Performance-Aware Interconnect Delay Insertion Against EM Side-Channel Attacks

Performance-Aware Interconnect Delay Insertion Against EM Side-Channel Attacks

Minmin Jiang* and Vasilis F. Pavlidis†
Advanced Processor Technologies group, Department of Computer Science, University of Manchester
Email: *minmin.jiang@manchester.ac.uk, †vasileios.pavlidis@manchester.ac.uk

***Abstract*—Random Delay Insertion (RDI) has been shown to be an effective countermeasure to side-channel attacks (SCAs) on on-chip power networks. RDI effectively reduces the correlation between the power dissipation and the processed data. However, random delay insertion can degrade circuit performance. Considering the theoretical benefits of delay insertion, this paper proposes a novel methodology that adds delay to interconnect buses to mitigate electromagnetic (EM) SCAs without degrading bus latency. The methodology comprises an efficient delay insertion scheme that hinders electromagnetic attacks, where the delay is inserted into the *boundary lines* of the bus. As the worst-case bus latency is determined by the lines that drive the maximum cross-coupling capacitance, inserting delay at the *boundary lines* does not affect the circuit performance as these lines always drive a lower capacitance. The inserted delay improves the security strength of the bus to EM attacks due to the reduction of the correlation between EM emissions and transmitted data, making the methodology effective and directly applicable with negligible overhead. The technique is applied to interposer based off-chip memory buses due to the increasing adoption of 2.5-D integrated systems (although the method is effectively applicable to any interconnect bus). Simulation results show that the technique decreases *SNR* below 1, which makes EM attacks unsuccessful, and do not increase the (worst-case) bus latency, sustaining the overall circuit performance. Consequently, the proposed method provides a superior EM SCA mitigation method compared to the state-of-the-art. Indeed, theoretical analysis and simulation results demonstrate that the new technique can offer the same level of protection against SCAs with better performance than other hardware RDI countermeasures.**

***Index Terms*—interconnection, delay insertion, crosstalk, coupling capacitance, electromagnetic emission, side-channel attack**

I. Introduction

Advanced embedded systems, such as smart cards often use cryptographic algorithms to protect their data. Although software algorithms can provide high security to the sensitive data, the side channels of the systems allow the attackers to measure the voltage fluctuations, power consumption, temperature, timing or EM emissions and compromise the system [1]. Side-channel attacks (SCAs) can exploit the measured information and recover the key at low cost [2]. Among these side-channel attacks, correlation power attack (CPA) is one of the most powerful and efficient attack methods. Countermeasures against CPA at circuit level can be classified into two categories [3]: 1) **flattening** the power consumed within critical clock cycles; 2) **randomizing** the processing power to reduce the correlation between the processed data and the consumed power. RDI is an effective technique against CPA which belongs to the latter category [3].

RDI countermeasures have been utilized into the datapaths of microprocessors [4], FPGA platforms [5], ASIC designs [6], and mask encryption algorithms on microprocessor [7]. These countermeasures effectively reduce the correlation between the assumed power model and measured power consumption, thereby preventing potential correlation power attacks. However, these RDI implementations can degrade circuit performance, or make timing closure more challenging as the timing slack of paths decreases, which is an important limitation.

Inspired by the idea to randomize power consumption through RDI in CPAs, a novel technique is proposed to mitigate EM attacks by decorrelating EM emissions with data transferred on interconnect buses. Randomly delaying bit lines as in RDI, however, has a detrimental effect on bus performance. Hence, an alternative approach is devised that does not degrade the bus latency. This improvement is achieved by reducing the cross-coupling capacitance of specific interconnect bus wires. Indeed, at nanometer scale fabrication technologies, the transmission latency of the interconnects becomes largely data-dependant due to the dominant coupling capacitance [8]. For the 2.5-D packaging paradigm, thick off-chip interconnects form wide buses, that connect two or more dies on the same substrate [9], exhibiting high cross-coupling capacitance. By using the novel delay insertion methodology, the cross-coupling capacitance is reduced, along with the correlation between the EM emissions and the processed data.

Existing countermeasures for EM attacks on-chip, include techniques that introduce additional resources or adapt specific steps of the IC design flow, such as a low-level metal routing scheme to resist EM attacks. The cryptographic core is routed with low-level metal layers to suppress the critical signatures before the EM emission reaches the top metal layer [10]. However, low-level metal routing can lead to routing congestion, higher interconnect resistance, and, therefore, performance degradation. Additionally, these techniques are not applicable to off-chip buses on interposers as only few metal layers are available for routing compared to the on-chip interconnect stack that comprises over ten layers in modern fabrication processes. Comparing with state-of-the-art EM resilience methods, the proposed methodology offers a superior choice for interconnect buses.

The contributions of this paper are as follows:

This work is funded by European Commission under the Horizon 2020 Framework Programme for Research and Innovation through the EuroExa project under Grant 754337.

978-1-6654-0083-1/21/$31.00 ©2021 Crown

- A novel methodology of non-random delay insertion technique against EM attacks without degrading the circuit performance, applicable to any interconnect bus.
- A systematic theoretical analysis of the relationship between the added delay with the security metrics, such as correlation coefficient, *SNR*, etc..
- Specifically, an interposer-based interconnect is modeled and simulated to demonstrate the effectiveness of the delay insertion methodology.

The paper is structured as follows. Preliminaries about AES algorithm, correlation EM attack, *SNR* and EM attack flow are introduced in Section II and the delay insertion methodology is presented in Section III. The architecture of the delay insertion scheme and interconnect bus model are described in Section IV. Simulation results are analyzed in Section V. Finally, conclusions are drawn in Section VI.

II. Preliminaries

Fundamental information on advanced encryption standard (AES), correlation electromagnetic attack (CEMA), and signal-to-noise ratio (*SNR*) are introduced in this section. Furthermore, the block diagram for the CEMA analysis on 128-bit AES is presented in this section.

A. AES

AES is a symmetric block cipher that encrypts messages segmented into blocks [11]. The encryption is symmetric because the same key is used for both encryption and decryption. AES is currently widely used for the encryption of sensitive data and is, therefore, the most commonly targeted algorithm of side-channel attacks. When implemented for blocks of 128 bits and a 128-bit (or 256-bit) key, AES normally has ten encryption rounds. For each round, a different sub-key is generated from the key-generator, as shown in Fig. 1. The sub-key used in the first round is the attack target.

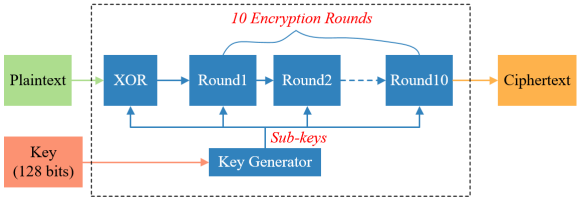


Fig. 1. Block diagram of the AES encryption algorithm.

B. CEMA

The current flowing in the interconnections generates a magnetic field around each wire. When the magnetic field is coupled by a near-field probe, the induced voltage at the probe terminal is correlated with the rate of change of the current flowing in the wire. Furthermore, the sensitive information transmitted in the wire can be attacked by adversaries through statistical methods [12]. The correlation attack to retrieve the useful message normally has three steps:

1) *Calculate the information leakage*: The assumed EM leakage is computed from the number of output transitions where the number of transitions ($0 \rightarrow 1$ or $1 \rightarrow 0$) effectively determine the magnitude of the EM field. By injecting m pre-processed n-bit plaintext vectors $I_m\,[n-1:0]$ into the AES circuit and sweeping all the possible subkeys, an output vector $H_m\,[n-1:0]$ can be computed, which is the Hamming Weight of the XOR result of the plaintext and guessed subkey in the first round of the AES encryption process [15].
2) *Collect the EM traces*: When using near-field coupling techniques, there is no direct electrical connection between the probe and the circuit under test, leading to efficient contactless measurements. The known stream of input vectors $I_m\,[n-1:0]$ and a fixed secret key $K\,[n-1:0]$ produce n-bit output vectors $O_m\,[n-1:0]$. The EM traces generated by the output vectors are measured at the probe terminal as sampled coupled voltages $V_m\,[n-1:0]$.
3) *Calculate the correlation coefficient and attack the key*: In this step, the coupled voltage is sampled and the information leakage is determined, simultaneously, at time instance t within the clock period T. Note that an encrypted word is assumed to be transferred on the interconnect in each clock cycle with clock period T. For each guessed key k, the correlation between the assumed leakage and the measured EM traces is calculated by,

$$\rho_{k,t} = \frac{E\left[\left(V_m^t - \overline{V_m^t}\right)\left(H_m^k - \overline{H_m^k}\right)\right]}{\sqrt{Var\left(V_m^t\right) Var\left(H_m^k\right)}}, \quad k \in \{0, 2^n - 1\}, t \in \{0, T\}, \tag{1}$$

where the numerator demonstrates the co-variance between the coupled voltage and assumed leakage and Var is the variance of the dataset. When evaluating all of the correlation coefficients $\rho_{k,*}$ at any time point $(*)$ for each guessed key, a maximum $\rho_{k,\max}$ is determined. With this process, a vector $\{\rho_{k_1,\max}, \rho_{k_2,\max}, ..., \rho_{k_m,\max}\}$ is formed for all m possible keys. The key with the highest correlation coefficient corresponds to the correct key.

C. SNR

In this and other security related papers, *SNR* is regarded as the ratio between the correlation coefficient of the correct key ρ_{corr} and the maximum correlation coefficient of the incorrect key ρ_{incorr} for all samples in time, as follows [13],

$$SNR = \frac{\rho_{corr}}{\rho_{\max,incorr}}. \tag{2}$$

When *SNR* drops below 1, a system is considered to provide high immunity to side-channel attacks in real-world scenarios where noise is considered [13].

D. EM Attack Flow on 128-bit AES

The EM side-channel attack flow, which exploits multiple EM traces on the symmetric 128-bit AES algorithm is shown in Fig. 2. The substitution block (SBox) is the only nonlinear part of the AES algorithm, typically implemented by using look-up tables (LUTs) or logic operations, such as XORing. However, either implementation approach causes an

overhead in area or dynamic power of the circuit that supports encryption. Therefore, in 2.5-D ICs, a LUT based SBox is implemented on a customized off-chip read-only ROM [15], as shown in Fig. 2. The address and substitution data can be sent and received through the interconnections routed within the distribution layers (RDLs) of the interposer.

The near-field probe can optimally be placed to perform an EM attack using the method described in [14] [15]. When the probe is placed at the optimal position above the bus, a series of plaintexts $\{pt_1, pt_2, ...pt_{N_e}\}$ and a fixed key is fed into the system for encryption. The amplitude of the coupled voltage at the probe terminal is correlated with the number of transitions (Hamming Weight) happening on the bus. By repeating the CEMA method mentioned in Section II-B, the entire key can be recovered byte by byte.

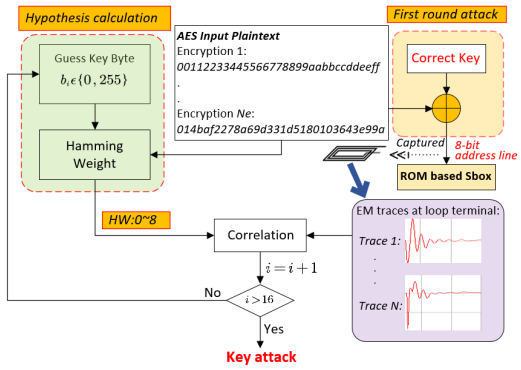


Fig. 2. CEMA analysis of 128-bit AES algorithm.

III. Delay Insertion Methodology

In this section, a performance-aware delay insertion strategy to resist EM off-chip memory bus attacks is presented. The core idea of the method is to delay the *boundary lines* by a constant delay determined during the design process. This approach can help improve the security and does not sacrifice the performance of the circuit, a twofold advantage that makes the new method highly beneficial. First, the concept of performance-aware delay insertion is introduced in this section. Then, the effect of the delay insertion on the total bus latency and attack efficiency are analyzed, respectively.

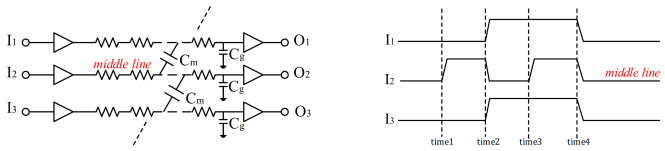
A. Performance-aware Delay Insertion

As shown in Fig. 3(a), where the bit lines are treated as *RC* interconnects, the transmission latency of a bus is proportional to the wire resistance and capacitance. In this model, the bus latency is effectively determined by the capacitance driven by each line, which depends on the switching conditions of the adjacent lines, as its resistance depends on the geometric characteristics and, nominally, is the same for all lines. The capacitance for each line is, respectively, composed of the ground capacitance (C_g) and coupling capacitance (C_m), as shown in Fig. 3(a). The interconnection capacitance of the *middle line* (I_2) can be calculated by [16],

$$C_{middle} = C_g + C_m \left| \frac{\Delta V_{12}}{V_{dd}} \right| + C_m \left| \frac{\Delta V_{23}}{V_{dd}} \right|, \qquad (3)$$

where V_{dd} is the voltage of the power supply, and ΔV_{12}, ΔV_{23} is the voltage difference between the *middle line* (I_2) with its two neighbours (I_1, I_3), respectively.

For an example pattern shown in Fig. 3(b), where the three lines switch in the same direction (*time4*), in opposite direction (*time2*), and only a single line transitions (*time1, time3*), the capacitance of the *middle line* corresponds to C_g, $C_g + 4C_m$, and $C_g + 2C_m$, respectively. For interconnect buses due to the Miller effect, the highest capacitance is driven when a line switches to the opposite direction of its neighbours (*time2* in Fig. 3(b)). Thus, the worst bus latency among all of the bits is proportional to $C_g + 4C_m$ (D_2 in Fig. 4).



(a) (b)

Fig. 3. (a) Interconnect model of three bit lines of a *k*-bit bus, and (b) three data transition scenarios: same direction transition, single transition, and opposite direction transition.

As illustrated in Fig. 4, the delay insertion strategy aims to ***skew the transition time of the two boundary wires.*** This intentional skew does not increase the latency of the bus, as the latency of these lines is always smaller than the worst-case latency of the bus. As the edge lines have only one adjacent line, the capacitance that these lines can drive range from C_g to $C_g + 2C_m$ when switching in the same and opposite direction, respectively, to the neighbouring line. Inserting some delay to these lines decreases (increases) the coupling capacitance for switching in the opposite (same) direction. For the case of *time2* in Fig. 4, when the delay Δt inserted into the *boundary lines* (I_1, I_3) is greater or equal to their transition time (t_{tran}), the coupling capacitance of the *boundary lines* is reduced. If the skew is selected appropriately, the bus latency of the *boundary lines* (denoted as D_1, D_3) can still be smaller than the worst-case latency (denoted as D_2), as shown in Fig. 4. The precise delay to be added is determined by analysing these cases during design time and can be inserted with negligible overhead. Furthermore, the inserted delay decreases the correlation between the processed data and the coupled voltage at the probe terminal, which helps improve the resilience against EM side-channel attacks. Therefore, the performance-aware delay insertion can serve as a security countermeasure without degrading the circuit performance.

In the 8-bit bus case, the worst-case switching pattern is where all adjacent bit lines switch either from 1 to 0 or from 0 to 1, alternatively, between two successive pieces of data. In this pattern, all but the bit lines at the edges of the bus drive the maximum capacitance $C_g + 4C_m$, while the lines at both edges drive a capacitance $C_g + 2C_m$. Consequently, by inserting a delay that is less than the delay incurred by driving a capacitance of $2C_m$ into two *boundary lines*, which drive a

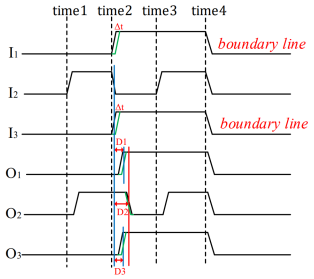


Fig. 4. Performance-aware delay insertion into *boundary lines* to improve the resilience against EM side-channel attacks.

lower capacitance, the latency of the bus does not decrease but security resiliency can be offered by decreasing the correlation coefficient. The available range of delays and the effect on the security resilience of the interconnect are discussed for a specific bus on a silicon interposer in the following sections.

B. Quantifying Effect of Added Delay on Bus Latency

In this subsection, in order to quantify the effect of the delay insertion on the total bus latency, only the worst case is considered. That is, the bus latency is determined by the switching condition that leads to the highest delay for a bit line. This condition happens where all adjacent bit lines switch in the opposite direction. Using the typical 50% delay metric in static logic circuits, the transmission latency of an interconnect is estimated by [17],

$$\begin{aligned} T &= 0.4R_tC_t + 0.7\left(R_{buffer}C_t + R_{buffer}C_L + R_tC_L\right) \\ &\approx \left(0.7R_{buffer} + 0.4R_t\right)C_t, \quad if\ C_L \ll C_t, \end{aligned} \tag{4}$$

where R_t, C_t are the resistance and capacitance of the wire, respectively. R_{buffer} is the on-resistance of the driver and C_L is the load capacitance. According to Eq. (3) and Eq. (4), when delay Δt is inserted into the *boundary lines*, the capacitance of the *boundary line* is

$$\begin{aligned} C_{boundary} &= C_g + C_m\left|\frac{\Delta V_2\left(0\right) - \Delta V_1\left(\Delta t\right)}{V_{dd}}\right| + \\ C_m&\left|\frac{\Delta V_2\left(0\right) - \Delta V_3\left(\Delta t\right)}{V_{dd}}\right| = C_g + nC_m,\ 1 \leqslant n \leqslant 2, \end{aligned} \tag{5}$$

where $n = 1$ for $\Delta t \geqslant t_{tran}$ and $n = 2$ for $\Delta t = 0$.

By substituting Eq. (5) into Eq. (4), the total bus latency of the *boundary line* (T_b) can be estimated by,

$$T_b = \Delta t + \left(0.7R_{buffer} + 0.4R_t\right)\left(C_g + nC_m\right),\ 1 \leqslant n \leqslant 2. \tag{6}$$

When delay Δt is inserted into the *boundary lines*, the coupling capacitance of the line next to the *boundary line* is also reduced, whose latency (T_{nextb}) can be approximated as

$$T_{nextb} = \left(0.7R_{buffer} + 0.4R_t\right)\left(C_g + nC_m\right),\ 3 \leqslant n \leqslant 4, \tag{7}$$

where $n = 3$ for $\Delta t \geqslant t_{tran}$ and $n = 4$ for $\Delta t = 0$.

The bus latency of the rest middle lines (T_{middle}) can be estimated by,

$$T_{middle} = \left(0.7R_{buffer} + 0.4R_t\right)\left(C_g + nC_m\right),\ n = 4. \tag{8}$$

When no delay is added, the transmission latency of the middle lines, described by Eq. (8) is the greatest across the entire bus since the middle lines drive the highest capacitance ($C_g + 4C_m$). From Eqs. (6)-(8), if delay Δt, inserted into the *boundary lines*, is properly selected, the latency of the *boundary line* (T_b) does not surpass the bus latency of the middle lines (T_{middle}). Consequently, the overall speed of the bus is not degraded.

Note that inserting delay into the middle lines can also help reduce the coupling capacitance. However, extensive simulations have shown that inserting delay into middle lines always degrades bus performance. Therefore, emphasis is given to the *boundary lines*.

C. Statistical Analysis of the Attacks

To establish the link between the added delay and the security figures of merit, such as the correlation coefficient and *SNR*, a systematic and theoretical analysis is offered in this subsection.

When Δt is inserted into the bus to temporally shift the transition of the lines, the total captured leakage by the probe V_{total}, is denoted as $V_{total} = V_{t-\Delta t} + V_{noise}$, where V_{noise} is the uncorrelated noise generated from neighbouring wires. As shown in Fig. 5, $V_{t-\Delta t}$ can be replaced by $V_{max} + V_\Delta$, where V_{max} is the maximum coupled voltage at time t, correlated with the leakage model, and V_Δ is the voltage difference due to the shifting.

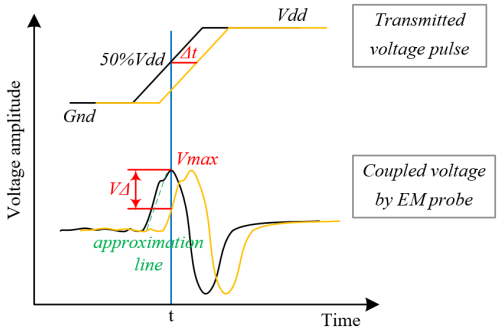


Fig. 5. The peak value of coupled voltage at the probe terminal has a V_Δ difference when delay Δt is inserted to an interconnection (not to scale).

If $Var\left(V_\Delta\right) \gg Var\left(V_{\max}\right)$ is assumed, the correlation between the assumed leakage H and V_{total} can be calculated by [18],

$$\begin{aligned} \rho\left(H, V_{total}\right) &= \frac{E\left(H, V_{total}\right) - E\left(H\right)E\left(V_{total}\right)}{\sqrt{Var\left(H\right)Var\left(V_{total}\right)}} \\ &= \frac{\rho\left(H, V_{\max}\right)}{\sqrt{1 + \frac{1}{SNR}}}\frac{1}{\sqrt{1 + \frac{Var(V_\Delta)}{Var(V_{\max})}}} \\ &\approx \frac{\rho\left(H, V_{\max}\right)}{\sqrt{1 + \frac{1}{SNR}}}\frac{1}{\sqrt{\frac{Var(V_\Delta)}{Var(V_{\max})}}}. \end{aligned} \tag{9}$$

From Eq. (9), $\rho\left(H, V_{total}\right)$ is inversely proportional to $\sqrt{Var\left(V_\Delta\right)}$. According to [9], the pulse voltage coupled at the probe terminal is denoted as $-MI_p\frac{8t}{\tau^2}\exp\left(-\frac{4t^2}{\tau^2}\right)$, where

M, I_p, and τ is the mutual inductance between the probe and the interconnects, peak value of current transmitted on the bus, and pulse width, respectively.

It is assumed that data on all bit lines are launched from a register followed by an I/O buffer since an off-chip bus is investigated. In other words, the register to output path (R2O) is reasonably assumed to exclude combinational paths that may lead to glitches. Furthermore, the inclusion of an I/O buffer can suppress such glitches.

Additionally, the modelling process is used to analytically link the correlation coefficient with the inserted delay. The slew and delay of signal based on the 10%-90% range. Within this range, V_Δ is assumed to change linearly with Δt, and the voltage waveform is approximated by the dashed green line in Fig. 5.

The inserted delay Δt used in this paper is generated from delay lines and uniformly distributed in group $[0, \Delta_{\min}, 2\Delta_{\min}, ...\Delta_{\max}]$, where $\Delta_{\min}$ is the minimum delay and $\Delta_{\max} = k\Delta_{\min}$ (the delay of a line for driving a capacitance of $2C_m$). More details about the circuits that can produce these delays with low overhead are described in Subsection IV-A. Therefore, the variance of V_Δ can be estimated by,

$$\begin{aligned} &Var\left(V_\Delta\right) \approx Var\left(\Delta_t\right) = E\left(\Delta_t^2\right) - \left[E\left(\Delta_t\right)\right]^2 \\ &= \frac{k\left(2k+1\right)\Delta_{\min}^2}{6} - \frac{k\Delta_{\min}^2}{2} = \frac{k\left(k+2\right)}{12}\Delta_{\min}^2. \end{aligned} \tag{10}$$

According to Eq. (10), Eq. (9) can be further approximated as,

$$\begin{aligned} \rho\left(H, V_{total}\right) &\propto \frac{1}{\sqrt{Var\left(V_\Delta\right)}} \approx \sqrt{\frac{12}{k\left(k+2\right)\Delta_{\min}^2}} \\ &\propto \frac{1}{\sqrt{k^2\Delta_{\min}^2 + 2k\Delta_{\min}^2}} = \frac{1}{\sqrt{\Delta_{\max}^2 + 2k\Delta_{\min}^2}}. \end{aligned} \tag{11}$$

From Eq. (11), $\Delta_{\max}$, k, and $\Delta_{\min}$ are critical parameters for the delay insertion technique. Increasing either of the three parameters can help reduce the correlation between the assumed leakage and captured EM emissions. Theoretically, $\Delta_{\min} = 0$, where no delay is inserted and the correlation coefficient is not affected and $\Delta_{\max} = t_{tran}$ as delaying the transition of the *boundary line* by longer than the transition time of the middle lines does not further alter the coupling capacitance. However, for both improving circuit security and sustaining performance, the lower bound, $\Delta_{\min}$ and upper bound, $\Delta_{\max}$ need to be properly determined.

As there are diverse side-channel mitigation techniques that have been proposed, for this new *boundary-line* delay insertion technique (even if deterministic), it will be practically infeasible or prohibitively time-consuming for an attacker to guess what delay-insertion pattern has been adopted in the circuit. Therefore, even though deterministic, the delay insertion strategy is unknown to the attacker, and can provide the off-chip memory bus the resilience against EM side-channel attacks.

IV. Testbench

The architecture of the proposed delay insertion scheme is described in this section. The diagram of the bus data transmission and the delay line are discussed in subsection IV-A, and the interposer-based interconnection model is presented in subsection IV-B, respectively.

A. Hardware Architecture of Delay Insertion Scheme

The proposed delay insertion scheme can be implemented as shown in Fig. 6. Delay is added to two *boundary lines* where the added delay is generated by a delay line and is applied to these interconnect lines. The added delay Δt should be chosen such that the resulting bit line latency does not exceed the worst-case latency where $n = 4$ (see Eq. (6) and (8)).

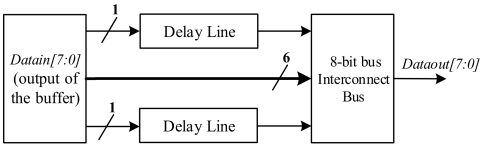


Fig. 6. The two *boundary lines* are selected as the target lines to be delayed with the inserted delay Δt generated by the delay line.

There has been some research relating to the implementation of the delay logic. A tuneable delay line rather than Random D-FF (RDFF), Random Wait-state D-FF (RWDFF), and Random Number Generator (RNG), is assumed here to produce the desired delay as the implementation of these RDI circuits is complex and can easily induce significant overhead in power and/or area.

With the delay line, the propagated data pulse is delayed at a low power cost. A more detailed description about the delay line can be found in [20]. The inserted delay is determined during the design process with the steps described in Section III (specifically using Eqs. (6)-(8) and (11)). Once the desired delay Δt has been determined, the transistors of the delay line can be suitably sized to produce this delay for the worst transition case.

B. Interposer based Interconnect Model

The proposed technique is applied to an interposer-based off-chip memory bus due to the increasing adoption of 2.5-D integrated systems, as shown in Fig. 7. The off-chip memory bus implemented in a silicon interposer is modelled to connect the AES encryption chip (chip1) and the memory chip (chip2), placed on the same substrate, through microbumps. The wire dimensions are designed as the top global interconnection with 65 *nm* technology, as silicon interposers at this technology node have appeared in literature [21]. The *RLC* parameters of the wires are listed in Table. I.

TABLE I
RLC PARAMETERS OF THE INTERCONNECTIONS.

R (Ω/mm)	L (nH/mm)	C_g(fF/mm)	C_c(fF/mm)	C_{total}(fF/mm)
30.56	1.641	41.34	157.64	356.62

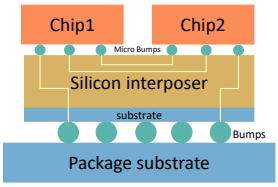


Fig. 7. Interposer based off-chip memory bus [9].

V. Simulation Results

In this section, the simulation setting is initially provided. Then the EM attack results and circuit performance results are shown where delay is inserted into the *boundary lines*. Furthermore, the robustness of the new delay insertion technique is verified.

A. Simulation Environment Settings

An 8-bit bus, implemented in the top metal layer of the redistribution layers of the interposer, is modeled in *ANSYS HFSS*, according to Fig. 7. The near-field probe is modeled as a single turn rectangle coil with 100 μm length and 50 μm width, which exhibits the maximum normalized standard deviation (NSD) of emissions [15]. When the probe is placed vertically over the bus, the S-parameters are generated with frequency sweep, exported from *HFSS*, and imported into *Spectre* for transient analysis of the interconnect in the time domain. The overall design is simulated using a 65 *nm* technology and the nominal voltage V_{dd} is 1.8 V (typical I/O voltage for 65 *nm* technology). To help with the demonstration of the simulation results in the following subsection, the 8-bit bus is depicted in Fig. 8 with annotated bit lines.

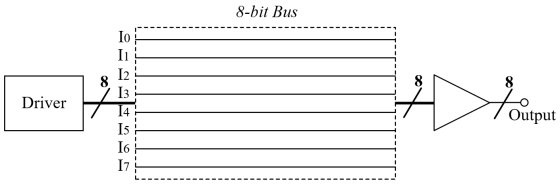


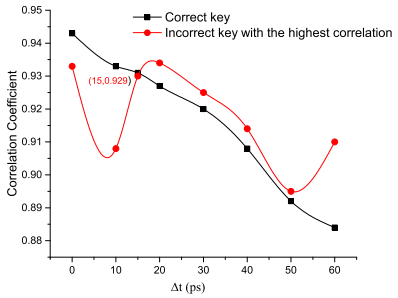
Fig. 8. Structure of the interposer-based bus with annotated bit lines.

Noise generated by the physical devices is not considered here, rather, the effect of critical delay parameters on the circuit security and performance is the main exploration target. *SNR*, as defined in Subsection III-C, are regarded as the figure of merit to evaluate the security and the security results in this paper are based on sweeping all 256 possibilities of the 8-bit input.

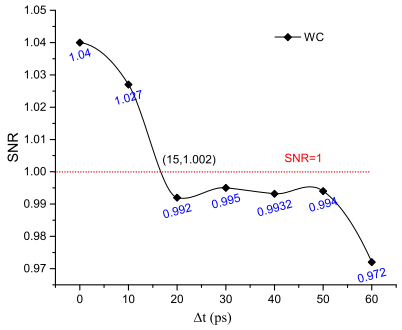
B. SNR and Performance Simulation Results

In this subsection, the efficiency of inserted delay Δt on *SNR* and bus latency is explored. Meanwhile, the lower bound, Δ_{min} and upper bound, Δ_{max} can be determined according to the simulation results.

In the worst-case scenario (all adjacent bit lines switch simultaneously in the opposite direction), Δt is inserted into both *boundary wires* (I_0, I_7 in Fig. 8) to shift in time the transmitted data. EM attacks are performed to extract the secret key. The AES algorithm is repeated 256 times for all possible 8-bit plaintexts and a fixed 8-bit key. The correlation coefficient (for both correct key and incorrect key) and *SNR* are, subsequently, plotted as a function of Δt, as depicted in Fig. 9.



(a)



(b)

Fig. 9. (a) Calculation of correlation coefficient for both the correct key and incorrect key when Δt increases, and (b) for the worst-case (wc) scenario, with the increase in inserted delay Δt, *SNR* decreases.

As illustrated in Fig. 9(a), the lower bound, Δ_{min} should be greater than 15 ps for the delay insertion to be effective, which means the correlation coefficient of the correct key is lower than that of the incorrect key. Furthermore, the plot of *SNR* in Fig. 9(b), used as the figure of merit to evaluate security, demonstrates that the inserted delay Δt provides resistance against EM attacks. *SNR* decreases with increasing Δt. When Δt increases over 15 ps, the *SNR* drops below 1 (dashed red line), meanwhile, the EM attacks fail.

Note that increasing Δt helps improve circuit security, however, how much Δt can be added without degrading the circuit performance needs also to be addressed. As shown in Fig. 10, the total bus latency is calculated from the 50% point of the earliest transition of the output of *Inv1* to the 50% point of the latest transition of the signals at the input of the receiver circuit.

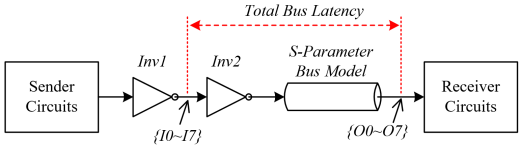


Fig. 10. The start and end point of the measurement of total bus latency.

The simulated total bus latency with delay insertion is shown in Table II and Fig. 11, where the x-axis is the delay added into I_7 (Δt) and the y-axis is the bus latency. If no delay is added, the total bus latency is 240 ps, which is determined by the worst-case switching pattern scenario of middle lines (e.g., I_3). When the inserted delay Δt increases, the total bus latency remains almost unchanged as the coupling capacitance of I_3 does not change, whose latency still dominates the total bus latency. When Δt is greater than 70 ps, the *boundary line* I_7 (added with Δt) starts taking over middle line I_3 and dominates the bus latency. Consequently, if Δt increases over 70 ps, for the specific setup, the speed of the bus starts to degrade. Thus, the intersecting point of the two curves (annotated with the square and circle markers) sets the useful upper bound of Δt (Δ_{max}).

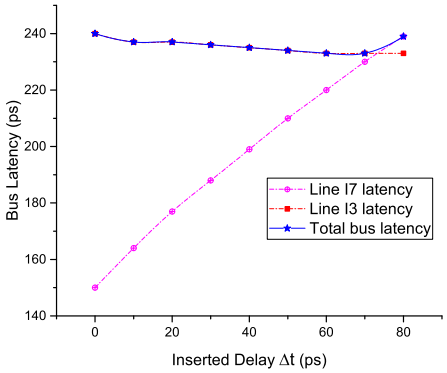


Fig. 11. Total bus latency *vs.* delay inserted into I_7.

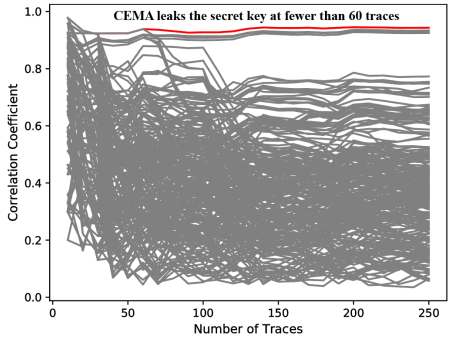
TABLE II
ADDED DELAY INTO I_7 VS. TOTAL BUS LATENCY.

Added delay (ps)	Latency of I_3 (ps)	Latency of I_7 (ps)	Total bus latency (ps)
0	240	150	240
10	237	164	237
20	237	177	237
30	236	188	236
40	235	199	235
50	234	210	234
60	233	220	233
70	233	230	230
80	233	239	239

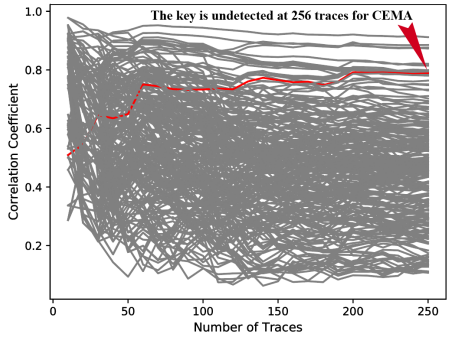
As shown in Fig. 9(b), 11 and Table II, when $\Delta t = 60\,ps$, the *SNR* drops by 6.5% (decrease below 1) and, meanwhile, the total bus latency remains unchanged to sustain the circuit performance (compared with the scenario where no delay is added).

The number of traces needed to attack the secret key for both scenarios, which is widely used in the hardware security field [22]– [24], are illustrated in Fig. 12. The x-axis is the number of traces needed for a successful attack and the y-axis is the correlation coefficient. The 256 traces in each sub-figure correspond to the probability of the corresponding 8-bit key value.

As depicted in Fig. 12(a), when no delay is inserted, the line that corresponds to the correct key (red line) can be distinguished from other guessed key lines with fewer than 60 traces, while if $\Delta t = 60\,ps$ is inserted into selected interconnects, more than 250 traces are recorded and yet the correct key (that corresponds to the red line in Fig. 12(b)) shows a low correlation coefficient and is not detected.



(a)



(b)

Fig. 12. Correlation coefficient vs. number of traces (a) where no delay is added, and (b) where 60 ps is added to the *boundary lines*.

To demonstrate that the proposed methodology is unbiased to encryption keys, the average *SNR* of unprotected interconnects (with no delay added) and protected interconnects ($\Delta t \in (15\,ps, 70\,ps)$) with different keys is, respectively, depicted in Fig. 13. In the interest of space, only ten randomly generated encryption keys are listed here. As shown in Fig. 13, the proposed technique is not biased to a fixed key, where different delays are inserted into the interconnects, *SNR* for all listed keys falls below 1.

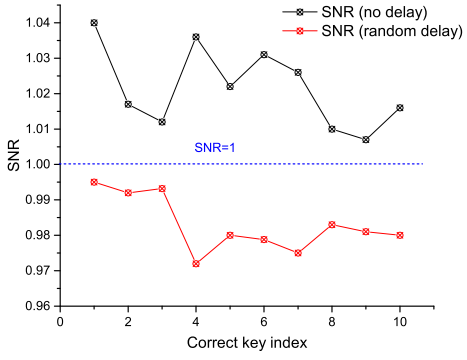


Fig. 13. *SNR* comparison between interconnects with no delay and interconnects with a random delay ($\Delta t \in (15\,ps, 70\,ps)$) inserted, where ten different keys are also generated randomly.

VI. Conclusion

In this paper, a novel delay insertion methodology, applied to interconnect buses to mitigate EM attacks without degrading bus latency, is proposed. The core idea is based on temporally shifting the transition of *boundary lines*, which have the lowest coupling capacitance across the bus when all adjacent bit lines switch in opposite directions simultaneously. In this case, the correlation between the EM emissions and processed data is reduced as well, to achieve *SNR* lower than 1. For the off-chip 8-bit interconnect bus scenario, when a delay of up to 70 ps is inserted into selected interconnects, the *SNR* decreases below 1 and the total bus latency does not increase, demonstrating that the proposed delay insertion methodology offers a superior choice in the resilience against EM side-channel attacks for interconnect buses.

References

[1] F. X. Standaert, "Introduction to side-channel attacks," *Proc. of Secure integrated circuits and systems conference*. pp. 27–42, Dec. 2010.

[2] R. Novak, "Side-channel attack on substitution blocks," *International Conference on Applied Cryptography and Network Security*, pp. 307–318, Oct. 2003.

[3] I. Levi *et al.*, "Data-dependent delays as a barrier against power attacks," *IEEE Transactions on Circuits and Systems I*, Vol. 62, No. 8, pp. 2069–2078, Aug. 2015.

[4] C. Clavier, J. S. Coron, and N. Dabbous, "Differential power analysis in the presence of hardware countermeasures," *International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 252–263, Aug. 2000.

[5] M. Bucci *et al.*, "A countermeasure against differential power analysis based on random delay insertion," *IEEE International Symposium on Circuits and Systems*, pp. 3547–3550, May 2005.

[6] S. Moore *et al.*, "Balanced self-checking asynchronous logic for smart card applications," *Microprocessors and Microsystems*, Vol. 27, No. 9, pp. 421–430, Oct. 2003.

[7] M. Tunstall and O. Benoit, "Efficient use of random delays in embedded software," *IFIP International Workshop on Information Security Theory and Practices*, pp. 27–38, May 2007.

[8] J. S. Yim and C. M. Kyung, "Reducing cross-coupling among interconnect wires in deep-submicron datapath design," *Proceedings Design Automation Conference*, pp. 485–490, Jun. 1999.

[9] V. F. Pavlidis, I. Savidis, and E. G. Friedman, *Three-Dimensional integrated circuit design*, 2^{nd} Ed. Morgan Kaufmann Publishers, 2017.

[10] D. Das *et al.*, "STELLAR: A generic EM side-channel attack protection through ground-up root-cause analysis," *International Symposium on Hardware Oriented Security and Trust*, pp. 11–20, May 2019.

[11] J. Daemen and V. Rijmen, "AES proposal: Rijndael," 1999.

[12] D. Real, F. Valette, and M. Drissi, "Enhancing correlation electromagnetic attack using planar near-field cartography," *Design, Automation & Test in Europe Conference & Exhibition*, pp. 628–633, April 2009.

[13] I. Levi, A. Fish, and O. Keren, "CPA secured data-dependent delay-assignment methodology," *IEEE Transactions on Very Lar Scale Integration Systems*, Vol. 25, No. 2, pp. 608–620, Feb. 2017.

[14] C. Teegarden, M. Bhargava, and K. Mai, "Side-channel attack resistant ROM-based AES S-Box," *Proceedings of IEEE International Symposium on Hardware-Oriented Security and Trust*, pp. 124–129, Jun. 2010.

[15] M. Jiang and V. F. Pavlidis, "A Probe Placement Method for Efficient Electromagnetic Attacks," *International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design*, April 2021.

[16] K. Hirose and H. Yasuura, "A bus delay reduction technique considering crosstalk," *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, Vol. 85, No. 1, pp. 24–31, Jan. 2002.

[17] H. B. Bakoglu and J. D. Meindl, "Optimal interconnection circuits for VLSI," *IEEE Transactions on Electron Devices*, Vol.32, No. 5, pp. 903–909, May 1985.

[18] S. Mangard, "Hardware countermeasures against DPA–a statistical analysis of their effectiveness," *Cryptographers' Track at the RSA Conference*, pp. 222–235, Feb. 2004.

[19] Z. Zhang, A. A. Ding, and Y. Fei, "A Fast and Accurate Guessing Entropy Estimation Algorithm for Full-key Recovery," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 26–48, Mar. 2020.

[20] M. Fujino and V. G. Moshnyaga, "An efficient Hamming distance comparator for low-power applications," *International Conference on Electronics, Circuits and Systems*, pp. 641–644, Sep. 2002.

[21] E. Beigne *et al.*, "An asynchronous NOC architecture providing low latency service and its multi-level design framework," *IEEE Int. Symposium on Asynchronous Circuits and Systems*, pp. 54-63, March 2005.

[22] M. Alioto *et al.*, "Effectiveness of leakage power analysis attacks on DPA-resistant logic styles under process variations," *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 61, No. 2, pp. 429–442, Aug. 2013.

[23] N. Chawla *et al.*, "Extracting side-channel leakage from round unrolled implementations of lightweight ciphers," *IEEE International Symposium on Hardware Oriented Security and Trust*, pp. 31–40, May 2019.

[24] A. Singh *et al.*, "Enhanced power and electromagnetic SCA resistance of encryption engines via a security-aware integrated all-digital LDO," *IEEE Journal of Solid-State Circuits*, Vol. 55, No. 2, pp. 478–493, Oct. 2019.