Journal of
*Low Power Electronics
and Applications*

Article

Towards Energy-Efficient and Secure Computing Systems

Zhiming Zhang and Qiaoyan Yu *

Department of Electrical and Computer Engineering, University of New Hampshire, Durham, NH 03824, USA; zz1017@wildcats.unh.edu

* Correspondence: qiaoyan.yu@unh.edu; Tel.: +1-603-862-1546

Received: 30 October 2018; Accepted: 21 November 2018; Published: 27 November 2018

Abstract: Countermeasures against diverse security threats typically incur noticeable hardware cost and power overhead, which may become the obstacle for those countermeasures to be applicable in energy-efficient computing systems. This work presents a summary of energy-efficiency techniques that have been applied in security primitives or mechanisms to ensure computing systems' resilience against various security threats on hardware. This work also uses examples to discuss practical methods for securing the hardware for computing systems to achieve energy efficiency.

Keywords: energy efficiency; hardware security; computing system; hardware Trojan; side-channel attack; IP piracy; reverse engineering; three-dimensional integrated circuit; network-on-chip

1. Introduction

Depending on the scale, we can categorize computing systems into two types: stand-alone and networked computing systems, as shown in Figure 1. A stand-alone computing system is typically implemented with application-specific integrated circuits (ASICs), field programmable gate arrays (FPGAs), embedded systems, or multiprocessor systems-on-chip (MPSoCs). A stand-alone computing system could also be a node in networked computing systems, such as the Internet of Things (IoTs).

Increasing security issues are concerned in the supply chain of integrated circuits (ICs) [1,2], embedded systems [3], IoTs [4], and cloud computing systems [5]. Intellectual property (IP) piracy, IC overbuilding, reverse engineering, physical attack, counterfeiting chip, and hardware Trojans (HTs) are recognized as critical security threats in maintaining the integrity and trust of ICs. Compared to traditional computing systems, embedded systems are likely to operate in a more hostile environment and could suffer from a larger number of potential attackers and their persistent attacks. Embedded systems are prone to security attacks due to their limited resources available for self-protection and unsafe languages used for application programming. For instance, the high-level languages like C/C++ are prone to control flow attacks (CFAs), such as stack buffer overflows and pointer corruptions [6,7]. IoT devices increasingly interact with the physical world, such as medical networks, power grids, transportation, and government service. Due to the various security policies and mechanisms applied in the ubiquitous IoT devices, it is difficult to track the root cause of the system vulnerabilities. Security threats on cloud computing include data breaches, data loss, account hijacking, malicious insiders, and abuse of cloud services [5].

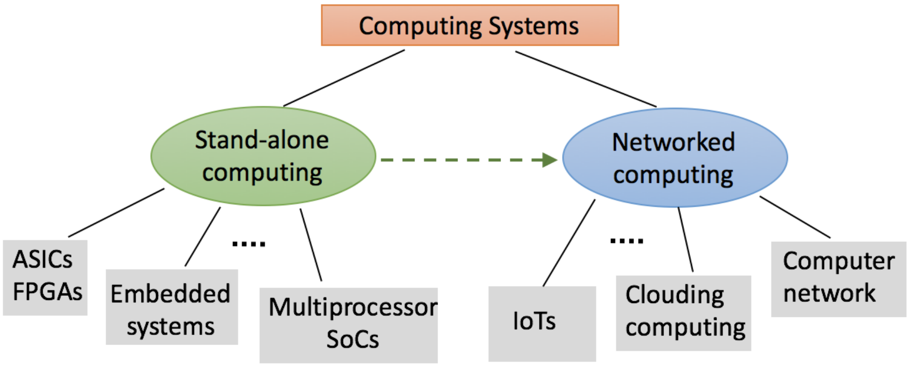


Figure 1. Categories of computing systems.

Besides security requirements, energy efficiency is also desired in computing systems, especially in many embedded systems and IoT devices. Unfortunately, the employment of security policies and mechanisms typically results in large hardware and power overhead. It is challenging to maintain system energy efficiency and high resistance against various security threats at the same time. In this work, we provide a survey on the circuit-level low power and energy efficiency techniques, which are suitable for security primitives and countermeasures against various security threats in integrated circuits and systems.

The remainder of this work is organized as follows. In Section 2, we briefly review the existing techniques for energy efficiency. In Section 3, we conduct a survey on how low power methods have been applied in the security primitives, security mechanisms, and secure computing systems. Detailed design examples are also provided in Section 4 to showcase how power/energy minimization is considered in the implementation of security mechanisms. In Section 5, we conclude this work and foresee the possible design guidelines for circuit-level methods for energy-efficient and secure computing systems.

2. Low-Power and Energy-Efficient Techniques

Techniques for low power and energy efficiency have been developed for a few decades. Those techniques are applicable at the technology, circuit, architecture, system, and application levels [8]. We summarize the representable energy efficiency techniques in Figure 2. Transistor sizing and logic optimization are common practice to minimize power consumption. Clock and input gating techniques shut down the unused hardware components to reduce dynamic power. Dynamic voltage and frequency scaling techniques adapt the supply voltage and frequency to achieve the desired performance without sacrificing energy efficiency. However, not all the low power techniques are suitable in the context of security. For example, near-threshold computing operates the system with a supply voltage near the threshold voltage to reduce the switching power. As demonstrated in [9], near threshold computing makes the system vulnerable to timing attacks.

Some low power techniques have been exploited to either implement security primitives. A large amount of effort has been done to assess the feasibility of applying emerging technologies, such as memristor, polarity-controllable silicon nanowire, and phase change memory in hardware security [10]. Approximate computing leverages fault tolerance in applications to adopt low-precision computation for the purpose of power saving. In Table 1, we highlight the representable techniques that have been employed in security primitives and mechanisms. In the next sections, we discuss those suitable methods in detail.

Table 1. Summary of typical energy efficiency techniques in security measures.

Technique Name	Authentication and IP Protection	Crypto Engine	Trojan Detection and Mitigation	Side-Channel Security	Reverse Engineering
Voltage/Frequency Scaling	[11]	[12–14]		[12–14]	
Emerging Technologies	[15–18]		[19]		
Approximate Computing	[20,21]	[21]			
Three-Dimensional Integration		[22]		[22–24]	[24]
Clock/Input Gating			[25]	[26]	

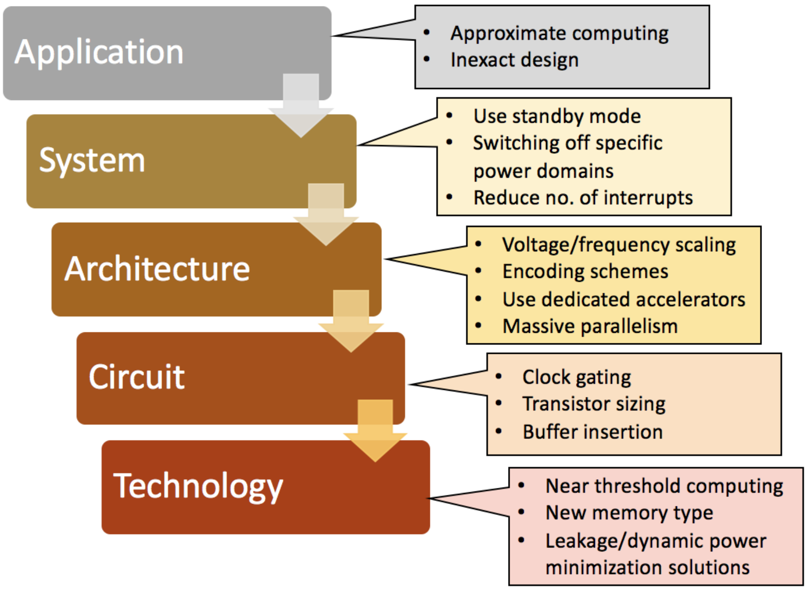


Figure 2. Energy efficiency techniques at different abstraction levels.

3. Low Power Techniques in Hardware Security

Not all low power techniques are suitable for hardware security. In this section, we review the representable low power techniques that have been applied to implement security primitives and reduce the power overhead induced by security mechanisms. The techniques such as voltage and frequency scaling, nanotechnology, approximate computing, and clock gating are useful to save power and strengthen system security at the same time. However, there are several techniques that are effective at power saving, but could make the system vulnerable to security threats. We use three-dimensional integration and near-threshold computing as two examples to draw prospective readers' attention.

3.1. Voltage/Frequency Scaling

Lowering supply voltage is an effective way to reduce the power consumption of a circuit. The study in [11] indicated that the uniqueness of the physical unclonable function (PUF) operating in the subthreshold region is drastically higher than that of the PUF with a normal supply voltage. Voltage/frequency scaling is a commonly-used technique for integrated circuits and systems to achieve energy efficiency. To reduce power consumption without sacrificing performance, the scaling of supply voltage and frequency can be managed in a dynamic fashion. Random dynamic voltage and frequency scaling (RDVFS) and aggressive voltage and frequency scaling (AVFS) have been employed in the on-chip switched-capacitor (SC) voltage converter to save power consumption and meanwhile reduce the correlation coefficient between the measured and hypothesized power, thus mitigating power analysis attacks [14]. A dynamic voltage and frequency switching (DVFS) scheme was proposed in [12]. In that work, a DVFS scheduler (DVFSS) facilitated providing a random voltage and frequency for the cryptoprocessor and generated a random output power signal, which increases the difficulty for attackers to perform side-channel analysis attacks. The scaling on voltage and frequency in [12] helped the system to obtain a 27% reduction on energy. The work [13] applied DVFS in reliable and aggressive designs to reduce the correlation between the correct key and practical power measurement. This mechanism breaks the one-to-one mapping between voltage and frequency to eliminate the security vulnerability existing in conventional DVFS. From the literature above, we can see that dynamic power and frequency scaling techniques are effective for power reduction and can strengthen the computing systems against power-based side-channel attacks.

3.2. Emerging Technologies

The memristor has been widely applied in designing PUFs, which generate unique digital signatures to protect hardware IP privacy. The memristor-based PUFs have a higher complexity and greater scalability than CMOS PUFs and thus offer better resistance against modeling attacks [17]. The highly nonlinear process variation effect of memristors inherently randomizes the PUF outputs and makes them less vulnerable to security attacks. Some nanotechnologies [15,16] consume less power than the conventional CMOS technology. Modern nanotechnology-based computing focuses on molecular device synthesizing, the randomness of which can generate unclonable components. Hence, it is suitable for PUF design [18]. Nanocell, the nanotechnology used in [18], is a good candidate for the design of public physical unclonable functions (PPUFs). This type of PUFs can be used in trusted remote sensing where the Nanocell plays an important role in improving power consumption. The nanotechnologies including memristors and nanowires can be leveraged to design a bidirectional polyomino partitioned PPUF to avoid high power usage [16].

3.3. Approximate Computing

Approximate arithmetic computing sacrifices computation accuracy to improve system energy efficiency. The work [21] proposed to hide confidential information in the least significant bits (LSBs) of operands. The LSBs are ignored in approximate computing, but can be used as IP watermarking, digital fingerprinting, or encryption keys. Approximate computing can also be leveraged to implement a lightweight authentication mechanism [20]. The voltage over-scaling (VOS) in [20] lowers the operating voltage to cause a timing error, which contains information of the process variation. This timing error can be profiled to generate a device signature for authentication.

3.4. Clock Gating

Clocking gating is another technique for power saving. The latch-based random clock gating (LRCG) [26] is used to mitigate side-channel attacks by obfuscating the power signature. The power consumption of LRCG is only 0.5% more over the baseline. The hardware Trojan is a predominant security threat on hardware. Typically, hardware Trojan detection yields significant power overhead. The P-VIRUSmonitor proposed in [25] captures the improper behavior caused by the Trojan inside power management units (PMU) by comparing the abnormal voltage with the standard supply voltage. Clock gating can be applied to the Trojan detection unit to reduce the power overhead associated with frequent abnormal behavior checking.

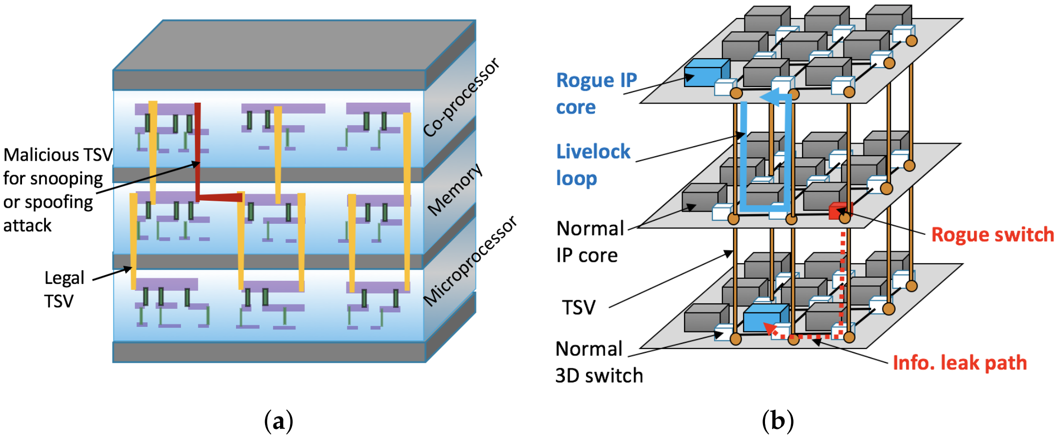
3.5. Near-Threshold Computing

Not every low power technique is suitable for the implementation of secure computing systems. Recently, there has emerged research efforts to investigate the increasing security vulnerability in low power systems. Near-threshold computing (NTC) reduces the power consumption of a circuit via the utilization of a supply voltage approximately equal to the threshold voltage of transistors. In spite of being a promising technique for low power systems, NTC is vulnerable to timing fault attacks, which exploit timing violation faults to activate the behaviors defined in the malicious malware [9]. Compared to super-threshold computing (STC), the same percentage of the variation on supply voltage in NTC could lead to 6× more delay variation. Chips operating at NTC suffer from 20× more process variation-induced delay variation, with respect to those at STC [27,28]. The increased sensitivity on delay variation in NTC makes the low power system vulnerable to timing fault attacks.

3.6. Three-Dimensional Integration

3D integration performed at different granularities enables different design trade-offs [29]. 3D integration at the logic gate level (block folding) can be utilized to save power and delay [24]. In block folding, a single functional block is spread out in multiple layers that minimize the intra-block delays and power. In a case study of a large-scale commercial-grade microprocessor (OpenSPARC T2) using the block folding and face-to-face bonding style, the power consumption was reduced by 20% compared to the 2D design [30]. Stack-based 3D integration is envisioned to resist side-channel analysis and reverse engineering attacks [24]. A low power adjustment technique named the power profile equalizer is presented to leverage power grid-induced noise to counteract power analysis attacks [23]. The work [22] combined two methods to thwart the power analysis attacks in an AES crypto engine. The first method is noise addition. The second one is isolating the power supply from the encryption system to minimize the leak of side-channel signal (leakage power). Because of the lesser amount of current required to resist power analysis attacks, the work [22] reduced the power overhead by 10× compared to the noise injection method alone [31].

3D integration is a double-edge sword. The 3D noise could make the side-channel analysis easier or more difficult, depending on the power distribution network topology [32]. The survey [33] further highlighted that the 3D integration leaves more room to attackers to hide hardware Trojans, which are more stealthy and difficult to detect due to limited and immature 3D testing approaches. The undetected Trojans, especially the Trojan remaining in the through-silicon-vias (TSVs), could perform snooping or spoofing attacks on multiple 3D tiers, as shown in Figure 3. A rouge switch in a 3D network-on-chip (NoC) could facilitate the cross-tier information leakage. The recent literature [34] presented the security threats in dense-wavelength-division-multiplexing-based photonic NoCs, in which a hardware Trojan manipulates the electrical driving circuit of its microring resonators to snoop data from the neighboring wavelength channels in a shared photonic waveguide.



(**a**) (**b**)

Figure 3. Security threats in 3D integrated circuits: (**a**) spoofing attack and (**b**) rogue 3D switch. TSV, through-silicon-via.

4. Power/Energy-Aware Security Mechanisms

There are many research efforts that have been made to counteract the arising security threats, including side-channel attacks, reverse engineering, IP piracy, and hardware Trojan insertion. Besides the low power techniques discussed in Section 3, other methods have been investigated in practical applications. In this section, we use several examples to showcase how the power and energy are reduced in the implementation of security mechanisms applied in computing systems.

4.1. Adjusting Redundancy in Security Primitives

4.1.1. Information Redundancy

In our work [35], we examined how hardware redundancy affects the crypto module's capability of resisting the power-based side-channel analysis attack. We used the accumulated partial guessing entropy (APGE)—sum of all PGE—as a metric to assess the speed of key retrieval. Three cyclic redundancy check (CRC) codes, CRC2, CRC4, and CRC8, were applied in the Sboxof an AES module to resist fault attacks. The digits after CRC stand for the number of check bits. As the crypto unit could suffer from correlation power analysis (CPA) attack, we evaluated the key retrieval speed of three AES protected with three CRC codes. The experimental results shown in Figure 4 indicate that more redundancy applied to the AES will require less power traces to retrieve the crypto key. When we compared the hardware cost of each method, we observed that the hardware cost was the primary factor responsible for the different speed of key recovery. For the same type of linear error detection code, more check bits mean that more exclusive-OR gates (i.e., hardware cost) are typically needed for check bit calculation. Hence, given the same linear code, a higher degree of information redundancy speeds up the CPA key retrieval. As shown in Table 2, CRC8 consumes the most area in FPGA, with respect to CRC2 and CRC4. From this example, we conclude that more information redundancy will lead to more hardware cost and also worse system resistance against CPA attack.

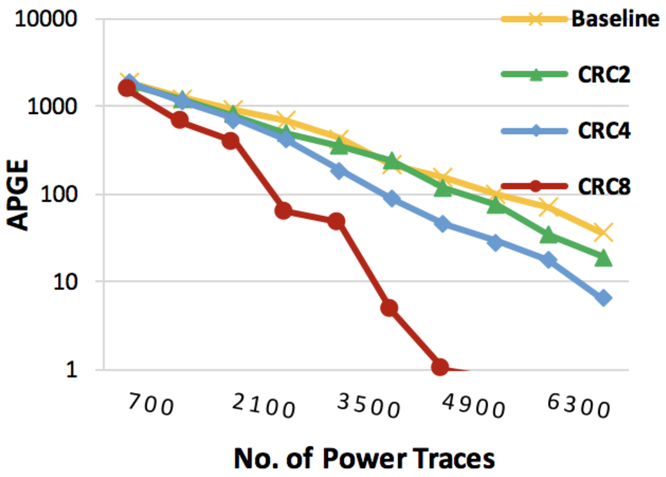


Figure 4. Average accumulated partial guessing entropy (APGE) for cyclic redundancy check (CRC) code applied to SBoxin AES.

Table 2. Hardware cost of CRC-based fault detection methods applied in AES S-box.

Design	Number of LUTs in FPGA
Baseline AES	2663
AES with CRC2 protected Sbox	3012
AES with CRC4 protected Sbox	3144
AES with CRC8 protected Sbox	3390

4.1.2. Hardware Redundancy

To thwart hardware tampering, modular redundancy is utilized in the implementation of computing systems. If the tampering does not happen to each replica, the difference between the outputs of replicas will indicate the occurrence of malicious modification on hardware. The limitation of hardware redundancy-based tampering resistance methods is significantly increased area cost and power overhead. To tackle the overhead issue, the adaptive triple modular redundancy (ATMR) proposed in [36] makes minor changes on the classic TMR and effectively reduces the power consumption. Instead of applying comparison among all three replicas, ATMR only compares two of the replicas at a time. The third one is only activated when mismatch is detected.

The work [37] exploits the principle of moving target defense (MTD) to address simultaneously the power overhead and the security threats originating from the malicious FPGA computer-aided design (CAD) tools. The proposed FPGA-oriented MTD (FOMTD) method in [37] makes the output of FPGA placement and routing unpredictable, such that attackers who mount a malicious program on the original FPGA design suite cannot easily and successfully alter the original implementation on an FPGA. One defense line in FOMTD is the hot-swappable submodule-assembling technique, as shown in Figure 5. The original design is partitioned into multiple submodules, and each submodule is duplicated by several times. Only one replica of each submodule will be assembled into a complete design. The pseudo-random selector is utilized to determine which replica to choose at runtime. After a period of time, the selection of submodule replicas will be changed without stopping the normal operation (i.e., hot-swappable assembling). Input gating is used to mute the unused replica. In addition, the comparison between replica outputs is removed to save power consumption without noticeably scarifying the Trojan detection rate.

The FOMTD method is compared with a static Trojan detection method, double or triple modular redundancy (DMR). Assume attackers could search which FPGA lookup tables (LUTs) will be configured for a given design and then place the hardware Trojan on those occupied slices. As the DMR method duplicates the design, it is very likely that attackers could search for the same FPGA configuration and thus duplicate the hardware Trojans on the identical LUT configurations. In contrast, the FOMTD method configures the FPGA slices in a dynamic fashion, which increases the unpredictability from the attackers' point of view. As a result, the FOMTD obtains a much lower Trojan hit rate than DMR, as shown in Figure 6. Due to the input gating and hot-swappable assembling techniques, the FOMTD only activates one design copy (DMR runs two active design copies). Consequently, the FOMTD significantly reduces the power consumption compared to DMR, as shown in Figure 7.

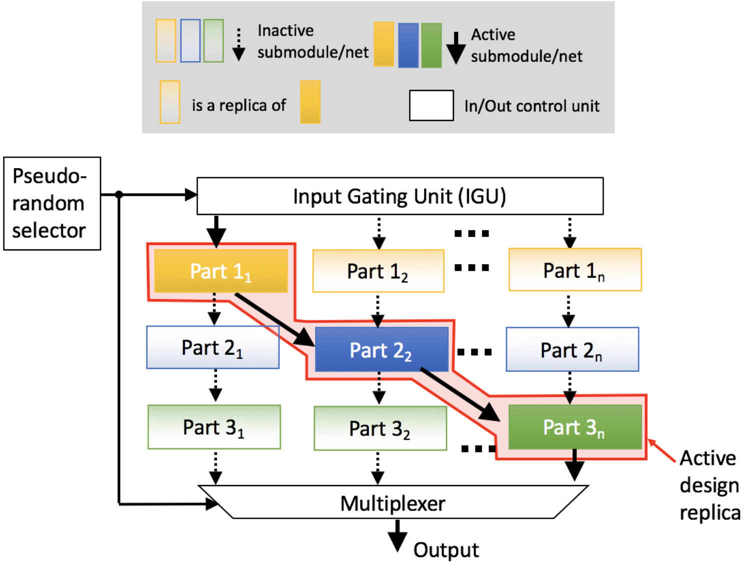


Figure 5. Schematic diagram of the hot-swappable submodule-assembling technique in FPGA-oriented moving target defense (FOMTD) [37]. HT, hardware Trojan.

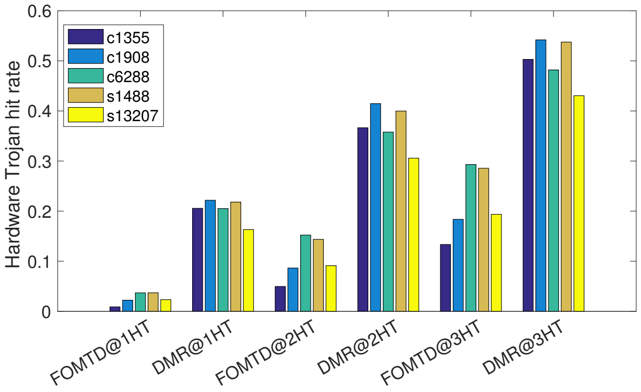


Figure 6. Comparison of hardware Trojan hit rate for FOMTD and double modular redundancy (DMR) affected by different numbers of Trojans.

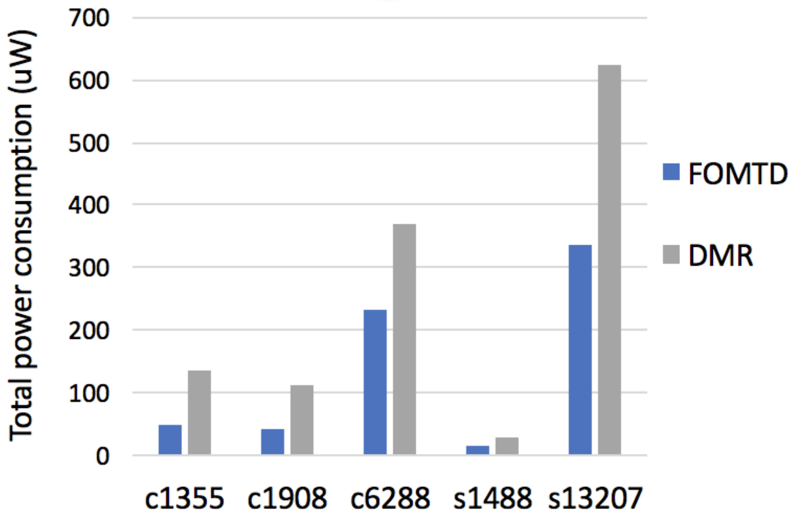


Figure 7. Comparison of power consumption between FOMTD and DMR.

4.2. Selecting Proper Key Size and Locking Locations in Design Obfuscation

State obfuscation has been demonstrated as a promising countermeasure to thwart reverse engineering and hardware intellectual property piracy. The general principle for state obfuscation is depicted in Figure 8. A correct key sequence K_1 ... K_7 will guide the finite state machine (FSM) to enter the normal operation mode; any wrong key sequence will lead the system to be stuck in a state dead loop.

Different design methodologies for obfuscated state transition provide various system resistance against reverse engineering attack and incur different hardware overhead. The work [37] recommended to obfuscate the circuit of interest with a target area overhead of 5%, in order to minimize the power and delay overhead. In this subsection, we use the dynamic state-deflection (DSD) method [2] as an example to discuss how power consumption is managed in state obfuscation methods. The concept of DSD is shown in Figure 9. Each normal state ST_x transition in the normal mode needs to pass the key authentication. If a wrong key (i.e., $! = KS$) is applied to the FSM, a normal state will be deflected to its black hole cluster B_x. The FSM never returns to the normal state once it enters the black hole cluster, each of which is composed of multiple states. Each black hole state corresponds to a unique wrong key. The state within the black hole cluster does not remain stable; instead, it constantly switches to other black hole states. More details of the DSD method are discussed in [2].

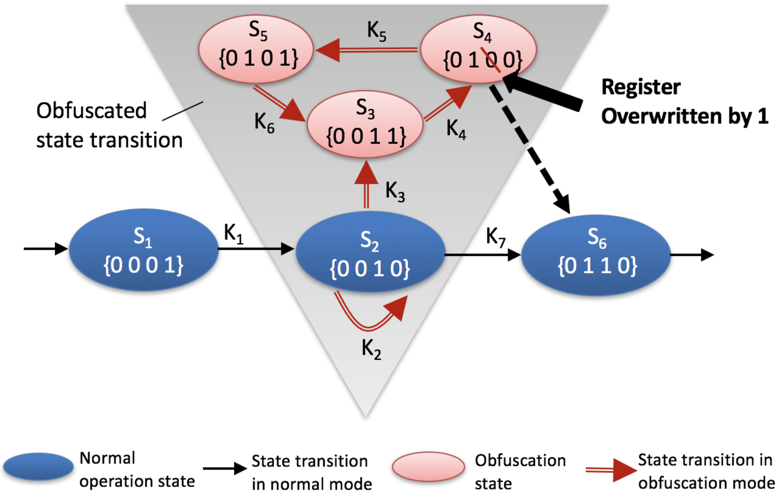


Figure 8. A simplified example of state obfuscation [2].

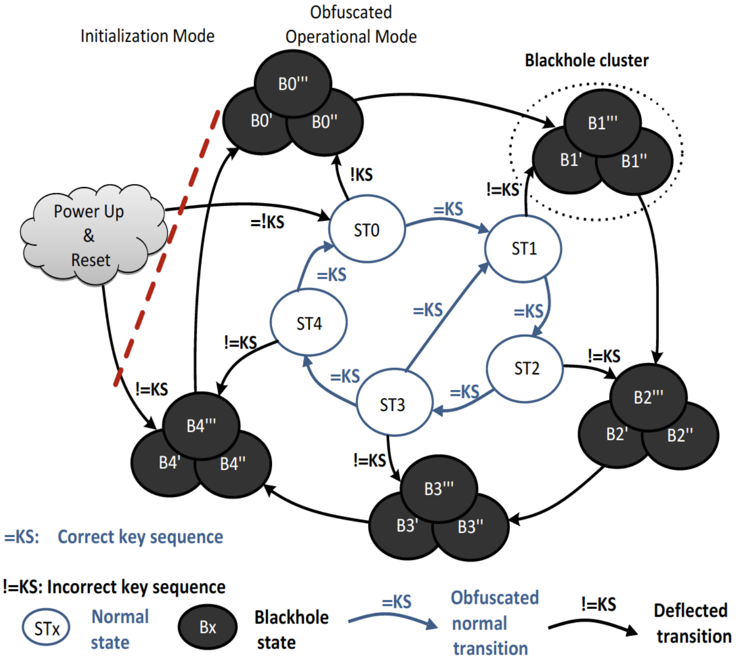


Figure 9. State transition graph for the dynamic state-deflection method for netlist obfuscation [2].

The size of key sequence and the location of key-based obfuscation have a direct impact on the area and power overhead associated with the obfuscation method. We normalized the area and power consumption for the obfuscated ISCASbenchmark circuits to that for the original ones. As shown in Figure 10a, the area consistently increases with the size of the key vector; the amount of increased area varies with the circuit under protection. Interestingly, the power overhead does not always steadily increase with the key size, as illustrated in Figure 10b. This result indicates that there is a large optimization space available in the design for us to tackle the power overhead. The number of obfuscated states and the location of locked states are the dependent factors as far as energy efficiency is concerned.

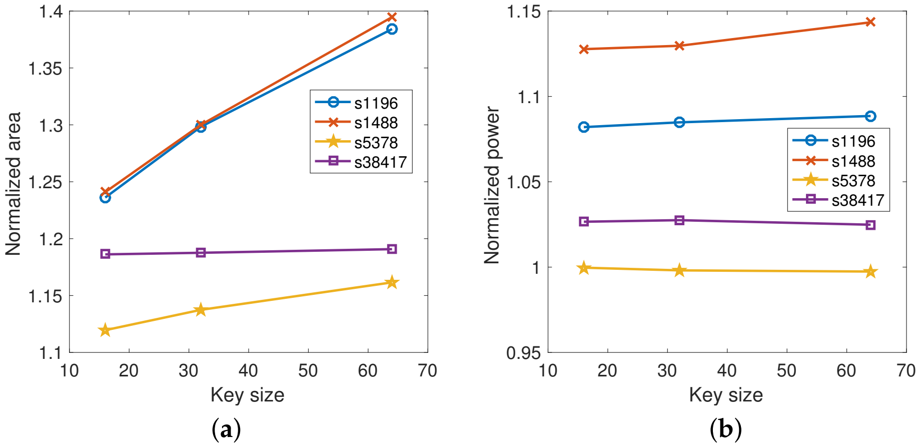


Figure 10. Impact of key size on (**a**) area overhead and (**b**) power increase.

4.3. Localizing Hardware Trojan Detection and Mitigation

Network-on-chip (NoC) is an energy-efficient communication infrastructure for on-chip communication. A compromised NoC, for example hardware Trojan (HT) injected NoC, will lead to information leakage, unauthorized memory access, and denial-of-service attacks, such as incorrect path routing, deadlock, and livelock. HT detection methods for NoCs aim to detect and mitigate the HTs that modify the flit type, change the legal packet destination address to a unauthorized one, or sabotage the integrity of a packet. Some HT detection mechanisms detect the Trojans at IP cores and the packet's final destination; some HT mitigation methods detect HTs and terminate the contaminated packets in the middle of the routing path. In the following subsection, we discuss how different Trojan detection/mitigation methods will affect the overall NoC power consumption.

Figure 11 illustrates a generic mesh-NoC, which is composed of links, a network interface (NI), and routers with five ports (north, east, south, west, and local input/output ports). The effect of injected HTs in NoCs will lead to NoC bandwidth depletion. As a result, packets injected from IP cores cannot reach their destinations in a timely manner. We use power consumption per effective packet delivery as a metric to compare different Trojan detection and mitigation methods for NoCs. In [38], a counter-based traffic adjustment NI is presented to address the packet loss and thus bandwidth depletion due to hardware Trojans. Hereafter, we name this method as termination at NI. The work [39] examined the feasibility of applying address filters to NoC routers; if the address pair is not legal, that packet is filtered out by the router. Hereafter, we refer to this method as address filtering at the router. The work [40] proposed a novel router design to detect and mitigate HTs at the NoC router level (detection and mitigation at the router). The zoom-in view of Figure 11 depicts the high-level view of the HT countermeasures (HTC) applied in the south input port and the east output port. In the input port, the HTC1 module dynamically permutes the incoming NoC flit bits before reaching the FIFO. The NoC flit saved in the input FIFO is scrambled. The goal of permutation is to reduce the probability of an HT inserted in the FIFO successfully modifying the critical bits in the NoC packet. Consequently, it is difficult for an attacker to change the flit content into something meaningful through HTs. The HTC2 module is used to examine the integrity of flits, which could be sabotaged by the HTs placed in the input FIFO. The application of HTC2 will prevent the malicious flits from entering the rest of the network by dropping the flit and flushing the input FIFO if necessary. As HTs could also be located in the flit processing unit and the FIFO in the east output port, an HTC3 module placed after the FIFO will stop the malicious flit from leaving the current router. The three HTC modules in Figure 11 cooperatively thwart the potential HTs that harm the NoC's flit integrity.

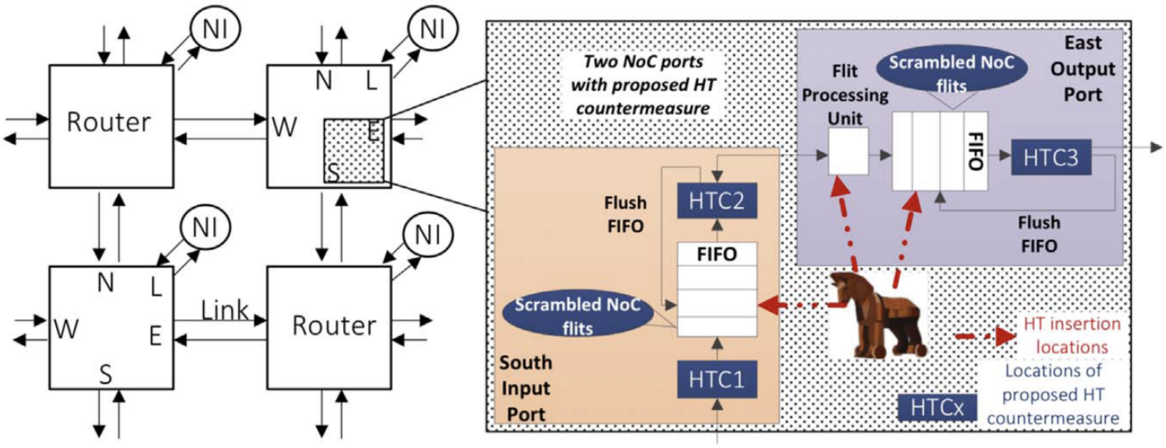


Figure 11. High-level view of the runtime router hardening method against Trojan injection in NoCs [40]. HTC, HT countermeasure.

The baseline network is a 4 × 4 mesh NoC that used an XY routing algorithm. The link width is 32 bits, and the input and output FIFOs have a buffer depth of eight. The crossbar uses a round robin arbiter for directional priority. A Hamming (15, 11) code is used as the error control code. Twenty thousand clock cycles are used in the simulation to achieve a steady state in all methods. The 4 × 4 NoCs with the compared HT detection/mitigation methods were implemented with Verilog HDL. The source codes were synthesized in Synopsys Design Vision with a 65-nm TSMCtechnology. Trojan detection and mitigation mechanisms result in power overhead. The power consumption per packet for different NoC scenarios is shown in Figure 12. If the method terminates the packet transmission until the acknowledgment signal returns or the time-out moment arrives (e.g., the method termination at NI), the entire NoC may switch logic, but not performing effective packet transfer, thus consuming the highest power. We recommend detecting and terminating HTs as soon as they are found. As shown in Figure 12, the methods address filtering at the router and detection and mitigation at the router significantly reduce the required power for each packet transfer. Hence, we conclude that distributed hardware Trojan detection is preferred to obtain energy efficiency. Combining detection and mitigation into a single unit is another good choice for the purpose of power saving.

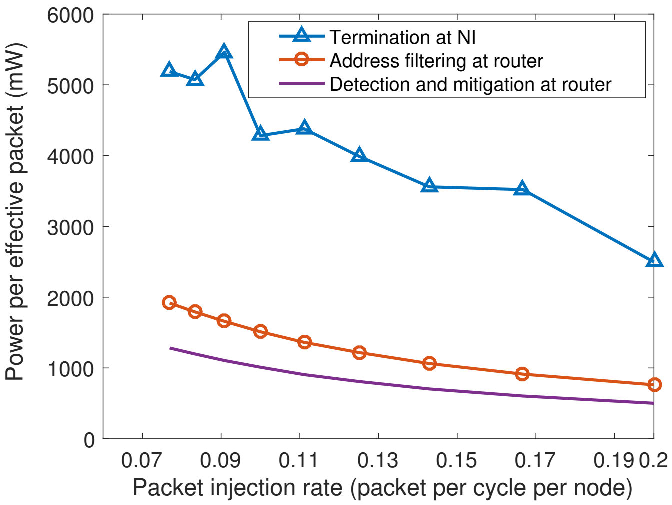


Figure 12. Comparison of power consumption per received packet in NoCs with different hardware Trojan detection and mitigation methods. NI, network interface.

4.4. Cross-Plane Obfuscation for Three-Dimensional Integrated Circuits

As highlighted in Section 3.6, 3D integration brings in potential security threats on the inter-plane communication. Due to the frequent communication between planes, the application of encryption and decryption is not an energy-efficient approach. The work [41] proposes a network-on-chip-based shielding plane (NOC-SIP) between two planes to thwart reverse engineering attacks in 3D ICs. The countermeasure NOC-SIP shown in Figure 13a obfuscates the communication among the adjacent planes so that attackers have a low success rate to perform sniffing attacks through vertical connection in the 3D chip. Figure 13b provides an example of how the vertical communication signal is detoured in the NOC-SIP. Hence, the direct communication between planes is obfuscated. The communication between Nodes A and B will be modified based on the routing algorithms defined in the NOC-SIP. As reported in [41], the source routing-based NOC-SIP only consumes 21% of the power used in a 2D NoC with the same size. This is because the NOC-SIP does not require implementing the complete NOC router. Compared to a conventional 2D NOC, there is significant hardware reduction on the inputoutput FIFO and network interface in NOC-SIP. The main hardware cost of NOC-SIP lies in packetization for cross-plane communication.

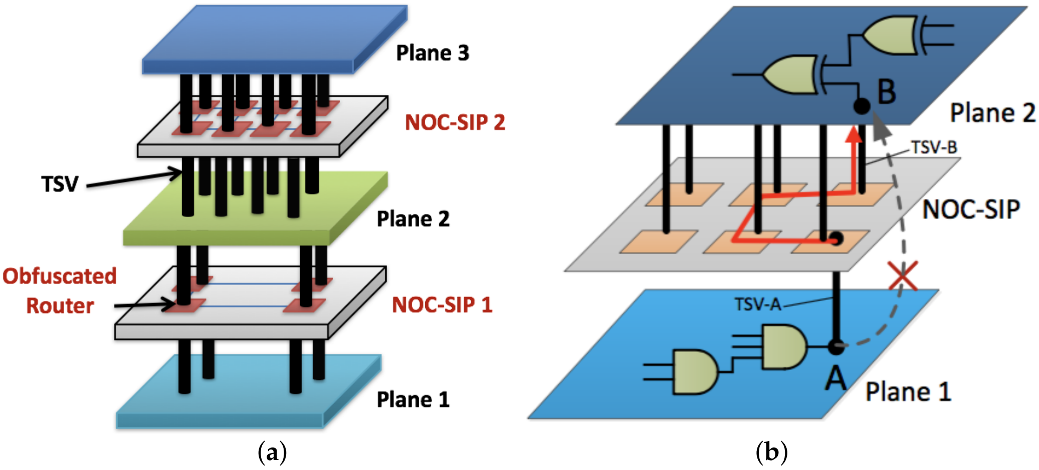


Figure 13. Network obfuscation to thwart hardware Trojans in 3D ICs. (**a**) Obfuscated cross-plane communication and (**b**) detoured data transmission on the obfuscated layer [41]. NOC-SIP, network-on-chip-based shielding plane.

5. Summary

Voltage and frequency scaling is effective at power reduction and maintaining resilience against side-channel attacks. As machine learning algorithms are being applied in hardware attacks, existing voltage and frequency scaling techniques may not lose their strengths in thwarting power analysis attacks. Machine learning techniques also challenge the PUF implementation. The security threat on approximate computing is not an open area. We envision that the reduced precision due to approximate computing will leave an exploration space for attackers to hide the effect of malicious design modification, which only slightly alters the original design to develop covert channels. It is worth further investigation. 3D integration is a double-edge sword, which could provide a natural defense against security threats on 2D ICs, but also bring in increased noise on process, voltage, and thermal variation. Potential solutions for 3D security include the vertical-dimensional obfuscation method, precise 3D modeling, and 3D masking techniques.

6. Conclusions

The emerging security threats raise serious concerns for computing systems. For some computing systems (e.g., IoTs and embedded systems), energy efficiency is typically demanded due to limited battery lifetime and power budget. In this work, we survey the existing low power techniques, which are feasible for the security primitives or security mechanisms, and also highlight the techniques that could worsen the system's resilience against certain types of security attacks. In real applications, the degree of different redundancies applied in the security mechanism should be investigated to obtain high security and energy efficiency.

Author Contributions: Conceptualization, Z.Z. and Q.Y.; Investigation, Z.Z.; Validation, Z.Z.; Writing, Z.Z. and Q.Y.; Supervision, Q.Y.; Project Administration, Q.Y.

Funding: This research was funded by National Science Foundation CAREER Award No. 1652474.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Guin, U.; Huang, K.; DiMase, D.; Carulli, J.; Tehranipoor, M.; Makris, Y. Counterfeit integrated circuits: A rising threat in the global semiconductor supply chain. *Proc. IEEE* **2014**, *102*, 1207–1228. [CrossRef]
2. Dofe, J.; Yu, Q. Novel dynamic state-deflection method for gate-level design obfuscation. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2018**, *37*, 273–285. [CrossRef]
3. Serpanos, D.; Voyiatzis, A. Security challenges in embedded Systems. *ACM Trans. Embed. Comput. Syst.* **2013**, *66*, 1–66. [CrossRef]
4. Alur, R.; Berger, E.; Drobnis, A.W.; Fix, L.; Fu, K.; Hager, G.D.; Lopresti, D.; Nahrstedt, K.; Mynatt, E.; Patel, S.; et al. Systems computing challenges in the Internet of things. *arXiv* **2016**, arXiv:1604.02980.
5. Srivastava, S.; Sudhish, P. Security in cloud computing systems: A review of challenges and solutions for security in distributed computing environments. In Proceedings of the 39th National Systems Conference (NSC), Noida, India, 14–16 December 2015; pp. 1–5.
6. Cowan, C.; Wagle, F.; Pu, C.; Beattie, S.; Walpole, J. Buffer overflows: Attacks and defenses for the vulnerability of the decade. In Proceedings of the Foundations of Intrusion Tolerant Systems, Hilton Head, SC, USA, 25–27 January 2003; pp. 227–237.
7. Davi, L.; Koeberl, P.; Sadeghi, A. Hardware-assisted fine-grained control-flow integrity: Towards efficient protection of embedded systems against software exploitation. In Proceedings of the 51st ACM/EDAC/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 1–5 June 2014; pp. 1–6.
8. Energy-Efficient Computing Systems, Dynamic Adaptation of Quality of Service and Approximate Computing. Available online: http://goo.gl/eJvfYs (accessed on 25 November 2018).
9. Basu, P.; Pandey, P.; Bal, A.; Rajamanikkam, C.; Chakraborty, K.; Roy, S. Titan: Uncovering the paradigm shift in security vulnerability at near-threshold computing. *IEEE Trans. Emerg. Top. Comput.* **2018**. [CrossRef]
10. Tehranipoor, M.; Forte, D.; Rose, G.; Bhunia, S. *Security Opportunities in Nano Devices and Emerging Technologies*; CRC Press: Boca Raton, FL, USA, 2018.
11. Vivekraja, V.; Nazhandali, L. Circuit-level techniques for reliable physically uncloneable functions. In Proceedings of the IEEE International Workshop on Hardware-Oriented Security and Trust, Francisco, CA, USA, 27 July 2009; pp. 30–35.
12. Yang, S.; Wolf, W.; Vijaykrishnan, N.; Serpanos, D.; Xie, Y. Power attack resistant cryptosystem design: A dynamic voltage and frequency switching approach. In Proceedings of the Design, Automation and Test in Europe, Munich, Germany, 7–11 March 2005; Volume 3, pp. 64–69.
13. Avirneni, N.; Somani, A. Countering power analysis attacks usingreliable and aggressive designs. *IEEE Trans. Comput.* **2014**, *63*, 1408–1420. [CrossRef]
14. Yu, W.; Kose, S. Exploiting voltage regulators to enhance various power attack countermeasures. *IEEE Trans. Emerg. Top. Comput.* **2018**, *6*, 244–257. [CrossRef]
15. Potkonjak, M.; Goudar, V. Public physical unclonable functions. *Proc. IEEE* **2014**, *102*, 1142–1156. [CrossRef]
16. Wendt, J.; Potkonjak, M. The bidirectional polyomino partitioned ppuf as a hardware security primitive. In Proceedings of the IEEE Global Conference on Signal and Information Processing, Austin, TX, USA, 3–5 December 2013; pp. 257–260.
17. Mazady, A.; Rahman, M.; Forte, D.; Anwar, M. Memristor puf—A security primitive: Theory and experiment. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2015**, *5*, 222–229. [CrossRef]
18. Wendt, J.; Potkonjak, M. Nanotechnology-based trusted remote sensing. In Proceedings of the IEEE SENSORS, Limerick, Ireland, 28–31 October 2011; pp. 1213–1216.

19. Rathor, V.; Garg, B.; Sharma, G. An energy-efficient trusted FSM design technique to thwart fault injection and Trojan attacks. In Proceedings of the 31st International Conference on VLSI Design and 2018 17th International Conference on Embedded Systems (VLSID), Pune, India, 6–10 January 2018; pp. 73–78.
20. Arafin, M.; Gao, M.; Qu, G. Volta: Voltage over-scaling based lightweight authentication for IoT applications. In Proceedings of the 22nd Asia and South Pacific Design Automation Conference (ASP-DAC), Chiba, Japan, 16–19 January 2017; pp. 336–341.
21. Gao, M.; Wang, Q.; Arafin, M.; Lyu, Y.; Qu, G. Approximate computing for low power and security in the internet of things. *Computer* **2017**, *50*, 27–34. [CrossRef]
22. Das, D.; Maity, S.; Nasir, S.; Ghosh, S.; Raychowdhury, A.; Sen, S. High efficiency power side-channel attack immunity using noise injection in attenuated signature domain. In Proceedings of the IEEE International Symposium on Hardware Oriented Security and Trust (HOST), McLean, VA, USA, 1–5 May 2017; pp. 62–67.
23. Wang, C.; Yan, M.; Cai, Y.; Zhou, Q.; Yang, J. Power profile equalizer: A lightweight countermeasure against side-channel attack. In Proceedings of the IEEE International Conference on Computer Design (ICCD), Boston, MA, USA, 5–8 November 2017; pp. 305–312.
24. Xie, Y.; Bao, C.; Serafy, C.; Lu, T.; Srivastava, A.; Tehranipoor, M. Security and vulnerability implications of 3d ics. *IEEE Trans. Multi-Scale Comput. Syst.* **2016**, *2*, 108–122. [CrossRef]
25. JayashankaraShridevi, R.; Rajamanikkam, C.; Chakraborty, K.; Roy, S. Catching the flu: Emerging threats from a third party power management Unit. In Proceedings of the 53rd Annual Design Automation Conference, Austin, TX, USA, 5–9 June 2016; pp. 1–6.
26. Tanimura, K.; Dutt, N. Lrcg: Latch-based random clock-gating for preventing power analysis side-channel attacks. In Proceedings of the Eighth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis, Tampere, Finland, 7–12 October 2012; pp. 453–462.
27. Karpuzcu, U.; Kolluru, K.; Kim, N.; Torrellas, J. Varius-ntv: A microarchitectural model to capture the increased sensitivity of manycores to process variations at near-threshold voltages. In Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012), Boston, MA, USA, 25–28 June 2012; pp. 1–11.
28. Seo, S.; Dreslinski, R.; Woh, M.; Park, Y.; Charkrabari, C.; Mahlke, S.; Blaauw, D.; Mudge, T. Process variation in near-threshold wide simd Architectures. In Proceedings of the DAC Design Automation Conference, San Francisco, CA, USA, 3–7 June 2012; pp. 980–987.
29. Loh, G.; Xie, Y.; Black, B. Processor design in 3d die-stacking Technologies. *IEEE Micro* **2007**, *27*, 31–48. [CrossRef]
30. Jung, M.; Song, T.; Wan, Y.; Peng, Y.; Lim, S. On enhancing power benefits in 3d ics: Block folding and bonding styles perspective. In Proceedings of the 51st Annual Design Automation Conference, San Francisco, CA, USA, 1–5 June 2014; pp. 1–6.
31. Güneysu, T.; Moradi, A. Generic side-channel countermeasures for reconfigurable devices. In Proceedings of the 13th International Workshop, Nara, Japan, 28 September–1 October 2011; pp. 33–48.
32. Dofe, J.; Zhang, Z.; Yu, Q.; Yan, C.; Salman, E. Impact of Power Distribution Network on Power Analysis Attacks in Three-Dimensional Integrated Circuits. In Proceedings of the GLSVLSI'17, Banff, AB, Canada, 10–12 May 2017; pp. 327–332.
33. Dofe, J.; Gu, P.; Stow, D.; Yu, Q.; Kursun, E.; Xie, Y. Security Threats and Countermeasures in Three-Dimensional Integrated Circuits. In Proceedings of the on Great Lakes Symposium on VLSI 2017, Banff, AB, Canada, 10–12 May 2017; pp. 321–326.
34. Chittamuru, S.V.R.; Thakkar, I.G.; Bhat, V.; Pasricha, S. SOTERIA: Exploiting process variations to enhance hardware security with photonic NoC, architectures. In Proceedings of the 55th ACM/ESDA/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 24–28 June 2018.
35. Dofe, J.; Pahlevanzadeh, H.; Yu, Q. A comprehensive fpga-based assessment on fault-resistant aes against correlation power analysis attack. *J. Electron. Test.* **2016**, *32*, 611–624. [CrossRef]
36. Mal-sarkar, S.; Krishna, A.; Ghosh, A.; Bhunia, S. Hardware Trojan Attacks in FPGA Devices: Threat Analysis and Effective Countermeasures. In Proceedings of the GLSVLSI '14 Proceedings of the, Houston, TX, USA, 21–23 May 2014; pp. 287–292.
37. Zhang, Z.; Yu, Q.; Njilla, L.; Kamhoua, C. FPGA-oriented moving target defense against security threats from malicious FPGA tools. In Proceedings of the IEEE International Symposium on Hardware Oriented Security and Trust (HOST), Washington, DC, USA, 30 April–4 May 2018; pp. 163–166.

38. Baron, S.; Wangham, M.; Zeferino, C. Security mechanisms to improve the availability of a network-on-chip. In Proceedings of the 20th International Conference on Electronics, Circuits, and Systems (ICECS), Abu Dhabi, UAE, 8–11 December 2013; pp. 609–612.
39. Evain, S.; Diguet, J. From NoC security analysis to design solutions. In Proceedings of the IEEE Workshop on Signal Processing Syst. Design and Implementation, Athens, Greece, 2–4 November 2005; pp. 166–171.
40. Frey, J.; Yu, Q. A hardened network-on-chip design using runtime hardware Trojan mitigation methods. *Integr. VLSI J.* **2017**, *56*, 15–31. [CrossRef]
41. Dofe, J.; Yu, Q.; Wang, H.; Salman, E. Hardware Security Threats and Potential Countermeasures in Emerging 3D ICs. In Proceedings of the 26th edition on Great Lakes Symposium on VLSI, Boston, MA, USA, 18–20 May 2016; pp. 69–74.