM space by 50 % for multiplexer-based redundancy schemes, as the RSs at both sides of a TSV set are equal. If possible, choosing a location where an NVM macro is either-way required, further decreases the design complexity. Also, a controller for the global NVM macro, which serially shifts the RSs into the RRs, while minimizing the number of inter-die connections (*i.e.*, TSVs), is proposed in [127]. The major drawback of using a single global macro is that all RRs must be connected to one long shift-register chain, spanning over all dies of the system.

Traditional fixed error-correcting codes (*e.g.*, *Hamming* [128]) require large amounts of redundant TSVs, making them impractical to address TSV yield issues due to the enormous TSV dimensions and parasitics. Hence, a coding-based approach, illustrated in Figure 9.1b, has been proposed, which requires

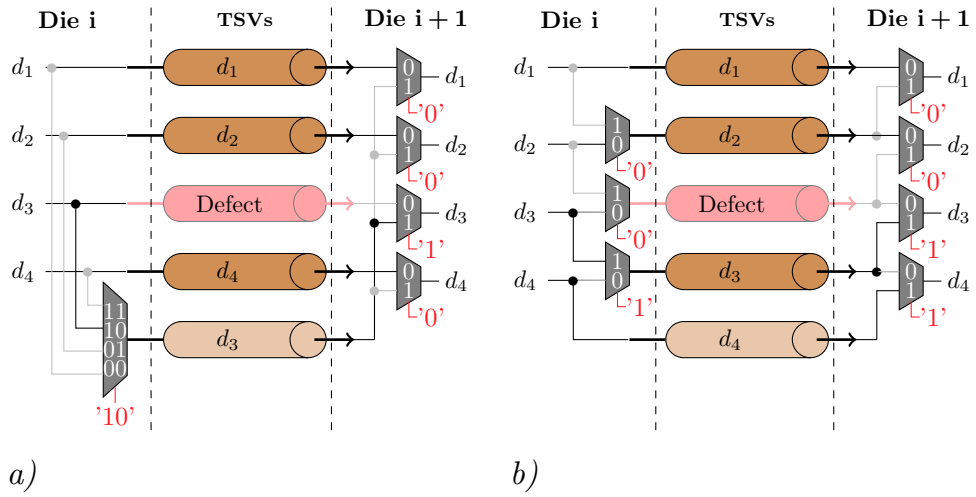


Fig. 9.2: Variants of multiplexer-based TSV redundancy schemes: *a) Signal-Reroute* [37]; *b) Signal-Shift* [9, 37].

only a reconfiguration of the decoder circuit, while it still minimizes the number of required redundant TSVs [38]. Thus, using a single redundant line, the approach can recover the information words in the case of any single static defect by modifying the decoding. Since the redundancy technique proposed in this chapter is also based on coding, in Section 9.4 and 9.5, the concept of coding-based redundancy schemes is explained in more detail. The main advantage of the existing coding-based approach is that configuration is only required at the receiver side of a link. This reduces the number of required RSs in the individual dies by 50 %. Thereby, the length of the RR chain for the global-NVM-macro approach is halved, which reduces the area overhead and the wiring complexity. For distributed NVM cells, the approach decreases the number of required NVM cells by 50 %, which is a drastic improvement.

9.2. Preliminaries—Logical Impact of TSV Faults

The common TSV manufacturing defects—outlined in Subsection 2.3.1 on Page 21—can be classified as:

1. voids;
2. delamination at the interface;
3. material impurities;
4. TSV-to-substrate shorts.

In this subsection, the logical effects of the four different TSV defect types on the transmitted digital signals are outlined.

The first three defect types result in an increased TSV resistance, which increases the signal propagation delay (see Equation (3.42) on Page 50) [25, 26].

Hence, the defects entail logical delay faults. A TSV-to-substrate short results in a resistive electrical connection between the TSV conductor and the substrate-bias/bulk contact through the conductive substrate. Hence, a TSV-to-substrate short draws the potential of the TSV toward the substrate-bias potential [25]. Thus, for typical grounded p-doped substrates, the defect type entails logical stuck-at-0 (SA0) faults, while for an n-doped substrate, biased at the power-supply potential, the defect type entails stuck-at-1 (SA1) faults. Furthermore, substrate shorts result in increased leakage currents when the TSV-conductor potential differs from the substrate potential (*e.g.*, when, for a p-doped grounded substrate, a logical 1 is on the TSV implying a conductor potential of V_{dd}).

In summary, a redundancy technique has to cope with stuck-at (SA) and delay errors. Typically, only one SA type (SA1 or SA0) occurs, depending on the substrate doping. Nevertheless, for increased robustness, a scheme that can repair both stuck-at fault types is also presented in this chapter.

9.3. Fundamental Idea

In this section, first, drawbacks of existing TSV redundancy schemes are outlined. Based on this, the fundamental idea of the proposed technique is outlined. Three major drawbacks of existing TSV reliability schemes can be identified.

First, current redundancy techniques do not consider the implications of heterogeneous integration. Digital logic elements are more expensive (in terms of area, power consumption, and delay) when integrated into a less aggressively scaled technology. This creates a demand for heterogeneous redundancy schemes. In dies with larger feature sizes, a redundancy scheme should result in a possibly low hardware overhead.

The second drawback is that recent techniques require repair signatures (RSs) in every die, which increases the manufacturing complexity. For example, consider a system consisting of three different dies, illustrated in Figure 9.3a. In the example, the previous coding-based redundancy approach is already applied to reduce the total number of required RSs. Thus, an RS is only required at each decoder. However, since data flows in both directions, RSs are still required in every die. Thus, the repair register (RR) chain—through which the RSs are distributed at run time—still spans over all dies of the system when a global NVM macro is used. Alternatively, with distributed NVM cells, expensive NVM programming circuits are required in all dies, resulting in high costs.

To overcome this bottleneck, a redundancy technique where the reconfigurable circuit is always located in the more aggressively scaled technology, at the interface of two heterogeneous dies, is proposed in this chapter. Furthermore, this reduces the RR hardware costs for the global NVM macro.

Thus, for data flowing toward a die fabricated in a bigger technology node, a scheme with a configurable encoding and a fixed decoding with minimal

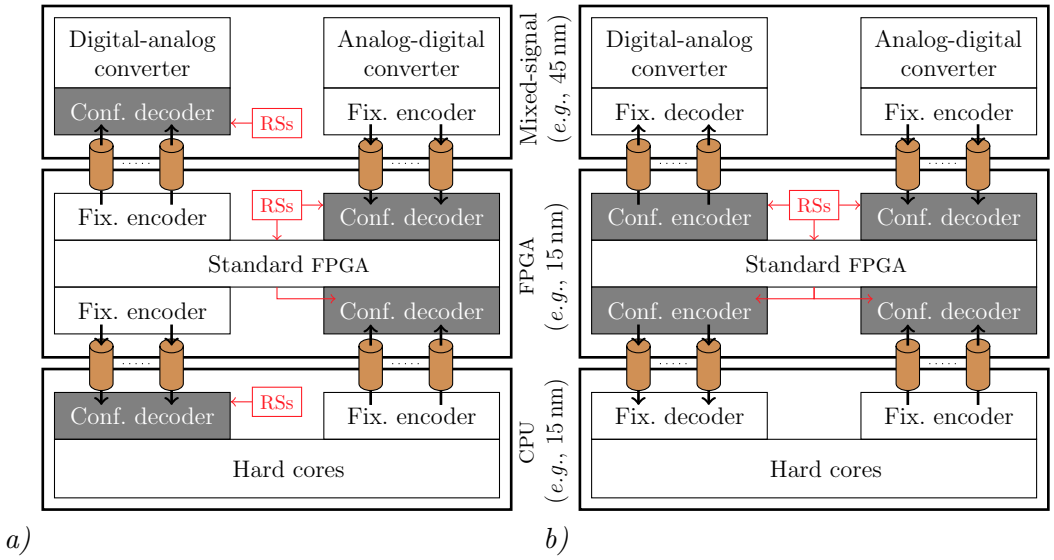


Fig. 9.3: Heterogeneous 3D IC with: *a)* the previous coding-based TSV redundancy scheme [38]; *b)* the proposed TSV redundancy scheme.

circuit complexity is proposed. If data flows toward a die fabricated in a more aggressively scaled technology, the decoding should be configurable, while the encoding should be of minimal complexity.

One exception is made for systems using FPGA blocks to integrate semi-custom digital blocks after manufacturing. Redundancy schemes are designed to cope with manufacturing defects. Thus, a dynamic reconfiguration of the encoding or decoding at run-time is generally not required. Thus, if one of the two dies contains an FPGA, the placement and routing of the FPGA soft-components, executed after TSV manufacturing and testing, is adaptable on the defect characteristics of the links. This allows eliminating high costs due to NVM and RR cells, storing RSs on-chip for the proposed as well as all existing redundancy schemes as outlined in detail in the later Section 9.5. Thus, at the interface of a die containing an FPGA (referred to in the following as an FPGA die) and another die, it is proposed to place all fixed encoders and decoders in the non-FPGA side/die.

For the example illustrated in Figure 9.3b, the proposed technique requires no RSs in the mixed-signal and the CPU die anymore. Furthermore, the need for NVM and RR cells vanishes completely. This is an extreme scenario, but it represents existing products, such as the ones presented in [51, 125]. For systems without FPGA dies and a large number of layers, the number of dies that require RSs is still reduced by 50 %, compared to all previous approaches.

Here, the proposed technique prevents the necessity of an NVM programming circuit for distributed NVM cells in every second die. For the global NVM macro, the RR chain has to cover 50 % fewer dies.

The last drawback of existing TSV redundancy schemes is that they are a waste of resources in the most likely event of a defect-free TSV manufacturing. Thus, in this chapter, a redundancy technique is presented, which optimizes the TSV power consumption—especially in scenarios where redundancy is not required for yield enhancement. Therefore, the encoder-decoder pairs are moved from the boundaries of the TSV links to the boundaries of the full 3D links, including the metal wires. This implies another requirement for the fixed encoding and the fixed decoding. Besides being of minimal complexity, the fixed encoder and decoder circuits also have to be usable as a low-power encoder and decoder for TSVs and metal wires.

Generally, the requirements for a coding technique to serve as an efficient LPC depends on various scenarios. At first, the coupling phenomenon changes drastically for TSV arrays and metal wire due to the different capacitance structures outlined in Chapter 4. Also temporal misalignment affects the coupling behavior in a complex way, as shown in Chapter 8. This makes it hard to identify what is a suitable low-power code for long 3D interconnects made up of several metal-wire and TSV segments when the coupling is considered explicitly. However, in the previous chapter, it was shown that reducing the self-switching activities is either-way the more efficient coding approach if nets switch (even only slightly) temporally misaligned.

The previous considerations, combined with the fact that considering coupling results in a much larger CODEC complexity, indicate that the proposed coding technique should focus on reducing the power consumption of the 3D interconnects by minimizing the switching activities of the transmitted bits. In addition to the switching activities, an optimization in the bit probabilities is desirable in order to exploit the TSV MOS effect. Exploiting the MOS effect demands increased logical 1-bit probabilities for typical p-doped substrates and increased 0-bit for n-doped substrates, as shown in Chapter 4. However, the bit probabilities should only be fine-tuned if this can be achieved at a reasonable increase in the coding complexity for mainly two reasons. First, only the TSV but not the metal-wire power consumption can be optimized through the MOS effect, and second, the magnitude of the TSV MOS effect varies strongly for different technologies, as shown in Chapter 4.

In summary, the fundamental idea is to design two coding architectures, which exploit heterogeneity to reduce implementation costs by having either a minimal fixed encoder or decoder (one each), allow for TSV yield enhancement, and optimize the 3D-interconnect power consumption by reducing the switching activities. Moreover, the possibility to extend the techniques in a way that the bit probabilities can be additionally optimized should be investigated.

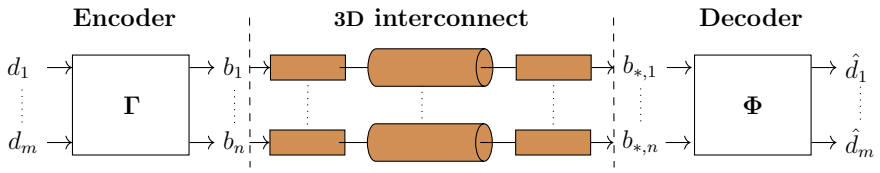


Fig. 9.4: Arrangement of the redundancy encoder-decoder pair enclosing the full 3D-interconnect structure.

9.4. Formal Problem Description

The formal problem description considering decodability (*i.e.*, defect fixing) and circuit complexity is presented in the following. Therefore, a link composed of n lines over which m bits have to be transmitted is considered. Thus, the number of redundant lines, r, is $n - m$.

9.4.1. Decodability

To guarantee a fault-free transmission in the case of a defect in a TSV group, a data encoding is integrated, as illustrated in Figure 9.4. A fault is corrected by only modifying the encoding or the decoding while the counterpart is fixed.

In this subsection, the analysis is restricted to linear block codes in the binary Galois field, $\mathbb{F}_2$, as these codes allow for a systematic analysis and result in low hardware complexities. Thus, the encoding of the m-bit information word, $\vec{d} = [d_1, \ldots, d_m] \in \mathbb{B}^{1\times m}$, to the n-bit codeword, $\vec{b} = [b_1, \ldots, b_n] \in \mathbb{B}^{1\times n}$, is expressed as a matrix multiplication in the Galois field 2:

$$\vec{b} = \vec{d} \cdot \mathbf{\Gamma}, \tag{9.1}$$

where $\mathbf{\Gamma} \in \mathbb{B}^{m\times n}$ is the encoding matrix. The effect of the physical link on the received word, $\vec{b}_*$, is expressed by function f_L:

$$\vec{b}_* = f_\mathrm{L}(\vec{b}) = f_\mathrm{L}(\vec{d} \cdot \mathbf{\Gamma}). \tag{9.2}$$

In the defect-free case, the received word, $\vec{b}_*$, is always equal to transmitted one, $\vec{b}$. However, the received and the transmitted words differ for a defect in the link.

A linear decoding, expressed by a multiplication with $\mathbf{\Phi} \in \mathbb{B}^{n\times m}$, has to recover the information word, $\vec{d}$, out of $\vec{b}_*$. Hence, the decoding has to satisfy

$$\vec{d} \stackrel{!}{=} \hat{\vec{d}} = f_\mathrm{L}(\vec{b}) \cdot \mathbf{\Phi} = f_\mathrm{L}(\vec{d} \cdot \mathbf{\Gamma}) \cdot \mathbf{\Phi}. \tag{9.3}$$

The set of considered errors is symbolized as $\mathbb{S}_\mathrm{E}$. Consequently, a fixed encoding, expressed by $\mathbf{\Gamma}_\mathrm{fix}$, in combination with a configurable decoding, expressed by

$\mathbf{\Phi}_{\text{conf}}$, must satisfy

$$\forall e \in \mathbb{S}_{\text{E}} \; \exists \; \mathbf{\Phi}_{\text{conf},e} : f_{\text{L},e}(\vec{d} \cdot \mathbf{\Gamma}_{\text{fix}}) \cdot \mathbf{\Phi}_{\text{conf},e} = \vec{d}. \tag{9.4}$$

For a fixed decoding, expressed by $\mathbf{\Phi}_{\text{fix}}$, the constraint is

$$\forall e \in \mathbb{S}_{\text{E}} \; \exists \; \mathbf{\Gamma}_{\text{conf},e} : f_{\text{L},e}(\vec{d} \cdot \mathbf{\Gamma}_{\text{conf},e}) \cdot \mathbf{\Phi}_{\text{fix}} = \vec{d}. \tag{9.5}$$

In the following, it is outlined how the encoder-decoder pairs cope with the two logical fault types, stuck-at (SA) and delay. First, faults are masked in a way that they do not affect the decoded information, $\hat{d}_i$. If the encoding ($\mathbf{\Gamma}$) is fixed, a fault on line i is masked by configuring the decoding in a way that all elements in the i^{th} column of $\mathbf{\Phi}$ are equal to 0. Thereby, the decoded output becomes independent of the value on the defective line.

This kind of masking is not possible if the decoding is fixed. Here, in case of a fault, the encoding is configured such that a constant value is assigned to a faulty line, irrespective of the transmitted information word. This value is logical 0 for an SA0, and logical 1 for an SA1. Thus, a faulty line is set to constant logical 0 and 1 for n-doped and p-doped substrates, respectively.[1] If, for example, a constant logical 0 is assigned to the input of a line that might have either an SA0 or a delay fault, also the output at the far end is constant logical 0, independent of the presence of the fault and its type (*i.e.*, SA0 or delay). Thus, the transmitted and the received word are always equal, despite the defect (*i.e.*, $\vec{b}_* = \vec{b}$). Assigning constant/stable values to faulty lines furthermore enables us to treat delay faults in the same way as SA faults. Hence, the remainder of this discussion can be limited to SA faults.

In the following, SA0 faults are considered as the substrate is typically p doped. However, an analysis of SA1 faults results in the same constraints. For a single SA0 on line i, the effect of the link is expressed by a multiplication with a matrix similar to identity, but with a 0 on the i^{th} diagonal entry, represented by $\mathbf{E}_{\text{sa}_i}$. Thus, for an SA0 fault, Equation (9.3) is expressed as

$$\vec{d} \stackrel{!}{=} f_{\text{L}}(\vec{b}) \cdot \mathbf{\Phi} = \vec{d} \cdot \mathbf{\Gamma} \cdot \mathbf{E}_{\text{sa}_i} \cdot \mathbf{\Phi}. \tag{9.6}$$

Hence, to cope with every single SA fault, a fixed encoding has to satisfy

$$\forall i \in \{1, 2, \ldots, n\} \; \exists \; \mathbf{\Phi}_{\text{conf,sa}_i} : \mathbf{\Gamma}_{\text{fix}} \cdot \mathbf{E}_{\text{sa}_i} \cdot \mathbf{\Phi}_{\text{conf,sa}_i} = \mathbf{I}_m, \tag{9.7}$$

where $\mathbf{I}_m$ is the $m \times m$ identity matrix. The resulting equation systems can

[1] This forbids later inversions for the nets, as proposed in Chapter 6, to exploit the MOS effect since this would destroy the error masking. Thus, for nets that belong to the output of a configurable encoder, the technique proposed in Chapter 6 can only be applied without inversions (resulting in a mere reordering of the net-to-TSV assignment).

be solved if the rank of the result of $\mathbf{\Gamma}_{\text{fix}} \cdot \mathbf{E}_{\text{sa}_i}$ is at least m (*i.e.*, right inverse exists). The multiplication with $\mathbf{E}_{\text{sa}_i}$ simply sets all entries of the i^{th} column to 0. Since all entries of the i^{th} column of the decoding matrix $\mathbf{\Phi}_{\text{conf}}$ are also 0 to mask the error, the equation systems from Equation (9.7) have unique solutions if all subsets of m columns in $\mathbf{\Gamma}_{\text{fix}}$ are linearly independent. Mathematically this is expressed as

$$\text{spark}(\mathbf{\Gamma}_{\text{fix}}) > m. \tag{9.8}$$

Analogously, one can show that decodability for up to s arbitrary errors, using r redundant lines, requires that in all subsets of $m + r - s$ columns of the encoding matrix, $\mathbf{\Gamma}_{\text{fix}}$, at least m columns are linearly independent.

For a fixed decoding, $\mathbf{\Phi}_{\text{fix}}$, the equation systems are as follows:

$$\forall i \in \{1, 2, \ldots, n\} \; \exists \; \mathbf{\Gamma}_{\text{conf,sa}_i} : \mathbf{\Gamma}_{\text{conf,sa}_i} \cdot \mathbf{E}_{\text{sa}_i} \cdot \mathbf{\Phi}_{\text{fix}} = \mathbf{I}_m. \tag{9.9}$$

In this equation, the multiplication with $\mathbf{E}_{\text{sa}_i}$ sets all entries of the i^{th} row to 0. Thus, the left inverse of $\mathbf{E}_{\text{sa}_i} \cdot \mathbf{\Phi}_{\text{fix}}$ always exists, if all subsets of m rows of $\mathbf{\Phi}_{\text{fix}}$ are linearly independent, mathematically expressed as

$$\text{spark}(\mathbf{\Phi}_{\text{fix}}^T) > m. \tag{9.10}$$

Consequently, the general constraint to be able to handle arbitrary combinations of m errors with a fixed decoder and r redundant lines is that, in all subsets of $m + r - s$ rows of $\mathbf{\Phi}_{\text{fix}}$, at least m rows are linearly independent.

9.4.2. Circuit Complexity

In this subsection, the lower bounds for the circuit complexities for the fixed encoding and the fixed decoding are derived. To take TSV costs into account, only redundancy schemes where the number of simultaneously repairable errors is equal to the number of redundant TSVs (*i.e.*, $s = r$) are considered.

Again, fixed encoding and decoding are expressed by Galois-field-2 multiplications with $\mathbf{\Gamma}_{\text{fix}}$ and $\mathbf{\Phi}_{\text{fix}}$, respectively. The number of ones in the i^{th} column of $\mathbf{\Gamma}_{\text{fix}}$ and $\mathbf{\Phi}_{\text{fix}}$ defines how many input bits are XORed by the encoding and decoding to generate the i^{th} output bit, respectively. Thus, the maximum number of ones in a single column of $\mathbf{\Gamma}_{\text{fix}}$ and $\mathbf{\Phi}_{\text{fix}}$ defines the maximum delay of the encoder and the decoder circuit, respectively.

Assuming that only two-input gates are available, representing the worst case, the delay of an XOR operation on i inputs is equal to $\lceil \log_2(i) \rceil \cdot T_{\text{xor}}$, where T_{xor} is the delay of a single XOR gate. Thus, the maximum delay of the fixed encoder is

$$\hat{T}_{\mathbf{\Gamma}_{\text{fix}}} = \left\lceil \log_2 \left(\max_j \left(\sum_{i=1}^{m} \mathbf{\Gamma}_{\text{fix},i,j} \right) \right) \right\rceil \cdot T_{\text{xor}}, \tag{9.11}$$

while the maximum delay of the fixed decoder is

$$\hat{T}_{\mathbf{\Phi}_{\text{fix}}} = \left\lceil \log_2 \left(\max_j \left(\sum_{i=1}^{n} \mathbf{\Phi}_{\text{fix},i,j} \right) \right) \right\rceil \cdot T_{\text{xor}}. \tag{9.12}$$

The area for an XOR operation on i inputs, employing only two-input gates, is equal to $(i-1)A_{\text{xor}}$, where A_{xor} is the XOR-gate area. Thus, the overall area of the fixed encoder circuit is expressed as follows:

$$A_{\mathbf{\Gamma}_{\text{fix}}} = \left(-n + \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbf{\Gamma}_{\text{fix},i,j} \right) \cdot A_{\text{xor}}. \tag{9.13}$$

Consequently, the overall area of the fixed decoder circuit is

$$A_{\mathbf{\Phi}_{\text{fix}}} = \left(-m + \sum_{i=1}^{n} \sum_{j=1}^{m} \mathbf{\Phi}_{\text{fix},i,j} \right) \cdot A_{\text{xor}}. \tag{9.14}$$

First, the lower bound for the circuit complexity of a fixed encoder is derived. Since all m input bits have to affect at least $s+1$ bits of the transmitted $n = m + s$ codeword bits to cope with every s-bit error, the minimum number of ones in the encoder matrix, $\mathbf{\Gamma}_{\text{fix}}$, is $m \cdot (s+1)$. Thus, the optimal area is

$$A_{\text{opt},\mathbf{\Gamma}_{\text{fix}}} = (m \cdot (s+1) - n) \cdot A_{\text{xor}} = s \cdot (m-1) \cdot A_{\text{xor}}. \tag{9.15}$$

The optimal delay is obtained if the $m \cdot (s+1)$ ones are equally distributed on the $n = m + s$ columns of $\mathbf{\Gamma}_{\text{fix}}$. In this scenario, the maximum number of ones in a column is $\lceil {}^{m\cdot(s+1)}/_{m+s} \rceil$. Thus, the optimal delay of a fixed encoder is

$$\hat{T}_{\text{opt},\mathbf{\Gamma}_{\text{fix}}} = \left\lceil \log_2 \left(\frac{m \cdot (s+1)}{m+s} \right) \right\rceil \cdot T_{\text{xor}}. \tag{9.16}$$

For a fixed decoding, $\mathbf{\Phi}_{\text{fix}}$, each of the m output bits must be a Boolean expression of at least $s+1$ input bits to cope with up to s entropy-less/stable input bits (*i.e.*, masked defect lines). Thus, the optimal delay and area are:

$$\hat{T}_{\text{opt},\mathbf{\Phi}_{\text{fix}}} = \lceil \log_2(s+1) \rceil \cdot T_{\text{xor}}; \tag{9.17}$$

$$A_{\text{opt},\mathbf{\Phi}_{\text{fix}}} = \big(m \cdot (s+1) - m\big) \cdot T_{\text{xor}} = m \cdot s \cdot T_{\text{xor}}. \tag{9.18}$$

As a sanity check, an extensive search of all matrices in the Galois field 2 that full-fill the decodability constraints is performed for m equal to 4 and 5 and s/r equal to 1 and 2. The check validates the correctness of the derived formulas for the minimum possible circuit complexities.

9.5. TSV Redundancy Schemes

The proposed TSV redundancy schemes are outlined in this section. In Subsection 9.5.1, a fixed decoding and the associated configurable encoding are presented. Subsection 9.5.2 covers the required counterpart: A fixed minimal low-power encoding with a configurable decoding. The proposed schemes are designed to repair a single error per group (*i.e.*, $r = s = 1$). A minimal/optimal circuit complexity for one redundant line implies a maximum delay of only T_{xor} for both circuits, fixed encoder and fixed decoder. The optimal circuit area of the fixed encoder and the fixed decoder are $(m-1) \cdot A_{\text{xor}}$ and $m \cdot A_{\text{xor}}$, respectively.

9.5.1. Fixed-Decoding Scheme

The proposed TSV redundancy scheme with a fixed decoding is illustrated in Figure 9.5. In the right part of Figure 9.5a, the fixed decoder is illustrated, which simply is a controllable inverter where the MSB of the received word controls if the decoded m-bit output is equal to the first m bits of the received word, $\vec{b}_*$, or equal to its bit-wise negation. The Boolean equation describing the decoding is

$$\hat{d}_i = b_{*i} \oplus b_{*m+1} \quad \text{for } i \in \{1, 2, \ldots, m\}, \tag{9.19}$$

where $\oplus$ is the Boolean XOR operator. For m equal to 4, the decoding matrix is

$$\mathbf{\Phi}_{\text{fix}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \tag{9.20}$$

Defect Fixing

In the following, it is outlined how the minimal invert decoding and a configurable encoding handle a defect on an arbitrary line with the index j. Thus, the received value b_{*j} is faulty, while all other received values are correct (*i.e.*, $b_{*i} = b_i$ for $i \in \{1, \ldots, m{+}1\} \backslash \{j\}$). A correct transmission of $\vec{d}$ despite the defect requires a transmission of information bit d_j over the redundant line (*i.e.*, $b_{m+1} = d_j$). Over all other lines, the information bits XORed with the value on the redundant line are transmitted:

$$b_i = d_i \oplus b_{m+1} = d_i \oplus d_j \text{ for } i \in \{1, 2, \ldots, m\}. \tag{9.21}$$

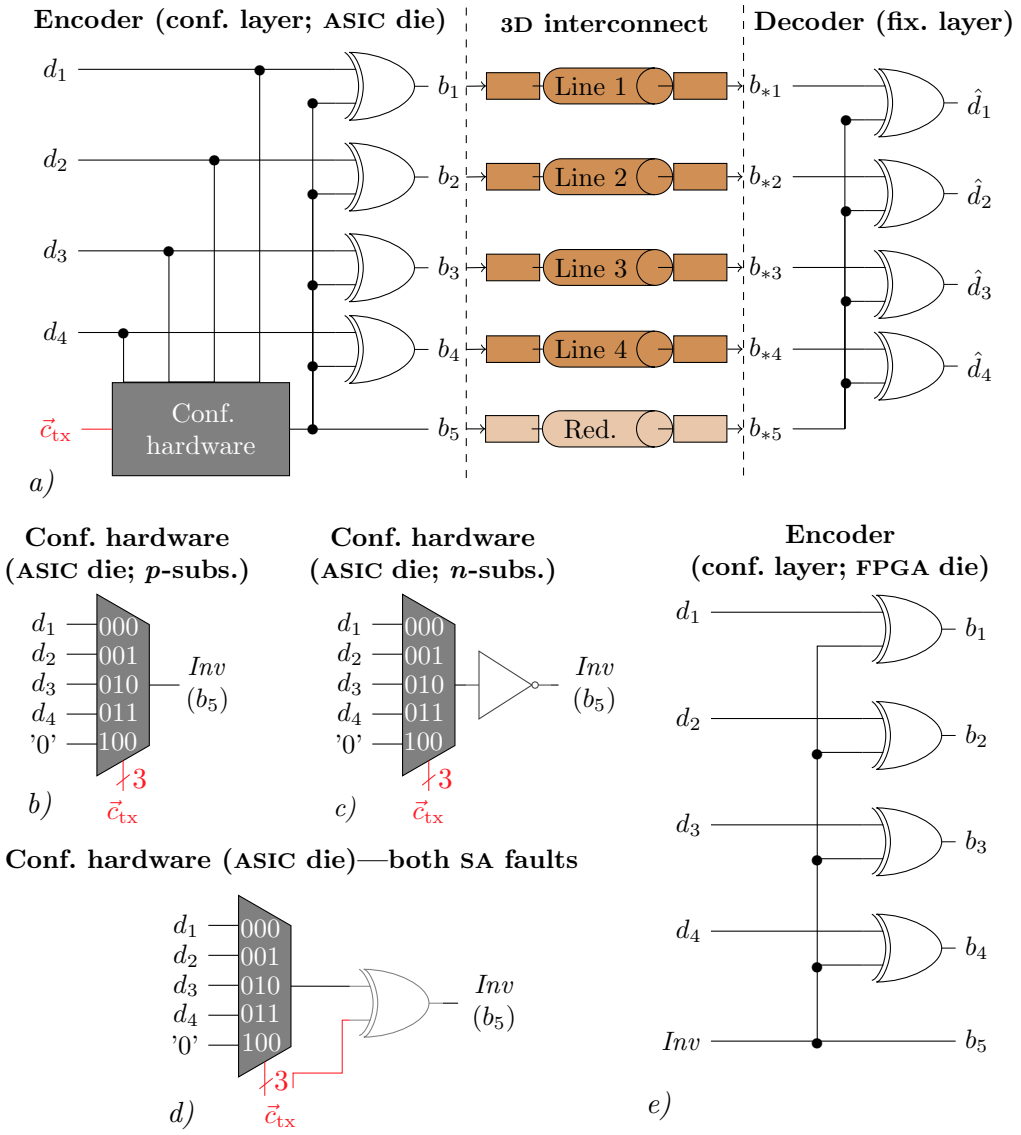


Fig. 9.5: Proposed redundancy scheme with a fixed decoding for a 4-bit input: *a)* General structure; *b)* Configurable-hardware structure for p-doped substrates and an implementation into an ASIC die; *c)* Configurable hardware structure for n-substrates and an implementation into an ASIC die; *d)* Configurable hardware structure to repair both stuck-at fault types and an implementation into an ASIC die; *e)* Encoder for an implementation into an FPGA die.

Therefore, the encoder also contains a controllable inverter. As required, the encoding masks the delay or SA0 error on line j:

$$b_j = d_j \oplus b_{m+1} = d_j \oplus d_j = 0. \tag{9.22}$$

Also, the correct information is decoded despite the fault:

$$\hat{d}_i = b_{*i} \oplus b_{*m+1} = \begin{cases} (d_i \oplus d_j) \oplus d_j & = d_i \quad \text{for } i \neq j \\ \underbrace{0}_{b_{*j} \text{ has SA0}} \oplus d_j & = d_j \quad \text{for } i = j. \end{cases} \tag{9.23}$$

A constant 0 is assigned to the redundant invert line for an SA0 fault on the $(m+1)^{\text{th}}$ interconnect. Afterward, the first m bits of the transmitted and the received word are equal to the original information word, $\vec{d}$, since the inverter is turned off permanently.

If the substrate is n-doped—entailing SA1 instead of SA0 faults—the codewords only have to be logically negated. The negation does not affect the decoding since the invert line is also negated. To obtain the negated codewords at the encoder output, the value for the invert line of the encoder simply has to be negated (*i.e.*, $\neg d_j$ instead of d_j is transmitted over the redundant line for a defect on line j). Thus, for an SA1 or delay fault, the Boolean equation describing the encoding is

$$b_i = d_i \oplus b_{m+1} = d_i \oplus \neg d_j \text{ for } i \in \{1, \ldots, m\}. \tag{9.24}$$

Again, the requirement of masking the fault is satisfied:

$$b_j = d_j \oplus b_{m+1} = d_j \oplus \neg d_j = 1. \tag{9.25}$$

The decoding is mathematically expressed as

$$\hat{d}_i = b_{*i} \oplus b_{*m+1} = \begin{cases} (d_i \oplus \neg d_j) \oplus \neg d_j & = d_i \quad \text{for } i \neq j \\ \underbrace{1}_{b_{*j} \text{ has SA1}} \oplus \neg d_j & = d_j \quad \text{for } i = j. \end{cases} \tag{9.26}$$

Here, a constant logical 1 is assigned to the invert line if the redundant line is faulty. The encoding for an exemplary SA0 on line 3 and a group size of four is described by

$$\vec{b} = \vec{d} \cdot \mathbf{\Gamma}_{\text{conf,SA}_3} = \vec{d} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{9.27}$$

The required encoder hardware, for an implementation into an ASIC die containing hard components, is shown in Figure 9.5a for m equal to 4. Besides the controllable inverter, made up of m XOR gates, a configurable block is required. This configurable block outputs the invert signal, equal to the signal transmitted over the redundant line. For TSVs traversing a p-doped substrate, the minimum configurable block simply consists of a multiplexer shown in Figure 9.5b. Signal $\vec{c}_{\text{tx}}$, equal to the RS stored in an RR, controls the multiplexer. If a data line i is defect (*i.e.*, integer($\vec{c}_{\text{tx}}$) $< m$), the multiplexer connects-through d_i. Otherwise, the multiplexer connects-through a constant logical 0. An inversion of the multiplexer output is added if the TSVs traverse an n-doped substrate, as shown in Figure 9.5c. Such an inversion does not affect the hardware complexity.

In contrast, the hardware complexity increases by one XOR gate and one configuration bit if the system should be capable of repairing SA0 as well as SA1 faults, as shown in Figure 9.5d. Here, in case of a stuck-at fault, the RS does not only have to indicate the location of the defective line but also the defect type (*i.e.*, SA1 or SA0), using an additional bit. For a delay instead of an SA fault on a line, both configurations—the one that treats an SA0 as well as the one to treats an SA1 on the according line—ensure a correct functionality of the link. Thus, here one of the two configurations can be chosen arbitrarily.

The encoder hardware for an implementation into an FPGA die, illustrated in Figure 9.5e, has a lower complexity. Here, just the controllable inverter is physically required. This makes the encoder hardware requirements equal to the decoder hardware requirements (*i.e.*, minimum delay). For FPGA dies, the RSs should be used offline during the placement and routing of the FPGA soft-components, performed after TSV testing. Therefore, after manufacturing and testing, the TSV-defect information is stored not on-chip, but in a file readable for the EDA tool of the FPGA. During the place and route stage of the FPGA soft-components, the tool reads the RS of the link and then routes the information bits that have to be transmitted over the TSVs to the encoder inputs such that a correct transmission is ensured. If the i^{th} ($i \leq m$) line of the link has an SA0 fault, the tool will route d_i to the invert input, *Inv*, to ensure a correct transmission. $\neg d_i$ is assigned to the invert input for an SA1 fault. The invert input is connected to logical 0 and logical 1 if the redundant line has an SA0 and SA1, respectively. Again, delay faults can be treated either as an SA0 or as an SA1 fault.

In case of no defect, an arbitrary value can be assigned to the redundant invert line for all encoder implementations. This freedom is exploited to effectively reduce the 3D-interconnect power consumption through the proposed redundancy technique, which operates in a low-power configuration in that case.

Low-Power Configuration

In the following, the low-power configurations for the fixed-decoding scheme are presented. The same encoding as for a defect on line m, expressed by $\mathbf{\Gamma}_{\text{conf,SA}_m}$, reduces the 3D-interconnect power consumption if the data words are normally distributed. This is because of the specific characteristics of the MSB region of normally distributed data words, outlined in Subsection 5.1.2 on Page 91. In brief, the more-significant bits of normally distributed data show a strong spatial correlation. Hence, if one of the correlated MSBs toggles, the remaining ones toggle as well in the same direction. The switching activity of the correlated bits in the MSB regions varies between 0 and 1, depending on the word-level correlation of the data. However, if, for each pattern $\vec{d}$ of a normally distributed data stream, the bits are XORed with the MSB d_m ($\mathbf{\Gamma}_{\text{conf,SA}_m}$ for p-doped substrates), the strong spatial correlation of the higher-value bits result in codeword bits nearly stable on logical 0. An XOR operation with the negated MSB $\neg d_m$ ($\mathbf{\Gamma}_{\text{conf,SA}_m}$ for n-doped substrates), results in codeword bits nearly stable on 1. Thus, independent of the word-level correlation, the switching of the more-significant lines is reduced to almost 0 when the encoding is $\mathbf{\Gamma}_{\text{conf,SA}_m}$ for p-doped as well as n-doped substrates.

This coding scheme is based on the same LPC concept as the Gray-coding technique used in Chapter 6, which showed to improve the TSV power consumption effectively. However, the technique in Chapter 6 also employs inversions to optimize the TSV power consumption through the bit probabilities further. These inversions were realized by swapping inverting with non-inverting drivers. This is not possible for the proposed technique, as it would destroy the error masking for SA faults.

Nevertheless, an alternative approach is presented in the following, which allows optimizing the TSV power consumption through the bit probabilities. The lowest power consumption is obtained for p-doped substrates if the 1-bit probabilities are maximized, as shown in Chapter 4. For n-doped substrates, the 1-bit probabilities should be minimized. Both can be achieved by assigning the negated MSB for p-doped substrates, and the non-negated MSB for n-doped substrates, to the invert line. Thus, the enhanced low-power configuration—optimizing the bit probabilities besides the switching activities—is the same as for an SA1 on line m for p-doped substrates (typically entail SA0 faults). For n-doped substrates, the enhanced low-power configuration is the same as for an SA0 on line m. The improved configuration comes at slightly increased hardware costs if an encoder that is only capable of fixing one SA fault type is integrated into an ASIC die. Here, the multiplexers of the two configurable circuits, shown in Figure 9.5b and 9.5c, have to be extended by one input $\neg d_m$.

The fixed-decoder scheme can also reduce the power consumption for arbitrary distributed data words. In the defect-free case, the invert line can be used

to implement any low-power invert-coding. For example, the widely known classical bus invert (CBI) technique, presented in [112], has already shown to effectively reduce the power consumption of TSVs (besides metal wires as intended) in previous chapters of this thesis. The multiplexers in Figure 9.5b–9.5d need an additional input, Inv_{lp}, to integrate an invert-coding technique into an ASIC die. Additionally, a block that checks the invert condition and sets Inv_{lp} if required has to be implemented. This block is cut from the power supply if it is not used (erroneous link). If the block is implemented by means of an FPGA soft-component, it can be simply erased when it is not needed. Please note, CBI is just the most well-known of many possible invert-codes (*e.g.*, [79, 80]). Generally, all invert-codes can be combined with the proposed redundancy scheme. However, CBI encoding typically results in the best trade-off between the 3D-interconnect power-consumption improvement and the CODEC complexity. Hence, other invert-coding methods are not investigated for the proposed technique.

9.5.2. Fixed-Encoding Scheme

Two TSV redundancy schemes based on a fixed encoding and a configurable decoding were proposed in [38]. Although not noticed by the authors, one even has the smallest possible encoder complexity. This fixed encoder, illustrated in Figure 9.6a, is a simple Gray encoder with the extension that the LSB of the information word, d_1, is attached as the codeword LSB. Thus, it also has good potential to reduce the 3D-interconnect power consumption. Consequently, the fixed-decoding scheme proposed in this section is based on this existing technique.

The fixed encoding of the technique is expressed as

$$b_i = \begin{cases} d_1 & \text{for } i = 1 \\ d_i \oplus d_{i-1} & \text{for } i \in \{2, \ldots, m\} \\ d_m & \text{for } i = m+1. \end{cases} \tag{9.28}$$

For a group size of four, the according encoding matrix is equal to

$$\mathbf{\Gamma}_{\mathrm{fix,prev}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}. \tag{9.29}$$

As the decoding is already extensively discussed in [38], it is only briefly summarized in the following. The decoder circuit is shown in Figure 9.7a. The

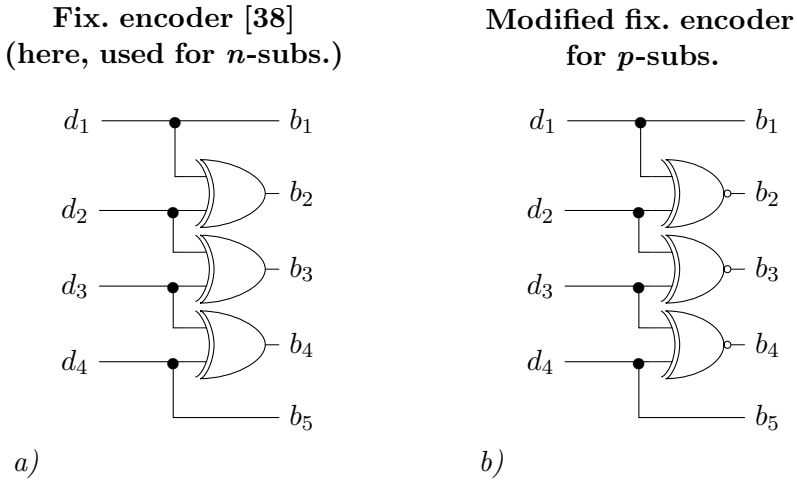


Fig. 9.6: Fixed-encoder circuits for a group size of four: *a)* Existing structure presented in [38] (reused in the proposed technique for n-doped substrates); *b)* Proposed structure for p-doped substrates.

decoding is formally expressed as:

$$\hat{d}_i = \begin{cases} \sum_{l=1}^{i} b_{*l} & \text{for } c_{\mathrm{rx},i} = 0 \\ \sum_{l=i+1}^{m+1} b_{*l} & \text{for } c_{\mathrm{rx},i} = 1. \end{cases} \tag{9.30}$$

The m-bit configuration signal, $\vec{c}_{\mathrm{rx}}$, controls m output multiplexers. If all configuration bits for the link are 0, the multiplexers connect through the higher inputs and the output is independent of line $m+1$. An error on any line with index j is bypassed by setting all bits $c_{\mathrm{rx},i}$ with $i \geq j$ to 1. Thus, $\vec{c}_{\mathrm{rx}}$ has to be "0011" to treat an exemplary error on the third line (*i.e.*, $b_{*,3}$ is faulty). This results in the following decoding:

$$\begin{aligned} \hat{d}_1 &= b_{*1} = d_1; \\ \hat{d}_2 &= b_{*1} \oplus b_{*2} = d_1 \oplus (d_1 \oplus d_2) = d_2; \\ \hat{d}_3 &= b_{*5} \oplus b_{*4} = d_4 \oplus (d_4 \oplus d_3) = d_3; \\ \hat{d}_4 &= b_{*5} = d_4. \end{aligned} \tag{9.31}$$

Hence, the correct information word is decoded independent of the logical value on the faulty line.

Furthermore, the specific behavior of the more-significant bits of normally distributed data words, in combination with the XORing of neighbored bits leads to codeword bits that are almost stable on logical 0. Hence, the previous technique is optimal for n-doped substrates, as it reduces the switching activities

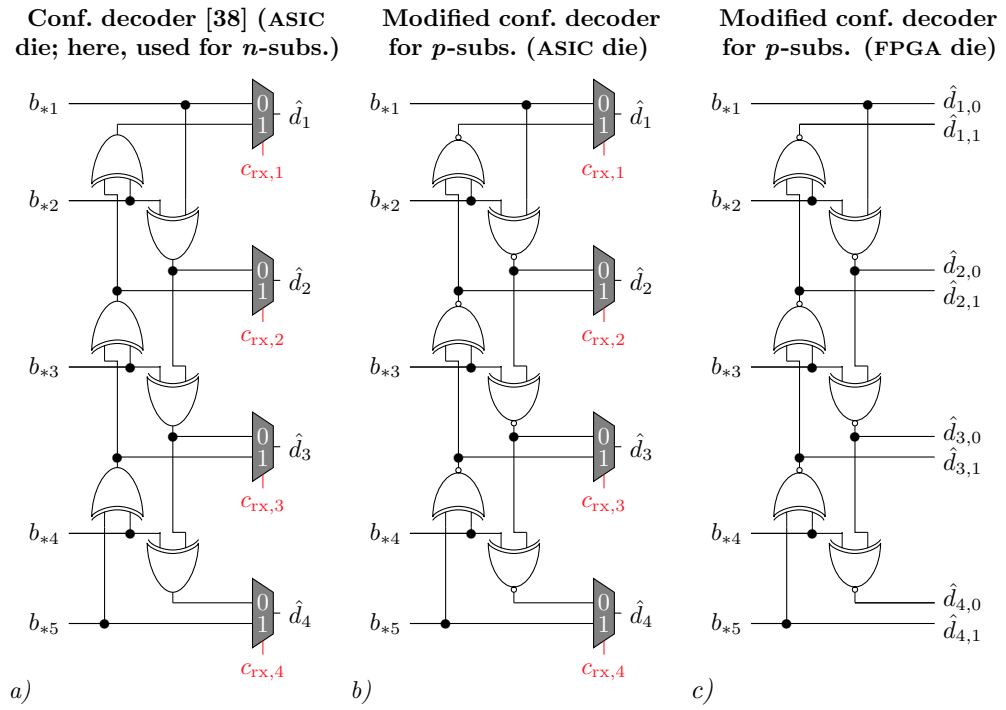


Fig. 9.7: Decoder circuits in the fixed encoder scheme for a group size of four: *a)* Previous structure [38] (used here for n-doped substrates); *b)* Proposed structure for p-doped substrates; *c)* Proposed structure for FPGA dies and p-doped substrates.

as well as the 1-bit probabilities. Consequently, the previously proposed fixed encoder scheme is reused for n-doped substrates. However, for p-doped substrates, this previous encoder results in increased TSV capacitance quantities, which makes the approach suboptimal.

Consequently, a small extension in the existing scheme is presented in the following, which maximizes the 1-bit probabilities for typical p-doped substrates, without the need to invert nets through a driver swapping.[2] To maximize the 1-bit probabilities, the XOR gates are swapped with XNOR gates, as shown in Figure 9.6b. Afterward, the codeword bits are negated for the indexes 2 to m, compared to the previous encoding architecture. This modification does not affect the switching properties; it only maximizes the 1-bit probabilities instead of the 0-bit probabilities. Thus, it results in a lower TSV power consumption, without affecting the power-consumption reduction for metal wires.

[2]For the fixed-encoder scheme, inverting nets through automatic driver swapping, as proposed in Chapter 6, is possible. The extension is presented to enable the use of the proposed technique in the best possible way, without the need to implement the technique from Chapter 6.

The modified circuit is still of minimal complexity, as the area and the delay of an XOR and an XNOR gate are generally equal. To decode the correct information word, $\vec{d}$, XOR and XNOR gates also have to be swapped for the decoder. The resulting decoder structure for a p-doped substrate is illustrated in Figure 9.7b.

Again, the multiplexers and the configuration signals are not required if the decoder is implemented in an FPGA die, as illustrated in Figure 9.7c. Here, during placement and routing of the FPGA soft-components, the EDA tool routes the decoder outputs that are independent of the broken lines to the input of the FPGA soft-components.

9.6. Evaluation

In this section, the proposed technique is evaluated in terms of yield enhancement, power-consumption reduction, and hardware complexity. Furthermore, a comparison with existing multiplexer-based and coding-based redundancy schemes is drawn. Thereby, only the values for the previous coding-based scheme based on a Gray encoding are presented, as it significantly outperforms the other one, also presented in [38], in all analyzed metrics.

9.6.1. Yield Enhancement

In order to evaluate the yield enhancement, first, the repair limitations of the proposed and existing redundancy schemes are analyzed. Furthermore, it is outlined that the shielding approach presented in Chapter 7 can be used to overcome the limitation of the proposed technique partially. Afterward, the overall TSV-yield enhancement is quantified.

Repair limitations

Not all single line defects can be fixed with the proposed, nor existing, redundancy techniques. Pinhole defects entail a resistive connection between the TSV conductor and the substrate-bias contacts [25]. The bigger the pinhole, the lower the resistance. Thus, leakage current flows through the conductive substrate if the potential on a TSV with a pinhole defect differs from the substrate-bias potential. These leakage currents can dramatically increase the static power consumption. Furthermore, the resistive connection affects the body-bias potential of active elements located near the defective TSV. With increased pinhole size, this will eventually result in malfunctions of active circuit elements.

This cannot be fixed by existing redundancy schemes, nor by the proposed fixed-encoding scheme, since the schemes do not allow to permanently set the potential of a faulty TSV to the potential of the substrate bias. However, the proposed fixed-decoding scheme overcomes this limitation. The configurable encoding sets a faulty line permanently on the potential of the substrate bias

to mask the error. Consequently, no leakage current flows and the body bias of the active circuits is not impaired.

However, the proposed fixed-decoding scheme has a different limitation: A too significant increase in a TSV resistance makes the signal at the far end sensitive to coupling noise. The previously used equivalent circuit for TSV defect modeling does not include coupling effects [25]. Thus, this model cannot identify any limitation for the proposed fixed-decoding scheme.

Hence, a more detailed setup is required to outline the limitations of the scheme. Coupling effects are maximized for large, densely bundled, TSVs and low significant-frequencies, as shown in Chapter 4. Thus, a 3π-RLC equivalent circuit of a 3×3 array with the minimum TSV spacing is extracted by means of the scaleable TSV-array model presented in Subsection 2.4.1 and the *Q3D Extractor*.[3] The TSV radius and length are set to $2\,\mu$m and $50\,\mu$m, respectively, which are typical global TSV dimensions. For this radius, the minimum TSV spacing is $8\,\mu$m [41]. A TSV in the middle of an array experiences the highest coupling, as shown in Chapter 4. Thus, a resistive open in the middle TSV is analyzed. The extracted TSV resistance is increased by a parameterizable value R_o to model a defect, analogously to [25, 26].

Spectre circuit simulations are used to analyze worst-case TSV coupling for the fixed decoding scheme. Thereby, inverters of strength 1×, stemming from the aggressively scaled 15-nm standard-cell library *NanGate15* [57], are used to drive/load the signal TSVs, as small-sized cells are most sensitive to coupling noise. Contact resistances of $0.1\,\mathrm{k\Omega}$ are included between the drivers and the TSVs to obtain realistic values for the path resistance. Ramp voltage sources with a rise time of 10 ps, in combination with an additional inverter for realistic signal shaping, are used to generate input stimuli.

Due to the *p*-doped substrate, the faulty TSV (noise victim) is set to constant logical 0 for error masking. All other signal TSVs switch from logical 0 (*i.e.*, 0 V) to logical 1 (*i.e.*, $V_\mathrm{dd} = 0.8\,\mathrm{V}$), which results in the highest coupling noise on a stable victim line [129]. Over multiple simulation-runs, the value of the defect resistance, R_o, is increased from $0\,\Omega$ to $1\,\mathrm{M\Omega}$, since this is the maximum reported increase in the TSV resistance due to void or delamination defects [26]. Reported is the output waveform of the load driver at the far end of the faulty TSV. Since the load driver is an inverter, while the faulty TSV is grounded for error masking, the analyzed signal is constant on V_dd (*i.e.*, 0.8 V) in the ideal/noise-free scenario.

The results are illustrated in Figure 9.8. In the first analyzed scenario, all remaining eight TSVs in the array are signal TSVs, which switch from logical 0 to 1. Here, the large coupling capacitances in the TSV array affect the

[3] All TSV parasitics in this evaluation section are extracted for a significant frequency of 6 GHz.

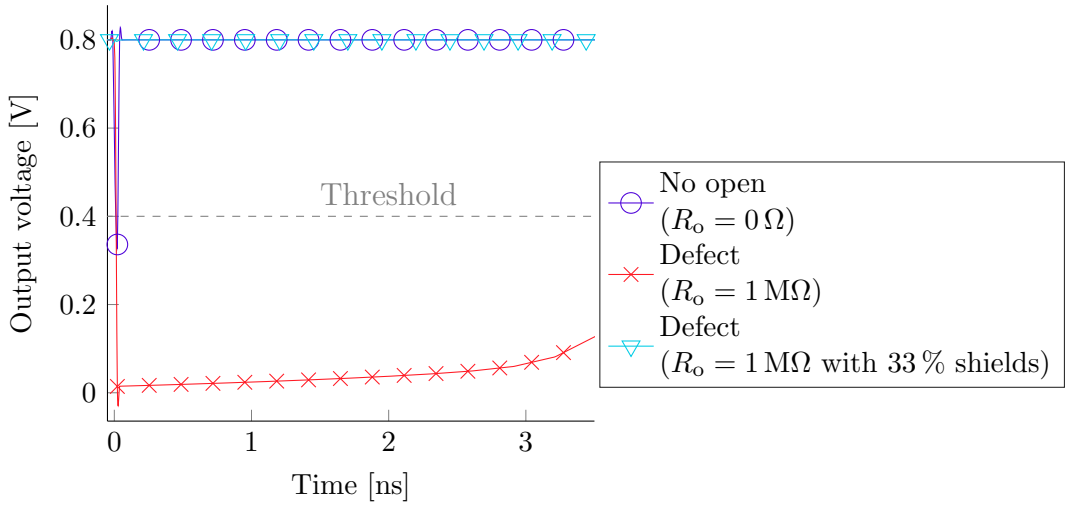


Fig. 9.8: Worst-case coupling noise on the load-driver output of a stable and faulty TSV line.

output signal even if the victim TSV has no open defect (*i.e.*, for $R_o = 0\,\Omega$). The maximum induced noise for no manufacturing defect is 0.46 V. Thus, the noise even pushes the signal for a few picoseconds below the threshold voltage ($V_{dd}/2 = 0.4\,V$), resulting in a logical glitch in the signal. With an increase in resistance, R_o, also the coupling noise increases. For an open resistance of 1 MΩ, the noise-peak magnitude is in the range of V_{dd}, and the maximum duration of a logical glitch is several nanoseconds. This exceeds the typical timing budget for a TSV link. Thus, delay faults can even result in an erroneous transmission for a stable input signal if coupling effects are considered.

However, the shielding technique presented in Chapter 7 of this thesis can be used to overcome this bottleneck. Thus, stable power or ground (P/G) are positioned within the TSV array using the proposed performance-optimal assignment technique. Adding stable shields does not impair the overall TSV yield itself if the shields are on the substrate-bias potential (*i.e.*, all errors are masked anyhow). Thus, for typical p-doped substrates, ground-shields must be used, and, for n-doped substrates, V_{dd}-shields.

Hence, the coupling-noise analysis is repeated after placing 33 % stable ground TSVs in the array (*i.e.*, *3C* shielding) in a performance-optimal way. As shown in Figure 9.8, the shield lines reduce the maximum coupling noise at the output in case of a defect to about zero. Thus, it completely overcomes the repair limitations for the analyzed technology. However, logical glitches might still occur for future technologies. In this case, a higher shield density could be used to increase the yield further. For the extreme scenario where eight shields

enclose every signal TSV, no coupling noise occurs. Here, the fixed-decoding approach can repair faults as long as the open resistance, R_o, is several times lower than the gate-to-substrate resistance, which exceeds the GΩ range.

By integrating shields, not only yield issues due to coupling noise are addressed, but also the maximum TSV performance is improved. Consequently, the proposed technique enables low-power, yield-enhanced, and yet high-performance 3D interconnects. However, since the performance improvement of the proposed shielding technique was already evaluated in Chapter 7, the interconnects performance is not further investigated in the present chapter. Summarized, no redundancy technique is capable of fixing all possible faults. However, shielding techniques can effectively increase the number of repairable errors for the proposed fixed-decoding scheme.

Overall TSV Yield

Previous works use the following formula for the manufacturing yield of an m-bit wide link with r redundant lines [38, 127]:

$$Y_{\text{prev}}(m,r) = \sum_{i=0}^{r} \binom{m+r}{i} (1-p_{\text{def}})^{m+r-i} p_{\text{def}}^{i}, \tag{9.32}$$

where p_{def} is the TSV-defect probability. However, this formula is based on the assumption that the redundancy scheme can cope with every defect, as long as the total number of defects in the set does not exceed r. This is not precise, as outlined previously. The actual yield is slightly lower:

$$Y(m,r,\kappa_d) = \sum_{i=0}^{r} \binom{m+r}{i} (1-p_{\text{def}})^{m+r-i} (\kappa_{\text{repair}} \cdot p_{\text{def}})^{i}, \tag{9.33}$$

where κ_{repair} is the conditional probability that an occurring defect is treatable. Thus, for the technique based on two redundancy schemes, the overall TSV manufacturing yield is:

$$\begin{aligned} Y_{\text{t}} = {} & Y(m_{\text{e}},1,\kappa_{\text{e}})^{\left\lfloor \frac{N_{\text{e}}}{m_{\text{e}}} \right\rfloor} \cdot Y(\text{mod}(N_{\text{e}},m_{\text{e}}),1,\kappa_{\text{e}}) \\ & \cdot Y(m_{\text{d}},1,\kappa_{\text{d}})^{\left\lfloor \frac{N_{\text{d}}}{m_{\text{d}}} \right\rfloor} \cdot Y(\text{mod}(N_{\text{d}},m_{\text{d}}),1,\kappa_{\text{d}}). \end{aligned} \tag{9.34}$$

In this equation, m_{e} and m_{d} are the group sizes for the fixed-encoding and the fixed-decoding scheme, respectively. N_{e} and N_{d} are the numbers of overall required functional data TSVs for the two schemes. κ_{e} and κ_{d} indicate the repair capabilities of the schemes. The modulo terms (*i.e.*, terms with "$\text{mod}(N_{\text{e}},m_{\text{e}})$" or "$\text{mod}(N_{\text{e}},m_{\text{e}})$") are required to represent the grouping for the remaining TSVs, in case of no integer ratio between the number of required data TSVs and the group sizes.

For an integer ratio between N_e and m_e as well as N_d and m_d, the formula for the overall yield simplifies as follows:

$$Y_\mathrm{t} = Y(m_\mathrm{e}, 1, \kappa_\mathrm{e})^{\frac{N_\mathrm{e}}{m_\mathrm{e}}} \cdot Y(m_\mathrm{d}, 1, \kappa_\mathrm{d})^{\frac{N_\mathrm{d}}{m_\mathrm{d}}}. \tag{9.35}$$

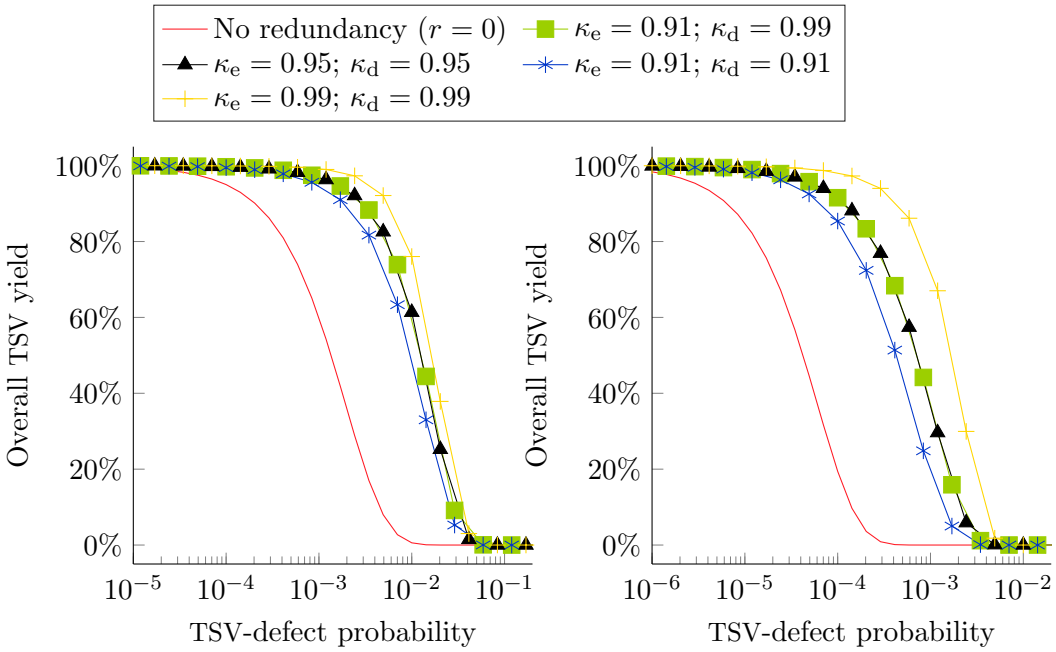
Previous works (*e.g.*, [38,127]) already analyzed how the yield is reduced with increasing group size, m, and the overall number of required functional TSVs, N. The following analysis quantifies the effect of varying repair capabilities for the redundancy schemes. In the analysis, the number of required data TSVs, as well as the group sizes, are equal for both schemes (*i.e.*, $N_\mathrm{e} = N_\mathrm{d} = N/2$; $m_\mathrm{e} = m_\mathrm{d} = m$). Once a system with a moderate amount of required TSVs ($N = 512$) and a group-size, m, of eight is considered, and once a system with a high amount of required TSVs ($N = 16,384$) and a group size, m, of 16. 0.91, 0.95 and 0.99 are analyzed as the repair capabilities, κ_e and κ_d.

The overall TSV yields over the TSV defect probability are plotted in Figure 9.9 for all analyzed scenarios. In accordance with previous works, an increase in the TSV count and the group size generally results in decreased yield quantities. As expected, with increasing repair capabilities of the two schemes, κ_e and κ_d, the overall TSV yield increases.

An interesting observation is that the mean of κ_e and κ_d, represented by $\bar{\kappa}_\mathrm{repair}$, seems to be the determining factor for the overall yield. Thus, increasing just the repair capability of one of the two redundancy schemes by x typically is as effective as increasing both by $x/2$. As outlined previously, shielding can enhance the repair capabilities of the proposed fixed-decoder scheme with relatively low effort. Thus, shielding allows to effectively increase the overall yield for the proposed technique employing two different redundancy schemes.

Reported TSV defect rates are not smaller than 1×10^{-5} [130]. A TSV defect rate of 1×10^{-5} results in an unacceptably low yield of 84.89 % for the complex system. However, integrating the proposed low-power redundancy scheme enhances the yield to above 99.13 %, even for a pessimistic mean repair capability of 95 % (*i.e.*, $\bar{\kappa}_\mathrm{repair}$ equal to 0.95, which implies that 5 % of the occurring single-line defects are not treatable, despite having a redundant TSV left). This yield enhancement of the proposed high-level technique is even higher than the one for an improvement in the TSV defect rate by a factor of 10 ×, resulting in an overall TSV manufacturing yield of 98.37 %. This underlines the effectiveness of the proposed technique in terms of yield enhancement.

The use of the proposed redundancy technique for a repair capability of 95 % and the improved TSV defect rate of 1×10^{-6} results in an overall TSV manufacturing yield, which is as high as 99.91 %. Consequently, the proposed technique paves the way for three-sigma TSV production for complex systems (requires a yield of 99.73 %).



a) $N = 512$; $m = 8$ *b)* $N = 16,384$; $m = 16$

Fig. 9.9: Effect of the proposed redundancy technique on the overall TSV manufacturing yield, over the TSV defect probability, for different repair capabilities of the fixed encoder (κ_e) and decoder (κ_d) scheme. The yield is analyzed for: *a)* a system with 512 required functional/data TSVs, grouped in sets of eight; and *b)* a system with 16 k required functional/data TSVs, grouped in sets of 16.

For the smaller system with 512 required data TSVs, the overall yield without a redundancy scheme is 99.49 % for a TSV defect rate of 1×10^{-5}. Integration of a redundancy technique with a mean repair capability of 95 % results in a yield that is higher than 99.97 %. Thus, the proposed technique enables three-sigma production for the smaller system without the need for an improvement in the TSV defect rate. Furthermore, the yield improvement implies a reduction in the number of systems that have to be discarded after manufacturing due to TSV defects by a factor of 17 ×.

Applying the proposed yield-enhancement technique for an improved TSV defect rate of 1×10^{-6} results in an overall yield of 99.9994 %, when 99 % of the occurring TSV defects are generally treatable. This is even close to six-sigma production, which requires a yield of 99.9997 %. Hence, the proposed technique drastically boosts the process of overcoming TSV-related yield issues for complex 3D systems.

9.6.2. Impact on the Power Consumption

The effect of the redundancy schemes on the bit-level characteristics, which determine the 3D-interconnect power consumption, is investigated in this subsection for various data types and a common p-doped substrate. Therefore, normally distributed 16-bit data streams, each containing 1×10^5 words, with different word-level correlations, ρ, and standard deviations, σ, are generated.

For the various data streams, the information words are either negatively correlated ($\rho = -0.95$), uncorrelated ($\rho = 0$), or positively correlated ($\rho = 0.95$). The logarithm to the base two of the standard deviation, $\log_2(\sigma)$, is varied between 1 and 15 for all three correlation quantities. Subsequently, the effects of the proposed and previous redundancy schemes—all adding one redundant line—on the transmitted codewords are included for the case of a defect-free link. Afterward, the total/accumulated switching activity of the bits ($\alpha_{\text{total}} = \sum_i \alpha_i$) and the total/mean 1-bit probability for all bits, ($\bar{p} = 1/n \sum_i p_i$), is determined.[4] Investigated are the total switching activity and 1-bit probability as these metrics quantify by how much the approach is generally capable of reducing the power consumption of TSV and metal-wire structures. In the case study (Section 9.7), the effect of the techniques on the true power consumption of an exemplary TSV and metal-wire structure is investigated.

Three different encoder variants are considered for the proposed fixed-decoding scheme. In the first one, the invert line is equal to the MSB of the transmitted information word in the low-power configuration (*i.e.*, minimal encoder implementation for an ASIC die). In the second variant, the invert line is equal to the negated MSB of the transmitted data word, in order to optimize the 1-bit beside the switching probabilities (*i.e.*, extended ASIC or FPGA implementation). In the last one, the invert line is used for CBI coding.

The results of the analysis are plotted in Figure 9.10. Both multiplexer-based techniques, *Signal-Shift* and *Signal-Reroute*, always increase the interconnect power consumption, as they increase the total switching activity, compared to the case where no redundancy technique is integrated, due to the added redundant line, transmitting a data bit even in the defect-free scenario.

In contrast, all other redundancy schemes enable a drastic improvement in the interconnect power consumption. The maximum decrease in the total switching activity for the proposed scheme with the fixed encoding and any analyzed data stream is 84.2 %, while the overall 1-bit probability can be increased by up to 81.8 %. On average, for all analyzed data streams, the fixed encoding scheme results in an improvement in the total switching activity and 1-bit probability by 37.1 % and 53.4 %, respectively.

[4] The accumulated switching activity instead of the mean switching probability is investigated to take the added redundant line/bit for the yield-enhancement techniques into account. This ensures a fair comparison.

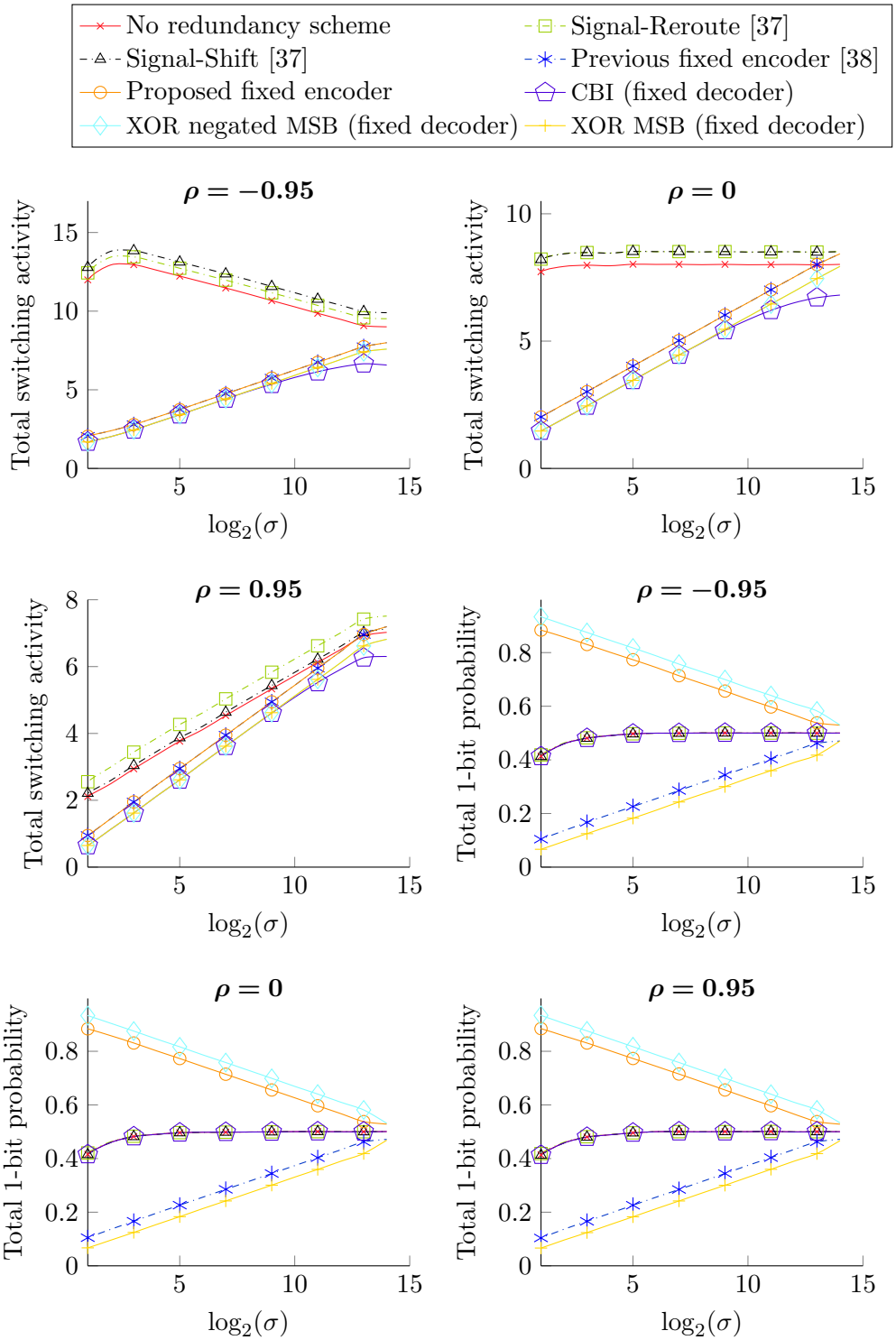


Fig. 9.10: Effect of the proposed and previous redundancy schemes on the power-related bit-level statistics for various synthetic data streams.

The same reduction in the switching activities is obtained for the existing coding-based redundancy scheme, but it has the drawback for typical p-doped substrates that it decreases the 1-bit probability by 77.6 % in the peak, and 42.2 % in the mean. Thus, as expected, the previous coding-based technique shows the same power savings for metal wires as the proposed/modified version, but lower power savings for TSVs.

The fixed-encoding schemes only optimize the power consumption for normally distributed data words. Hence, the more the data-word distribution tends to be uniformly distributed (*i.e.*, high σ), the lower the improvement in the power consumption. Hence, both fixed encoding schemes result in an increase in the total switching activity by 6.3 % for uniformly distributed data due to the added redundant line. This increase is roughly the bit overhead of the encoding technique (*i.e.*, the reciprocal of the group size, $1/m$).

For all three variants of the proposed fixed-decoding scheme, the switching reduction reaches values as high as 86.3 %. Using the MSB or the negated MSB of the information word as the invert line reduces the switching by 36.2 % in the mean. The total 1-bit probability is additionally increased by 87.7 % in the peak, and 62.5 % in the mean, for the negated MSB as the invert bit. In contrast, using the non-negated MSB for inversion reduces the 1-bit probability by at least 5.81 % (50.5 % in the mean). Hence, the non-minimal ASIC implementation can result in a significantly lower TSV power consumption if the TSVs show a significant MOS effect.

The negated or non-negated MSB as the invert signal, again, is only effective for normally distributed data. Thus, the power-consumption reduction vanishes with increasing standard deviation, σ. However, in both variants of the proposed fixed-decoding scheme, one line of the link is always stable. Thus, the scheme never results in an increase in the interconnect power consumption compared to the scenario without any integrated redundancy scheme, even if all redundant lines are required for yield enhancement and the data is completely random. This is another major advantage of the proposed redundancy scheme compared to all previous ones.

Integrating CBI coding in the proposed fixed-decoder scheme enables a power-consumption reduction also for uniformly distributed data (here, about 11 %). The toggle reduction with CBI coding for normally distributed data is about the same as for the two more hardware-efficient approaches. Thus, the two non-CBI alternatives are generally more efficient for normally distributed data due to their lower CODEC complexities.

In summary, the proposed technique can significantly reduce the power consumption of 3D interconnects compared to all previous techniques. If the TSV MOS effect is not too significant and the data words tend to be normally distributed, even the simplest proposed configurable ASIC encoder structure already results in a drastic improvement of the interconnect power consumption.

With increasing TSV MOS effect, adding the small extension to optimize the bit probabilities becomes reasonable. Adding an extra invert-coding is only required for data that tends to be uniformly distributed.

9.6.3. Hardware Complexity

The hardware costs associated with loading the repair signatures (RSs) from the NVM cells into the repair registers (RRs) are discussed in this subsection. Furthermore, the costs associated with the actual redundancy schemes are quantified.

NVM Cells and Controller

The lower bound for the overall number of bits of the compressed concatenated repair signatures (RSs) is

$$NB_{\mathrm{rs}} = \frac{N}{m} \cdot \lceil \log_2 (m+1) \rceil. \tag{9.36}$$

In the following, the worst case, a system without FPGA dies, is considered. The number of NVM cells for the multiplexer-based schemes is equal to $2\,NB_{\mathrm{rs}}$ when the cells are distributed with the transmitter (Tx) and receiver (Rx) components. This also requires the programming of NVM cells in every die of the system. The number of required NVM cells can be minimized to NB_{rs} with the coding-based scheme presented in [38]. However, an expensive programming circuit is still required in every die. With the proposed technique, the number of required NVM cells and the number of dies that require NVM programming are minimized. Consequently, considering distributed NVM cells strongly favors the technique presented throughout this chapter.

Thus, a global NVM macro, from which the RSs are loaded into the RRs during start-up, is considered in the following for a fair comparison. In this scenario, the number of NVM cells is minimized for all techniques, and at most, one NVM-programming circuit has to be integrated. No extra programming circuit is required if one die includes one-time-programmable NVM cells either way (*e.g.*, for memory repair).

An advantage of coding-based techniques over multiplexer-based techniques for a global NVM macro is that the implementation cost of the NVM controller, which serially loads the RSs from the NVM cells into the RRs at start-up, is reduced. This NVM controller, proposed in [127] along with an approach to connect the RRs to a shift register while minimizing the inter die connections, is briefly reviewed in the following. The repair signatures are stored in ascending order in the NVM macro. Afterward, the controller has to shift out the stored RSs twice: First in ascending and subsequently in descending order. This results in a relatively low-complex controller architecture.

For the proposed and the previous coding-based approach, every RS only

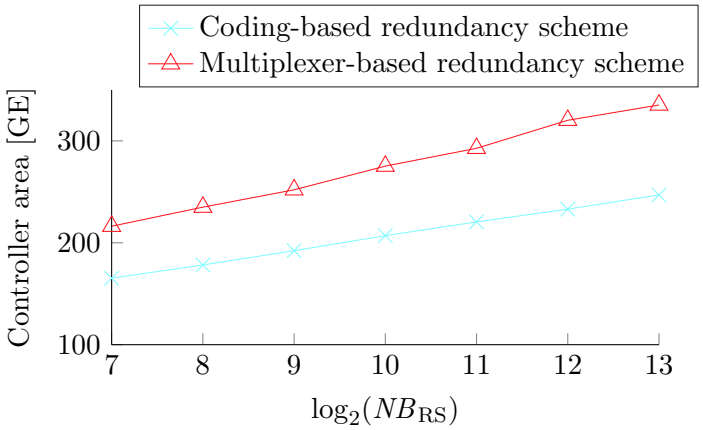


Fig. 9.11: Area in gate equivalents of the synthesized controller, which loads the RSs from the global NVM macro into the distributed RRs for the two redundancy approaches, multiplexer based and coding based.

has to be loaded once, since it is only required at the encoder (*i.e.*, Tx) or the decoder (*i.e.*, Rx) side. This asymptotically reduces the length of the shift register chain and the number of inter-die connections by 50 %. The RSs are stored in the NVM macro in the order the according RRs are connected. This results in a further reduced controller complexity as the values only have to be shifted out once. Furthermore, the number of cycles to load the configuration data is reduced by 50 %.

To quantify the controller complexity for multiplexer-based and coding-based redundancy techniques, generic RTL implementations are created. At start-up, the controllers serially shift the RSs, stored in a global NVM macro with a word width of NB_{nvm}, into the RRs using a serial output, *rr_in*. Therefore, the signal *shift_rr* is set, which enables shifting the values of the RR chain. The completion of loading the RSs into the RRs is acknowledged by a flag *rr_ready*. After completion, the controllers go to an idle state until they receive a request to reload the RSs via a signal *load_rb* (*e.g.*, after a reset of a block).

Except during start-up, an NVM controller is idle. Thus, it is cut from the power supply afterward. Consequently, the power consumption of a controller does not have a noticeable impact. Furthermore, the *shift_rr* signal, which must be routed to all RRs, is the limiting factor for the speed at which the RSs can be loaded, not the controller. Thus, only the controller area is a major concern. The area of the controllers, depending on NB_{rs}, is determined by means of gate-level syntheses in the *NanGate15* technology using *Synopsys* tools. Thereby, an NVM-word width of 32 bit is considered.

In Figure 9.11, the resulting controller areas in gate equivalents are plotted over $\log_2(NB_{\text{rs}})$. The results show that both NVM-controller areas increase

linearly with $\log_2(NB_{\mathrm{rs}})$. Using a coding-based redundancy technique, instead of a multiplexer-based technique, reduces the controller complexity by about 25 %, independent of the length of the concatenated RS.

Redundancy Scheme

The complexities of the encoder-decoder pairs are assessed in the following. Using the area of typical standard cells is a straightforward approach to obtain formulas for the area of the components, including the RRs. Such formulas are reported in Table 9.1. Note that no differentiation is made between gates of equal complexity (*e.g.*, XOR and XNOR, or AND and NOR). Thus, the hardware complexities of the previous and the extended fixed-encoding scheme are reported in a single row. To analyze heterogeneity, component complexities for the source (Tx) and the destination (Rx) die are reported individually. A unary encoding or a one-hot encoding of the compressed RS, stored in the RR, is required to generate the configuration signals, $\vec{c}_{\mathrm{tx}}$ or $\vec{c}_{\mathrm{rx}}$, for some redundancy schemes. Such an encoding requires at least m gates of the size of a standard AND gate, which is considered for area estimation.

For the proposed fixed-decoding scheme, four alternative implementations are considered for an ASIC die. In the first one, just the minimal configurable hardware illustrated in Figure 9.5b or 9.5c is implemented. In the second and third one, an extra input is added to the multiplexers, which is either an invert line for CBI coding or the negated MSB for low-power coding of DSP signals. Both structures have the same hardware complexity, and in both cases, the hardware is extended to improve the low-power configuration. Thus, the values for both alternatives are reported in a single row of the table, using the identifier "Fix. decoder low-power". In the fourth one, an XOR gate and a configuration bit are added, as shown in Figure 9.5d, in order to cope with both SA fault types.

For an implementation into an FPGA die, the minimal circuit (shown in Figure 9.5e) can operate in all low-power configurations and repair all fault types. The configuration hardware (*i.e.*, multiplexers and RRs) vanishes in FPGA dies for all proposed and previous redundancy schemes. Redundancy schemes, realized in ASIC dies by means of multiplexers, have no hardware complexity at all in FPGA dies. Here, the place-and-route tool for the FPGA soft-components just assigns the signals to the functional lines and skips defective ones.

All paths starting at the RRs are not considered for delay analysis, as the RR signals are constant during normal operation (after start-up). Thus, the formulas presented in Table 9.2 are used to estimate the component delays.

Power estimation is not as straight forward for several reasons. At first, the configuration signals are only updated once during start-up. Afterward, they are stable. Furthermore, this allows effective clock gating of all RRs with the shift signal of the NVM controller, *shift_rr*, which is only active during start-up. This

Table 9.1.: Area requirements of the transmitter (Tx) and receiver (Rx) units of the redundancy schemes.

Scheme	**Area Tx***	**Area Rx***
Fix. decoder minimal (ASIC)	$m(A_{\text{xor}}{+}A_{\text{mux}}) + \lceil \log_2(m{+}1) \rceil A_{\text{ff}}$	mA_{xor}
Fix. decoder low-power (ASIC)	$m(A_{\text{xor}}{+}A_{\text{mux}}) + A_{\text{mux}} + \lceil \log_2(m{+}2) \rceil A_{\text{ff}}$	mA_{xor}
Fix. decoder both SAs (ASIC)	$m(A_{\text{xor}}{+}A_{\text{mux}}) + A_{\text{xor}} + \lceil \log_2(2(m{+}1)) \rceil A_{\text{ff}}$	mA_{xor}
Fix. decoder (FPGA)	mA_{xor}	mA_{xor}
Fix. encoder (ASIC)	$(m-1)A_{\text{xor}}$	$m(2A_{\text{xor}}{+}A_{\text{mux}}{+}A_{\text{and}}) - 2A_{\text{xor}} + \lceil \log_2(m{+}1) \rceil A_{\text{ff}}$
Fix. encoder (FPGA)	$(m-1)A_{\text{xor}}$	$2(m{-}1)A_{\text{xor}}$
Signal-Reroute [37] (ASIC)	$(m{-}1)A_{\text{mux}} + \lceil \log_2(m{+}1) \rceil A_{\text{ff}}$	$m(A_{\text{mux}}{+}A_{\text{and}}) + \lceil \log_2(m{+}1) \rceil A_{\text{ff}}$
Signal-Shift [37] (ASIC)	$m(A_{\text{mux}}{+}A_{\text{and}}) - A_{\text{mux}} + \lceil \log_2(m{+}1) \rceil A_{\text{ff}}$	$m(A_{\text{mux}}{+}A_{\text{and}}) + \lceil \log_2(m{+}1) \rceil A_{\text{ff}}$
Signal-Reroute/Shift [37] (FPGA)	0	0

* A_{mux}, A_{xor}, and A_{and} are the areas of two-input multiplexer, XOR and AND cells in the die, respectively. A_{ff} is the area of a flip-flop.

Table 9.2.: Delay of the transmitter (Tx) and receiver (Rx) units.

Scheme	**Delay Tx***	**Delay Rx***
Fix. decoder minimal (ASIC)	$T_{\text{xor}} + \lceil \log_2 (m{+}1) \rceil T_{\text{mux}}$	T_{xor}
Fix. decoder low-power (ASIC)	$T_{\text{xor}} + \lceil \log_2 (m{+}2) \rceil T_{\text{mux}}$	T_{xor}
Fix. decoder both SAs (ASIC)	$2T_{\text{xor}} + \lceil \log_2 (m{+}1) \rceil T_{\text{mux}}$	T_{xor}
Fix. decoder (FPGA)	T_{xor}	T_{xor}
Fix. encoder (ASIC)	T_{xor}	$(m{-}1)T_{\text{xor}} + T_{\text{mux}}$
Fix. encoder (FPGA)	T_{xor}	$(m{-}1)T_{\text{xor}}$
Signal-Reroute [37] (ASIC)	$\lceil \log_2 (m) \rceil T_{\text{mux}}$	T_{mux}
Signal-Shift [37] (ASIC)	T_{mux}	T_{mux}
Signal-Reroute/Shift [37] (FPGA)	0	0

* T_{mux}, T_{xor}: Delay of two-input multiplexer and XOR cells, respectively.

asymptotically reduces the dynamic power consumption associated with the RRs and the configuration signals to zero. Second, the power consumption depends on the switching activities, which again depend on the signal characteristics. Third, the power consumption will be determined mainly by the interconnects in between, and not by the encoder and the decoder, as shown in the case study of this work. Thus, the power consumption of the components is not considered in this subsection. An in-depth power analysis is included in the case study of this chapter.

The standard cell libraries *NanGate15* [57] (15 nm) and *NanGate45* [131] (45 nm) are used to compare the complexity of the redundancy techniques for a bidirectional link with varying group size, m. Thereby, the integration of the extended low-power configuration for the fixed-decoder scheme is considered for the proposed technique, employing two redundancy schemes. In order to obtain pessimistic values, only two-input standard cells with a drive strength of 1× are considered. Reported are the products of the total cell area and the maximum component delay, obtained by means of the formulas reported in Table 9.1 and 9.2.

Four different die interfaces are analyzed. The first one—representing a heterogeneous scenario—is made up of an FPGA die using a 15-nm technology and a mixed-signal die made-up of 45-nm standard cells. In the second scenario, the 15-nm technology is used for an ASIC instead of an FPGA die. The third scenario is the interface between a 15-nm ASIC die and a 15-nm FPGA die. As the last scenario, a homogeneous interface between two 15-nm ASIC dies is considered.

Figure 9.12 illustrates the results of the analysis. Compared to the best previous technique, *Signal-Shift*, the coding-based approach from [38] results in an increase in the complexity by a factor of 3× for a group size of four,

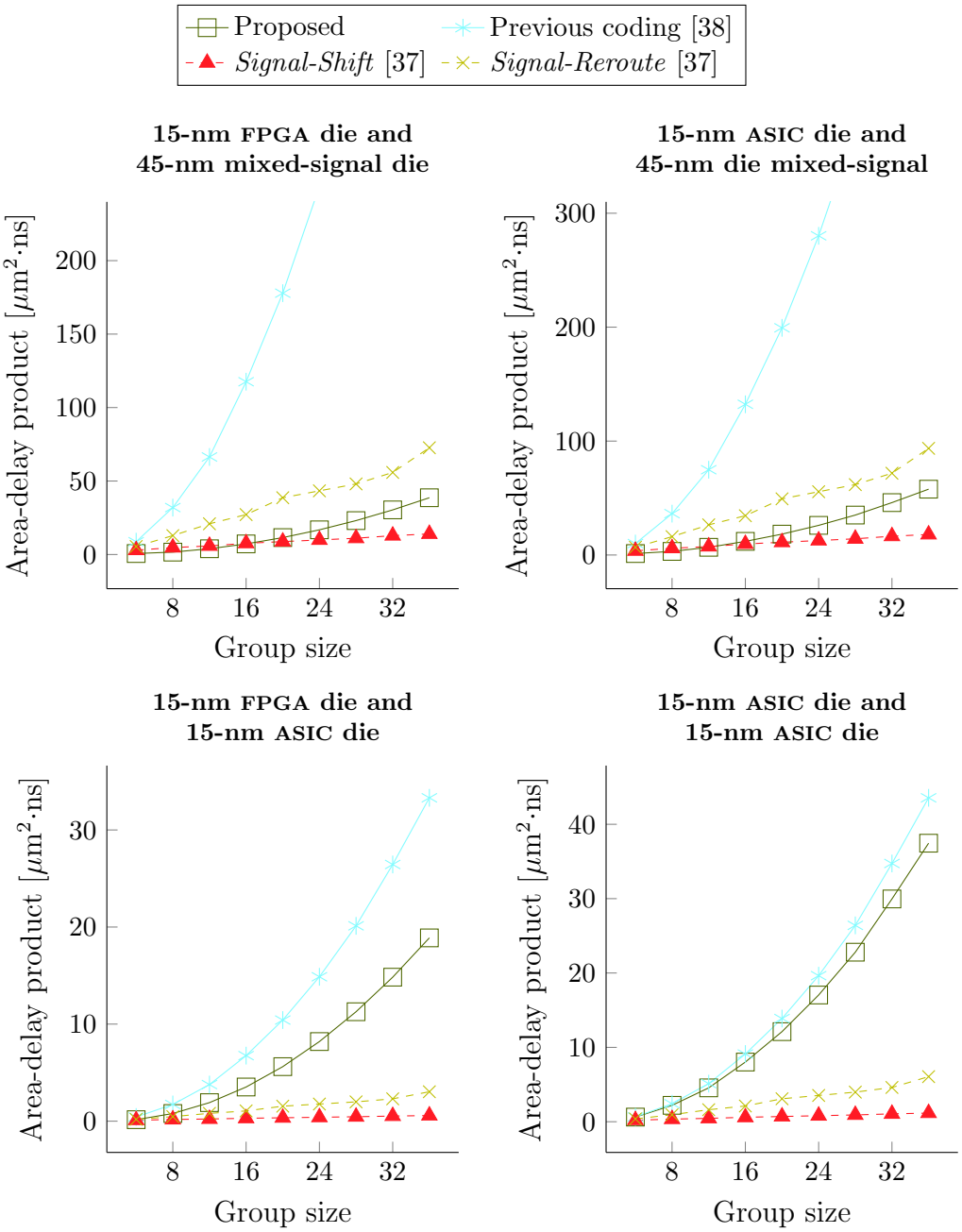


Fig. 9.12: Area-delay products of the redundancy techniques over the group size for a bidirectional 3D link and different die interfaces.

and a factor of 34× for a group size of 32, for the first strongly heterogeneous interface. Thus, the previously outlined reduction in the complexity of the NVM controller, as well as the reduction in the interconnect power consumption, come at high expenses. This is due to the delay of the XOR chains of the decoder located in the slower mixed-signal technology. The length of the XOR chains, and thereby the delay, increases linearly with the group size, m. Since the hardware complexity also increases with the group size, the area-delay product increases quadratically.

However, the proposed technique partially overcomes this bottleneck. At first, the critical XOR chain is only present in the faster layer, while, in the slower layer, the delay is minimal. Additionally, integrating just fixed circuits of minimal complexity in the mixed-signal die reduces the overall area overhead drastically. For a group size of four, the proposed technique even reduces the circuit complexity compared to the best previous technique, *Signal-Shift*, by a factor of 4×.

Albeit slower, the maximum delay of the proposed technique still increases with the group size. Consequently, the proposed technique scales worse than the *Signal-Shift* method with a constant delay. Thus, from a group size of 20 or higher, the proposed technique results in a higher circuit complexity. However, commercially available systems typically use significantly smaller group sizes in order to achieve the highest yield [9]. A similar trend can be observed for the second interface (15-nm digital and 45-nm mixed-signal), with the exception that here the complexity of the proposed redundancy technique exceeds the complexity of the *Signal-Shift* scheme for group sizes of 16 or higher.

If the same technology is used for both dies of the interface, the die location of the XOR chains does not affect the overall circuit delay. Thus, the only advantage of the proposed technique for the third analyzed scenario is that the configuration hardware vanishes. Hence, the area is reduced at the cost of a delay increase, compared to the best previous technique. However, configuration-hardware costs increase only logarithmically. Thus, in the third analyzed scenario, the *Signal-Shift* approach results in a significantly lower hardware complexity for bigger group sizes than the proposed technique with a linearly increasing delay.

In the last scenario, no heterogeneity remains, resulting in a generally higher complexity for the proposed technique. The same accounts for the interface between two FPGA dies. Hence, this scenario is not investigated here. Nevertheless, even for such homogeneous interfaces, the proposed technique outperforms the existing coding-based approach in terms of CODEC complexity. Furthermore, an integration of a low-power gray encoding together with the *Signal-Shift* method also requires a long XOR chain. Thus, if a low-power coding is either-way required, the proposed technique is also superior for homogeneous interfaces.

9.7. Case Study

A case study, considering the commercially available heterogeneous SoC from [125], in which a TSV redundancy technique has to be integrated, is presented in this section. Even-though the SoC integration is referred to as 3D, the actual integration method in [125] is 2.5D, as TSVs connect the dies through an interposer. Interposer-based 2.5D integration is an intermediate solution, mainly driven by cost concerns. Since technology has advanced since the publication date of [125], true heterogeneous 3D integration is considered in this thesis. The reference system consists of two 28-nm FPGA dies, and a 65-nm mixed-signal die containing sixteen 13-bit analog-to-digital converters (ADCs) and sixteen 16-bit digital-to-analog converters (DACs). Again, technology improvement is taken into account: The 15-nm technology *NanGate15*, and the 45-nm technology *NanGate45* are considered for the FPGA dies and the mixed-signal die, respectively.

The proposed technique is not suitable for the interface between the two FPGA dies, where the standard *Signal-Shift* approach requires no hardware at all. Thus, only the die interface between the data converters and the FPGA is extended by the proposed technique.

Compared is the integration of the proposed redundancy technique, without CBI coding, and the best previous coding-based and multiplexer-based approach. The group size is chosen to match the pattern width (*i.e.*, 13 bit for the down-stream from the ADCs to the FPGAs; 16 bit for the up-stream from the FPGAs to the DACs).[5] Thus, including the redundant lines, the total TSV count is: $16 \cdot (14 + 17) = 496$. In this analysis, the TSVs radius, pitch, and depth are $2\,\mu$m, $8\,\mu$m, and $80\,\mu$m, respectively.[6]

Crosstalk is minimized by the *2C* shielding technique from Chapter 7 to guarantee maximized repair capabilities and an improved TSV performance. For a fair comparison, an integration of *2C* shielding is also considered for previous techniques. With *2C* shielding, one 4×7 and one 5×7 TSV array are required per ADC and DAC, respectively. Furthermore, typical global metal wires with a minimum width/spacing of $0.45\,\mu$m and a length of $100\,\mu$m are added between the components and the TSVs to obtain full 3D-interconnect paths. The PTM interconnect tool [69] is used to obtain the metal-wire parasitics. *Q3D Extractor* and the TSV-array model are again used to obtain TSV parasitics for a significant frequency of 6 GHz. The logic areas of the redundancy techniques are obtained by RTL-to-gate-level syntheses. Furthermore, the syntheses results

[5] This maximum group size results in worst-case values for the proposed technique due to the inferior scaling of the fixed-encoder scheme, compared to previous multiplexer-based techniques, outlined in Subsection 9.6.3.

[6] The higher TSV depth, compared to previous analyses, is chosen for proper noise insulation between the mixed-signal and the FPGA dies.

Table 9.3.: Logic area, A_{logic}, maximum delay, $\hat{T}_{\text{total}}$, and required NVM space for different redundancy techniques and the heterogeneous SoC from [125].

Metric	**Proposed**	**Previous coding [38]**	***Signal-Shift* [37]**
A_{logic}	$1050\,\mu\text{m}^2$	$3715\,\mu\text{m}^2$	$3394\,\mu\text{m}^2$
$\hat{T}_{\text{total}}$	198 ps	559 ps	166 ps
NVM space	0 b	80 b	144 b

in combination with *Spectre* simulations—employing the extracted metal-wire and TSV parasitics—are used to obtain the accumulated maximum propagation delay, $\hat{T}_{\text{total}}$, of the encoder—3D-interconnect—decoder arrangements.

In Table 9.3, the results are presented. The proposed technique does not require any RRs for the analyzed system. This fact, combined with the minimal circuit complexity in the mixed-signal die, results in a decrease in the area requirements by more than a factor of 3×, compared to the best previous multiplexer-based or coding-based techniques.

Compared to the best coding-based technique, the total delay is reduced by a factor of 2.8×. The proposed technique induces a small delay increase of 32 ps (*i.e.*, 19 %), compared to the best multiplexer-based technique, *Signal-Shift*. A delay-increase only occurs for the down-stream path (*i.e.*, ADC to FPGA), which employs the fixed-encoder scheme with the long XOR chains. However, the ADC sample duration in [125] is 8 ns, while, it is only 625 ps for a DAC. Consequently, this delay penalty is not critical. The previous coding-based technique requires a 5-bit wide RR for each of the 16 decoders in the mixed-signal die. The *Signal-Shift* technique, additionally, requires a 4-bit wide RR for each of the 16 encoders in the mixed-signal die. Thus, for the reference techniques, RSs of a total length of 80 bit and 144 bit have to be loaded into the RRs during start-up. In contrast, not a single RS has to be loaded for the proposed technique. Although the NVM controllers are responsible for less than 5 % of the area requirements of the reference techniques, the previous analysis reveals a major advantage of the proposed technique: NVM is no longer required.

The publicly available RF data set *Deepsig* [132] is used to quantify the data-dependent power consumption of the logic and the interconnects. The modulation schemes QAM64, QAM16, and BPSK are analyzed to consider RF signals with low and high modulation possibilities. The mean power consumption for the transmission of 1,028 different example signals, each 1,000 bit-patterns long, and a signal-noise ratio of 18 dB, is investigated for each modulation scheme. Therefore, the switching and bit probabilities of the raw information bits and the transmitted (encoded) bits are determined by exact bit-level

Table 9.4.: Power consumption for different RF signals.

Metric	Proposed	Previous coding [38]	*Signal-Shift* [37]
$P_{\text{dyn,link}}$—QAM64	8.69 mW	9.49 mW	13.44 mW
$P_{\text{dyn,link}}$—QAM16	8.83 mW	9.52 mW	12.87 mW
$P_{\text{dyn,link}}$—BPSK	8.73 mW	9.48 mW	12.80 mW
$P_{\text{dyn,logic}}$—QAM64	0.98 mW	4.11 mW	1.44 mW
$P_{\text{dyn,logic}}$—QAM16	0.98 mW	4.10 mW	1.45 mW
$P_{\text{dyn,logic}}$—BPSK	1.00 mW	4.16 mW	1.50 mW
$P_{\text{leak,logic}}$	35.85 μW	85.42 μW	70.24 μW
mean(P_{total})	9.77 mW	13.70 mW	14.57 mW

simulations. This information is forwarded to the synthesis tool to obtain values for the power consumption of the logic circuits. Thereby, the clock-gate signal of the RRs, *shift_rr*, is defined as constant zero. The pattern-dependent interconnect power consumption is calculated by means of the precise high-level formulas presented in Part II of this thesis, employing the exact bit-level probabilities.

Both proposed redundancy schemes are based on low-power coding techniques that primary aim for a reduction in the switching activities instead of the coupling switching. Hence, as shown in Section 8.3, their relative power savings are lower the more the signal edges are temporally aligned. Thus, perfectly temporal aligned signal edges are considered in this case study to report pessimistic values for the proposed technique.

The results, including the leakage power consumption of the cells, are presented in Table 9.4. Compared to the best multiplexer-based scheme, the proposed technique results in a reduction in the interconnect and logic power consumption by over 30 % for all analyzed modulation schemes. Leakage currents are even reduced by about 50 %. The previous coding-based technique results in about 5 % lower reductions in the 3D-interconnect power consumption. However, here, the complex configurable decoders in the mixed-signal die result in an increased power consumption of the logic cells compared to the *Signal-Shift* technique. Hence, the overall power-consumption reduction for the previous coding-based technique is, in the mean, only 6 % and thus more than five times lower than for the proposed technique.

In summary, the proposed technique significantly outperforms previous TSV redundancy schemes in nearly all metrics for the existing heterogeneous SoC. Hence, the system is an ideal system for the integration of the proposed technique. The reason is that the proposed technique can heavily exploit the technological heterogeneity between the dies and the normal distribution of the data words transmitted over the 3D links of the RF SoC.

9.8. Conclusion

In this chapter, a novel TSV redundancy technique was proposed, designed to improve the manufacturing yield and the power consumption of 3D interconnects. The technique is based on two optimal coding-based redundancy schemes, used in combination, which minimize the complexity of redundancy techniques in heterogeneous 3D-integrated systems. Furthermore, the coding techniques are designed in a way that the 3D-interconnect power consumption can be reduced effectively. Moreover, using the proposed technique in combination with the shielding technique from Chapter 7 does not only result in an additional improvement in the 3D-interconnect performance, it also further enhances the TSV yield.

Despite its low complexity and the capability to significantly improve the power consumption, the proposed technique shows to improve the overall TSV yield by a factor of 17 $\times$. Furthermore, an extensive set of analyses, as well as a case study for a commercial SoC, show that the proposed technique significantly outperforms previous approaches in nearly all metrics.

Part IV: Finale

CHAPTER 10

Conclusion

Previous optimization techniques show rather limited capabilities to improve the TSV performance and are furthermore not capable of reducing the critical TSV power consumption. This work has shown that these severe limitations of existing approaches are mainly due to the absence of precise models to estimate the TSV power consumption and performance on higher abstraction levels. Optimization techniques can only effectively overcome interconnect-related issues in TSV-based 3D ICs if the techniques are derived based on physically precise, and yet universally valid, high-level models.

To address this problem, a set of abstract models have been proposed in this thesis, which enable a precise estimation of the power consumption and the performance of 3D interconnects solely on high abstraction levels. The proposed models are physically precise since they were derived while considering the lowest abstraction levels. Nevertheless, the methods are universally valid and provide the required level of abstraction. Evaluations have shown that the proposed models allow for an estimation of the power consumption and performance of modern 3D interconnects with relative errors below 3 %. Moreover, it was demonstrated that the proposed models enable the derivation of efficient optimization techniques for 3D interconnects.

Four novel optimization techniques have been proposed in this thesis—derived based on the contributed high-level formulas—which drastically outperform all previous methods. At first, a low-power technique for TSV structures was

presented. To the best of the author's knowledge, the technique is the first-ever low-power approach tailored for TSV-based interconnects. The technique improves the TSV power consumption by an intelligent, physical-effect-aware, local net-to-TSV assignment, exploiting the bit-level statistics of the transmitted patterns. Analyses for a broad set of real and synthetic data streams have underlined the efficiency of the proposed low-power technique, which can reduce the TSV power consumption by over 45 %, without inducing noticeable overhead costs. Furthermore, the proposed technique enables us to use any existing low-power approach for metal wires in the most efficient way for TSVs. This allows for improving the power consumption of both structures effectively through a single data-encoding technique.

To provide a further improvement in the performance of 3D interconnects, two optimization techniques for low-power and yet high-performance 3D interconnects have been proposed in this thesis. The first technique exploits the bit-level switching characteristics of traditional 2D-CAC-encoded data by a net-to-TSV assignment, which results in a simultaneous improvement in the power consumption and performance of TSVs as well as metal wires. Experimental results showed that this approach drastically improves the performance of modern TSV arrays and metal-wire buses by 25.6 % and 90.8 %, respectively, while it simultaneously reduces the TSV and metal-wire power consumption by up to 5.3 % and 20.4 %, respectively. In comparison, prior approaches improve the TSV performance by a maximum of 12.0 % while providing no optimization for the metal wires at all. Moreover, the best previous technique results in significantly higher hardware costs and a dramatic increase in the TSV power consumption by 50 %.

Since the first-proposed, as well as all previous, optimization techniques for high-performance 3D interconnects are only efficient for the case of a perfect temporal alignment between the signal edges on the lines, the second proposed technique is designed to improve the TSV performance for the case of an arbitrary temporal misalignment between the signal edges. More precisely, the technique exploits temporal misalignment between the signals by a physical-effect-aware TSV assignment, which again results in negligible overhead costs. An in-depth evaluation showed that the proposed technique can improve the TSV performance by over 65 %. Paired with a traditional low-power code for metal wires, a further improvement in the TSV power consumption by about 17 % can be achieved, despite resulting in a more than five times lower bit overhead than the best previous technique.

The last proposed low-power technique furthermore addresses the low manufacturing yield of TSV structures. This yield-enhancement technique is based on two optimal coding-based redundancy schemes that are used in combination. These coding techniques enable us to minimize the complexity of a redundancy scheme, by exploiting technological heterogeneity between the dies of a 3D IC

as much as possible. Furthermore, the coding techniques are designed in a way that they additionally decrease the power consumption of TSVs and metal wires effectively. An evaluation, considering a commercially available heterogeneous SoC, showed that the proposed technique decreases the area overhead and the power consumption compared to the best previous technique by 69.1 % and 32.9%, respectively. Despite the low hardware requirements and the large power savings it provides, the proposed yield-enhancement method can improve the overall manufacturing yield by a factor of about 17 × for typical TSV defect rates (same as previous techniques). Moreover, it was shown that the proposed yield-enhancement method can be combined with the proposed technique for high-performance 3D interconnects in order to achieve all objectives at once: An improved power consumption, an improved performance, and an improved overall yield.

In conclusion, all proposed techniques show to improve the quality of 3D interconnects drastically beyond the state-of-the-art, which is enabled by physically precise and yet abstract models for the TSV metrics. Thereby, this thesis will accelerate the process of overcoming interconnect-related issues in 3D-integrated systems through effective high-level optimization techniques. Using high-level techniques to address the power issues of 3D interconnects generally is of particular importance because just relying on advances in the manufacturing of TSV structures will likely not work due to the very poor scaling of the parasitics with shrinking geometrical TSV dimensions.

10.1. Impact on Future Work

This thesis is expected to have a significant impact on future work beyond its primary objectives. The contributed set of high-level models for a precise estimation of the TSV power consumption and performance pioneer the derivation of novel optimization techniques for future applications. New trends for 3D systems like machine learning will bring new types of data flow and thus might require a new set of low-power techniques. However, the derivation of the optimization techniques proposed in this thesis has shown a systematic way to approach this challenge based on precise high-level models.

Moreover, as shown through [CA7, CP2, CP5, CP6], the proposed models enable a technology-precise simulation of the performance and power requirements of 3D-integrated systems using high-level programming languages such as *SystemC*. This feature, for example, allows for an early design space exploration, considering the impact of the TSVs on the system's power consumption and performance.

Furthermore, the optimization techniques themselves will have an impact on future optimization methods, rather than just serving as reference approaches. For example, for each newly proposed low-power coding technique for metal

wires, the method from Chapter 6 can directly quantify how effective the technique could be used for TSVs—determining the usability of the technique for 3D integration. In the same way, one can precisely quantify the maximum efficiency of a newly proposed 2D CAC for 3D integrated systems by the formal method contributed through Chapter 7.

Also, the mathematical background provided throughout the derivation of the proposed low-power technique for yield-enhanced 3D interconnects is expected to have a significant impact on future work. At first, a precise mathematical formulation of the requirements for a reconfigurable encoding and decoding method to serve as an efficient yield-enhancement technique is contributed. By these formulas, new coding techniques can be systematically derived and verified. Second, the lower bounds for the minimum possible circuit complexities of a yield-enhancement encoder and decoder allow for a precise quantification of the hardware efficiency of novel coding approaches for yield-enhanced 3D interconnects.

In summary, the contributions of this thesis will not only significantly boost the process of overcoming TSV-related problems in 3D ICs through the provided optimization techniques. Moreover, the thesis also enables and assists new research projects covering various emerging domains of 3D integration.

References

[1] Mark T Bohr. Interconnect scaling-the real limiter to high performance ULSI. In *International Electron Devices Meeting*, pages 241–244. IEEE, 1995.

[2] Ruth Brain. Interconnect scaling: Challenges and opportunities. In *International Electron Devices Meeting (IEDM)*, pages 3–9. IEEE, 2016.

[3] Igor L Markov. Limits on fundamental limits to computation. *Nature*, 512(7513):147–154, 2014.

[4] Nir Magen, Avinoam Kolodny, Uri Weiser, and Nachum Shamir. Interconnect-power dissipation in a microprocessor. In *International Workshop on System Level Interconnect Prediction*, pages 7–13. ACM, 2004.

[5] M Mitchell Waldrop. The chips are down for Moore's law. *Nature News*, 530(7589):144, 2016.

[6] A 3D technology toolbox in support of system-technology co-optimization. `https://www.imec-int.com/en/imec-magazine/imec-magazine-july-2019`, July 2019. (Accessed on 15/08/2019).

[7] Sung Kyu Lim. *Design for High Performance, Low Power, and Reliable 3D Integrated Circuits*. Springer Science & Business Media, 2012.

[8] Patrick Leduc, Francois de Crecy, Murielle Fayolle, Barbara Charlet, Thierry Enot, Marc Zussy, Bob Jones, Jean-Charles Barbe, Nelly Kernevez, Nicolas Sillon, et al. Challenges for 3D IC integration: bonding quality and thermal management. In *International Interconnect Technology Conference*, pages 210–212. IEEE, 2007.

[9] Uksong Kang, Hoe-Ju Chung, Seongmoo Heo, Duk-Ha Park, Hoon Lee, Jin Ho Kim, Soon-Hong Ahn, Soo-Ho Cha, Jaesung Ahn, DukMin Kwon, et al. 8 Gb 3-D DDR3 DRAM using through-silicon-via technology. *IEEE Journal of Solid-State Circuits*, 45(1):111–119, 2009.

[10] Dae Hyun Kim and Sung Kyu Lim. Impact of TSV and device scaling on the quality of 3D ICs. In *More than Moore Technologies for Next Generation Computer Design*, pages 1–22. Springer, 2015.

[11] Chuan Xu, Hong Li, Roberto Suaya, and Kaustav Banerjee. Compact AC modeling and performance analysis of through-silicon vias in 3-D ICs. *IEEE Transactions on Electron Devices*, 57(12):3405–3417, 2010.

[12] Dae Hyun Kim, Saibal Mukhopadhyay, and Sung Kyu Lim. Fast and accurate analytical modeling of through-silicon-via capacitive coupling. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(2):168–180, 2011.

[13] A Ege Engin and Srinidhi Raghavan Narasimhan. Modeling of crosstalk in through silicon vias. *IEEE Transactions on Electromagnetic Compatibility*, 55(1):149–158, 2012.

[14] Stefano Piersanti, Francesco de Paulis, Antonio Orlandi, Madhavan Swaminathan, and Vittorio Ricchiuti. Transient analysis of TSV equivalent circuit considering nonlinear MOS capacitance effects. *IEEE Transactions on Electromagnetic Compatibility*, 57(5):1216–1225, 2015.

[15] Stefano Piersanti, Enza Pellegrino, Francesco De Paulis, Antonio Orlandi, Daniel H Jung, Dong-Hyun Kim, Joungho Kim, and Jun Fan. Algorithm for extracting parameters of the coupling capacitance hysteresis cycle for TSV transient modeling and robustness analysis. *IEEE Transactions on Electromagnetic Compatibility*, 59(4):1329–1338, 2016.

[16] Yarui Peng, Taigon Song, Dusan Petranovic, and Sung Kyu Lim. Silicon effect-aware full-chip extraction and mitigation of TSV-to-TSV coupling. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 33(12):1900–1913, 2014.

[17] Chang Liu, Taigon Song, Jonghyun Cho, Joohee Kim, Joungho Kim, and Sung Kyu Lim. Full-chip TSV-to-TSV coupling analysis and optimization in 3D IC. In *Design Automation Conference (DAC)*, pages 783–788. ACM/IEEE, 2011.

[18] Chenbing Qu, Ruixue Ding, Xiaoxian Liu, and Zhangming Zhu. Modeling and optimization of multiground TSVs for signals shield in 3-D ICs. *IEEE Transactions on Electromagnetic Compatibility*, 59(2):461–467, 2016.

[19] Zheng Xu and Jian-Qiang Lu. Three-dimensional coaxial through-silicon-via (TSV) design. *IEEE Electron Device Letters*, 33(10):1441–1443, 2012.

[20] Jonghyun Cho, Eakhwan Song, Kihyun Yoon, Jun So Pak, Joohee Kim, Woojin Lee, Taigon Song, Kiyeong Kim, Junho Lee, Hyungdong Lee, et al. Modeling and analysis of through-silicon via (TSV) noise coupling and suppression using a guard ring. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(2):220–233, 2011.

[21] Rajeev Kumar and Sunil P Khatri. Crosstalk avoidance codes for 3D VLSI. In *Design, Automation & Test in Europe Conference (DATE)*, pages 1673–1678. IEEE, 2013.

[22] Qiaosha Zou, Dimin Niu, Yan Cao, and Yuan Xie. 3DLAT: TSV-based 3D ICs crosstalk minimization utilizing less adjacent transition code. In *Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 762–767. IEEE, 2014.

[23] Xiaole Cui, Xiaoxin Cui, Yewen Ni, Min Miao, and Jin Yufeng. An enhancement of crosstalk avoidance code based on Fibonacci numeral system for through silicon vias. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 25(5):1601–1610, 2017.

[24] Nahid M Hossain, Ritesh Kumar Reddy Kuchukulla, and Masud H Chowdhury. Failure analysis of the through silicon via in three-dimensional integrated circuit (3D-IC). In *International Symposium on Circuits and Systems (ISCAS)*, pages 1–4. IEEE, 2018.

[25] Maoxiang Yi, Jingchang Bian, Tianming Ni, Cuiyun Jiang, Hao Chang, and Huaguo Liang. A pulse shrinking-based test solution for prebond through silicon via in 3-D ICs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 38(4):755–766, 2018.

[26] Yi Zhao, Saqib Khursheed, and Bashir M Al-Hashimi. Online fault tolerance technique for TSV-based 3-D-IC. *IEEE Transactions on Very Large Scale Integrated (VLSI) Systems*, 23(8):1567–1571, 2015.

[27] Daniel Arumí, Rosa Rodríguez-Montañés, and Joan Figueras. Prebond testing of weak defects in TSVs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 24(4):1503–1514, 2015.

[28] Sergej Deutsch and Krishnendu Chakrabarty. Contactless pre-bond TSV test and diagnosis using ring oscillators and multiple voltage levels. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 33(5):774–785, 2014.

[29] Li-Ren Huang, Shi-Yu Huang, Stephen Sunter, Kun-Han Tsai, and Wu-Tung Cheng. Oscillation-based prebond TSV test. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(9):1440–1444, 2013.

[30] Shi-Yu Huang, Yu-Hsiang Lin, Li-Ren Huang, Kun-Han Tsai, and Wu-Tung Cheng. Programmable leakage test and binning for TSVs with self-timed timing control. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(8):1265–1273, 2013.

[31] Brandon Noia, Shreepad Panth, Krishnendu Chakrabarty, and Sung Kyu Lim. Scan test of die logic in 3-D ICs using TSV probing. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 23(2):317–330, 2014.

[32] Young-Woo Lee, Hyeonchan Lim, and Sungho Kang. Grouping-based TSV test architecture for resistive open and bridge defects in 3-D-ICs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 36(10):1759–1763, 2016.

[33] Imed Jani, Didier Lattard, Pascal Vivet, Lucile Arnaud, and Edith Beigné. BISTs for post-bond test and electrical analysis of high density 3D interconnect defects. In *European Test Symposium (ETS)*, pages 1–6. IEEE, 2018.

[34] Chao Wang, Jun Zhou, Roshan Weerasekera, Bin Zhao, Xin Liu, Philippe Royannez, and Minkyu Je. BIST methodology, architecture and circuits for pre-bond TSV testing in 3D stacking IC systems. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 62(1):139–148, 2014.

[35] Liang-Che Li, Wen-Hsuan Hsu, Kuen-Jong Lee, and Chun-Lung Hsu. An efficient 3D-IC on-chip test framework to embed TSV testing in memory BIST. In *Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 520–525. IEEE, 2015.

[36] Erik Jan Marinissen. Testing TSV-based three-dimensional stacked ICs. In *Design, Automation & Test in Europe Conference (DATE)*, pages 1689–1694. IEEE, 2010.

[37] Ang-Chih Hsieh and TingTing Hwang. TSV redundancy: Architecture and design issues in 3-D IC. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 20(4):711–722, 2011.

[38] Christof Osewold, Wolfgang Büter, and Alberto García-Ortiz. A coding-based configurable and asymmetrical redundancy scheme for 3-D interconnects. In *International Symposium on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC)*, pages 1–8. IEEE, 2014.

[39] Chunjie Duan, Brock J LaMeres, and Sunil P Khatri. *On and off-chip crosstalk avoidance in VLSI design*. Springer, 2010.

[40] Anand Raghunathan, Niraj K Jha, and Sujit Dey. *High-level power analysis and optimization*. Springer Science & Business Media, 2012.

[41] 2015 International Technology Roadmap for Semicondunctors (ITRS) — Interconnects. `https://eps.ieee.org/images/files/Roadmap/ITRSIntercon2015.pdf`. (Accessed on 03/10/2019).

[42] Raguraman Venkatesan, Jeffrey A Davis, Keith A Bowman, and James D Meindl. Optimal n-tier multilevel interconnect architectures for gigascale integration (GSI). *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 9(6):899–912, 2001.

[43] James D Meindl, Jeffrey A Davis, Payman Zarkesh-Ha, Chirag S Patel, Kevin P Martin, and Paul A Kohl. Interconnect opportunities for gigascale integration. *IBM Journal of Research and Development*, 46(2.3):245–263, 2002.

[44] N Rohrer, C Akrout, M Canada, D Cawthron, B Davari, R Floyd, S Geissler, R Goldblatt, R Houle, P Kartschoke, et al. A 480 MHz RISC microprocessor in a 0.12μm L_{eff} CMOS technology with copper interconnects. In *International Solid-State Circuits Conference (ISSCC)*, pages 240–241. IEEE, 1998.

[45] M Naik, S Parikh, P Li, J Educato, D Cheung, I Hashim, P Hey, S Jenq, T Pan, F Redeker, et al. Process integration of double level copper-low k (k=2.8) interconnect. In *International Interconnect Technology Conference (IITC)*, pages 181–183. IEEE, 1999.

[46] Payman Zarkesh-Ha, Peter Bendix, William Loh, JinJoo Lee, and James D Meindl. The impact of cu/low/spl kappa/on chip performance. In *International ASIC/SOC Conference*, pages 257–261. IEEE, 1999.

[47] Andrew B Kahng, Sudhakar Muddu, and Egino Sarto. Interconnect optimization strategies for high-performance VLSI designs. In *International Conference on VLSI Design*, pages 464–469. IEEE, 1999.

[48] Luca Benini and Giovanni De Micheli. Networks on chips: A new SoC paradigm. *IEEE Computer*, 35(1):70–78, 2002.

[49] Mauro J Kobrinsky, Bruce A Block, Jun-Fei Zheng, Brandon C Barnett, Edris Mohammed, Miriam Reshotko, Frank Robertson, Scott List, Ian Young, and Kenneth Cadien. On-chip optical interconnects. *Intel Technology Journal*, 8(2), 2004.

[50] Vasilis F Pavlidis, Ioannis Savidis, and Eby G Friedman. *Three-dimensional integrated circuit design*. Newnes, 2017.

[51] Xianmin Chen and Niraj K Jha. A 3-D CPU-FPGA-DRAM hybrid architecture for low-power computation. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 24(5):1649–1662, 2015.

[52] Franz Laermer and Andrea Schilp. Method of anisotropically etching silicon, March 26 1996. US Patent 5,501,893.

[53] Bioh Kim, Charles Sharbono, Tom Ritzdorf, and Dan Schmauch. Factors affecting copper filling process within high aspect ratio deep vias for 3D chip stacking. In *Electronic Components and Technology Conference (ECTC)*, pages 6–pp. IEEE, 2006.

[54] C. S. Premachandran, Luke England, Sukeshwar Kannan, Rakesh Ranjan, Kong Boon Yeap, Walter Teo, Salvatore Cimino, Tan Jing, Haojun Zhang, Daniel Smith, et al. Impact of 3D via middle TSV process on 20nm wafer level FEOL and BEOL reliability. In *Electronic Components and Technology Conference (ECTC)*, pages 1593–1598. IEEE, 2016.

[55] C. S. Premachandran, Salvatore Cimino, Sean Ogdan, Zhuo-Jie George Wu, Daniel Smith, Sukesh Kannan, Linjun Cao, Manjunatha Prabhu, Walter Yao, Rakesh Ranjan, et al. Comprehensive 3D TSV reliability study on 14nm FINFET technology with thinned wafers. In *Electron Devices Technology and Manufacturing Conference (EDTM)*, pages 37–40. IEEE, 2018.

[56] Dingyou Zhang, Daniel Smith, Gopal Kumarapuram, Rudy Giridharan, Shinichiro Kakita, Mohamed A Rabie, Peijie Feng, Holly Edmundson, and Luke England. Process development and optimization for 3μm high aspect ratio via-middle through-silicon vias at wafer level. *IEEE Transactions on Semiconductor Manufacturing*, 28(4):454–460, 2015.

[57] NanGate15: 15nm open cell library. `https://www.silvaco.com/products/nangate/FreePDK15_Open_Cell_Library/index.html`. (Accessed on 15/03/2019).

[58] P Batude, M Vinet, A Pouydebasque, C Le Royer, B Previtali, C Tabone, J-M Hartmann, L Sanchez, L Baud, V Carron, et al. Advances in 3D

CMOS sequential integration. In *International Electron Devices Meeting (IEDM)*, pages 1–4. IEEE, 2009.

[59] P Batude, L Brunet, C Fenouillet-Beranger, F Andrieu, J-P Colinge, D Lattard, E Vianello, S Thuries, O Billoint, P Vivet, et al. 3D sequential integration: Application-driven technological achievements and guidelines. In *International Electron Devices Meeting (IEDM)*, pages 1–3. IEEE, 2017.

[60] L Brunet, C Fenouillet-Beranger, P Batude, S Beaurepaire, F Ponthenier, N Rambal, V Mazzocchi, JB Pin, P Acosta-Alba, S Kerdiles, et al. Breakthroughs in 3D sequential technology. In *International Electron Devices Meeting (IEDM)*, pages 2–7. IEEE, 2018.

[61] Pascal Vivet, Sébastien Thuriès, Olivier Billoint, Sylvain Choisnet, Didier Lattard, Edith Beigné, and Perrine Batude. Monolithic 3D: an alternative to advanced CMOS scaling, technology perspectives and associated design methodology challenges. In *International Conference on Electronics, Circuits and Systems (ICECS)*, pages 157–160. IEEE, 2018.

[62] Sudeep Pasricha. A framework for TSV serialization-aware synthesis of application specific 3D networks-on-chip. In *International Conference on VLSI Design*, pages 268–273. IEEE, 2012.

[63] Ying Wang, Yin-He Han, Lei Zhang, Bin-Zhang Fu, Cheng Liu, Hua-Wei Li, and Xiaowei Li. Economizing TSV resources in 3-D network-on-chip design. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 23(3):493–506, 2014.

[64] Wen-Pin Tu, Yen-Hsin Lee, and Shih-Hsu Huang. TSV sharing through multiplexing for TSV count minimization in high-level synthesis. In *International SOC Conference (SOCC)*, pages 156–159. IEEE, 2011.

[65] Dae Hyun Kim and Sung Kyu Lim. Design quality trade-off studies for 3-D ICs built with sub-micron TSVs and future devices. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2(2):240–248, 2012.

[66] Electromagnetics — Electronic Simulation Software — ANSYS. `https://www.ansys.com/products/electronics`. (Accessed on 13/03/2019).

[67] Kaustav Banerjee, Sungjun Im, and Navin Srivastava. Interconnect modeling and analysis in the nanometer era: Cu and beyond. In *Advanced Metallization Conference (AMC)*, pages 1–7, 2005.

[68] Thorsten Steinmetz, Stefan Kurz, and Markus Clemens. Domains of validity of quasistatic and quasistationary field approximations. *COMPEL-The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 30(4):1237–1247, 2011.

[69] PTM — Interconnect. `http://ptm.asu.edu/interconnect.html`. (Accessed on 15/03/2019).

[70] William C Elmore. The transient response of damped linear networks with particular regard to wideband amplifiers. *AIP Journal of Applied Physics*, 19(1):55–63, 1948.

[71] Nidhi Chandoke and Ashish Kumar Sharma. A novel approach to estimate power consumption using SystemC transaction level modelling. In *Annual IEEE India Conference (INDICON)*, pages 1–6. IEEE, 2015.

[72] Vincenzo Catania, Andrea Mineo, Salvatore Monteleone, Maurizio Palesi, and Davide Patti. Cycle-accurate network on chip simulation with Noxim. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 27(1):4, 2016.

[73] Sandeep Saini. *Low power interconnect design*. Springer, 2015.

[74] Dae Hyun Kim and Sung Kyu Lim. Through-silicon-via-aware delay and power prediction model for buffered interconnects in 3D ICs. In *International Workshop on System Level Interconnect Prediction (SLIP)*, pages 25–32. ACM/IEEE, 2010.

[75] Taigon Song, Chang Liu, Yarui Peng, and Sung Kyu Lim. Full-chip signal integrity analysis and optimization of 3-D ICs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 24(5):1636–1648, 2015.

[76] Yarui Peng, Taigon Song, Dusan Petranovic, and Sung Kyu Lim. On accurate full-chip extraction and optimization of TSV-to-TSV coupling elements in 3D ICs. In *International Conference on Computer-Aided Design (ICCAD)*, pages 281–288. ACM/IEEE, 2013.

[77] Shyh-Chyi Wong, Gwo-Yann Lee, and Dye-Jyun Ma. Modeling of interconnect capacitance, delay, and crosstalk in VLSI. *IEEE Transactions on Semiconductor Manufacturing*, 13(1):108–111, 2000.

[78] Chunjie Duan, Anup Tirumala, and Sunil P Khatri. Analysis and avoidance of cross-talk in on-chip buses. In *Symposium on High Performance Interconnects (HOT Interconnects)*, pages 133–138. IEEE, 2001.

[79] Maurizio Palesi, Giuseppe Ascia, Fabrizio Fazzino, and Vincenzo Catania. Data encoding schemes in networks on chip. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 30(5):774–786, 2011.

[80] Nima Jafarzadeh, Maurizio Palesi, Ahmad Khademzadeh, and Ali Afzali-Kusha. Data encoding techniques for reducing energy consumption in network-on-chip. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 22(3):675–685, 2014.

[81] Kareem Ali, Eslam Yahya, Alaa Elrouby, and Yehea Ismail. Different scenarios for estimating coupling capacitances of through silicon via (TSV) arrays. In *International Conference on Energy Aware Computing Systems & Applications (ICEAC)*, pages 1–4. IEEE, 2015.

[82] Tapobrata Bandyopadhyay, Ki Jin Han, Daehyun Chung, Ritwik Chatterjee, Madhavan Swaminathan, and Rao Tummala. Rigorous electrical modeling of through silicon vias (TSVs) with MOS capacitance effects. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 1(6):893–903, 2011.

[83] Paul E Landman and Jan M Rabaey. Power estimation for high level synthesis. In *European Conference on Design Automation with the European Event in ASIC Design*, pages 361–366. IEEE, 1993.

[84] Paul E Landman and Jan M Rabaey. Architectural power analysis: The dual bit type method. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 3(2):173–187, 1995.

[85] Luca Benini, Giovanni De Micheli, Enrico Macii, Donatella Sciuto, and Cristina Silvano. Address bus encoding techniques for system-level power optimization. In *Design, Automation & Test in Europe Conference (DATE)*, pages 861–867. IEEE, 1998.

[86] Enric Musoll, Tomás Lang, and Jordi Cortadella. Working-zone encoding for reducing the energy in microprocessor address buses. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 6(4):568–572, 1998.

[87] Satoshi Komatsu and Masahiro Fujita. Irredundant address bus encoding techniques based on adaptive codebooks for low power. In *Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 9–14. ACM, 2003.

[88] Youngsoo Shin, Kiyoung Choi, and Young-Hoon Chang. Narrow bus encoding for low-power DSP systems. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 9(5):656–660, 2001.

[89] Alberto García-Ortiz, Leandro S Indrusiak, Tudor Murgan, and Manfred Glesner. Low-power coding for networks-on-chip with virtual channels. *ASP Journal of Low Power Electronics*, 5(1):77–84, 2009.

[90] Alberto García-Ortiz, Leandro S Indrusiak, Tudor Murgan, and Manfred Glesner. PMD: A low-power code for networks-on-chip based on virtual channels. In *International Workshop on Power and Timing Modeling, Optimization and Simulation*, pages 219–228. Springer, 2008.

[91] Peter Petrov and Alex Orailoglu. Low-power instruction bus encoding for embedded processors. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 12(8):812–826, 2004.

[92] Wei-Chung Cheng and Massoud Pedram. Power-optimal encoding for a DRAM address bus. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 10(2):109–118, 2002.

[93] Luca Benini, Giovanni De Micheli, Enrico Macii, Donatella Sciuto, and Cristina Silvano. Asymptotic zero-transition activity encoding for address busses in low-power microprocessor-based systems. In *Great Lakes Symposium on VLSI (GLSVLSI)*, pages 77–82. IEEE, 1997.

[94] Yazdan Aghaghiri, Farzan Fallah, and Massoud Pedram. Transition reduction in memory buses using sector-based encoding techniques. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(8):1164–1174, 2004.

[95] Youngsoo Shin, Soo-Ik Chae, and Kiyoung Choi. Partial bus-invert coding for power optimization of application-specific systems. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 9(2):377–383, 2001.

[96] Sumant Ramprasad, Naresh R Shanbhag, and Ibrahim N Hajj. A coding framework for low-power address and data busses. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 7(2):212–221, 1999.

[97] Enric Musoll, Tomás Lang, and Jordi Cortadella. Exploiting the locality of memory references to reduce the address bus energy. In *International Symposium on Low Power Electronics and Design (ISLPED)*, pages 202–207. IEEE, 1997.

[98] Sudeep Pasricha and Nikil Dutt. *On-chip communication architectures: system on chip interconnect*. Morgan Kaufmann, 2010.

[99] Alberto García-Ortiz, Daniel Gregorek, and Christof Osewold. Analysis of bus-invert coding in the presence of correlations. In *Saudi International Electronics, Communications and Photonics Conference (SIECPC)*, pages 1–5. IEEE, 2011.

[100] Sumant Ramprasad, Naresh R Shanbhag, and Ibrahim N Hajj. Analytical estimation of transition activity from word-level signal statistics. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 16(7):718–733, 1997.

[101] S Bobba, IN Hajj, and NR Shanbhag. Analytical expressions for average bit statistics of signal lines in DSP architectures. In *International Symposium on Circuits and Systems (ISCAS)*, volume 6, pages 33–36. IEEE, 1998.

[102] Janardhan H Satyanarayana and Keshab K Parhi. Theoretical analysis of word-level switching activity in the presence of glitching and correlation. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 8(2):148–159, 2000.

[103] Asghar Havashki, Lars Lundheim, Per Gunnar Kjeldsberg, Oscar Gustafsson, and Geir E Oien. Analysis of switching activity in DSP signals in the presence of noise. In *IEEE EUROCON*, pages 234–239. IEEE, 2009.

[104] Alberto Garcia-Ortiz and Leandro S Indrusiak. Practical and theoretical considerations on low-power probability-codes for networks-on-chip. In *International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, pages 160–169. Springer, 2010.

[105] Alberto García Ortiz. *Stochastic Data Models for Power Estimation at High-Levels of Abstraction*. Shaker, 2004.

[106] Mostafa Said, Farhad Mehdipour, and Mohamed El-Sayed. Improving performance and fabrication metrics of three-dimensional ICs by multiplexing through-silicon vias. In *Euromicro Conference on Digital System Design (DSD)*, pages 925–932. IEEE, 2013.

[107] William J Dally. Virtual-channel flow control. *IEEE Transactions on Parallel & Distributed Systems*, 3(2):194–205, 1992.

[108] Henry Stark and John W Woods. *Probability, random processes, and estimation theory for engineers*. Prentice-Hall, Inc., 1986.

[109] David Harris and Sarah Harris. *Digital design and computer architecture*. Morgan Kaufmann, 2010.

[110] Nikolay Kavaldjiev, Gerard JM Smit, and Pierre G Jansen. A virtual channel router for on-chip networks. In *International SOC Conference (SOCC)*, pages 289–293. IEEE, 2004.

[111] Evgeny Bolotin, Israel Cidon, Ran Ginosar, and Avinoam Kolodny. QNoC: QoS architecture and design process for network on chip. *Elsevier Journal of Systems Architecture*, 50(2-3):105–128, 2004.

[112] Mircea R Stan and Wayne P Burleson. Bus-invert coding for low-power I/O. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 3(1):49–58, 1995.

[113] Fadoua Guezzi-Messaoud, Antoine Dupret, Arnaud Peizerat, and Yves Blanchard. Benefits of three-dimensional circuit stacking for image sensors. In *International New Circuits and Systems Conference (NEWCAS)*, pages 1–4. IEEE, 2013.

[114] Gray Frank. Pulse code communication, March 17 1953. US Patent 2,632,058.

[115] Richard A Brualdi. *Combinatorial matrix classes*, volume 13. Cambridge University Press, 2006.

[116] Peter JM Van Laarhoven and Emile HL Aarts. Simulated annealing. In *Simulated Annealing: Theory and Applications*, pages 7–15. Springer, 1987.

[117] Frank Niklaus and Andreas C Fischer. Heterogeneous 3D integration of MOEMS and ICs. In *International Conference on Optical MEMS and Nanophotonics (OMN)*, pages 1–2. IEEE, 2016.

[118] Sang-Yong Lee and Antonio Ortega. A novel approach of image compression in digital cameras with a Bayer color filter array. In *International Conference on Image Processing (ICIP)*, volume 3, pages 482–485. IEEE, 2001.

[119] Boris Vaisband and Eby G Friedman. Hexagonal TSV bundle topology for 3-D ICs. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 64(1):11–15, 2016.

[120] Chunjie Duan, Chengyu Zhu, and Sunil P Khatri. Forbidden transition free crosstalk avoidance CODEC design. In *Design Automation Conference (DAC)*, pages 986–991. ACM/IEEE, 2008.

[121] Chittarsu Raghunandan, KS Sainarayanan, and MB Srinivas. Process variation aware bus-coding scheme for delay minimization in VLSI interconnects. In *International Symposium on Quality Electronic Design (ISQED)*, pages 43–46. IEEE, 2008.

[122] Bret Victor and Kurt Keutzer. Bus encoding to prevent crosstalk delay. In *International Conference on Computer-Aided Design (ICCAD)*, pages 57–63. ACM/IEEE, 2001.

[123] Kan Xu and Eby G Friedman. Scaling trends of power noise in 3-D ICs. *Elsevier Integration*, 51:139–148, 2015.

[124] Nima Jafarzadeh, Maurizio Palesi, Saeedeh Eskandari, Shaahin Hessabi, and Ali Afzali-Kusha. Low energy yet reliable data communication scheme for network-on-chip. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 34(12):1892–1904, 2015.

[125] Christophe Erdmann, Donnacha Lowney, Adrian Lynam, Aidan Keady, John McGrath, Edward Cullen, Daire Breathnach, Denis Keane, Patrick Lynch, Marites De La Torre, et al. A heterogeneous 3D-IC consisting of two 28 nm FPGA die and 32 reconfigurable high-performance data converters. *IEEE Journal of Solid-State Circuits*, 50(1):258–269, 2015.

[126] Darren Anand, Bruce Cowan, Owen Farnsworth, Peter Jakobsen, Steve Oakland, Michael R Ouellette, and Donald L Wheater. An on-chip self-repair calculation and fusing methodology. *IEEE Design & Test of Computers*, 20(5):67–75, 2003.

[127] Yu-Jen Huang and Jin-Fu Li. Built-in self-repair scheme for the TSVs in 3-D ICs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 31(10):1600–1613, 2012.

[128] Richard W Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160, 1950.

[129] Matheos Lampropoulos, Bashir M Al-Hashimi, and Paul Rosinger. Minimization of crosstalk noise, delay and power using a modified bus invert technique. In *Design, Automation & Test in Europe Conference (DATE)*, volume 2, pages 1372–1373. IEEE, 2004.

[130] Jeffrey P Gambino, Shawn A Adderly, and John U Knickerbocker. An overview of through-silicon-via technology and manufacturing challenges. *Elsevier Microelectronic Engineering*, 135:73–106, 2015.

[131] NanGate45: 45nm open cell library. `https://www.silvaco.com/products/nangate/FreePDK45_Open_Cell_Library`. (Accessed on 20/10/2019).

[132] DeepSig Inc. `https://www.deepsig.io/datasets`. (Accessed on 18/02/2019).

Appendices

APPENDIX A

Notation

In this part of the appendices, the notation used throughout this thesis is summarized. The following notation is used for vectors and matrices. A vector is represented by a non-bold italic serif accented by a right arrow, as in $\vec{v}$. The i^{th} element of a vector $\vec{v}$ is denoted as v_i. Matrices are symbolized using bold upper-case letters, such as $\mathbf{M}$, and written in box brackets:

$$\mathbf{M} = \begin{bmatrix} M_{1,2} & M_{1,2} & \dots & M_{1,n} \\ M_{2,1} & M_{2,2} & \dots & M_{12,n} \\ \vdots & \vdots & \ddots & \vdots \\ M_{m,1} & M_{m,2} & \dots & M_{m,n} \end{bmatrix}. \tag{A.1}$$

Hence, the matrix entry at i^{th}-row and the j^{th}-column position of a matrix $\mathbf{M}$ is symbolized as $M_{i,j}$.

Physical quantities that occur throughout this thesis in a time-discrete (or even constant) as well as in a time-continuous manner are symbolized by upper-case and lower-case letters, respectively. For example, the constant power-supply level of the analyzed standard-cell library is symbolized as V_{dd}, while the time-continuous voltage between node i and the ground is symbolized as $v_i(t)$. A time-continuous variable is indicated by a t (symbol for time) in parentheses, such as in $v_i(t)$. In contrast, if the voltage is a discrete-sampled variable, it is denoted as $V_i[k]$.

Always a sampling with the system's clock frequency, represented by f_{clk}, is considered in this work for a time-discrete variable. If not stated otherwise, for each variable $x_i[k]$, the sampling times are $k \cdot T_{\text{clk}}$ with $k \in \mathbb{N}$ and T_{clk} equal to $1/f_{\text{clk}}$. An exception is made for the consideration of temporal misalign-

ment between the signal edges on the transmitted binary signals, $b_i[k]$. Here, constant skews, symbolized as $T_{\mathrm{edge},i}$, are added to the sampling times, which consequently are $k \cdot T_{\mathrm{clk}} + T_{\mathrm{edge},i}$, where i is the index of the variable. Hence, for misaligned signal edges, the input value for the i^{th} interconnect changes its logical value always $T_{\mathrm{edge},i}$ after the rising clock edge in the respective cycle at $k \cdot T_{\mathrm{clk}}$.

Although it is common to index the bits of an n-bit binary word from 0 to $n-1$ (starting at the least-significant bit with index 0), throughout this thesis, the indices range from 1 to n. Thus, the least-significant bit of a binary word $\vec{b}$ is symbolized as b_1, and the most-significant bit as b_n. The reason is the extensive use of matrix and vector notations in nearly all parts of this thesis, in which binary words are represented by vectors, while vectors and matrices in mathematics generally start at an index value of 1.

Multi-bit binary patterns are denoted throughout this thesis in quotes to distinguish them from decimal numbers, starting with the most-significant bit at the left-hand side. Hence, “1010” represents the decimal value ten, assuming that the binary pattern is interpreted as a 4-bit unsigned integer word.

APPENDIX B

Pseudo Codes

Algorithm 1: Generate the capacitance matrix of a metal-wire bus.

Input: Number of lines n;
Self-capacitance value $C_{\mathrm{mw,g}}$;
Coupling-capacitance value $C_{\mathrm{mw,c}}$.

Output: $n \times n$ capacitance matrix $\mathbf{C}$.

$\mathbf{C} := \mathrm{zeros}\,(n, n)$ // initialize matrix with zeros

$\mathrm{diag}\,(\mathbf{C}) := C_{\mathrm{mw,g}}$ // set ground capacitances on the matrix diagonal

for $i = 1$ *to* $n - 1$ **do**

  $\mathbf{C}_{i,(i+1)} := C_{\mathrm{mw,c}}$ // set coupling capacitance

  $\mathbf{C}_{(i+1),i} := C_{\mathrm{mw,c}}$

return C

Algorithm 2: Generate the capacitance matrix of a TSV array according to the previously used capacitance model.

Input: Array dimensions M, N;
Coupling-capacitance values $C_{\mathrm{n,prev}}$, $C_{\mathrm{d,prev}}$.
Output: $(M{\cdot}N) \times (M{\cdot}N)$ capacitance matrix $\mathbf{C}$.

$\mathbf{C} := \mathrm{zeros}\,(M{\cdot}N, M{\cdot}N)$ // initialize matrix with zeros

/* get matrix $\mathbf{D}$ with distance (normalized by d_{min}) between TSV$_i$ and TSV$_j$ on entry (i,j). Thus, $D_{i,j}$ is equal to 1 and $\sqrt{2}$ for direct and diagonally adjacent TSVs, respectively. Function ''distances()'' depends on how TSVs are indexed (*e.g.*, column-by-row) */

$\mathbf{D} := \mathrm{distances}\,(M, N)$
for $i = 1$ *to* $M \cdot N$ **do**
  for $j = 1$ *to* $M \cdot N$ **do**
    if $D_{i,j} = 1$ **then**
      $\mathbf{C}_{i,j} := C_{\mathrm{n,prev}}$ // directly adjacent TSV pair
    else if $D_{i,j} = \sqrt{2}$ **then**
      $\mathbf{C}_{i,j} := C_{\mathrm{d,prev}}$ // diagonally adjacent TSV pair
return C

Algorithm 3: Generate the capacitance matrix of a TSV array according to the capacitance model proposed in this thesis.

Input: Array dimensions M, N;
Capacitance values $C_{\mathrm{G,n}}$, $C_{\mathrm{G,d}}$, $C_{\mathrm{G,e0}}$, $C_{\mathrm{G,e1}}$, $C_{\mathrm{G,e2}}$, $C_{\mathrm{G,c0}}$, $C_{\mathrm{G,c1}}$, $C_{\mathrm{G,c2}}$;
Capacitance deviations ΔC_{n}, ΔC_{d}, ΔC_{e0}, ΔC_{e1}, ΔC_{e2}, ΔC_{c0}, ΔC_{c1}, ΔC_{c2}.

Output: $(M{\cdot}N) \times (M{\cdot}N)$ capacitance matrices $\mathbf{C}_{\mathrm{G}}$, $\mathbf{\Delta C}$.

$\mathbf{C}_{\mathrm{G}} := \mathbf{\Delta C} := \mathrm{zeros}\,(M{\cdot}N, M{\cdot}N)$ // initialize matrix with zeros
$\mathbf{D} := \mathrm{distances}\,(M, N)$ // get normalized TSV distances
$\mathbb{S}_{\mathrm{E}} := \mathrm{edge\_tsv\_set}(M, N)$ // set of all TSVs located at a single edge
$\mathbb{S}_{\mathrm{C}} := \mathrm{corner\_tsv\_set}(M, N)$ // set of the four corner TSVs
for $i = 1$ *to* $M \cdot N$ **do**
  if $i \in \mathbb{S}_E$ **then**
    $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,e0}}, \Delta C_{\mathrm{e0}}]$ // edge TSV self-capacitance
  else if $i \in \mathbb{S}_C$ **then**
    $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,c0}}, \Delta C_{\mathrm{c0}}]$ // corner TSV self-capacitance
  for $j = 1$ *to* $M \cdot N$ **do**
    if $D_{i,j} = 1$ **then**
      $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,n}}, \Delta C_{\mathrm{n}}]$ // directly adjacent TSV pair
      if $i \in \mathbb{S}_E$ *and* $j \in \mathbb{S}_E$ **then**
        $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,e1}}, \Delta C_{\mathrm{e1}}]$ // located at an edge
      else if $i \in \mathbb{S}_C$ *or* $j \in \mathbb{S}_C$ **then**
        $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,c1}}, \Delta C_{\mathrm{c1}}]$ // located at a corner
    else if $D_{i,j} = \sqrt{2}$ **then**
      $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,d}}, \Delta C_{\mathrm{d}}]$ // diagonally adjacent TSV pair
    else if $D_{i,j} = 2$ **then**
      /* indirectly adjacent TSV pair */
      if $i \in \mathbb{S}_E$ *and* $j \in \mathbb{S}_E$ **then**
        $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,e2}}, \Delta C_{\mathrm{e2}}]$ // located at an edge
      else if $i \in \mathbb{S}_C$ *or* $j \in \mathbb{S}_C$ **then**
        $[C_{\mathrm{G},i,i}, \Delta C_{i,i}] := [C_{\mathrm{G,c2}}, \Delta C_{\mathrm{c2}}]$ // located at a corner
return $\mathbf{C}_G, \Delta\mathbf{C}$

APPENDIX C

Method to Calculate the Depletion-Region Widths

The applied method to determine the width of the depletion regions for the parameterisable 3D model of a TSV array is outlined in this part of the appendices. An in-depth derivation of the formulas can be found in [11].

Since a TSV, its isolating oxide liner, and the conductive substrate form a MOS junction, a TSV oxide is surrounded by a depletion region. A depletion region is an area where the normally conductive substrate has a lack of free charge carriers, resulting in a silicon area with nearly zero conductivity. The width of the depletion region depends on the mean voltage on the TSV compared to the substrate (which is grounded for common p-doped substrates). Hence, the width of the depletion region can differ for the individual TSVs in an array arrangement. The method presented in the following is applied to determine the width of the depletion region of each TSV in the 3D model individually.

Since a TSV has a relatively large length-over-diameter ratio, a 2D approximation of the electrostatics can be used. Consequently, Poison's equation for the electrical field surrounding the TSV in a p-doped substrate can be expressed as

$$\frac{1}{r}\frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{\mathrm{d}\psi}{\mathrm{d}r}\right) = \begin{cases} \dfrac{qN_\mathrm{a}}{\epsilon_\mathrm{subs}} & \text{for } r_\mathrm{tsv} + t_\mathrm{ox} < r < r_\mathrm{tsv} + t_\mathrm{ox} + w_\mathrm{dep} \\ 0 & \text{for } r_\mathrm{tsv} < r < r_\mathrm{tsv} + t_\mathrm{ox}, \end{cases} \tag{C.1}$$

where w_dep is the width of the depletion region surrounding the TSV which has to be determined; $\psi(r)$ is the electrical potential at radial distance r from the center of the TSV conductor; N_a is the acceptor doping concentration in the substrate; q is the elementary charge; and ϵ_subs is the permittivity of the substrate.

The boundary conditions for the Poison equation are:

$$\begin{aligned}
&\psi(r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}}) = 0; \\
&\psi(r_{\mathrm{tsv}}) = \bar{V} + \phi_{\mathrm{MS}}; \\
&\left.\frac{\mathrm{d}\psi}{\mathrm{d}r}\right|_{r=r_{\mathrm{tsv}}+t_{\mathrm{ox}}+w_{\mathrm{dep}}} = 0; \\
&\epsilon_{\mathrm{subs}} \left.\frac{\mathrm{d}\psi}{\mathrm{d}r}\right|_{r=r_{\mathrm{tsv}}+t_{\mathrm{ox}}+\mathrm{d}r} = \epsilon_{\mathrm{ox}} \left.\frac{\mathrm{d}\psi}{\mathrm{d}r}\right|_{r=r_{\mathrm{tsv}}+t_{\mathrm{ox}}-\mathrm{d}r}.
\end{aligned} \tag{C.2}$$

In the boundary conditions, $\bar{V}$ is the mean voltage on the TSV, ϵ_{ox} is the permittivity of the TSV oxide, and ϕ_{MS} is the work function difference between the TSV metal (*i.e.*, copper) and the doped silicon substrate:

$$\phi_{MS} = \phi_{\mathrm{metal}} - \phi_{\mathrm{subs}}. \tag{C.3}$$

Furthermore, $r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}} + \mathrm{d}r$ and $r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}} + \mathrm{d}r$ indicate the conductive-substrate and the depletion-region side of the boundary between them (infinitesimal thin), respectively. By solving the Poison equation under consideration of the boundary conditions, w_{dep} can be determined. For the weak-inversion/depletion, where a negligible amount of charge exists at the oxide-silicon interface (a strong inversion does not occur due to the relatively low substrate conductivity), the total charge is expressed as

$$\begin{aligned}
&\pi \left((r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}})^2 - (r_{\mathrm{tsv}} + t_{\mathrm{ox}})^2 \right) \cdot q N_{\mathrm{a}} \\
&\qquad = \left(\bar{V} - \phi_{MS} - \psi(r_{\mathrm{tsv}} + t_{\mathrm{ox}}) \right) \cdot \frac{2\pi\epsilon_{\mathrm{ox}}}{\log_e \left(\dfrac{r_{\mathrm{tsv}} + t_{\mathrm{ox}}}{r_{\mathrm{tsv}}} \right)}.
\end{aligned} \tag{C.4}$$

This equation represents Gauss's law. The scalar potential at the interface can be expressed by integrating Poison's equation:

$$\begin{aligned}
\psi(r_{\mathrm{tsv}} + t_{\mathrm{ox}}) = \frac{q N_{\mathrm{a}}}{2\epsilon_{\mathrm{subs}}} \cdot \Bigg(&(r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}})^2 \cdot \log_e \left(\frac{r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}}}{r_{\mathrm{tsv}} + t_{\mathrm{ox}}} \right) \\
&- 0.5 w_{\mathrm{dep}}^2 - w_{\mathrm{dep}}(r_{\mathrm{tsv}} + t_{\mathrm{ox}}) \Bigg).
\end{aligned} \tag{C.5}$$

Two unknown variables exist in Equation (C.4)–(C.5): w_{dep} and $\psi(r_{\mathrm{tsv}} + t_{\mathrm{ox}})$. Therefore, the two equations can be solved self-consistently to determine w_{dep} for any given mean TSV voltage, $\bar{V}$.

Solving the equations shows that w_{dep} increases with the mean TSV voltage for p-doped substrates. However, a depletion-region width does not increase

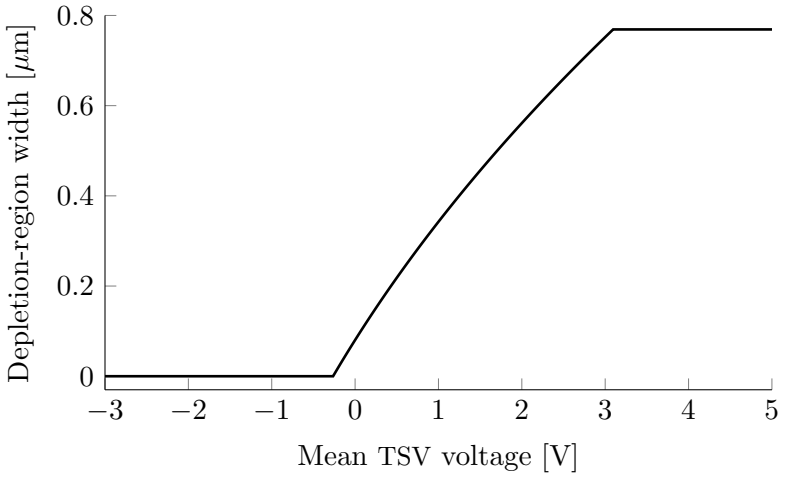


Fig. C.1: Depletion-region width over the mean TSV voltage for a grounded p-doped substrate. Thereby, the TSV parameters are: $r_{\mathrm{tsv}} = 2.5\,\mu\mathrm{m}$, $t_{\mathrm{ox}} = 0.5\,\mu\mathrm{m}$, $\phi_{\mathrm{metal}} = 4.6\,\mathrm{eV}$ (Copper TSVs), $\phi_{\mathrm{subs}} = 4.89\,\mathrm{eV}$, $\epsilon_{\mathrm{ox}} = 3.9\epsilon_0$, $\epsilon_{\mathrm{subs}} = 11.9\epsilon_0$, and $N_{\mathrm{a}} = 1.35 \times 10^{15}\,\mathrm{cm}^{-3}$ (*i.e.*, $\sigma_{\mathrm{subs}} = 10\,{}^{\mathrm{S}}/_{\mathrm{m}}$).

infinitely with the mean voltage, as it is limited to a maximum value. In the case where the maximum depletion-region width is reached, the electrical potential drop across the depletion region is

$$\frac{2k_{\mathrm{B}}T}{q} \cdot \log_e\left(\frac{N_{\mathrm{a}}}{n_i}\right) = \frac{qN_{\mathrm{a}}}{2\epsilon_{\mathrm{subs}}}\Bigg(-0.5w_{\mathrm{dep}}^2 - w_{\mathrm{dep}}(r_{\mathrm{tsv}} + t_{\mathrm{ox}}) + (r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}})^2 \cdot \log_e\left(\frac{r_{\mathrm{tsv}} + t_{\mathrm{ox}} + w_{\mathrm{dep}}}{r_{\mathrm{tsv}} + t_{\mathrm{ox}}}\right)\Bigg), \quad \text{(C.6)}$$

where n_i is the intrinsic carrier concentration of silicon, k_{B} is the Boltzmann constant, and T is the temperature (assumed to be 300 K). Equation (C.6) is derived from Equation (C.5) for a potential $\psi(r_{\mathrm{tsv}} + t_{\mathrm{ox}})$ that is fixed to $2\phi_F = {}^{2k_BT}/_q \cdot \log_e\left({}^{N_{\mathrm{a}}}/_{n_i}\right)$.

The width of the depletion region, determined by means of the derived methodology, over the mean TSV voltage, for a TSV radius of $2.5\,\mu\mathrm{m}$, is exemplarily shown in Figure C.1.

The same concept can be used to derive a method to determine the depletion-region widths for n-doped substrates, as shown in [11]. While the width of a depletion region increases with the mean voltage on the related TSV conductor for common p-doped substrates, it is simply *vice versa* for n-doped substrates (*i.e.*, the width decrease with an increased mean voltage).

APPENDIX D

Curriculum Vitae of the Author

NAME:	Lennart Bamberg
MAIL:	bamberg@uni-bremen.de
WEB:	www.ids.item.uni-bremen.de
ORCID:	0000–0003–4673–8310
DATE OF BIRTH:	5th of February, 1992

Work Experience

ITEM, University of Bremen **06/2016–11/2019**
Role: *Research and Teaching Associate*
Main project: *Technology-Aware Asymmetric 3D Interconnect Architectures: Templates and Design Methods* (funded by the DFG)

GTCAD, Georgia Tech **02/2019–07/2019**
Role: *Invited Research Scholar* (funded by a scholarship of the BMBF and the DAAD)
Project: *Design of a 3D-Integrated RISC-V Multi-Core System Employing STT MRAM* (cooperation with IMEC, Belgium)

GrAi Matter Labs **11/2019–Present**
Current Role: *Senior Computer Architect*

Education

Pankratiusschule Möhnesee **08/1998–06/2002**
Elementary school

Aldegrever Gymnasium **08/2002–05/2011**
Abitur

University of Bremen **10/2011–05/2016**
B.Sc. & M.Sc. in *Electrical and Computer Engineering*
Overall grade: *Passed with Distinction (1.1)*

University of Bremen **10/2016–11/2019**
Ph.D. in *Electrical and Computer Engineering*
Overall grade: *Summa Cum Laude (0.7)*

Awards

Best Paper Award, PATMOS 2017
27th IEEE International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS), Thessaloniki, Greece

Best Paper Award, PATMOS 2018
28th IEEE International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS), Platja d'Aro, Spain

List of Figures

List of Tables