

Edge effect aware low-power crosstalk avoidance technique for 3D integration

Lennart Bamberg^{a,1}, Amir Najafi^a, Alberto García-Ortiz^a

^a*Institute of Electrodynamics and Microelectronics (ITEM.ids), University of Bremen, Otto-Hahn-Allee NW 1, 28359 Bremen, Germany*

Abstract

Metal wires and through silicon vias (TSVs) are frequently performance bottlenecks of 3D ICs due to their high capacitive crosstalk which can be reduced using coding techniques. In this work we show that existing TSV crosstalk avoidance codes (CACs) are impractical for real applications due to the edge effects in TSV bundles. Additionally, these 3D CACs do not reduce the metal wire crosstalk and dramatically increase the power consumption of 2D and 3D interconnects. This work presents a 3D CAC which overcomes previous limitations. The method is based on an intelligent fixed mapping of the bits of existing 2D CACs onto rectangular or hexagonal TSV arrangements. Simulation results, obtained by circuit simulations in combination with an electromagnetic field solver, show that existing 3D CACs only reduce the TSV crosstalk by a maximum of 9.4%, provide no optimization of the metal wire crosstalk and induce an increase in the interconnect power consumption by about 50%. In contrast, the presented technique requires less hardware and reduces the maximum crosstalk of modern TSV and metal wire buses by 37.8% and 47.6%, respectively, while leaving their power consumption almost unaffected. Alternatively, our technique can reduce the TSV and metal wire crosstalk peaks by 20.3% and 47.7%, respectively, while additionally providing a reduction in the TSV and metal wire power consumption by 5.3% and 21.9%, respectively.

Keywords: Through silicon via, crosstalk, coding, high performance, low-power, 3D integration

1. Introduction

3D integration is one of the promising solutions to overcome the challenges that arise with the limit of Moore's law. As the interconnect structure between the dies of a 3D system on chip (3D SoC), through silicon vias (TSVs) are typically used as they yield to a high reliability [1]. TSVs are usually bundled together, rather than used in isolation [2]. By using regular TSV bundles, it is possible to generate wide I/O 3D components, such as stacked DRAM cells [3].

However, TSV bundles, as well as metal wires, suffer from crosstalk which is a threat to the delay, the power consumption and the signal integrity [4]. In recent years, crosstalk became the critical design issue for the traditional planar metal wire interconnects, due to their limited scaling [5]. Previous research shows that the crosstalk problem is not alleviated for TSV interconnects due to the relatively large TSV dimensions and the increased number of aggressors compared to the traditional planar metal wires [6, 7]. Thus, crosstalk is still an important design concern for 3D integrated circuit (3D IC) design and consequently caught the attention of academic as well as industrial experts (e.g. [6–18]). Most previous works deal with the theoretical analysis of crosstalk models [6–15], with some also proposing manufacturing techniques to suppress TSV

crosstalk noise [12–15]. On the downside, these manufacturing techniques significantly increase the production costs and further impair the already critical TSV manufacturing yield [19].

Thus, since crosstalk is a pattern dependent phenomena [4], data encoding approaches have been recently proposed which reduce the maximum TSV crosstalk without affecting the manufacturing process [16–18]. These crosstalk avoidance coding (CAC) approaches keep the crosstalk of each TSV in the middle of an array below a certain level by avoiding critical transitions. None of the existing methods ever analyzed the crosstalk of the TSVs which are located at the edges of an array. Previous works claim that, due to the reduced number of adjacent aggressors, the crosstalk of these edge TSVs is significantly lower than the crosstalk of the middle TSVs. However, this assumption is wrong. Due to the edge effects, the coupling between two edge TSVs is stronger than in the middle of the array [20]. Hence, the overall crosstalk of edge TSVs is only slightly lower than the crosstalk of middle TSVs. This fact heavily reduces the coding efficiency of existing 3D CACs. For modern TSV arrays, the actual crosstalk reductions of the CACs are less than 50% of the previously reported values, as shown in this work. The lower coding gains in combination with their high overhead costs make existing 3D CACs impractical for real applications. Thus, an efficient coding method needs to reduce the crosstalk of the middle and the edge TSVs simultaneously. The second limitation of existing 3D CACs is that they only aim to reduce the max-

*Corresponding author

Email addresses: bamberg@item.uni-bremen.de (Lennart Bamberg), amir.najafi@item.uni-bremen.de (Amir Najafi), agarcia@item.uni-bremen.de (Alberto García-Ortiz)

imum crosstalk of rectangular TSV arrays, even though the placement of TSVs within a hexagonal grid shows lower area requirements [21]. Furthermore, metal wires are not absent in 3D integration and their crosstalk is not negligible. Hence, an efficient technique should reduce the maximum crosstalk of metal wires and TSV simultaneously. Additionally, the high overhead costs of existing 3D CACs lead to a drastic increase in the interconnects power consumption by up to 50% [5]. This often strictly forbids the usage of the techniques as the advent of nanometric technologies has increased the criticality of the interconnect power consumption [5].

Existing 2D CACs have been shown to effectively reduce the maximum crosstalk and the power consumption of traditional metal wires [4]. Thus, to overcome all limitations, we focus on 2D CACs and present a solution to make them suitable for arbitrary TSV arrangements by an edge effect aware bit-to-TSV mapping. Thereby, the CACs retain their efficiency for the planar metal wires. The optimal bit-to-TSV mapping is determined by means of a cost function. Hereby, the mapping constraints can be fine-tuned to solely optimize the maximum TSV crosstalk (interconnect delay and noise) or the maximum TSV crosstalk and the TSV power consumption simultaneously.

We first presented the idea of an edge effect aware 3D CAC technique, suitable for TSVs as well as metal wires, in [22]. However, the present work provides major extensions. While the previous work is restricted to TSV arrays, this work additionally includes a modeling approach, an analysis and an optimization technique for hexagonal TSV arrangements. Despite its importance, a power analysis as well as a technique to optimize the power consumption beside the maximum crosstalk are also not included in the previous work. Moreover, for this work, we strengthened the mathematical formulations.

Analyses for modern TSV arrangements show that for all employed 2D CACs our method outperforms all existing 3D CACs. For example, for an underlying FTF 2D CAC [4], compared to the latest presented 3D CAC [18], we measured an improvement in the maximum TSV crosstalk/delay reduction by a factor of $2.95 \times$ and a simultaneous decrease in the TSV power consumption by 24.7%. Furthermore, our technique requires a 12.0% lower bit overhead and a 42.8% lower circuit area. Additionally it decreases the power consumption and delay of modern metal wire buses by 21.9% and 47.6%, respectively.

The remainder of this paper is organized as follows. In Section 2 an edge effect aware crosstalk classification is presented for mesh and hexagonal TSV topologies. The limitations of previous 3D CACs are precisely outlined in Section 3. Our proposed technique, which overcomes those limitations, is presented in Section 4. An evaluation of our technique which analyzes the maximum crosstalk reduction and also clarifies the trade-off between crosstalk peak and power reduction is presented in Section 6. The subsequent Section 7 includes simulation results. Finally, this paper is concluded in Section 8.

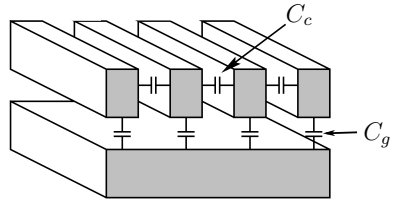


Fig. 1: Capacitances in a metal wire bus

2. Crosstalk classification including edge effects

For typical digital signals, capacitive coupling is the dominant crosstalk source of planar metal wires and vertical TSVs and dominates over inductive crosstalk [4, 16–18, 23]. This observation is again validated by the final analysis of our work (Section 7), which considers inductive effects. Consequently, only capacitive coupling is considered in the crosstalk classification presented in this paper.

The crosstalk of an interconnect i in clock cycle k is quantified by the effective capacitance to be reloaded [4]:

$$C_{eff,i}[k] = \Delta b_i^2[k]C_{i,0} + \sum_j \delta_{i,j}[k]C_{i,j}. \quad (1)$$

Here $C_{i,0}$ is the self/ground-capacitance of interconnect i and $C_{i,j}$ is the coupling-capacitance between the interconnects i and j . Δb_i determines the self switching of interconnect i :

$$\Delta b_i[k] = b_i[k] - b_i[k-1], \quad (2)$$

which is either -1 , 0 or 1 . Here, $b_i[k]$ is the binary value of interconnect i for the k^{th} clock cycle. $\delta_{i,j}$ determines the crosstalk switching between the interconnects i and j :

$$\delta_{i,j}[k] = \Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k]. \quad (3)$$

$\delta_{i,j}$ is equal to 2, if reverse signal transitions occur on the interconnects i and j (e.g. i switches from logical 0 to 1, while j switches from logical 1 to 0); if only interconnect i switches, $\delta_{i,j}$ is equal to 1, otherwise it is 0.

In the following we present simple, scalable and universal capacitance models for metal wires and TSVs. The combination of these models with Eq. 1 results in the crosstalk classification. The model for the metal wires is depicted in Fig. 1. Between every adjacent metal wire pair exists a coupling-capacitance C_c and every metal wire has a capacitance to the grounded substrate contacts C_g [5]. The actual ground-capacitances of a metal wire is equal to the sum of C_g and the interconnects load-capacitance. However, the load-capacitance is negligibly low in modern on-chip interconnects [5]. Combining this capacitance model with Eq. 1 results in a range for the effective capacitance of a metal wire of $[C_g, C_g+4C_c]$. Usually the ground-capacitances are neglected and the metal wire crosstalk is quantified in five

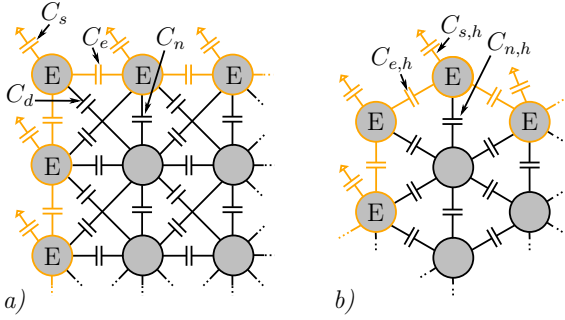


Fig. 2: Capacitances in TSV arrangements: a) array/mesh topology; b) hexagonal topology. Edge TSVs are marked **E**

classes as $0C_c$ to $4C_c$, since the self-capacitances are magnitudes smaller than the coupling-capacitances in modern metal wire buses [4].

Let us now discuss the capacitances inside TSV bundles. While previous 3D CACs are only applicable for rectangular TSV arrays, this work presents a technique which can be generally used for any TSV topology. However, mainly two different TSV topologies exist. Besides the well established rectangular TSV arrays (mesh topology), hexagonal TSV bundles have been recently proposed [21]. While hexagonal arrangements have the advantage of a reduced area per TSV, array structures are more regular and thus simplify the manufacturing as well as the place & route process. Here, we exemplarily investigate our approach for both arrangements. Thus, for hexagonal as well as mesh structures a capacitance model is required. As shown in Ref. [20], the TSV array capacitance model used for the derivation of previous 3D CACs leads to impractical CACs, due to the disregarded edge effects. Also, the capacitance model for the hexagonal arrangement, stated in Ref. [21], neglects these effects. Thus, we present new simple capacitance models for hexagonal and rectangular TSV bundles, which consider these effects.

The capacitance model for rectangular TSV arrays is depicted in Fig. 2.a. Due to the Faraday cage effect, only TSVs at the edges of an array have a significant ground-capacitance due to the grounded substrate contacts. Although the ground-capacitances of the TSVs at the four corners of the array are slightly bigger [20], for the sake of simplicity, we use one capacitance value (C_s) to describe the substrate-capacitances of all edge TSVs. Between any adjacent TSV pair exists a coupling-capacitance. Because of the different distances, a capacitance between diagonal adjacent TSVs C_d is several (3-5 [20]) times smaller than a capacitance between orthogonal adjacent TSVs. In the middle of the array, the capacitances between orthogonal adjacent TSVs are equal to C_n . Due to the E-field sharing effect, the capacitance between two orthogonal adjacent edge TSVs C_e is significantly bigger than C_n (ca. 30-45% [20]). Summarized, the effective capacitance of a middle TSV is in the range of 0 to $8C_n + 8C_d$, while the effective capacitance of an edge TSV is in the range of C_s to $C_s + 4C_e + 2C_n + 4C_d$. For the crosstalk classification,

the capacitance value between two orthogonal adjacent middle TSVs (C_n) is used as the reference value, which is subsequently referred to as C_{3D} . This results in $9 \cdot 9 = 81$ crosstalk classes for middle TSVs ($0C_{3D}$ to $(8+8\lambda_d)C_{3D}$), and $3 \cdot 5 \cdot 5 = 75$ crosstalk classes for edge TSVs ($\lambda_s C_{3D}$ to $(2+4\lambda_e+4\lambda_d+\lambda_s)C_{3D}$). The factors λ_d , λ_e and λ_s , equal to C_d/C_{3D} , C_e/C_{3D} and C_s/C_{3D} , respectively, are independent of the TSV length. Consequently, they provide an abstract and universal TSV crosstalk classification, similar to the traditional one for the planar metal wires.

The capacitance model for hexagonal TSV bundles is depicted in Fig. 2.b. Again, due to the Faraday cage effect only capacitances between direct adjacent TSVs are considered. In hexagonal arrangements the distance between every adjacent TSV pair is constant. Thus, the capacitance value between every adjacent TSV pair in the middle of the arrangement is $C_{n,h}$, which is used as a reference value $C_{3D,h}$ for the crosstalk classification. Only edge TSVs have a ground-capacitance $C_{s,h}$, and the capacitance value between two edge TSVs is $C_{e,h}$. $\lambda_{e,0}$ and $\lambda_{e,h}$ are used to describe the relation between the capacitance values at the edges and the reference value $C_{3D,h}$. This results in 13 crosstalk classes for a TSV in the middle of a hexagonal TSV arrangement: $0C_{3D}$ to $12C_{3D}$. The crosstalk of a TSV located at an edge of a hexagonal arrangement is classified in 35 classes: $\lambda_{s,h}C_{3D,h}$ to $(6+4\lambda_{e,h}+\lambda_{s,h})C_{3D,h}$.

Finally, we want to mention that for both TSV topologies the sizes of the self-capacitances C_s and $C_{s,h}$ depend on the surrounding of the TSV arrangements (Keep-out-Zone (KOZ) width [24], ground ring [15], etc.). For a KOZ equal to four times the minimum TSV pitch, and a substrate grounding after the KOZ, parasitic extractions result in normalized capacitance values λ_s and $\lambda_{s,h}$ of ca. 0.6-1.1, as shown in Section 3 of this work.

3. Limitations of previous CACs for 3D ICs

In this section we discuss the limitations of existing CAC techniques for 3D ICs. Crosstalk avoidance coding was initially proposed for metal wires. The idea of these traditional 2D CACs is to limit the overall maximum crosstalk of the metal wires as it determines the interconnect performance (delay, switching noise approx. $\sim \max_{k,i}\{C_{eff,i}\}$) [4]. For example, in contrast to an unencoded ($4C$) 2D bus, in a $3C$ and in a $2C$ encoded 2D bus the crosstalk a metal wire experiences never exceeds $3C_c$ and $2C_c$, respectively. However, besides the system performance, the power consumption is an important design metric. In contrast to the performance of the interconnects, determined by the maximum/peak crosstalk value, their dynamic power consumption is proportional to the sum of the mean crosstalk values ($P \sim \text{mean}_k\{\sum_i C_{eff,i}\}$) [8]. Thus, a wide set of low-power coding techniques have been proposed which reduce the metal wire power consumption by reducing the expected value of $\sum_i C_{eff,i}$. Some existing 2D CACs for metal wires also result in drastically reduced mean $C_{eff,i}$ -values [4]. Thus, these coding techniques are low-power 2D

Table 1: (Normalized) capacitance values in modern TSV bundles

TSV dim.		Array Model				Hexagonal Model		
r_{tsv} [μm]	d_{tsv} [μm]	C'_{3D} [aF/ μm]	λ_d	λ_e	λ_s	$C'_{3D,h}$ [aF/ μm]	$\lambda_{e,h}$	$\lambda_{s,h}$
1	4	82	0.32	1.35	1.03	72	1.25	0.69
1	4.5	79	0.34	1.36	1.14	71	1.28	0.71
2	8	94	0.30	1.35	0.90	81	1.25	0.62
2	8.5	90	0.31	1.35	0.96	78	1.26	0.62

CACs which can be used to optimize the power consumption and the performance of metal wires, which are used in 3D ICs for the planar interconnects. However, traditional 2D CACs initially fail to optimize the crosstalk of TSVs, which are used in 3D ICs for the vertical interconnects.

Thus, the idea of crosstalk avoidance coding was also investigated for TSVs (3D CACs) [16–18]. However, these existing 3D CACs only aim to reduce the maximum crosstalk in TSV arrays, while they leave metal wires and hexagonal TSV arrangements unoptimized. Furthermore, the large bit overheads of existing 3D CACs lead to a drastic increase in the TSV power consumption which often prohibits the use of these techniques. However, the biggest limitation of existing 3D CACs for TSV arrays is that their real coding gain is far less than previously reported. To outline why, we first analyze the (normalized) capacitance values, reported in Table 1, for modern TSV arrays, obtained by parasitic extractions with the Ansys Q3D EM wave solver [25]. The parasitics are extracted for 5×5 TSV arrays with a KOZ equal to four times the minimum TSV pitch. The analyzed TSV dimensions (pitch: d_{tsv} ; radius: r_{tsv}) correspond with the ones reported for the 2015-2018 time slot of the International Technology Roadmap for Semiconductors (ITRS) 2013. Since C_{3D} is reported per unit length ($C'_{3D} = C_{3D}/l_{tsv}$), the results can be used for all TSV lengths l_{tsv} . The ITRS did not report the expected TSV liner thickness t_{ox} . According to existing process nodes, we choose $t_{ox} = r_{tsv}/5$. The substrate is Boron (p) doped and has a conductivity of 10 S/m . A TSV, its dielectric and the substrate form a metal oxide semiconductor (MOS) junction. Thus, in the substrate, a TSV is surrounded by a depletion region [26]. For the parasitic extractions, depletion regions are modeled as areas where the substrate has no free charge carriers ($\sigma = 0$) [6]. Therefore, the width of a depletion region is calculated for every geometrical variation by means of the exact Poisson's equation under the assumption of an average TSV voltage of $V_{dd}/2 = 0.5 \text{ V}$. For the sake of completeness, also the model coefficients for hexagonal TSV bundles are reported in Table 1. To obtain these coefficients, parasitic extractions for arrangements containing 16 TSVs in three hexagons (see Hex3 in Fig. 7 on page 9) are analyzed.

According to Table 1, for random patterns, the maximum crosstalk in a modern TSV array is approx. $10.5C_{3D}$ for the middle TSVs ($(8+8\lambda_d)C_{3D}$) and approx. $9.7C_{3D}$ for the edge TSVs ($(2+4\lambda_e+4\lambda_d+\lambda_s)C_{3D}$). For the derivation of all previous TSV CACs, the edge effects are not considered and the efficiency of the coding approaches is

only evaluated for a TSV in the middle of an array. Existing CACs have in common that, for each TSV, they reduce the maximum amount of adjacent TSVs switching in the opposite direction. For example, the 6C 3D CAC [16] simply limits the maximum amount of orthogonal adjacent aggressor TSVs switching in the opposite direction to three. When three orthogonal adjacent aggressor TSVs switch in the opposite direction, the remaining one always switches in the same direction. Consequently, for a middle TSV, the maximum crosstalk is reduced to $(6+8\lambda_d)C_{3D}$ (approx. $8.5C_{3D}$). Edge TSVs have a maximum of three orthogonal neighbors/aggressors. Thus, the 6C 3D CAC [16] coding does not provide an optimization of the crosstalk of an edge TSV. Coded and uncoded, their maximum capacitive crosstalk is approx. $9.7C_{3D}$. Thus, the worst case crosstalk/delay for the encoded patterns occurs at the edges. Consequently, the edge effects reduce the actual coding efficiency. In the same way one can show that all other previous TSV CACs actually result in significantly lower reductions in the crosstalk delay than previously reported.

To quantify the actual coding efficiencies, we reanalyze the delay reductions of promising 3D CAC approaches [16–18]. The 4LAT coding [17] limits the number of maximum adjacent switching TSVs to four. Consequently, for each TSV i , maximum three $\delta_{i,j}$ -values can be two. The 6C-FNS coding [18] limits the crosstalk of each middle TSV to $6.5C$, where an orthogonal capacitance value (C_n or C_e) is equal to $1C$ and a diagonal capacitance value (C_d) is equal to $0.25C$. The 6C[16] and the 4LAT[17] coding are evaluated for the already discussed quadratic 5×5 array with $r_{tsv} = 1 \mu\text{m}$, $d_{tsv} = 4 \mu\text{m}$ and $l_{tsv} = 50 \mu\text{m}$. These TSV dimensions correspond with the minimum global TSV dimensions reported for the year 2018 by the ITRS. A drawback of the 6C-FNS[18] 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 reduction of a middle TSV is analyzed, we analyze the delay of the edge and middle TSVs. To determine the pattern dependent delays of the TSVs with the Spectre circuit simulator, we use extracted complete 3π -RLC lumped element circuits of the TSV arrays. Thus, inductance effects are considered. In the simulations, each TSV is driven by a two-inverter chain and the input slew rate is 1 ps. Driver strengths of the first and second inverter are $2 \times$ and $6 \times$, respectively. For the inverters, 22 nm Predictive Technology Model (PTM) is used. In Fig. 3 the Spectre simulations and the measurement of the propagation delay from the input of the second inverter to the output of the TSVs are illustrated for the 6C coding [16]. In this work, we distinguish between the maximum delay of a middle TSV ($T_{p,m}$) and the maximum delay of an edge TSV ($T_{p,e}$), for the encoded and the uncoded patterns. In Table 2, all measured propagation delays are shown. The table reveals that, as expected, the worst case delay, which determines the maximum allowed clock frequency, always occurs at the edges for the 3D CACs, while it occurs in the middle

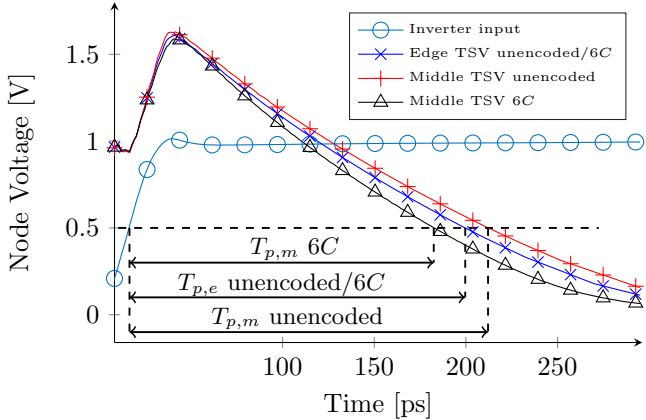


Fig. 3: Worst case propagation delay: inverter input and TSV output voltage waveform

Table 2: Maximum propagation delay for the middle ($T_{p,m}$) and the edge ($T_{p,e}$) TSVs, for CAC encoded and unencoded patterns, besides the CAC delay reduction of the middle TSVs ($\Delta T_{p,m}$) and the actual overall delay reduction (ΔT_p)

Pattern set	$T_{p,m}$ [ps]	$T_{p,e}$ [ps]	$\Delta T_{p,m}$ [%]	ΔT_p [%]
Unencoded	212.1	197.4	-	-
6C[16]	179.6	197.5	-15.3	-6.9
4LAT[17]	171.1	192.1	-19.3	-9.4
6C-FNS[18]	148.0	197.5	-30.0	-6.9

for the unencoded scenario. Table 2 also includes the delay reduction for middle TSVs as well as the overall TSV delay reduction. The overall delay is obtained by taking the maximum out of the delay values for edge and middle TSVs. The delay reductions for the middle TSVs conform with the ones reported in Ref. [16–18] for the overall delay reductions. However, the results show that the edge effects drastically decrease the true overall delay reduction and consequently the efficiency of the CACs. For example, the improvements in speed of the approaches 6C[16] and 4LAT[17] are less than 50% of their previous expected values. For the recently most promising 6C-FNS coding [18], the true delay reduction is less than one fourth of the previously expected value (6.9% instead of 30%). The first reason for this dramatic decrease in the coding efficiency is the neglected edge effects. The second one is the generally unconsidered edge TSVs. The 6C-FNS coding approach simply limits the maximum crosstalk of the middle TSVs to $6.5C$ (e.g. $6C_n + 2C_d$), while the maximum coupling of the edge TSVs remains unoptimized. If all surrounding TSVs of a TSV located at a single edge switch in the opposite direction, its coupling is equal to $2C_n + 4C_e + 4C_d$, which is equal to $7C$ according to the crosstalk classification in [18]. Therefore, the previously proposed 6C-FNS encoding is actually only a 7C-FNS encoding.

The (bit) overhead of an n -bit CAC is defined as:

$$OH(n) = \frac{m - n}{n}, \quad (4)$$

where n and m are the bit widths of the input and the

code words, respectively. The existing 3D CAC techniques require asymptotic overheads ($\lim_{n \rightarrow \infty} OH(n)$) of 44–80%. These high bit overheads are the source of the increased overall TSV power consumption for existing 3D CACs as they surpass the power savings per TSV [5].

In summary, we identified three limitations of current 3D CACs. Firstly, they only reduce the crosstalk of TSV arrays, while they leave hexagonal arrangements and metal wires unaffected. Secondly, due to the edge effects their actual crosstalk reduction is rather poor and does not justify their high overhead costs. And last but not least, the high overhead costs lead to a dramatic increase in the TSV power consumption. These considerations show the need for new crosstalk avoidance methods for 3D integration, which take the edge effects, the metal wires and the power consumption into account.

4. Proposed crosstalk avoidance technique

In this section we derive step by step a 3D CAC technique which overcomes all previously outlined limitations of existing techniques. In Subsection 4.1 we present a TSV CAC approach that overcomes the limitations due to the edge effects. Furthermore, the presented TSV CAC approach is no longer restricted to a specific TSV topology. In Subsection 4.2 we show how our TSV CAC can be extended to the first 3D CAC which optimizes the TSV and the metal wire crosstalk simultaneously. Finally, this 3D CAC is extended to low-power 3D CAC which can increase the performance of the TSVs and the metal wires while it simultaneously decreases their power consumption.

4.1. General TSV CAC approach

Here, we present a new CAC approach for TSV arrangements, called ω_m/ω_e TSV CAC. The presented coding technique overcomes the limitations of previous 3D CAC approaches which arise due to the edge effects. In detail: the presented coding approach does not only reduce the crosstalk of middle TSVs, it also reduces the crosstalk of edge TSVs. Furthermore, the presented approach is not limited to a specific topology and is thus applicable for rectangular arrays as well as hexagonal bundles.

The general idea is to reduce the maximum possible effective capacitance of each middle and each edge TSV by at least $\omega_m C_{3D,(h)}$ and $\omega_e C_{3D,(h)}$, respectively. Consequently, the maximum crosstalk class of a TSVs in the middle of an array is reduced to

$$(8 + 8\lambda_d - \omega_m)C_{3D}, \quad (5)$$

while the maximum crosstalk class of TSVs at an array edge is reduced to

$$(2 + 4\lambda_e + 4\lambda_d + \lambda_s - \omega_e)C_{3D}. \quad (6)$$

The maximum crosstalk class of TSVs in the middle of a hexagonal arrangement is reduced to

$$(12 - \omega_m)C_{3D,h}, \quad (7)$$

while the maximum crosstalk class of the TSVs at an edge of a hexagonal arrangement is reduced to

$$(6 + 4\lambda_{e,h} + \lambda_{s,h} - \omega_e)C_{3D,h}. \quad (8)$$

As shown in Table 1, for unencoded patterns, the maximum crosstalk of a middle TSV is slightly bigger than for an edge TSVs. Thus, in order to obtain the most efficient coding scheme, if possible, ω_m should be slightly bigger than ω_e .

4.2. 3D CAC technique

In this subsection we present a coding technique which combines the previously introduced ω_m/ω_e TSV coding approach with a traditional 2D CAC to obtain a 3D CAC which simultaneously reduces the TSV and the metal wire crosstalk. The technique is based on existing 2D CACs, as they show to effectively reduce the metal wire crosstalk. We propose to exploit the bit level properties of a 2D CAC encoded pattern set by a fixed mapping of the bits onto a TSV arrangement that results in a ω_m/ω_e TSV CAC.

Note that hereby the global signal-to-TSV-array mapping remains optimized. With other words, only the mapping within the individual TSV arrays is affected. The effect of this local routing is marginal as TSV parasitics are dominant. Additionally, due to KOZ restrictions no active components are located nearby TSV arrays. Thus, we do not face a metal-layer-utilization problem and the mapping does not lead in an area overhead. To quantify precisely the additional costs of the approach presented in this work, we analyze a 3×3 TSV array, including the local routing, for a commercial technology and TSVs with a radius of $2 \mu\text{m}$ and a minimum pitch of $8 \mu\text{m}$. The worst-case routing/mapping only increases the parasitics by a maximum of 0.5%, versus a routing which aims for a local metal wire length minimization. However, for our later proposed systematic mapping, the parasitic increase compared to a column-by-row mapping is not even noticeable. Thus, we can state that the overhead costs for a crosstalk aware bit-to-TSV mapping are negligible.

Besides this mapping, our approach is completely constructed of encoder/decoder pairs known from 2D crosstalk avoidance coding. An in-depth explanation of the implementation of these well known circuits can be found in Ref. [4], among others, and is thus not included in this work. However, for clarification, we repeat the fundamental ideas of the used 2D CACs. Memoryless $2C$ CACs are the most popular 2D CACs and have been extensively studied in the past [27–30]. Two different data encoding methods exist for $2C$ bus encoding: Forbidden-Pattern-Free (FPF) [29] and Forbidden-Transmission-Free (FTF) [30] encoding. For both methods the encoding/decoding process can be based on a Fibonacci numeral system (FNS) mapping, which leads to an encoder/decoder circuit (CODEC) complexity which is several magnitudes lower than for other $2C$ CODECs [4]. For the FPF CAC, bit vectors that contain a 010 or a 101 bit sequence are forbidden. For example

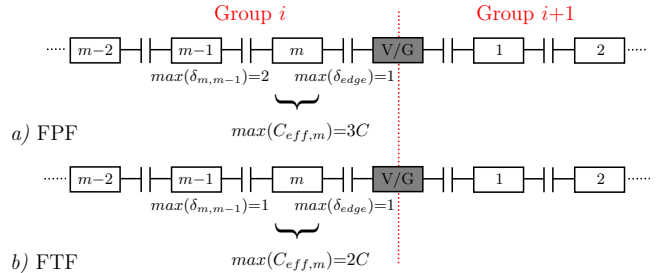


Fig. 4: Power/ground (V/G) lines used for 2D bus partitioning and its influence on the maximum crosstalk for: a) FPF encoding; b) FTF encoding

111000110 is a valid FPF codeword, while 110100011 is a forbidden bit vector. In [4] the authors prove that a FPF bus is a $2C$ bus (max. crosstalk $2C_c$), since, for all i , if $\delta_{i,i-1}$ is equal to 2, $\delta_{i,i+1}$ is always 0 and vice versa. For the FTF $2C$ CAC, all δ -values are limited to 1 by prohibiting adjacent bits from switching in the opposite directions. Hence, the forbidden transitions are 01→10 and 10→01. In [30] it is proven that the largest set of FTF codewords is generated by eliminating the 01 pattern from the $b_{2i+1}b_{2i}$ boundaries, and the 10 pattern from the $b_{2i}b_{2i-1}$ boundaries. The bit overheads for FPF and FTF coding, are equal and asymptotically reach about 44% [4].

We propose to use FTF data encoding, since it has some advantages over the FPF encoding. One advantage is that the CODEC of the FTF CAC requires a ca. 17% lower gate count and an almost 50% lower delay [4]. Nevertheless, for both encoding techniques, a FNS CAC CODEC still exhibits a quadratic growth in complexity with the size of the bus [4]. Thus, the CODEC complexity will quickly cancel out any coding savings with increasing input data widths. To overcome this limitation, the bus can be partitioned into small groups which are encoded individually. In this case, a difficulty arises due to undesired crosstalk transitions between adjacent lines of different groups. In [4], two techniques are designed to address this issue: Group Complement and Bit Overlapping. Both, again, cause a significant bit and CODEC overhead, which make them suboptimal. Here, we present a more effective technique for the FTF data encoding. In 3D ICs power/ground lines have to be spanned over the several dies of the system in order to build a 3D power network. Power/ground (V/G) lines are stable, and stable lines can be used in FTF encoding for the bus partitioning as illustrated in Fig. 4.a. The bus, containing N lines, is divided into NG groups which are encoded individually by a n to m -bit FTF code, where $m = N/NG$. Between each first bit of a group and the last (m^{th}) bit of the previous group, a stable (V or G) line is placed. This bus partitioning generally causes no overhead if the dynamic data lines as well as the stable lines are transmitted over one 2D/3D bus which is a common case [31, 32]. The crosstalk factor $\delta_{i,s}$ of a data line i and a stable line s is $\Delta b_i^2[k]$ (Eq. 3 with $\Delta b_j[k] = 0$), and thus limited to 1. Consequently, stable lines do not violate the

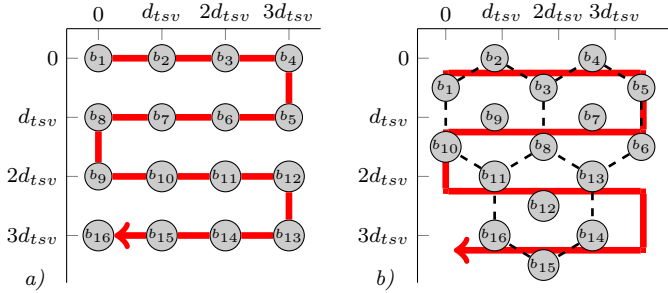


Fig. 5: Snake mapping of the bits b_i of 2D CAC encoded data for: a) TSV arrays; b) hexagonal TSV bundles

FTF condition. However, for FPF encoding, an additional stable line only leads to a $3C$ 2D bus instead of a $2C$ 2D bus, as illustrated in Fig. 4.b. For example, for m equal to five: $00001 \rightarrow 11110$ is a valid FPF CAC sequence. If the group is terminated by a power line (constant logical 1), the effective pattern sequence of the group, including the stable line, is $00011 \rightarrow 11110$. The second pattern is a forbidden pattern (includes 101 sequence). Thus, a stable line violates the FPF condition and the m^{th} -bit experiences a crosstalk of $3C_c$. Therefore, the metal wire crosstalk would no longer be limited to $2C_c$.

The second popular 2D crosstalk avoidance method is shielding, which adds stable (V/G) lines between the data lines to avoid worst case coupling. For the $2C$ shielding, the data signal lines (D) are regularly interleaved with stable shield lines (S), resulting in a $DSDSDS\dots$ 2D layout. In this case each data line i is shielded by two adjacent stable lines, resulting in a metal wire crosstalk of $2\Delta b_i^2[k]C_c \leq 2C_c$ for each data line i . For the $3C$ shielding, data line pairs are shielded by stable lines, resulting in a $DDSDDS\dots$ 2D layout. Consequently, the crosstalk of a metal wire, transmitting data bit i , is $(2\Delta b_i^2[k] - \Delta b_i[k]\Delta b_j[k])C_c \leq 3C_c$, where j is the adjacent data line.

A mapping of all consecutive bit pairs of a 2D CAC onto direct adjacent TSVs results in a ω_m/ω_e CAC for the TSVs. One possible systematic mapping, we refer to as Snake mapping, is illustrated in Fig. 5 for rectangular and hexagonal TSV bundles. A $3C$ CAC results in a guaranteed ω_m/ω_e TSV CAC with $\omega_m = \omega_e = 1$ (1/1 TSV CAC), and a $2C$ CAC results in a guaranteed ω_m/ω_e TSV CAC with $\omega_m = \omega_e = 2$ (2/2 TSV CAC), and so on. Thus, one can use any CAC, designed to reduce the crosstalk of planar metal wires, map its bits using the Snake mapping onto a TSV arrangement and obtain a 3D CAC. If a CAC is already used for the metal wires, this TSV CAC technique does not require any additional overhead costs.

4.2.1. Optimal 2D CAC to 3D CAC mapping

Although, the previously introduced Snake mapping of a 2D CAC results in a ω_m/ω_e TSV CAC, we need to investigate if another mapping leads to an actual lower maximum TSV crosstalk as, for example, a Snake mapping does not exploit the properties of stable lines. As already

discussed, the crosstalk switching δ between any data line and a stable line is always limited to one. Thus, stable lines should be mapped to the middle of the TSV arrangement in order to reduce the crosstalk of the maximum amount of data lines. This is not considered by the systematic Snake mapping which only exploits a reduced maximum crosstalk switching between direct adjacent bit pairs.

Therefore, in the following we derive a mathematical method to find the CAC optimal placement of the bits of a given pattern set onto an interconnect structure. By means of this mathematical method we can determine the perfect use of any 2D CAC for TSV arrays. For an initial mapping that maps bit i onto interconnect i , the vector of the effective capacitances C_{eff} , where the i^{th} vector entry is $C_{eff,i}$ (see Eq. 1), can be expressed as:

$$\mathbf{C}_{eff}[k] = \text{diag}\{\mathbf{T}[k]\mathbf{C}\}, \quad (9)$$

Here, $\text{diag}\{\}$ returns a vector containing the diagonal entries of a matrix. \mathbf{T} presents the switching properties of the interconnects with

$$\mathbf{T}_{i,j}[k] = \begin{cases} \Delta b_i^2[k] & \text{for } i = j \\ \delta_{i,j}[k] & \text{else} \end{cases}. \quad (10)$$

C is the capacitance matrix of the interconnects architecture containing N lines:

$$\mathbf{C} = \begin{bmatrix} C_{1,0} & C_{1,3} & \dots & C_{1,N} \\ C_{2,1} & C_{2,0} & \dots & C_{2,N} \\ \vdots & C_{3,2} & \ddots & \vdots \\ C_{N,1} & \dots & C_{N,N-1} & C_{N,0} \end{bmatrix}. \quad (11)$$

A new mapping of the bits onto the interconnects can be realized by swapping rows and the according columns of the switching matrix \mathbf{T} , mathematically expressed as:

$$\mathbf{T}'[k] = \mathbf{A}_\pi \mathbf{T}[k] \mathbf{A}_\pi^T, \quad (12)$$

where \mathbf{A}_π is a valid $N \times N$ permutation matrix [33]. A valid permutation matrix has exactly one 1 in each column/row while all other matrix entries are 0. The set of all valid $N \times N$ permutation matrices is denoted as S_N . To map the i^{th} bit of the data stream onto line j , $\mathbf{A}_{\pi,j,i}$ is set to one. For an exemplary 4-bit interconnect structure, if we want to map bit 4 onto line 1, bit 3 onto line 2, bit 1 onto line 3 and bit 2 onto line 4:

$$\mathbf{A}_\pi = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \quad (13)$$

Thus, the crosstalk for an arbitrary placement \mathbf{A}_π can be determined by:

$$\mathbf{C}'_{eff}[k] = \text{diag}\{\mathbf{A}_\pi \mathbf{T}[k] \mathbf{A}_\pi^T \mathbf{C}\}. \quad (14)$$

In the following we assume an independence of $\delta_{i,j}$ and $\delta_{i,l}$ for all i, j and l . Please note that this assumption is

true for random data and the 2D CACs discussed in this section but not for a FTF CAC, where: $\max_k(\delta_{i,i-1}) = \max_k(\delta_{i,i+1}) = 2$, but $\max_k(\delta_{i,i-1} + \delta_{i,i+1}) = 2 \neq 4$. For independent δ values, the vector constituting the maximum crosstalk quantity can be calculated by:

$$\hat{C}'_{eff} = \max \text{diag} \{ \mathbf{A}_\pi \mathbf{T}_w \mathbf{A}_\pi^T \mathbf{C} \}, \quad (15)$$

where $\max \text{diag}\{\}$ returns the maximum diagonal entry of a matrix. \mathbf{T}_w is the worst case crosstalk matrix with

$$\mathbf{T}_{w,i,j} = \begin{cases} \max_k(\Delta b_i^2) & \text{for } i = j \\ \max_k(\delta_{i,j}) = \max_k(\Delta b_i^2 - \Delta b_i \Delta b_j) & \text{else} \end{cases}. \quad (16)$$

The characteristics of a specific pattern set are captured by this matrix \mathbf{T}_w . For an unencoded pattern set $\mathbf{T}_{w,i,j}$ is equal to 2, except for the diagonal entries ($i = j$) which are equal to 1. For FTF patterns, $\mathbf{T}_{w,i,j}$ is equal to 2 except for the diagonal elements and its adjacent entries ($j = i+1$ and $j = i-1$) which are equal to 1. An additional stable line at position x ($\Delta b_x = 0$) leads to $\mathbf{T}_{w,x,j}$ equal to 0 for all j , and $\mathbf{T}_{w,i,x}$ equal to 1 for all $i \neq x$.

The optimal CAC mapping $\mathbf{A}_{\pi, \text{CACopt}}$ minimizes \hat{C}'_{eff} , which can mathematically be expressed as:

$$\mathbf{A}_{\pi, \text{CACopt}} = \arg \min_{\mathbf{A}_\pi \in S_N} (\max \text{diag} \{ \hat{C}'_{eff} \}). \quad (17)$$

In practice $\mathbf{A}_{\pi, \text{CACopt}}$ is determined with any of the several optimization tools available to reduce the computational complexity. Although, overall, up to several hundreds of TSVs exist in modern 3D ICs, the runtime of an optimization is negligibly low for our problem as it is executed for each TSV bundle individually whose size is relatively small. In this work, we use Simulated Annealing [34] to determine the optimal mapping.

5. Extension to a low-power 3D CAC

In this section we extend our 3D CAC approach to the first low-power 3D CAC technique, reducing the maximum crosstalk and the power consumption of the metal wires as well as the TSVs simultaneously.

From the underlying 2D CACs in our 3D CAC approach, only the FTF encoding effectively reduces the metal wire power consumption [4]. Therefore, here we only analyze our ω_m/ω_e TSV CAC based on the FTF encoding (with or without bus partitioning) since our objective is a simultaneous optimization of the metal wires and the TSVs. However, we would like to note that, as also shown in the later Section 7, on average shielding techniques leave the metal wire and TSV power consumption unaffected [4]. Thus, our 3D CAC technique based on shielding techniques still overcomes the dramatic power consumption increase of previous techniques.

Firstly, we derive a method to estimate the power consumption as a function of the bit-to-interconnect mapping

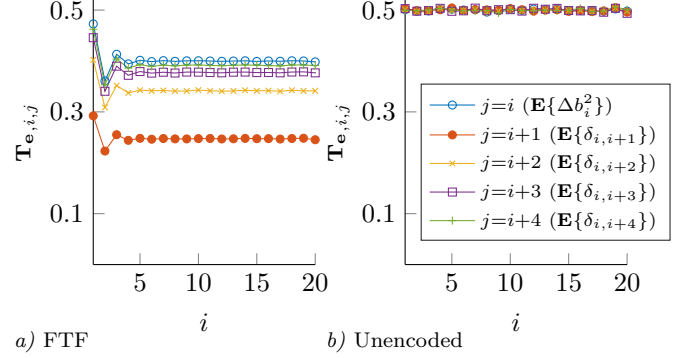


Fig. 6: \mathbf{T}_e -values for a FTF encoded and an unencoded random data stream

for a given data stream. The mean dynamic power consumption of an interconnect structure for an initial mapping ($b_i \rightarrow$ interconnect i) is equal to [8]:

$$\begin{aligned} P &= \frac{V_{dd}^2 f}{2} \sum_i \mathbf{E} \{ \Delta b_i^2 \} C_{i,0} + \sum_{i,j} \mathbf{E} \{ \delta_{i,j} \} C_{i,j} \\ &= \frac{V_{dd}^2 f}{2} \sum_i \mathbf{E} \{ C_{eff,i} \}, \end{aligned} \quad (18)$$

Where $\mathbf{E}\{\}$ is the expectation operator. The first term in the equation depends on the power supply voltage V_{dd} and the clock frequency f , which are not affected by the bit-to-interconnect mapping. Thus, we consider the mean power consumption normalized by this factor: $P_n = 2P/V_{dd}^2 f$.

P_n can be again transformed into a matrix form:

$$P_n = \text{tr} \{ \mathbf{A}_\pi \mathbf{T}_e \mathbf{A}_\pi^T \mathbf{C} \}. \quad (19)$$

For an arbitrary bit-to-interconnect mapping we obtain

$$P'_n = \text{tr} \{ \mathbf{A}_\pi \mathbf{T}_e \mathbf{A}_\pi^T \mathbf{C} \}. \quad (20)$$

In Eq. 19-20 the operator $\text{tr}\{\}$ calculates the trace (sum of diagonal entries) of a matrix. \mathbf{T}_e is a matrix containing the switching probabilities of the lines:

$$\mathbf{T}_{e,i,j} = \begin{cases} \mathbf{E} \{ \Delta b_i^2 \} & \text{for } i = j \\ \mathbf{E} \{ \delta_{i,j} \} = \mathbf{E} \{ \Delta b_i^2 \} - \mathbf{E} \{ \Delta b_i \Delta b_j \} & \text{else} \end{cases}. \quad (21)$$

In the following we will briefly discuss the switching probabilities for unencoded and FTF encoded data streams, captured by \mathbf{T}_e . Thereby, we consider the transmission of a random data stream, as it results in the highest interconnect power consumption [35]. $\mathbf{T}_{e,i,j}$ -values of both data streams are plotted in Fig. 6. For the unencoded data (Fig. 6.b) all \mathbf{T}_e -entries are about 0.5, as the toggle probability of each bit is 50% ($\mathbf{E}\{\Delta b_i^2\}=0.5$) and the bit pairs are spatially uncorrelated ($\mathbf{E}\{\Delta b_i \Delta b_j\} = 0 \rightarrow \mathbf{E}\{\delta_{i,j}\} = \mathbf{E}\{\Delta b_i^2\}$). The FTF encoding (Fig. 6.a) reduces the toggle activity to about 40% as it induces temporal bit correlations. Furthermore, due to an induced positive correlation ($\mathbf{E}\{\Delta b_i \Delta b_j\} > 0$)

between nearby bit pairs, the FTF encoding drastically reduces some \mathbf{T}_e -entries. For direct adjacent neighbors, the coupling switching probability $\mathbf{T}_{e,i,j}$ is about 1/2 of its value for the unencoded data. The further j is apart from i the more the bit pairs tend to be uncorrelated, resulting in increased $\mathbf{T}_{e,i,j}$ -values which asymptotically reach $\mathbf{E}\{\Delta b_i^2\} \approx 0.4$. An additional stable line at position x ($\Delta b_x[k]=0$) leads to $\mathbf{T}_{e,x,j} = 0$ and to $\mathbf{T}_{e,i,x} = \mathbf{E}\{\Delta b_i^2\}$ for all i, j (see Eq. 21 with Δb_i or Δb_j equal to 0).

These considerations imply that to effectively reduce the TSV power consumption for the transmission of FTF encoded patterns, neighboring bit pairs have to be mapped onto TSVs connected by a large coupling capacitance. Thus, neighboring bit pairs have to be mapped onto direct adjacent TSVs. This criteria is also satisfied by our Snake mapping, presented in the previous section as a plain CAC mapping. Thus, the Snake mapping of FTF patterns results in a low-power 3D CAC.

However, the Snake mapping will likely not lead to the optimum low-power 3D CAC, as it neither considers all variations in the $\mathbf{T}_{e,i,j}$ -entries (e.g. Fig. 6 shows a $\mathbf{T}_{e,i,j}$ increase for $i = 1$ and a decrease for $i = 2$), nor the properties of stable lines. Thus, in the following we derive a mathematical method to obtain the optimal mapping.

Here, two costs have to be optimized by the mapping: first the power consumption P'_n and second the maximum crosstalk \hat{C}'_{eff} . To combine these quantities into a single cost function $f_c[\mathbf{A}_\pi]$, we normalize the single costs P'_n and \hat{C}'_{eff} by the costs for the initial placement P_n and \hat{C}_{eff} :

$$f_c[\mathbf{A}_\pi] = \alpha \frac{P'_n}{P_n} + (1 - \alpha) \frac{\hat{C}'_{eff}}{\hat{C}_{eff}}, \quad (22)$$

where Eq. 19 and Eq. 14 are substituted for P'_n and \hat{C}'_{eff} , respectively. α , which is between 0 and 1, sets the optimization constraints. For α equal 0.5 the power optimization and the crosstalk peak optimization are considered as equivalent optimization goals. For bigger α the power consumption is more and more prioritized, while for smaller α the maximum crosstalk is more and more prioritized. Finally, the optimal mapping, can be determined by:

$$\mathbf{A}_{\pi,\text{opt}}(\alpha) = \arg \min_{\mathbf{A}_\pi \in S_N} \left(\alpha \frac{P'_n}{P_n} + (1 - \alpha) \frac{\hat{C}'_{eff}}{\hat{C}_{eff}} \right), \quad (23)$$

which is here solved by means of Simulated Annealing.

6. Evaluation of the proposed technique

In this section, our proposed technique is theoretically evaluated. Subsection 6.1 analyses the reduction in the maximum TSV crosstalk of our 3D CAC technique, for various rectangular and hexagonal TSV arrangements when the power consumption is neglected. In Subsection 6.2 the trade off when the power consumption and the crosstalk peak are simultaneously optimized is discussed.

1 Hexagon (Hex1) 2 Hexagons (Hex2) 3 Hexagons (Hex3)

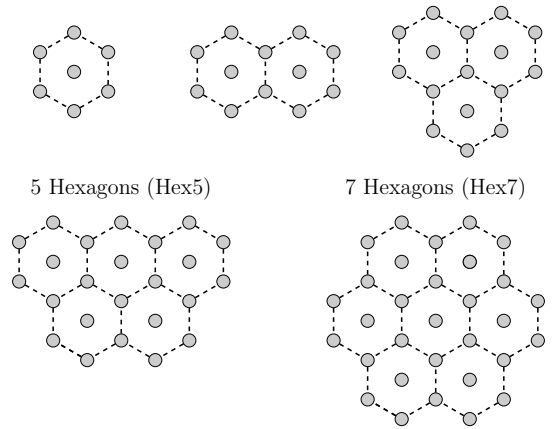


Fig. 7: Analyzed hexagonal TSV arrangements

6.1. Maximum crosstalk reduction

In this subsection we determine the effect of our 3D CAC technique on the maximum crosstalk for various TSV arrangements. Thereby, we assume that only the maximum crosstalk is the optimization goal. For the TSV capacitance matrices, we use the ones extracted for the TSV dimensions $r_{tsv} = 1 \mu\text{m}$; $d_{tsv} = 4 \mu\text{m}$ and $r_{tsv} = 2 \mu\text{m}$; $d_{tsv} = 8 \mu\text{m}$. The first TSV dimensions correspond with the minimum global TSV dimensions reported by the ITRS for the year 2018, while the second ones represent more typical TSV dimensions in modern 3D ICs. In all analyses the TSV length is $50 \mu\text{m}$ (corresponds with the typical substrate thickness) as the relative crosstalk reduction is independent of this quantity. The capacitance matrices are extracted for a wide range of rectangular and hexagonal TSV arrangements. For the rectangular arrays, we vary the dimensions M and N . More precisely, we analyze quadratic arrays from 3×3 to 7×7 and non-quadratic arrays with M equal 3 and N equal 6, 9 or 12. For the hexagonal topology, we analyze structures composed of 1, 2, 3, 5 and 7 TSV hexagons (Hex1-Hex7), shown in Fig. 7.

As the underlying 2D CACs in our 3D CAC technique, we analyze: $2C/3C$ shielding and the FTF data encoding. Besides the investigation of the plain FTF encoding, the CAC is additionally investigated for the scenario where approx. 10% stable TSVs are present in each TSV bundle, which are exploited for bus partitioning (FTF BP). For example, in a 4×4 array we assume two stable lines which are used to 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).

Fig. 8 shows, for all crosstalk avoidance methods, the maximum effective capacitance for the optimal and the Snake mapping, besides the maximum effective capacitance for random patterns. The figure reveals that, unencoded, the maximum crosstalk is almost independent of the topology (mesh or hexagonal) and the TSV count. In contrast, Ref. [21] reported a lower maximum TSV crosstalk for hexagonal TSV topologies. However, the according evaluation was only performed for TSV pitches bigger than

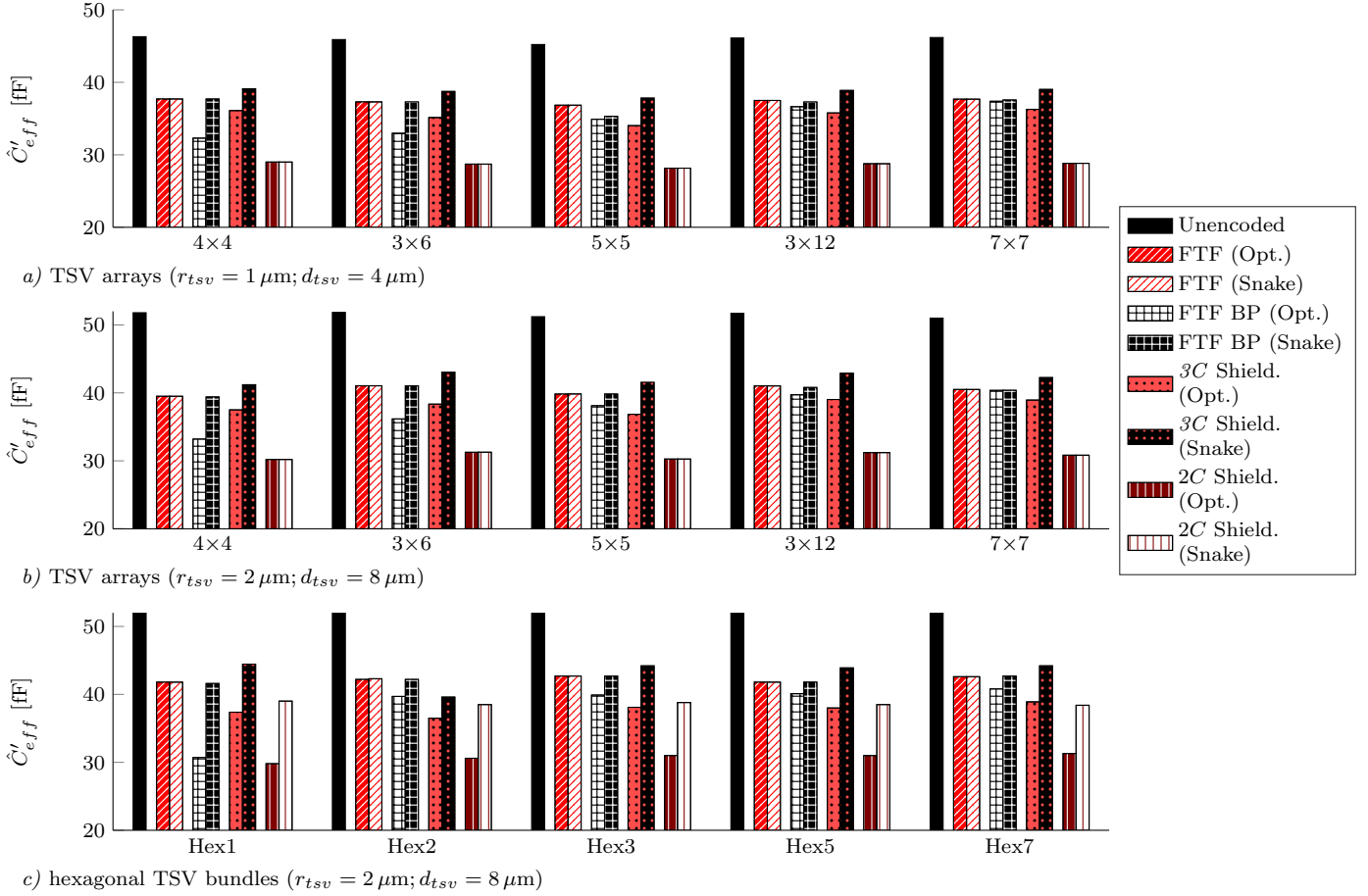


Fig. 8: Effect of the proposed technique on the maximum effective TSV capacitance $\hat{C}'_{eff,max}$, for different underlying 2D CACs. Compared are the two mapping methods, optimal and Snake, for: a)/b) TSV arrays, and c) hexagonal TSV bundles

10 μm . With a decreasing TSV pitch, the E-field sharing effect stronger impairs the capacitive crosstalk of TSVs in hexagonal bundles. Thus, for recent and future TSV dimensions, the maximum capacitance limit, presented in [36], is reached for hexagonal and array arrangements. Therefore, the crosstalk problem has about the same magnitude for array and hexagonal arrangements.

For all TSV arrangements, the Snake mapping leads to a $2/1+\lambda_{e(h)}$ (ω_m/ω_e) CAC for FTF patterns, and if no stable lines are present, the crosstalk reductions due to the Snake and the optimal mapping are equivalent (neglectable deviation). Compared to the unencoded pattern set, the FTF CAC without bus partitioning always leads to a TSV crosstalk reduction of 18-21%. While we obtain the bigger crosstalk reductions for arrays with thicker TSVs, the FTF CAC reduction is independent of the TSV count.

As expected, when stable lines are used for shielding or bus partitioning, the optimal TSV mapping generally results in a noticeably higher TSV crosstalk reduction than the Snake mapping. The only exception is the 2C shielding for TSV arrays, where the Snake mapping is equal to the optimal mapping. Here, the Snake mapping coincidentally results in a $4/2+\lambda_e$ TSV CAC and thus completely avoids

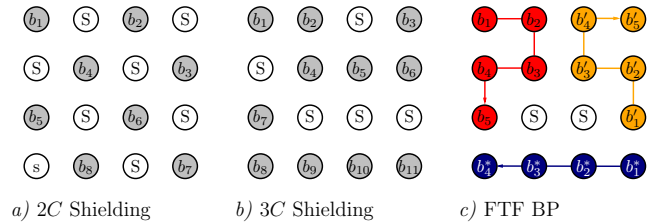


Fig. 9: CAC optimal mapping of several 2D CACs onto a 4x4 array

opposite switchings between any orthogonal adjacent TSV pair, as shown in Fig. 9.a. Thus, the maximum crosstalk is drastically reduced to ca. $6.5C_{3D}$ (reduction by ca. 40%). For all other analyzed scenarios, a remapping of the stable lines, executed by the optimal mapping, increases the efficiency of our 3D CAC, as we will show by an exemplary 4x4 array. However, the following discussion also applies to hexagonal TSV bundles. For TSV arrays, the Snake mapping of 3C shielded patterns leads to a $1+2\lambda_d/1+\lambda_d$ CAC (reduction ca. 15%). Here, a different mapping can lead to an increased ω_m , which increases the coding efficiency. As illustrated in Fig. 9.b, the shields are optimally placed in a way that ω_m is increased to $2+2\lambda_d$. As a result,

the $3C$ shielding results in a maximum crosstalk reduction of 22% instead of 15%. For the same reason, compared to the Snake mapping, an optimal mapping of the $2C$ and $3C$ shielded patterns decreases the maximum crosstalk in hexagonal TSV bundles by up to 43% and 31%, respectively, while the maximum crosstalk reductions due to the Snake mapping are 27% and 25%, respectively.

Also, for the FTF BP CAC, the crosstalk reduction of the optimal mapping differs from the reduction for the Snake mapping, which results in a $2/1+\lambda_{e(h)}$ CAC for rectangular and hexagonal TSV bundles. For the exemplary 4×4 array, the optimal mapping, shown in Fig. 9.c, increases the ω_m -value to $3+\lambda_d$ by mapping the two stable lines to the middle of the array. This boosts the crosstalk reduction from about 19% to above 30%. Here however, due to the small fraction of stable TSVs, a significant increase in ω_m is only possible for small TSV bundles. For bigger ones, the middle TSV over edge TSV ratio increases. Thus, for the FTF BP, the gain of the optimal mapping over the Snake mapping decreases with an increasing TSV count for the hexagonal and the array bundles.

In summary, the Snake mapping is equivalent to the optimal crosstalk mapping, if no stable lines/shields are present. Stable lines, located at the edges for the Snake mapping, are optimally remapped into the middle of a TSV bundle in order to shield the maximum amount of TSVs which generally increases the crosstalk reduction.

6.2. Simultaneous crosstalk peak and power reduction

In the following, we investigate the efficiency of our low-power 3D CAC technique for the Snake and the optimal bit-to-TSV mapping. Therefore, we analyze the simultaneous reduction in the TSV power consumption and the maximum crosstalk due to the two mapping approaches. The optimal bit-to-TSV mapping, determined by means of Eq. 23, is analyzed for different α -values between 0 and 1. For this analysis, we consider a hexagonal TSV bundle (Hex3 from Fig 6.) and two TSV arrays (4×4 and 5×5) with $r_{tsv} = 2\ \mu\text{m}$, $d_{tsv} = 8\ \mu\text{m}$ and $l_{tsv} = 50\ \mu\text{m}$ for the transmission of random patterns, FTF encoded random patterns, and FTF encoded random patterns with two stable lines which are used for bus partitioning (FTF BP). To take the overhead of the FTFs as well as the different numbers of data TSVs into account, mean power consumptions per effectively transmitted bit (P'_b) are compared. The resulting maximum crosstalk reductions ($\hat{C}'_{eff}/\hat{C}_{eff,unco}$) over the power reductions ($P'_{eff}/P_{b,unco}$) are plotted in Fig. 10.

For the FTF encoding without bus partitioning, if just the maximum crosstalk is considered for the optimal bit-to-TSV mapping ($\alpha = 0$), we obtain a crosstalk peak reduction of about 18-19%, but no reduction or even an increase in the TSV power consumption. However, for all α between 0.01 and 0.99, we obtain an equivalent maximum crosstalk reduction and furthermore a reduction in the power consumption by about 8.5%. Therefore, the low-power extension of our approach enables a significant reduction in the power consumption while providing the

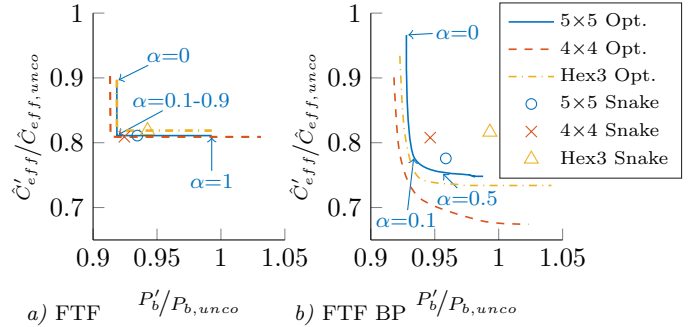


Fig. 10: Effect of our proposed low-power 3D CAC technique on the TSV crosstalk peak and power consumption for: a) FTF encoding; b) FTF encoding with two stable lines for bus partitioning (BP). Exemplary for the 4×4 array, the arrows indicate the locations of the α values

same reduction in the maximum crosstalk as the pure CAC approach. As already discussed in the previous subsection, the Snake mapping also leads to the minimum possible crosstalk peak for a FTF encoding without bus partitioning. Our analysis shows that the Snake mapping of FTF encoded data also decreases the power consumption by 5.7-7.5% and thus also results in a low-power 3D CAC. However, due to unexploited variations in the T_e -values, the Snake mapping does not result in the minimum possible power consumption and is therefore not optimal.

When stable lines are exploited for bus partitioning (FTF BP) there is a clearer trade-off between maximum crosstalk and power consumption. When the power consumption is neglected for the determination of the optimal placement ($\alpha = 0$) the crosstalk peak reduction is in between 25.7% and 32.6%, while the power consumption is often even increased. If both quantities are weighted equally ($\alpha = 0.5$), the crosstalk peak reduction is 25.4-28.7% and the power reduction is 6.1-6.3%. $\alpha = 0.5$ offers a very good trade-off for larger TSV counts and hexagonal arrangements. While for the smaller 4×4 array, the degradation in the crosstalk peak reduction compared to the optimal case is above 3 percentage points, for the hexagonal bundle and the 5×5 array, the degradation is less than 1 percentage point. For a power weight of 10% ($\alpha = 0.1$), the maximum crosstalk reduction is 25.5-32.3% and the power reduction is 1.1-2.2%. Thus, a small penalty in the crosstalk peak reduction by maximum 0.3 percentage points already results in a reduction, instead of a decrease in the power consumption. If the power weight is nine times higher than the maximum crosstalk weight, the power consumption reduction is 6.6-7.7% while the maximum crosstalk reduction is degraded to 18.5-22.2%. This higher trade-off 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, 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 power consumption, nor in the lowest maximum crosstalk.

Summarized, the evaluation reveals that the low-power extension presented in this work allows for a decrease in the power consumption while leaving the crosstalk almost unaffected, but only if no stable lines are present. If stable lines are present, there often is a clearer trade-off between power consumption and maximum crosstalk. However, the trade-off vanishes with increasing data line over stable line ratio. Thus, for big TSV arrangements with only a few stable TSVs, a power and crosstalk peak aware placement results in (almost) the same maximum crosstalk reduction as a mapping which only aims to reduce the crosstalk peaks. Therefore, the low-power extension presented in this work often allows for a decrease in the power consumption while leaving the maximum crosstalk reduction and the overhead costs unaffected.

7. Simulation results and analysis

In this section we compare the presented 2D CAC to ω_m/ω_e TSV CAC mapping approach with existing 3D CAC techniques [16–18] in terms of bit overhead, CODEC area, as well as interconnect delay and power reduction. We analyze the transmission of a 16-bit wide random data stream, including all possible bit transitions, over a metal wire bus and a TSV array or alternatively a hexagonal TSV bundle. In the analysis, the dimensions of the global TSVs are equal to the minimum ones reported for the year 2018 by the ITRS ($r_{tsv} = 1 \mu\text{m}$; $d_{tsv} = 4 \mu\text{m}$ and $l_{tsv} = 50 \mu\text{m}$). The Q3D extractor is again used to extract the TSV parasitics of the individual TSV arrangements, which are built as square as possible. The metal wires are assumed to be in Metal4 with a width and spacing of $0.15 \mu\text{m}$. The length of a metal wire segment is assumed as $100 \mu\text{m}$. The wire parasitics are obtained by the TSMC wire model, which is based on Synopsys Raphael. The chosen value for width and spacing corresponds with the minimum possible value.

As the underlying 2D CACs for our 3D CAC technique, we analyze the FTF CAC with bus partitioning (FTF BP) and the $2C/3C$ shielding. Here we analyze the FTF with bus partitioning, since one power and one ground TSV is included in each TSV bundle. This is a common case as at least two power/ground TSVs are required to set a power network. Thus, the FTF encoding is partitioned into three groups: two 5-bit to 7-bit FTF encoders and one 6-bit to 9-bit FTF encoder. Thus, arrangements consisting of 25 TSVs are required for the 23 FTF encoded data lines and the two stable lines. For the $2C$ and the $3C$ shielded data, we need TSV bundles containing 32 and 24 TSVs, respectively. For the shielding approaches, the two required power/ground TSVs are embedded into the TSV bundles as shields. For previous TSV CACs, except the $4LAT$ [17], the encoded data lines cannot be transmitted together with stable lines over one array without violating the pattern conditions. Thus, for the analysis of previous 3D CACs, we

assume that a separated bundle exists for the power/ground TSVs. To obtain the minimum overhead for the $6C$ [16], $4LAT$ [17] and $6C$ -FNS[18] coding, a 5×4 , 3×9 , and 3×8 TSV array is required, respectively. The power and delay quantities of the interconnects are determined with Spectre circuit simulations, employing drivers composed of two 22 nm PTM inverters of strength $2 \times$ and $6 \times$, and the full 3π - RLC circuits of the interconnects to consider possible inductance effects. To determine the CODEC complexity, all encoder/decoder pairs are synthesized in a commercial 40 nm technology, and the resulting gate equivalents (GE) are reported. Here, the CODEC delay is not reported, since it can be hidden in a pipeline. For the growth in coding complexity as well as the asymptotic overhead, a minimum of 5 % of stable lines is assumed in each TSV array.

The results are presented in Table 3. The table reveals that the presented coding approach outperforms all existing 3D CACs significantly in all metrics. Existing 3D CACs reduce the TSV delay by a maximum of 9.4 % ($4LAT$ [17]), while for the presented 3D CAC technique, the TSV delay reduction can be $4 \times$ larger ($2C$ shielding: 37.8 %). Additionally, the $2C$ shielding reduces the metal wire delay by 47.6 % and does not require a CODEC design. In comparison, the $4LAT$ [17] approach does not optimize the metal wire delay and requires a huge CODEC of 1,915 GE. Furthermore, a $4LAT$ [17] data encoding dramatically increases the TSV and metal wire power consumption by 50.1 % and 46.0 %, respectively. From the existing 3D CAC techniques only the $6C$ coding [17] leads to an acceptable interconnect power consumption as it increases the TSV power consumption by only 8.1 %, while it decreases the metal wire power consumption by 7 %. Here, the decrease in the metal wire power consumption is caused by the realization of the $6C$ 3D CAC encoder which is composed of M N-bit 2D FPF CACs (one for each row) [17]. In contrast, our proposed technique based on an FTF CAC reduces the metal wire power consumption by 21.9 % and, if extended to a low-power 3D CAC, it further decreases the TSV power consumption by over 5 %. Without the low-power extension, our FTF approach leaves the TSV power consumption almost unaffected but still shows the same metal wire power reduction (21.9 %). However, we think here it is not worthwhile to implement our 3D CAC technique, based on the FTF encoding, without the low power extension, as the only drawback is an increase in the TSV delay by less than 0.1 %, while it provides power saving by up to 6.5 %. As expected, our approach based on shielding techniques on average does not significantly affect the TSV or metal wire power consumption.

Finally, it is worth mentioning that the actual performance improvements, presented in this section, are consistent with the theoretical predicted ones from Section 6 (max. deviation 1 %). This shows the accuracy of the presented underlying crosstalk classification. Additionally, it once again validates that, in modern TSV arrays, inductance effects are neglectable for digital signals.

Table 3: Effect of the proposed and the existing 3D CACs on the TSV/metal wire delay ($\Delta T_{p,3D}/\Delta T_{p,2D}$), the TSV/metal wire power consumption ($\Delta P_{3D}/\Delta P_{2D}$), the overhead, the CODEC gate equivalent (GE) and the CODEC growth

Method	Pattern Set	Input Data Width $n = 16$ -bit					Asymptotic ($\lim_{n \rightarrow \infty}$)		
		$\Delta T_{p,3D}$	$\Delta T_{p,2D}$	Overhead	CODEC	ΔP_{3D}	ΔP_{2D}	Overhead	CODEC Growth
This Work: 2D to ω_m/ω_e TSV CAC for TSV Arrays	FTF BP ($\alpha=0$)	-20.4%	-47.6%	43.8%	411 GE	1.2%	-21.9%	44%	$O(n)$
	FTF BP ($\alpha=0.5$)	-20.3%	-47.6%	43.8%	411 GE	-5.3%	-21.9%	44%	$O(n)$
	2C Shielding	-37.8%	-47.6%	87.5%	0 GE	1.8%	0.0%	95%	\emptyset
	3C Shielding	-20.6%	-21.3%	37.5%	0 GE	-1.4%	0.0%	45%	\emptyset
This Work: 2D to ω_m/ω_e TSV CAC for Hex. TSV Bundles	FTF BP ($\alpha=0$)	-17.9%	-47.6%	43.8%	411 GE	-2.8%	-21.9%	44%	$O(n)$
	FTF BP ($\alpha=0.5$)	-17.9%	-47.6%	43.8%	411 GE	-5.2%	-21.9%	44%	$O(n)$
	2C Shielding	-36.7%	-47.6%	87.5%	0 GE	1.2%	0.0%	95%	\emptyset
	3C Shielding	-20.1%	-21.3%	37.5%	0 GE	-0.1%	0.0%	45%	\emptyset
Previous 3D CACs for TSV Arrays	6C [16]	-6.9%	0.0%	25.0%	181 GE	8.1%	-7.0%	44%	$O(n^{1.5})$
	4LAT [17]	-9.4%	0.0%	68.8%	1915 GE	50.1%	46.0%	80%	$O(e^n)$
	6C-FNS[18]	-6.9%	0.0%	50.0%	718 GE	25.7%	19.1%	50%	$O(n)$

Summarized, an FTF encoding can be exploited to effectively reduce the crosstalk delay and the power consumption simultaneously. A 2C shielding leads to the biggest (crosstalk) delay reduction but it leaves the power consumption almost unaffected. Furthermore, in contrast to the FTF encoding, the high line/TSV overhead of the 2C shielding often makes the approach unsuitable, due to the still relatively large TSV dimensions and limitations in the available silicon area. Thus, a good overall compromise offers the presented remapping of the FTF CAC with bus partitioning (highlighted in Table 3), as it results in: the lowest asymptotic overhead (44%), the highest metal wire crosstalk reduction, the second biggest TSV crosstalk reduction and the lowest power consumption. Furthermore, in contrast to most previous 3D CACs, the encoder overhead scales linearly when stable lines are exploited for bus partitioning which makes the technique even applicable for large TSV arrangements.

8. Conclusion

In this work we presented the first low-power crosstalk avoidance technique for 3D integration, called 2D CAC to ω_m/ω_e TSV CAC mapping. In the first part of this work we have proven theoretically and by means of experimental results that the edge effects make previous TSV crosstalk avoidance techniques inefficient. We also outlined that an efficient crosstalk avoidance method should also reduce the metal wire, not only the TSV, crosstalk peak and furthermore should not increase the power consumption drastically. Our proposed technique overcomes all limitations. Additionally, our approach results in the first CAC applicable for hexagonal as well as quadratic TSV arrangements. In our technique, the switching characteristics of 2D CAC pattern sets are exploited by a bit-to-TSV mapping that results in a simultaneous reduction of the TSV and metal wire crosstalk. For our approach, we analyzed different underlying 2D CACs, with the result that our technique always significantly outperforms all existing 3D CACs. Our technique shows a maximum TSV and metal wire delay reduction of 37.8% and 47.6%, respectively. Hereby, the technique induces a negligible increase in the

TSV power consumption while it decreases the metal wire power consumption by 21.9%. In comparison, previous approaches reduce the TSV delay by a maximum of 9.4% (4LAT CAC), while providing no optimization of the metal wire delay. Additionally, the 4LAT CAC results in significantly higher hardware costs and a dramatic increase in the interconnect power consumption (ca. 50%). Furthermore, we presented an extension to our 3D CAC technique which allows for a simultaneous power consumption and crosstalk noise reduction of the TSVs and the metal wires in 3D ICs. This low-power 3D CAC technique reduces the delay and the power consumption of modern metal wire buses by 47.6% and 21.9%, respectively, while it simultaneously reduces the delay and power consumption of modern TSV arrangements by up to 20.4% and 5.3%, respectively.

Acknowledgment

This work is funded by the German Research Foundation (DFG) project PI 447/8-1.

References

- [1] V. F. Pavlidis, E. G. Friedman, Three-dimensional Integrated Circuit Design, Morgan Kaufmann, San Francisco, California, 2009.
- [2] S. Pasricha, Exploring serial vertical interconnects for 3D ICs, in: 2009 46th ACM/IEEE Design Automation Conf., 2009, pp. 581–586.
- [3] D. U. Lee et al., A 1.2 V 8 Gb 8-Channel 128 GB/s High-Bandwidth Memory (HBM) Stacked DRAM With Effective I/O Test Circuits, IEEE Journal of Solid-State Circuits 50 (1) (2015) 191–203.
- [4] B. J. L. C. Duan, S. P. Khatri, On and Off-Chip Crosstalk Avoidance in VLSI Design, 2010th Edition, Springer, 2014.
- [5] A. Garcia-Ortiz, L. Bamberg, A. Najafi, Low-power coding: Trends and new challenges, Journal of Low Power Electronics 13 (3) (2017) 356–370.
- [6] C. Xu, H. Li, R. Suaya, K. Banerjee, Compact AC Modeling and Performance Analysis of Through-Silicon Vias in 3-D ICs, IEEE Trans. Electron Devices 57 (12) (2010) 3405–3417.
- [7] D. H. Kim, S. Mukhopadhyay, S. K. Lim, Fast and Accurate Analytical Modeling of Through-Silicon-Via Capacitive Coupling, IEEE Trans. Components, Packaging and Manufacturing Technology 1 (2) (2011) 168–180.

- [8] L. Bamberg, A. Garcia-Ortiz, High-level energy estimation for submicrometric TSV arrays, *IEEE Trans. Very Large Scale Integration (VLSI) Systems* PP (99) (2017) 1–11.
- [9] S. Piersanti et al., Algorithm for extracting parameters of the coupling capacitance hysteresis cycle for TSV transient modeling and robustness analysis, *IEEE Trans. Electromagn. Compatibility* 59 (4) 1329–1338.
- [10] S. Piersanti, F. de Paulis, A. Orlandi, M. Swaminathan, V. Ricciuti, Transient Analysis of TSV Equivalent Circuit Considering Nonlinear MOS Capacitance Effects, *IEEE Trans. Electromagn. Compatibility* 57 (5) (2015) 1216–1225.
- [11] A. E. Engin, S. R. Narasimhan, Modeling of crosstalk in through silicon vias, *IEEE Trans. Electromagn. Compatibility* 55 (1) 149–158.
- [12] C. Qu, R. Ding, X. Liu, Z. Zhu, Modeling and optimization of multiground TSVs for signals shield in 3-D ICs, *IEEE Trans. Electromagn. Compatibility* 59 (2) 461–467.
- [13] Y. Peng, T. Song, D. Petranovic, S. K. Lim, Silicon Effect-Aware Full-Chip Extraction and Mitigation of TSV-to-TSV Coupling, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems* 33 (12) (2014) 1900–1913.
- [14] Z. Xu, J. Q. Lu, Three-dimensional coaxial through-silicon-via (TSV) design, *IEEE Electron Device Letters* 33 (10) 1441–1443.
- [15] J. Cho et al., Modeling and Analysis of Through-Silicon Via (TSV) Noise Coupling and Suppression Using a Guard Ring, *IEEE Trans. Components, Packaging and Manufacturing Technology* 1 (2) (2011) 220–233.
- [16] R. Kumar, S. P. Khatri, Crosstalk avoidance codes for 3D VLSI, in: *Design, Automation Test in Europe Conf. Exhibition (DATE)*, 2013, 2013, pp. 1673–1678.
- [17] Q. Zou, D. Niu, Y. Cao, Y. Xie, 3DLAT: TSV-based 3D ICs crosstalk minimization utilizing Less Adjacent Transition code, in: *2014 19th Asia and South Pacific Design Automation Conf. (ASP-DAC)*, 2014, pp. 762–767.
- [18] X. Cui, X. Cui, Y. Ni, M. Miao, J. Yufeng, An enhancement of crosstalk avoidance code based on Fibonacci numeral system for through silicon vias, *IEEE Trans. Very Large Scale Integration (VLSI) Systems* PP (99) (2017) 1–10.
- [19] U. Kang et al., 8Gb 3D DDR3 DRAM using through-silicon-via technology, in: *2009 IEEE Int. Solid-State Circuits Conf. - Digest of Technical Papers*, pp. 130–131.
- [20] L. Bamberg, A. Najafi, A. García-Ortiz, Edge effects on the TSV array capacitances and their performance influence, *Integration, the VLSI Journal*, In Press.
- [21] B. Vaisband, E. G. Friedman, Hexagonal TSV bundle topology for 3-D ICs, *IEEE Trans. Circuits and Systems II: Express Briefs* 64 (1) (2017) 11–15.
- [22] L. Bamberg, A. Najafi, A. García-Ortiz, Edge effect aware crosstalk avoidance technique for 3D integration, in: *2017 27th Int. Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)*.
- [23] G. Katti, M. Stucchi, K. Meyer, W. Dehaene, Electrical Modeling and Characterization of Through Silicon via for Three-Dimensional ICs, *IEEE Trans. Electron Devices* 57 (1) (2010) 256–262.
- [24] M. Tsai et al., Investigation on Cu TSV-induced KOZ in silicon chips: Simulations and experiments, *IEEE Trans. Electron Devices* 60 (7) (2013) 2331–2337.
- [25] T. Ramadan, E. Yahya, Y. Ismail, M. Dessouky, Coupling capacitance in through-silicon vias: non-homogeneous medium effect, *Electronics Letters* 52 (2) (2016) 152–154.
- [26] T. Bandyopadhyay et al., Rigorous Electrical Modeling of Through Silicon Vias (TSVs) With MOS Capacitance Effects, *IEEE Trans. Components, Packaging and Manufacturing Technology* 1 (6) (2011) 893–903.
- [27] C. Duan, C. Zhu, S. P. Khatri, Forbidden transition free crosstalk avoidance CODEC design, in: *2008 45th ACM/IEEE Design Automation Conf.*, 2008, pp. 986–991.
- [28] C. Raghunandan, K. S. Sainarayanan, M. B. Srinivas, Process variation aware bus-coding scheme for delay minimization in VLSI interconnects, in: *9th Int. Symposium on Quality Electronic Design (isqed 2008)*, 2008, pp. 43–46.
- [29] C. Duan, A. Tirumala, S. P. Khatri, Analysis and avoidance of cross-talk in on-chip buses, in: *HOT 9 Interconnects. Symposium on High Performance Interconnects*, 2001, pp. 133–138.
- [30] B. Victor, K. Keutzer, Bus encoding to prevent crosstalk delay, in: *IEEE/ACM Int. Conf. on Computer Aided Design. ICCAD 2001. IEEE/ACM Digest of Technical Papers (Cat. No.01CH37281)*, 2001, pp. 57–63.
- [31] K. Xu, E. G. Friedman, Scaling trends of power noise in 3-D ICs, *Integration, the VLSI Journal* 51 (C) (2015) 139–148.
- [32] Y. H. Chen, C. P. Chiu, R. Barnes, T. Hwang, Architectural evaluations on TSV redundancy for reliability enhancement, in: *Design, Automation Test in Europe Conf. Exhibition (DATE)*, 2017, 2017, pp. 566–571.
- [33] R. A. Brualdi, *Combinatorial Matrix Classes -*, 1st Edition, Cambridge University Press, Cambridge, 2006.
- [34] V. Granville, M. Krivanek, J. P. Rasson, Simulated annealing: a proof of convergence, *IEEE Trans. Pattern Analysis and Machine Intelligence* 16 (6) (1994) 652–656.
- [35] P. E. Landman, J. M. Rabaey, Architectural power analysis: The dual bit type method, *IEEE Trans. Very Large Scale Integration (VLSI) Systems* 3 (2) (1995) 173–187.
- [36] T. Song, C. Liu, Y. Peng, S. K. Lim, Full-chip signal integrity analysis and optimization of 3-D ICs, *IEEE Trans. Very Large Scale Integration (VLSI) Systems* 24 (5) (2016) 1636–1648.



Lennart Bamberg received the B.Sc. and M.Sc. degree in Electrical and Information Engineering from the University of Bremen, Germany, in 2014 and 2016, respectively. He is currently working towards the Ph.D. degree in the Institute of Electrodynamics and Microelectronics at the University of Bremen, Germany. His research interests focus

mainly on low-power interconnect architectures for heterogeneous 3D integrated circuits. Mr. Bamberg is awarded for the best paper at PATMOS 2017.



Amir Najafi received the B.Sc. and M.Sc. degree in Electronics Engineering from Azad University of Qazvin, Iran, in 2010 and 2014, respectively. He is currently working towards the Ph.D. degree in the Institute of Electrodynamics and Microelectronics at the University of Bremen, Germany. His research inter-

ests focus mainly on low-power interconnect architectures.



Alberto García-Ortiz is currently full professor for the chair of integrated digital systems at the University of Bremen, Germany. Dr. García-Ortiz received the Outstanding dissertation award in 2004 from the European Design and Automation Association. In 2005, he received from IBM an innovation award for contributions to leakage

estimation. He serves as editor of JOLPE and is reviewer of several conferences, journals, and European projects.

His interests include low-power design and estimation, communication-centric design, SoC integration, and variations-aware design.