A high-efficient and low-cost secure AMBA framework utilizing configurable data encryption modeling against probe attacks

Kai Wang¹, Wei Li^{1, a)}, Yanjiang Liu¹, Tao Chen¹, Longmei Nan², and Xianzhao Xia³

Abstract In this paper, a high-efficient and low-cost secure AMBA framework utilizing the bus data encryption modeling is proposed to resist the probe attacks. By encrypting the confidential data flowing through the bus, the proposed configurable encryption model meets the security requirement of total SoC. Further, a data encryption pipeline with the third-level branch predictor is proposed to accelerate the encryption process. Finally, an SoC with the 32-bit proposed AMBA framework is established and validated with 55 nm technology. Experimental results show that the proposed framework achieves 6152 Mbps throughput, consumes 39547 μm^2 area, and provides a stronger resistance compared to the other countermeasures.

Keywords: probe attacks, SoC, branch predictor, data encryption

Classification: Integrated circuits (memory, logic, analog, RF, sensor)

1. Introduction

With an ever-growing of integrated circuit design technology, system on chip (SoC) can build a complete system by integrating the general-purpose processor, memory, and interface IPs into a single chip, which has become the main trend during the integrated circuit design stage [1]. Such a trend dramatically lowers the design cost and reduces the time to market, and makes SoC widely deployed into critical applications and sensitive fields.

As the fundamental part of SoC, several system buses, such as AMBA, CoreConnect, Wishbone, and Avalon, have been proposed to transfer the instruction sets, address, and data to the slave over the past decades [2]. Although the system bus establishes a channel of each module in the SoC, it also provides an opportunity for the attackers to leak confidential information. Of all the existing attack approaches, the probe attacks capture the values or states of the critical wires through the electronic probe, which is one of the most efficient ways to obtain the secret information of the circuit [3]. Fig. 1 shows four typical probe attacks in the SoC. The on-chip storage, system bus, the peripheral interfaces of the external device, and off-chip storage are vulnerable to probe attacks [4, 5].

Among all the system buses, the AMBA is widely applied into the SoC design. Therefore, it is crucial to design a secure AMBA framework against the probe attacks. Recently, vari-

Fig. 1 Four probe attacks in the SoC

ous countermeasures, including the active shields [6, 7, 8, 9], probe sensors [10, 11, 12, 13], and data encryption methods [14, 15, 16, 17], have been explored by researchers. For the active shields, numerous patterns, including the parallel equipotential, Picano curve, Hilbert curve, Moore curve, random Hamiltonian path, and so on, have been introduced to prevent the attackers from accessing the internal wires. But those approaches occupy one or more metal layers, which will degrade the performance during the placement and routing process [6, 7, 8]. Moreover, several sensors have been described to sense the variations of charge or capacitance value caused by the probe [10, 11]. However, the detection sensitivity is still the main problem of those approaches.

Of all the existing prevention approaches, data encryption methods are the most promising ones for the practical application by encrypting all the data [14, 15]. Such countermeasures can resist both intrusive probe attacks and non-intrusive probe attacks. The main challenges of such approaches are the trade-off between encryption efficacy, area overhead, and security capability.

To addressing this issue, a high-efficient and low-cost AMBA framework is proposed, which can achieve all the desired properties with improved metrics of area, throughput, and security. A configurable bus data encryption modeling is set up to encrypt the data flowing through the bus. For the modeling, a data encryption pipeline is proposed to balance the encryption cost and data transmission efficacy. Furthermore, a dynamic branch predictor is introduced into the data encryption pipeline to further improve the encryption efficacy. The main contributions are listed as follows.

- A configurable data encryption engine is proposed to encrypt the data flowing through the AMBA, and the encryption parameters can be configured by the users.
- A third-level dynamic branch predictor targeted for the

¹ Information Engineering University, China

² Fudan University, China

³ China Automotive Technology and Research Center, China

^{a)} try_1118@163.com

data transmission in the SoC is proposed to increase the hit rate of prediction, which improves the efficacy of the data encryption pipeline.

- A secure AMBA framework is proposed to balance the security and performance simultaneously. To the best of our knowledge, this is one of the first pioneer attempts to design a secure AMBA framework.

The following of this paper is organized as follows. Section 2 introduces the probe attacks prevention techniques. Section 3 gives the configurable bus data encryption modeling. Section 4 presents the proposed AMBA framework. Section 5 analyzes the experimental results. Section 6 concludes this paper.

2. Probe attacks prevention approaches

Considering the serious consequences caused by probe attacks, various prevention approaches are proposed over the past decades. It can be broadly classified into 1) active shield, 2) probe sensor and 3) data encryption. For the active shield, the complicated pattern is utilized to mask or obscure the critical wires, which makes the probe attacks more difficult to be implemented. Ngo et al. propose a grid line shield for IP protection in [6], and Wang et al. introduce an ILP-based tile routing algorithm to generate complex shield lines [7]. From [8], a vulnerability evaluation approach of active shield against probe attacks is proposed. Further, a FIB-aware anti-probing physical design flow is described in [9].

Moreover, various sensors are designed to sense the probe. A time-sensitive XOR gate for probe attempt detectors is presented in [10], and a low area probing detector is proposed in [11]. Weiner et al. design a calibratable lightweight invasive attacks detector to detect probe attacks [12]. Lee et al. propose an efficient detection architecture that can detect probe attacks and FIB attacks [13].

Of all the prevention approaches, data encryption is the most promising method towards the practical application. Fan et al. introduce the AES encryption/decryption hardware engine instead of LFSR into the solid-state disk controller [14], and Angizi et al. propose a low-energy reconfigurable data encryption module to encrypt the data in non-volatile memory [15]. In [16], a data transmission protection model is introduced to encrypt the data flowing through the scan chain. Further, a programmable data encryption accelerator is proposed in [17].

While these countermeasures are efficient, it occupies a large area and reduces the data throughput to prevent the sensitive information leakage from the probe attacks. It may not be acceptable in high-speed and area-constrained applications [18]. Accordingly, a high-efficient and low-cost secure AMBA framework based on the configurable bus data encryption modeling is proposed to resist the probe attacks.

3. Configurable bus data encryption modeling

For the traditional data encryption methods presented in [19, 20, 21], it requires additional time to encrypt the data using the block, stream, or public-key cryptography algorithms.

There is no doubt that the increased time cost will lower the data throughput of secure system bus. In this section, the effect of data throughput caused by data encryption is analyzed and the configurable bus data encryption model is proposed.

The data throughput v of bus is expressed in Eq. (1). Where the L_{bus} is the bus length, the f_{bus} is the clock frequency, and the β is the bus bandwidth utilization ratio.

$$v = \beta L_{\text{bus}} f_{\text{bus}} \quad (1)$$

The L_{bus} and f_{bus} are constant after the circuit is fabricated, thus the v is depended on the β [22]. The β is presented in Eq. (2). Where T_D is the time of data transmission and is a constant, and T_W is the waiting time of data transmission.

$$\beta = \frac{T_D}{T_D + T_W} \times 100\% \quad (2)$$

Combining the Eqs. (1) and (2), the v is negatively correlated with the T_W . The cost function f is used to evaluate the quality of the bus transmission, which is expressed as Eq. (3). Therefore, the minimum T_W is searched to reduce the loss of data throughput.

$$f = \text{argmin } T_W \quad (3)$$

In general, numerous data are sent from the master to the slave through the system bus. However, the majority of data are public, such as interface configuration information and part of external transmission data, which does not require enhanced security through data encryption. There exist a few sensitive data in the SoC, including the user identification, authentication password, key, and so on, which should be protected against the probe attacks [23, 24]. If all the data flowing through the system bus are encrypted, the data throughput will decrease dramatically, which may not satisfy the specific applications.

Therefore, a configurable bus data encryption method is proposed to encrypt the sensitive data only according to the configuration parameters. The configurable data encryption method is presented in Eq. (4). Where $\gamma = \{\gamma_{\text{WS}}, \gamma_{\text{WM}}, \gamma_{\text{RS}}, \gamma_{\text{RM}}\}$ is the configuration parameter vector, Ψ is the data sent from the master, Γ is the data to the slave, $\mathcal{K}$ is the encryption/decryption key, E is the encryption function, and D is the decryption function. The AES with the CTR mode [25, 26] is selected as a data encryption engine for the high speed, and the address of data instead of counter value during the data encryption process.

$$\Gamma = \begin{cases} \Psi, & (\gamma_{\text{WS}}, \gamma_{\text{WM}}) = 0 \\ E_{\mathcal{K}}(\Psi), & (\gamma_{\text{WS}}, \gamma_{\text{WM}}) = 1 \\ D_{\mathcal{K}}(\Psi), & (\gamma_{\text{WS}}, \gamma_{\text{WM}}) = 2 \\ D_{\mathcal{K}}(E_{\mathcal{K}}(\Psi)), & (\gamma_{\text{WS}}, \gamma_{\text{WM}}) = 3 \end{cases} \quad (4)$$

Both the γ_{WM} and γ_{WS} are equal to 0, the public data Ψ is transferred transparently. The Ψ is encrypted using the function E only when the γ_{WM} is equal to 1, while the Ψ is decrypted only if the γ_{WS} is equal to 1. If both the $\gamma_{\text{WM}} = 1$ and $\gamma_{\text{WS}} = 1$ are satisfied, the data transferring to the bus is encrypted and the data transmitting to the slave is decrypted. Similarly, the reading process from the slave to the master

Fig. 2 Data encryption pipeline

consistent with the writing process.

After the data encryption, the β is changed as Eq. (5). Where ΔT_W is the additional waiting time caused by data encryption. The ΔT_W is equal to 0 when the Ψ is the public data, while the $\Delta T_W > 0$ if the $\gamma_{WM} \neq 0$ or $\gamma_{WS} \neq 0$.

$$\beta' = \frac{T_D}{T_D + T_W + \Delta T_W} \times 100\% \quad (5)$$

Regarding the Eq. (5), ΔT_W should be further decreased to minimize the impact of throughput caused by data encryption. Therefore, a data encryption pipeline is proposed to reduce the ΔT_W by adding the pre-encryption process. The data encryption pipeline is shown in Fig. 2, which is consisted of the address fetching (AF), address pre-encryption (APE), predicted address arbitration (PAA), and data encryption (DE).

The AF fetches the predicted address $\mathcal{R}$ of the sensitive data, while the APE encrypts the $\mathcal{R}$ with the key $\mathcal{K}$. The ciphertext $\mathcal{P}$ of $\mathcal{R}$ is expressed as Eq. (6).

$$\mathcal{P} = E_{\mathcal{K}}(\mathcal{R}) \quad (6)$$

The PAA validates whether the predicted address $\mathcal{R}$ matches with actual address $\mathcal{A}$ or not, which is described as Eq. (7). If the $\mathcal{R}$ is equal to the $\mathcal{A}$, the DE implements the XOR operation between the ciphertext $\mathcal{P}$ and Ψ . Otherwise, the APE encrypts the actual address $\mathcal{A}$, and implements the XOR operation with the Ψ and the ciphertext of $\mathcal{A}$.

$$\Gamma = \begin{cases} \mathcal{P} \oplus \Psi & \mathcal{R} = \mathcal{A} \\ E_{\mathcal{K}}(\mathcal{A}) \oplus \Psi & \mathcal{R} \neq \mathcal{A} \end{cases} \quad (7)$$

The operation of AF and APE should be executed before the data transmission to improve the encryption efficacy. Branch predictor is an efficient way to increase the speed of instruction execution in the conventional processor architectures by completing the instruction fetching and decoding in advance [27, 28]. Inspired by this idea, a branch predictor is designed to predict the next address and an address arbiter is used to verify the address prediction results. If the prediction is correct, the ΔT_W is the sum of the time of PAA and DE executed, otherwise, the ΔT_W is the total time of the data pipeline consumed. In this paper, the hit rate η of prediction is used to express the prediction accuracy. The expected value of ΔT_W for a set of data transmission is expressed as Eq. (8). Where the t_{AF} , t_{APE} , t_{PAA} and t_{DE} are the time consumption of AF, APE, PAA and DE respectively.

$$E(\Delta T_W) = \eta(t_{PAA} + t_{DE}) + (1 - \eta)(t_{AF} + t_{APE} + t_{PAA} + t_{DE}) \quad (8)$$

Fig. 3 The state transition diagram of dynamic branch predictor

Regarding the Eq. (8), the t_{AF} , t_{APE} , t_{PAA} and t_{DE} are constant value, and the $t_{AF} + t_{APE} \gg t_{PAA} + t_{DE}$. Both the AF and APE are combinational operations, thus the $t_{PAA} + t_{DE} \approx 0$. The ΔT_W is approximated as 0 if the $\eta \approx 100\%$. In summary, the ΔT_W induced by data encryption can be decreased by increasing the η .

The traditional branch predictors have two branches, which is hard to cover all the cases of data transmission in the SoC. Actually, the address of data transmission in the SoC can be divided into current address, incremental address, and jumped address. Therefore, a third-level dynamic branch predictor is proposed to enhance the address prediction accuracy and increase the hit rate of prediction η .

The state transition diagram of the proposed branch predictor is shown in Fig. 3. Supposing that a set of data are written to the contiguous address space of the on-chip memory consecutively, the next address is predicted as the incremental address and the current state remains at the 100. The next address is changed as the jumped address when an external interrupt is generated, and the current state is switched to 101. Then, the branch predictor loads the current jumped address and continues to implement the incremental prediction. If the next prediction is correct, the state will go back to 100, otherwise, the branch predictor will jump to the state 001 or 011 according to the actual address. While the interrupt comes from the peripheral interfaces (e.g. USB, PCIe, and Ethernet), the actual address is a constant value [29]. In this case, the state switches to 001 and the branch predictor will predict the next address as the address of the interrupt sources. Otherwise, the next address is a jumped address, and the state will switch to the 011.

4. Secure AMBA framework utilizing configurable bus data encryption modeling

The proposed AMBA framework is shown in Fig. 4, which is consisted of the configurable encryption management module (denoted as CEMM), address encryption engine (denoted as AEE), branch predictor (denoted as BP), data encryption controller (denoted as DEC), address arbiter (denoted as AA), data encryption (denoted as DE) and data decryption (denoted as DD).

The CEMM configures the encryption/decryption parameters $\gamma = \{\gamma_{WS}, \gamma_{WM}, \gamma_{RS}, \gamma_{RM}\}$ and controls the data en-

Fig. 4 The secure AMBA framework

Algorithm 1 Writing procedure of the proposed framework**Input:** $R, A, \Psi, \mathcal{K}, \gamma_{wm}, \gamma_{ws}$;**Output:** F ;

```

1  For  $i < N$                                      %  $N$  is the number of  $\Psi$ ;
2      If ( $\gamma_{wm}=1$ )
3          If ( $R=A$ )
4               $D \leftarrow E_{\mathcal{K}}(R) \oplus \Psi$ ;
5          Else
6               $D \leftarrow E_{\mathcal{K}}(A) \oplus \Psi$ ;
7          Endif
8      Else
9           $D \leftarrow \Psi$ ;
10     Endif
11     If ( $\gamma_{ws}=1$ )
12         If ( $R=A$ )
13              $\Gamma \leftarrow E_{\mathcal{K}}(R) \oplus D$ ;
14         Else
15              $\Gamma \leftarrow E_{\mathcal{K}}(A) \oplus D$ ;
16         Endif
17     Else
18          $\Gamma \leftarrow D$ ;
19     Endif
20 EndFor

```

encryption/decryption flow. The BP predicts the next address of data transmission in advance and sends it to the AEE, while the AEE encrypts the address predicted by the BP. The DEC executes the encryption or decryption operation according to the configuration parameters given by the CEMM. The AA judges whether the predicted address R is consistent with the actual address A or not. If $R = A$, the encrypted data D is transferred to the bus after the implementation of the DE. Otherwise, the prediction is wrong and the actual address A is encrypted. The decryption process is consistent with the encryption process and the details are not presented in this section. The writing procedure of the proposed framework is shown in algorithm 1, and the reading procedure is similar to the writing procedure.

5. Result

5.1 Security analysis

A 32-bit AMBA with the proposed framework is realized, and an SoC, including the RISC-V processor [30], 32-bit AMBA, DMA, eFlash, SRAM, JTAG, UART, USB, Flash, and so on, is established as the experimental setup. Moreover, the SoC is tested on the Xilinx Virtex-7 FPGA platform, and the Chipscope IP is integrated into the project to capture the data of AMBA, SRAM, Flash, and USB. To prevent the key leaking from the data exchanging process, a physical

Table I The results of security analysis

Ref.	Bus data transmission	Storage on-chip	Storage off-chip	External communication
[8]	√	√	--	--
[9]	√	√	--	--
[12]	√	√	--	--
[13]	√	√	--	--
[14]	--	--	√	--
[15]	--	√	√	--
[16]	√	--	--	√
[17]	--	√	√	√
This work	√	√	√	√

Table II The performance evaluation results

Ref.	Technology (nm)	Area (μm^2)	Max throughput (Mbps)
[14]	180	3847744	2080
[15]	45	30943	2267
[16]	65	50315	--
[17]	16	26632	4186
This work	55	39547	6152

unclonable function based on the SRAM is used to generate the key for the data encryption engine [31, 32].

The confidential data flowing through the bus or storing to the on-chip storage medium (e.g. SRAM and eFlash) are encrypted using the encryption algorithm, and it is extremely difficult to reveal the confidential data captured by intrusive probe attacks according to the cryptography theory. Besides, the proposed approach also provides encrypted data to the peripheral interfaces or off-chip storage medium, which can resist the non-intrusive probe attacks. However, the approaches presented in [8, 9, 12] and [13] only resist the intrusive probe attacks, and the Ref. [14] and [15] enhance the security of storage medium. The Ref. [16] prevent the information leakage from the scan attacks, and a high-efficient encryption scheme is proposed in [17]. Such approaches are hard to cover all the confidential data in the SoC. The security results are shown in Table I. In summary, the proposed framework achieves a higher level of security compared to the other countermeasures.

5.2 Performance evaluation

The established SoC is synthesized with 55 nm CMOS technology, and the timing analysis is implemented using the Prime Time. The proposed framework can reach 769 MHz. In general, the maximum frequency of an AMBA 2.0 cannot exceed 300 MHz. Therefore, the frequency losses caused by the proposed framework can be ignored. Moreover, the area consumption and data throughput are calculated, and the results are shown in Table II. The third column lists the area consumption, while the fourth column shows the maximum data throughput compared to the existing approaches. The Ref. [14, 15, 16], and [17] are the data encryption methods, thus we only compare the performance of the proposed framework with those approaches in this section.

As shown in Table II, the area of the proposed approach is $39547 \mu\text{m}^2$, which is greater than the Ref. [15] and [17], and smaller than the Ref. [14] and [16]. In this paper, the AMBA runs 200 MHz to calculate the maximum data throughput. The data throughput of the proposed approach can reach 6152 Mbps, which is greater than the other countermeasures. To sum up, the proposed framework has great advantages

Fig. 5 The relationship between the $\overline{\Delta T_W}$ and η

in performance and area overhead compared to the other countermeasures.

For the bus, it is impossible to keep at the busy state all the time. Moreover, the hit rate η of branch predictor cannot reach 100% actually. Considering the above factors, the averaged waiting time is more suitable to quantify the data transmission efficacy of the proposed framework. Therefore, the relationship between the averaged clock cycles $\overline{\Delta T_W}$ and the hit rate η is calculated, and the result is shown in Fig. 5. The length of data transmission is denoted as LEN. As shown in Fig. 5, $\overline{\Delta T_W}$ is decreased with the increment of η . The $\overline{\Delta T_W}$ is approximated as 0 when all the prediction is correct ($\eta = 100\%$). In this case, the proposed framework achieves a higher level of security without any performance degradation. Moreover, the descending ratio of $\overline{\Delta T_W}$ decreases with the increasing of LEN, which shows the data transmission efficacy is improved with the length improvement of data.

6. Conclusion

In this paper, a high-efficient and low-cost secure AMBA framework utilizing the configurable bus data encryption modeling is proposed to resist the probe attacks. An SoC with the proposed framework is designed and the performance is analyzed with the 55 nm technology. The simulation results show that the proposed approach achieves high-level security for the confidential information against the probe attacks. Besides, the data encryption modeling is not only limited to the AMBA. The proposed framework is also applied to the other system buses. The performance of the other buses with the proposed framework will be analyzed in the future.

References

- [1] J. Li, *et al.*: "An area-efficient microprocessor-based SoC with an instruction-cache transformable to an ambient temperature sensor and a physically unclonable function," *IEEE J. Solid-State Circuits* **53** (2018) 728 (DOI: [10.1109/JSSC.2018.2791460](https://doi.org/10.1109/JSSC.2018.2791460)).
- [2] J. Romero, *et al.*: "Energy efficient peripheral and system buses for low-area and low-power SoC applications," *IEEE Trans. Circuits Syst. II, Exp. Briefs* **67** (2020) 866 (DOI: [10.1109/TCSII.2020.2984018](https://doi.org/10.1109/TCSII.2020.2984018)).
- [3] G. Qiu, *et al.*: "A dual dynamic key chaotic encryption system for industrial cyber-physical systems," *IEICE Electron. Express* **17** (2020)

- 20200389 (DOI: [10.1587/elex.17.20200389](https://doi.org/10.1587/elex.17.20200389)).
- [4] S. Manich, *et al.*: "Detection of probing attempts in secure ICs," 2012 IEEE Int. Symp. Hardware-Oriented Secur. Trust (2012) 134 (DOI: [10.1109/HST.2012.6224333](https://doi.org/10.1109/HST.2012.6224333)).
- [5] S. Skorobogatov: "How microprobing can attack encrypted memory," 2017 Euromicro Conf. Digital Syst. Des. (DSD) (2017) 244 (DOI: [10.1109/DSD.2017.69](https://doi.org/10.1109/DSD.2017.69)).
- [6] X.T. Ngo, *et al.*: "Cryptographically secure shield for security IPs protection," *IEEE Trans. Comput.* **66** (2017) 354 (DOI: [10.1109/TC.2016.2584041](https://doi.org/10.1109/TC.2016.2584041)).
- [7] K. Wang, *et al.*: "Multi-pair active shielding for security IC protection," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.* **38** (2019) 2321 (DOI: [10.1109/TCAD.2018.2878175](https://doi.org/10.1109/TCAD.2018.2878175)).
- [8] H. Wang, *et al.*: "Probing assessment framework and evaluation of antiprobing solutions," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **27** (2019) 1239 (DOI: [10.1109/TVLSI.2019.2901449](https://doi.org/10.1109/TVLSI.2019.2901449)).
- [9] H. Wang, *et al.*: "A physical design flow against front-side probing attack by internal shielding," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.* **39** (2020) 2152 (DOI: [10.1109/TCAD.2019.2952133](https://doi.org/10.1109/TCAD.2019.2952133)).
- [10] S. Manich and M. Strasser: "A highly time sensitive XOR gate for probe attempt detectors," *IEEE Trans. Circuits Syst. II, Exp. Briefs* **60** (2013) 786 (DOI: [10.1109/TCSII.2013.2278126](https://doi.org/10.1109/TCSII.2013.2278126)).
- [11] M. Weiner, *et al.*: "The low area probing detector as a countermeasure against invasive attack," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **26** (2018) 392 (DOI: [10.1109/TVLSI.2017.2762630](https://doi.org/10.1109/TVLSI.2017.2762630)).
- [12] M. Weiner, *et al.*: "A calibratable detector for invasive attack," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **27** (2019) 1067 (DOI: [10.1109/TVLSI.2019.2892408](https://doi.org/10.1109/TVLSI.2019.2892408)).
- [13] Y.-W. Lee, *et al.*: "Robust secure shield architecture for detection and protection against invasive attack," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.* **39** (2020) 3023 (DOI: [10.1109/TCAD.2019.2944580](https://doi.org/10.1109/TCAD.2019.2944580)).
- [14] L. Fan, *et al.*: "Data security concurrent with homogeneous by AES algorithm in SSD controller" *IEICE Electron. Express* **11** (2014) 20140535 (DOI: [10.1587/elex.11.20140535](https://doi.org/10.1587/elex.11.20140535)).
- [15] S. Angizi, *et al.*: "Design and evaluation of a spintronic in-memory processing platform for nonvolatile data encryption," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.* **37** (2018) 1788 (DOI: [10.1109/TCAD.2017.2774291](https://doi.org/10.1109/TCAD.2017.2774291)).
- [16] M. Da Silva, *et al.*: "Preventing scan attacks on secure circuits through scan chain encryption," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.* **38** (2019) 538 (DOI: [10.1109/TCAD.2018.2818722](https://doi.org/10.1109/TCAD.2018.2818722)).
- [17] M. Bahadori and K. Järvinen: "A programmable SoC-based accelerator for privacy-enhancing technologies and functional encryption," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **28** (2020) 2182 (DOI: [10.1109/TVLSI.2020.3010585](https://doi.org/10.1109/TVLSI.2020.3010585)).
- [18] A. Karati, *et al.*: "Provably secure and generalized signcryption with public verifiability for secure data transmission between resource-constrained IoT devices," *IEEE Internet Things J.* **6** (2019) 10431 (DOI: [10.1109/JIOT.2019.2939204](https://doi.org/10.1109/JIOT.2019.2939204)).
- [19] O. Kara and M.F. Esgin: "On analysis of lightweight stream ciphers with keyed update," *IEEE Trans. Comput.* **68** (2019) 99 (DOI: [10.1109/TC.2018.2851239](https://doi.org/10.1109/TC.2018.2851239)).
- [20] Y. Zhang, *et al.*: "High performance AES-GCM implementation based on efficient AES and FR-KOA multiplier," *IEICE Electron. Express* **15** (2018) 20180559 (DOI: [10.1587/elex.15.20180559](https://doi.org/10.1587/elex.15.20180559)).
- [21] D.B. Roy and D. Mukhopadhyay: "High-speed implementation of ECC scalar multiplication in GF(p) for generic Montgomery curves," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **27** (2019) 1587 (DOI: [10.1109/TVLSI.2019.2905899](https://doi.org/10.1109/TVLSI.2019.2905899)).
- [22] Y. Zhao, *et al.*: "A resource and timing optimized PCIe DMA architecture using FPGA internal data buffer," *IEICE Electron. Express* **16** (2019) 20180858 (DOI: [10.1587/elex.15.20180858](https://doi.org/10.1587/elex.15.20180858)).
- [23] W. Shen, *et al.*: "Enabling identity-based integrity auditing and data sharing with sensitive information hiding for secure cloud storage," *IEEE Trans. Inf. Forensics Security* **14** (2019) 331 (DOI: [10.1109/TIFS.2018.2850312](https://doi.org/10.1109/TIFS.2018.2850312)).
- [24] T. Agarwal, *et al.*: "Multi-level time-sensitive networking (TSN) using the data distribution services (DDS) for synchronized three-phase measurement data Transfer," *IEEE Access* **7** (2019) 131407

- (DOI: [10.1109/ACCESS.2019.2939497](https://doi.org/10.1109/ACCESS.2019.2939497)).
- [25] R. Ueno, *et al.*: “High throughput/gate AES hardware architectures based on datapath compression,” *IEEE Trans. Comput.* **69** (2020) 534 (DOI: [10.1109/TC.2019.2957355](https://doi.org/10.1109/TC.2019.2957355)).
 - [26] N.-P. Tran, *et al.*: “High throughput parallelization of AES-CTR algorithm,” *IEICE Trans. Inf. Syst.* **E96–D** (2013) 1685 (DOI: [10.1587/transinf.E96.D.1685](https://doi.org/10.1587/transinf.E96.D.1685)).
 - [27] M. Mohammadi, *et al.*: “Energy efficient on-demand dynamic branch prediction models,” *IEEE Trans. Comput.* **69** (2020) 453 (DOI: [10.1109/TC.2019.2956710](https://doi.org/10.1109/TC.2019.2956710)).
 - [28] M. Das, *et al.*: “Shared pattern history tables in multicomponent branch predictors with a dealiasing cache,” *IEEE Embedded Syst. Lett.* **12** (2020) 95 (DOI: [10.1109/LES.2019.2957512](https://doi.org/10.1109/LES.2019.2957512)).
 - [29] C. Li, *et al.*: “Priority-based PCIe scheduling for multi-tenant multi-GPU systems,” *IEEE Comput. Archit. Lett.* **18** (2019) 157 (DOI: [10.1109/LCA.2019.2955119](https://doi.org/10.1109/LCA.2019.2955119)).
 - [30] R. Uytterhoeven and W. Dehaene: “Completion detection-based timing error detection and correction in a near-threshold RISC-V microprocessor in FDSOI 28 nm,” *IEEE Solid-State Circuits Lett.* **3** (2020) 230 (DOI: [10.1109/LSSC.2020.3006626](https://doi.org/10.1109/LSSC.2020.3006626)).
 - [31] L. Kusters and F.M.J. Willems: “Secret-key capacity regions for multiple enrollments with an SRAM-PUF,” *IEEE Trans. Inf. Forensics Security* **14** (2019) 2276 (DOI: [10.1109/TIFS.2019.2895552](https://doi.org/10.1109/TIFS.2019.2895552)).
 - [32] S.K. Cherupally, *et al.*: “A smart hardware security engine combining entropy sources of ECG, HRV, and SRAM PUF for authentication and secret key generation,” *IEEE J. Solid-State Circuits* **55** (2020) 2680 (DOI: [10.1109/JSSC.2020.3010705](https://doi.org/10.1109/JSSC.2020.3010705)).