CRFlex: A Flexible and Configurable Cryptographic Hardware Accelerator for AES Block Cipher Modes

Pietro Nannipieri, Luca Baldanzi, Luca Crocetti, Stefano Di Matteo, Francesco Falaschi, Luca Fanucci, and Sergio Saponara

University of Pisa, Dept. of Information Engineering, Via G.Caruso, 16, Pisa, Italy

Abstract. This paper presents a System-on-Chip (SoC) implementation of a cryptographic hardware accelerator supporting multiple AES based block cypher modes, including the more advanced CMAC, CCM, GCM and XTS modes. Furthermore, the proposed design implements in hardware advanced features for AES key secure storage. A flexible interface allows the communication between the hardware accelerator and the chosen processor and makes this implementation suitable to be easily integrated into a generic embedded system. The system has been prototyped and characterized on a Xilinx Zynq 7000 platform. Synthesis results on a 7 nm CMOS Standard-Cell library are proposed too, showing competitive performances and resource usage respect to the State of Art and assessing the portability in different technology libraries of the proposed design. Furthermore, power consumption data are extracted to prove the suitability of the hardware acceleration also in the case of power-constrained devices.

Keywords: Security; Advanced Encryption Standard (AES); SoC; Real-time; Hardware Accelerator

1 Introduction

Generally, an embedded system is equipped with a microprocessor and a pure software solution can be a valid approach to perform security algorithms. Nevertheless, in specific application cases the performance of the embedded microprocessor could be insufficient to respect the latency and/or throughput requirements, for example in real-time IoT applications [1]. In these cases, a valid solution could be to privilege hardware (HW) domain solutions rather than software (SW) domain solutions, for security algorithms. Furthermore, a HW-based acceleration of cybersecurity functionalities may be needed to minimize power consumption [2], [3], [4]. Symmetric key algorithms are preferred over asymmetric ones due to their performance advantage, and the Advanced Encryption Standard (AES)[5] is the most popular of the symmetric key algorithms. HW implementations of the AES have demonstrated to have better performance than SW ones [6], to the point that pure SW implementations have

become uncommon in performance and power critical environments. This work presents an implementation of a configurable AES-based hardware accelerator whose differentiating aspects are reported herein. Multiple block cypher modes have been implemented, ensuring not only confidentiality but also integrity and Authenticated Encryption with Associated Data (AEAD). The proposed Intellectual Property-core (IP) has been named CRFlex core, which stands for Crypto-Flexible core. The flexibility refers to the possibility to select only part of the accelerator architecture at synthesis time, including only the modes that are strictly required for the application, hence minimizing the logic resources. The internal interconnection and the interface to interact with the SW layer are flexible as well and automatically generated depending on the selected features. The solution aims to be complete and easy to integrate in a SoC, where fast and context-aware security approaches are required. Moreover, also advanced policies for secure keys storage are implemented in hardware. The accelerator has been synthesized on 28 nm FPGA device and on a 7nm Standard-Cell technology. To the best of the authors knowledge, this work presents the first public data of AES implementation on extremely scaled technology (i.e. 7nm).

2 AES acceleration: hardware architecture

In literature, multiple approaches to achieve an acceleration of cryptographic algorithms can be found, ranging from dedicated ASIC solutions to customized central processing units, passing through a myriad of possible HW/SW hybrid implementations. In [7] a brief comparative analysis between different approaches and algorithms has been presented. Some notable HW/SW acceleration methods are Instruction Set Extension (ISE) [8], flexible dedicated crypto processors [9] and hardware accelerators. Among these methods, the most suitable to be generally employed is the hardware accelerator solution. In [10] the authors propose an AES IP core based on Altera Avalon bus interfacing with a NIOS II softcore, able to support ECB, CBC, OFB, CFB and CTR modes. A similar approach is followed in [11] where the authors have achieved a lower gate count at the cost of a lower speed by using a Xilinx Microblaze processor coupled with a custom accelerator using the Processor Local Bus (PLB). In the mentioned cases, the stand-alone AES modules are capable to maintain very high throughputs but suffer a considerable decrease in performance when connected to the processor, typically of some orders of magnitude. This is due to the bus interconnect that represents the system bottleneck and limits the communication speed between CPU and hardware accelerator. Our proposed design is a SoC based hardware accelerator that performs the basic block cypher modes such as ECB, CBC, OFB, CFB and CTR. Then, it offers advanced block cypher modes to guarantee integrity and authenticity along with confidentiality, such as CMAC, GCM and CCM modes. Furthermore, CRFlex performs also the XTS block cypher mode for disk encryption support. Our solution has been conceived to give the user the maximum flexibility in the selection of the minimum subset of AES-based cypher block modes to be included in the finally implemented hardware accel-

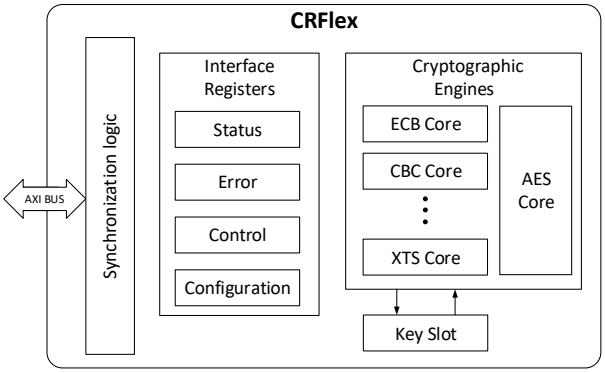


Fig. 1. CRFlex hardware accelerator architecture.

erator, making it suitable for different applications while minimizing the area overhead. The architecture of the CRFlex accelerator is shown in Figure 1. The main system components and features include: a **Memory mapped interface** to connect the hardware module with the CPU via AXI bus, including a dedicate synchronization logic to manage clock domain crossing; a **Interface registers** to handle the interpretation of the input commands (OPcodes); an **Internal key slot register** to perform a secure key storage and reduce system latency in case of multiple AES operations with the same cipher key; a **Multiple AES cipher block** modules, conditionally instantiated and interconnected; a **Globally shared AES Core**. The operations are controlled by the set of interface registers, that receives as input the 32-bit control register and generates the control signals that drive the underlying sub-modules. The micro-controller shall read each output before providing to it a new input.

AES core

Our proposed architecture instantiates a single AES core, to be shared by all the others implemented block cipher modes accelerator. The drawback is that a single mode at a time can be driven from the processor; the advantage is that the resource utilization is optimized. During the design phase, we carried out a trade-off between performance and complexity to properly implement the AES module. We implemented hardware functions belonging to one round, which are then used iteratively for several rounds (10 to 14, depending on the key size). To further save logic resources, we decided to support only 128 and 256-bit key lengths. It is important to observe that the AES-256 algorithm is considered secure against post-quantum attacks [12], which makes this solution also applicable to future designs. The AES core is based on the architecture presented in [13].

ECB, CBC, OFB, CFB, CTR cores

All the basic AES modes of operations described in the National Institute of Standards and Technology NIST special publication 800-38A [14] have been implemented. These modes are similar to each other and the convenience to use one instead of the other depends on the application. Among CBC, CFB, OFB and CTR, only the CTR mode does not present any dependencies among encryption/decryption results of current and previous data blocks. These block cypher modes (ECB, CBC, OFB, CFB and CTR) share a single AES core.

CMAC core

The Cipher-based MAC (CMAC) mode from NIST special publication 800-38B [15] is included in the CRFlex to provide a mode capable to guarantee the integrity of information. It performs customized processing of the input message using a cypher block chaining technique and returns a bit string called Message Authentication Code (MAC), also known as Tag.

CCM and GCM cores

The NIST special publications 800-38C and 800-38D, [16] and [17], describe the CCM mode and the GCM mode respectively, two advanced cryptosystems to achieve simultaneously confidentiality, integrity and authentication over sensible data, or rather the Authenticated Encryption (AE).

XTS core

The XTS mode, described in NIST special publication 800-38E [18], is included inside the CRFlex design. With the support of XTS mode, the CRFlex is a suitable solution also in the case of applications connected with external storage devices (e.g. hard disk). XTS-AES provides confidentiality of data, with the security strength of the AES algorithm, for block-oriented storage devices.

3 Implementation on FPGA and standard-cell technology

All data presented in this section refer to the folded architecture of AES core inside CRFlex design, i.e. instancing only one stage of AES core. Our design can instantiate an unfolded architecture with multiple stages of the internal AES (from 2 up to 14 in the case of AES-256), using synthesis parameters. In this work, we report only the single-stage case because our focus is a compact and low power design. The CRFlex module was synthesized on the Xilinx Zynq-7000 xc7z045ffg900-2 FPGA using Xilinx Vivado. All data herein reported referring to post-implementation (i.e. place and route) results. The HW frequency chosen for the synthesis is 166.66 MHz. In order to assess the portability of the proposed system, the CRFlex module was synthesized also on two different standard-cell

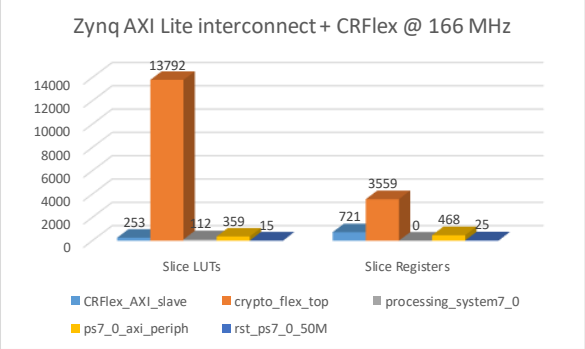


Fig. 2. System complexity: Slice LUTs and Slice Registers for 1 signle AES core with all modes of operation.

technologies (i.e. 45nm and 7nm). On 45nm, the CRFlex core can reach 620 MHz of clock frequency and a maximum throughput of 7.93 Gbps. Table 1 shows the results in terms of logic resources usage, expressed in kilo Gate Equivalent (kGE), and power consumption for two different configurations of the CRFlex, taken as representative examples of the multiple configuration of supported by our IP core: the AES-ECB mode and the AES-GCM mode. The synthesis was performed under *typical* conditions: 1.1V of supply voltage and 25° C for environmental temperature. The frequency chosen for the power characterization on standard-cell is 100 MHz, because this frequency is a good trade-off between throughput and power consumption. The power analysis was performed also considering the switched activity of the hardware modules, extracted by means of dedicated testbenches. The Electronic Design Automation (EDA) tools used for this characterization are: *Design Compiler* by Synopsys for netlist extraction, *Questa* by Mentor Graphics for post synthesis simulation and *Prime Time* by Synopsys for power extraction.

Cipher Mode	**Logic Usage**	**Leakage power**	**Dynamic Power @ 100 MHz**
AES-ECB	14.02 kGE	0.27 mW	7.33 mW
AES-GCM	61.01 kGE	0.96 mW	25.1 mW

Table 1. Synthesis on 45 nm Standard-Cell technology: cases of CRFlex only instancing AES-ECB and AES-GCM respectively (AES single stage architecture).

The power reports show that the CRFlex hardware accelerator is a valid and suitable solution also for power-constrained applications (e.g. mobile, IoT, etc.).

AES-ECB-256		
# Stage(s)	**Logic Usage**	**Throughput**
1 Stage	28 kGE	27.4 Gbps
2 Stages	55.7 kGE	55 Gbps
7 Stages	195 kGE	192 Gbps
14 Stages	370 kGE	384 Gbps

Table 2. Synthesis on 7 nm Standard-Cell technology: case of CRFlex only instancing AES-ECB (AES single stage architecture).

Synthesis on 7 nm technology On 7 nm technology the CRFlex core can reach 3.0 GHz of clock frequency and a maximum throughput of 27.4 Gbps with a single AES stage instantiated. Table 2 shows the results in terms of logic resources usage and throughput for all the possible number of cascaded stages for the AES-256 algorithm. The synthesis was performed with the maximum frequency reachable by the CRFlex. The modifications to increase the throughput would be suitable for contexts where system speed is more important than flexibility and chip size. On 7 nm technology, the CRFlex is a valid solution also for high speed applications, such as hardware acceleration in the new generation of High Performance Computing (HPC) systems.

4 Comparison to the State of the Art

In this section we compare our CRFlex against other AES hardware accelerators implemented on 45 nm Standard-Cell technology. To our knowledge, no results of AES hardware implementations on 7 nm technology can be found in other academic and commercial works. The comparison with other architectures is very complex due to the differences in supported functionalities among the different implementations and the lack of sufficient performance data in the literature. We compared the performance respect only to the CBC mode, instantiating only one AES core inside the CRFlex.

Table 3 shows the comparison among different works in terms of supported modes of operation, area and throughput. Area is measured in Gate Equivalent (GE) and throughput refers only to CBC mode. The work in [19] is a commercial AES hardware IP that supports different modes of operation. The circuit area is around 49-53 kGE and the information about throughput are not published. Work in [20] is an AES hardware accelerator that support only ECB and CBC modes. It is aimed to be used for content protection for high-performance applications. Work in [9] is a flexible crypto-processor that supports different symmetric key algorithms. Respect to the other works, our CRFlex is the more complete in terms of supported modes of operation, with performance in line with the State of the Art.

	Our Work	**[19]**	**[20]**	**[9]**
Key lenght	128-256	128-256	128-256	128
Supported Modes of Operation	ECB,CBC,OFB CFB,CTR,CCM GCM,CMAC,XTS	ECB,CBC CTR,GCM	ECB,CBC	ECB,CBC OFB,CFB
Technology	7nm/45nm	45nm	45nm	45nm
Area [kGE]		49-53	187.67	7925.31
Throughput CBC [Gbps]	29.45/7.93	–	16.74	6.40

Table 3. Comparison among different implementations of AES

5 Conclusions

This work presented a SoC implementation of a co-processor for AES-based block cypher modes, suitable to be employed in embedded applications. The accelerator was implemented in a 28 nm FPGA device and 7 nm standard-cell technologies. A complete system with an ARM® Cortex A9 and the proposed accelerator has been prototyped on a Zynq 7000 SoC. Complexity and performance have been thoroughly characterized for all the AES modes of operations in all the above mentioned technologies. To the best of the author's knowledge, this work is the first presenting in public literature AES implementation results on 7nm technology. In addition, its flexibility allows for a selection of the minimum resources needed for the chosen application, thus minimizing the cost of the resources, thanks to the shared AES engine. Even in the case in which only a single block cypher mode is selected the overhead of the flexibility is negligible since the crypto-arbiter module slice LUT occupation is 0.03% of the AES one (the registers composing the memory map interface is just 30 32-bits registers, thus again negligible); moreover, the hardware accelerator interface can be shared with other accelerators on the bus. The system interface allows easy integration in embedded systems that require high-performance cryptographic acceleration. The CRFlex interface can be easily modified to match the specific bus used. The module can be accessed either as a common memory-mapped device or as using the DMA engine, depending on the required throughput.

References

1. Fahim Rahman, Mohammad Farmani, Mark Tehranipoor, and Yier Jin. Hardware-Assisted Cybersecurity for IoT Devices. *IEEE 18th International Workshop on Microprocessor and SOC Test and Verification*, 2017.
2. P. Nannipieri, M. Bertolucci, L. Baldanzi, L. Crocetti, S. Di Matteo, F. Falaschi, L. Fanucci, and S. Saponara. Sha2 and sha-3 accelerator design in a 7 nm technology within the european processor initiative. *Microprocessors and Microsystems*, 2020.
3. P. Nannipieri, S. Di Matteo, L. Baldanzi, L. Crocetti, J. Belli, L. Fanucci, and S. Saponara. True random number generator based on fibonacci-galois ring oscillators for fpga. *Applied Sciences (Switzerland)*, 11(8), 2021.

4. Stefano Di Matteo, Luca Baldanzi, Luca Crocetti, Pietro Nannipieri, Luca Fanucci, and Sergio Saponara. Secure elliptic curve crypto-processor for real-time iot applications. *Energies*, 14(15), 2021.
5. NIST. FIPS 197: Advanced Encryption Standard (AES). *Federal Information Processing Standards Publication*, 197(441):0311, 2001.
6. Luca Baldanzi, Luca Crocetti, Stefano Di Matteo, Luca Fanucci, Sergio Saponara, and Patrice Hameau. Crypto accelerators for power-efficient and real-time on-chip implementation of secure algorithms. In *2019 26th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 775–778. IEEE, 2019.
7. Muhammad Rashid, Malik Imran, and Atif Raza Jafri. Comparative analysis of flexible cryptographic implementations. In *Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC), 2016 11th International Symposium on*, pages 1–6. IEEE, 2016.
8. N. Ben Hadjy Youssef, W. El Hadj Youssef, M. Machhout, R. Tourki, and K. Torki. nstruction set extensions of AES algorithms for 32-bit processors. In *2014 International Carnahan Conference on Security Technology (ICCST)*, pages 1–5, 2014.
9. Gokhan Sayilar and Derek Chiou. Cryptoraptor: High Throughput Reconfigurable Cryptographic Processor. In *Proceedings of the 2014 IEEE/ACM International Conference on Computer-Aided Design*, pages 154–161. IEEE Press, 2014.
10. Xi C. Tao, Di L. Zhang, and Yi K. Song. An Implementation of Configurable and Small-Area AES IP Core Oriented Avalon Bus. 2015.
11. K. Chang, Y. Chen, C. Hsieh, C. Huang, and C. Chang. Embedded a low area 32-bit AES for image encryption/decryption application. In *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, pages 1922–1925. IEEE, 2009.
12. Vasileios Mavroeidis, Kamer Vishi, Mateusz D. Zych, and Audun Jøsang. The Impact of Quantum Computing on Present Cryptography. *(IJACSA) International Journal of Advanced Computer Science and Applications*, 9(3), 2018.
13. Rei Ueno, Sumio Morioka, Naofumi Homma, and Takafumi Aoki. A High Throughput/Gate AES Hardware Architecture by Compressing Encryption and Decryption Datapaths - Toward Efficient CBC-Mode Implementation. *Cryptology ePrint Archive, Report 2016/595*, 2016.
14. Morris Dworkin. NIST Special Publication 800-38A. Technical report, 2001.
15. Morris Dworkin. *NIST Special Publication 800-38B*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2005.
16. Morris Dworkin. *NIST Special Publication 800-38C*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2004.
17. Morris Dworkin. *NIST Special Publication 800-38D*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2007.
18. Morris Dworkin. *NIST Special Publication 800-38E*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, 2008.
19. Crypt-ip-120 aes crypto, rambus. `https://www.rambus.com/security/crypto-accelerator-hardware-cores/basic-crypto-blocks/crypt-ip-120/`. Accessed: 2021-06-04.
20. Sanu K Mathew, Farhana Sheikh, Michael Kounavis, Shay Gueron, Amit Agarwal, Steven K Hsu, Himanshu Kaul, Mark A Anders, and Ram K Krishnamurthy. 53 gbps native $gf(2^4)^2$ composite-field aes-encrypt/decrypt accelerator for content-protection in 45 nm high-performance microprocessors. *IEEE Journal of Solid-State Circuits*, 46(4):767–776, 2011.