On the Hardware Implementation Efficiency of CAESAR Authentication Ciphers for FPGA Devices

Nicolas Sklavos [I], Paris Kitsos [II], Artemios G. Voyiatzis [III]

[I] SKYTALE Group,
Computer Engineering & Informatics Department,
University of Patras, Hellas

[II] Computer Informatics & Engineering Department,
Technological Educational Institute of Western Greece, Hellas

[III] SBA Research, Vienna, Austria

***Abstract*— Ciphers, also known as authenticated encryption methods, are the outcome of the marriage between the fields of mathematics and logic. The association of ciphers and data comprises Cryptography, the science that assures security and secrecy during a discussion of two parties in the presence of a third one; authentication, integrity and confidentiality are the values of the field in which we trust. In this work, four Caesar Round-Two variants are developed with the register transfer-level (RTL) abstraction, described by a hardware design language (HDL), simulated and implemented on Xilinx FPGAs. COLM, SCREAM, POET and Minalpher variants of the contest are all following an indistinguishable process to ensure the aftermath accuracy, competing each other in the meanings of throughput, area, and throughput-to-area (T/A) quota. Results are being presented and discussed over these aspects.**

Keywords*— *RTL Implementations; field programmable gate arrays; cryptography; authenticated ciphers; CAESAR; COLM; SCREAM; POET; Minalpher

I. INTRODUCTION

Authenticated ciphers assure certain qualities while encrypting; confidentiality, integrity and authenticity. The encrypting procedure combines input and the secret key under a way given by the protocol in use. A plaintext message and associated data, the number of public messages N_{pub} and, at times, the number of secret messages N_{sec} are considered to be the input, whereas, the key is secret and known exclusively between the two parties. Once the encrypted message, known as ciphertext, is received by the targeted party, it has to be decrypted, in order to have accessible content for the recipient. Decrypting the ciphertext is an inverse scheme, where the receiver uses the key, the ciphertext and associated data, the numbers N_{pub} and encrypted N_{sec}, if existing, to reproduce the initial message sent.

But how is confidentiality, integrity and authenticity certified during the whole process? Confidentiality is profoundly assured, as the ciphertext is computed by a function of the initial input and the secret key. As far as the remaining qualities are concerned, there is a part during encryption, right after the transformation of the plaintext, where a quantity, known as Tag, is created and adhered to the end of the ciphertext. Tag is mainly a hash function that uses every detail given as input plus the key to be computed. Consequently, when the encrypted message is received, there is a quantity that corresponds to Tag, marked as Tag'; if and only if Tag and Tag' are the equal, integrity and authenticity are proven.

Well-known cryptography competitions as CAESAR, urge researchers to improvise innovative ciphers [1]. The procedure is divided into knock-out rounds, where candidates are asked to submit parts of their work, under pre-defined specifications. Ciphers submitted are descripted in a hardware design language (HDL), such as VHDL. It is usual that the early rounds do not require any proof for the hardware efficiency of the ciphers, even if its importance is well-acknowledged in order to prevent cases of high complexity and/or presence of redundant components. Based on the volume of nominees, examining each and every hardware implementation, that could likely have undesired flaws even in its logic, can be really time-consuming. Although, there are times that candidates present estimated hardware efficiency along with cipher description, that come out compared to their default characteristics [2].

In this work, four CAESAR contestants that have already made it to Round Two, have been implemented, simulated and examined concerning throughput, area, and throughput-to-area (T/A) ratio in Xilinx Virtex FPGAs [1]. The dominant factor to specifically choose SCREAM, POET, Minalpher and COLM over the total of Round-Two ciphers is the cryptographic primitive 'Tweakable Block Cipher', in which first three are based and the similar structure of the fourth one; fact that allows a fertile ground to conduct any kind of comparison [3]. Under obligation towards trustworthy observations, hardware implementation is also developed in the universal hardware CAESAR Application Programming Interface (API) for authentication based on symmetric encryption algorithms [4].

II. CAESAR COMPETITION

Back to 1997, the United States National Institute of Standards and Technology (NIST) conducted a competition for a new Advanced Encryption Standard (AES); in 2004, ENCRYPT targeted in finding a new stream cipher suitable for widespread adoption, while, in 2007 another competition was looking for a new hash standard to agree on. As tradition holds, in 2013, CAESAR competition was announced [1].

CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) is aiming to the invention of authenticated symmetric-encryption algorithms. Contestants design and develop ciphers that satisfy certain pre-settled directions. CAESAR is structured into three knock-out rounds, followed by the phase of the finalists. The procedure is planned to last approximately four years; commencing with the first Round in January 2013 until May 2014, participants submit a kind of premature version of their cipher, which is describing the cipher software-wise. Once successful to Round Two submissions are announced, in July 2015, their researchers are allowed to import any applicable tweaks and, simultaneously, are asked to attach a description for the hardware layout of their cipher. By the time being (June 2016), it is known that COLM, SCREAM, POET and Minalpher, are successfully selected to proceed to Round Two. In August 2016, announcement for Third Round candidates will take place; chosen participants will enter the phase of the finalists. CAESAR winner will be publicly disclosed, on 15th December 2017.

III. ANALYSIS OF THE COMPARATIVE BASIS

Principal purpose of this work, is to focus on comparing and contrasting the efficiency of four authenticated ciphers, under the same conditions of simulation and implementation. For this cause, all ciphers –updated to any applicable Round Two tweaks- will once be developed according to published standardizations and a universal, fertile for comparison, interface. The CAESAR Hardware API is specially designed for the situation by George Mason University (GMU) [5]. What CAESAR API basically is an input/output interface (I/O) that allows flexible reception, segmentation and formatting of the data processing. In case of Serial Input/Parallel Output (SIPO) or Parallel Input/Serial Output (PISO), not only they are supported by the API, but it is also proved to require less interconnections; considering that, the span of applications implemented on FPGAs can significantly augment.

Hardware API favors candidates, given that is offering some extra features that are applied universally [6]. During the implementations, all ciphers used the padding technique; when incomplete blocks of associated data exist, then '1' followed by the needful total of '0's is appended to the block. Though, there is a chance for an undesired drawback to occur; the upper process might cause the creation of an extra block anew. Truncation of the final block is also a service that happens frequently enough; when truncation length is known after synthesis time, the procedure could turn out extremely pricey. CAESAR Hardware API cater for all upper services automatically, resulting simultaneously in great financial savings. Candidates need to put their encryption/decryption main unit into cipher core datapath and cipher core cotroller, so as to be in position of using the CAESAR API. A meaningful remark can be made at this point, regarding the design of the controller and the signal management, which is flexible and efficient in encrypting as much as in decrypting, too.

Before the universal implementation of the CAESAR API, candidates used their own, custom-made interfaces, which of course did not meet any common requirements among three ciphers. Top-level interfaces are mentioned in detail in [7], [8], [9] and [10] for COLM, SCREAM, POET and Minalpher respectively.

IV. EXAMINATION OF THE AUTHENTICATED CIPHERS

In this Section, main aspects of the chosen CAESAR ciphers will be presented. Comprehending the basic procedures of encryption and decryption, as depicted in Figure 1, proves to be much advantageous for the target of this work. Firstly, COLM will be examined, followed by SCREAM, POET, and then Minalpher.

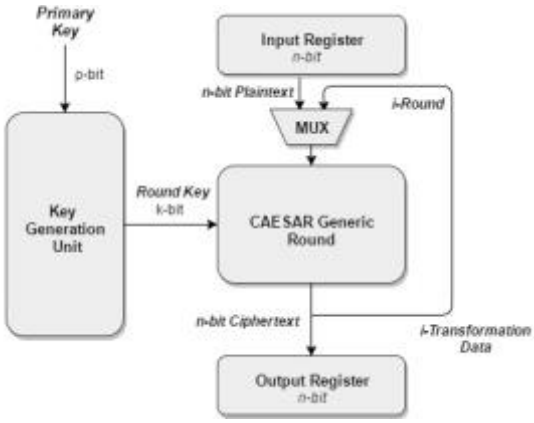

Fig. 1. *Generic System Architecture*

COLM cipher is actually formulated as a mixture of characteristics adopted from COPA and ELmD ciphers [7]. COLM family is parameterized according to value τ, known as the enumeration of blocks after which intermediate tags will be created; for instance, $COLM_0$ has no intermediate tags whereas $COLM_{127}$ does. Encryption key K, associated data, original message, N_{pub} and a set of parameters are combined under specified ways to generate ciphertext C and contingent intermediate tags T, which will be used during decryption to retrieve original message M. Once retrieval is complete, tag verification is held, to validate authenticity.

SCREAM (Side-Channel Resistant Authenticated Encryption with Masking) is an authenticated cipher, which was also successfully selected as a Round Two candidate. Its logic is based on the idea of Tweakable Authenticated Encryption (TAE) by Liskov et al [3]. What is basically happening during encryption is a repeatedly procedure for each step, commencing with calculating a tweak key TK_i; TK_i is an operation among the secret key, a nonce and the counter i, which indexes the number of the block. After TK_i is known, the 128-bit status word passes through a Substitution Box (S-Box), and in the way out a round constant RC (ρ, σ) is added

to its value. RC is a variable of round number $\rho \in \{0,1\}$ and step number $\sigma \in \{0...N_S\text{-}1\}$. The amount that results from the addition goes through a permutation (L-Box). Round Two version of SCREAM cipher includes a lightly updated S-Box, which allows better performance during differential calculations and is decreasing algebraic degree. Furthermore, throughput-to-area (T/A) ratio is improved when S-Box and L-Box are apart, so this is also the approach that is examined [11].

POET encryption cipher (Pipelineable On-line Encryption with authentication Tag) is an independent, authenticated cipher that allows user to select any hash function or cipher block as long as they fulfil expectations of [12]. Round Two version is importantly improved; the amount of subkeys required for encryption is lessened to 3, from the initial total of 5, while, simultaneously, the final header block can be calculated with 3 to 5 times fewer multiplications. Subkeys K and L are computed as a function of key via AES-main; K is used for all operations, while L is used to mask the blocks of associated data. In the latest version of POET, designers used two distinguished subkeys for AES-Top and AES-Bottom, which are now merged as one shared subkey K_f, an action ending up in great area savings. As long as all subkeys are computed, a portion known as hash value is encrypted and is now called authentication value τ. Registers AES-top and AES-bottom are initialized and the procedure continues as described in [12]. The authenticated cipher uses a verification tag, which is created from a part of the encrypted final block. This is one of the most expensive steps of the whole procedure, as it demands the existence of variable shifters and several truncations.

Minalpher, the third chosen authenticated cipher of this work, is specifically a cryptographic core, also known as a Tweakable Even-Mansour (TEM). Four alterations can happen in a TEM while encrypting or decrypting [13]. For each time there is a call towards TEM, a new tweak is being computed. While combining the secret key K, a flag and a nonce, tweak is figured out as a function of three other portions; the initial tweak value L, the root y of a composite Galois Field polynomial, and the variable exponents i and j that increase by 0,1 or 2 with each block. As L is the initial tweak that depends on K, flag and N_{pub} it needs to update right before every TEM round. A slight point where Minalpher differs from the other two ciphers is that it uses the 10* padding technique not only in associated data but also in plaintext processing. In the case that the last plaintext occurs to be a full block, the number of blocks will expand to one more, as 10* padding has to take place under obligation. Encrypting procedure begins with the processing of first plaintext, which turns into the first ciphertext through a TEM, and consequently the latest passes through another TEM, whose exit is finally added to the current hash. Once all plaintexts have turned into ciphertexts, the latest hash added to the conclusive plaintext block contribute to the calculation of the tag. During decrypting two cores are being used, as in encrypting too, but in this case tag can be computed at the same time with the decryption of the final ciphertext. Unfortunately, the size of the primitive plaintext can only have been known after decryption is complete, as the 10* padding has applied. In this implementation, examines ciphertext inch by inch to discover the specific point where 10* padding commenced; this will also be the length of the ciphertext.

V. DESIGN, COMPARISON AND RESULTS

Implementations of the four examined authenticated ciphers, were described in VHDL hardware description language. Formulating an unbiased base for all comparisons and remarks, demands one common testing field and careful examination under predetermined conditions; for this cause, CAESAR Hardware API, and Xilinx Vivado Design Suite, were used. Measurements and results of the above applications were all checked regarding validity and proper functionality via Xilinx tools. All input plaintexts and key vectors are by default the size of 128-bit, unless indicated otherwise.

	COLM	***SCREAM***	***POET***	***MINALPHER***
T_{CLK} *(ns)*	9,2592	8,9816	4,2655	4,4064
f_{CLK} *(MHz)*	108	118,141	211,565	236,232
Slices	2633	859	2003	1640
LUTs	6878	2993	7401	6073
Registers	4200	1591	3631	4045
Throughput (Mbps)	1257	1374	2708	3182
Throughput to Area (T/A)	0,477	0,459	0,366	0,524

Table 1. FPGA Implemenations Synthesis Results

Table 1 presents in detail interesting aspects of CAESAR ciphers that relate directly with their effectiveness. According to the axis of T/A ratio of authenticated ciphers in an FPGA Virtex device, values –calculated as the quotient of throughput to slices- for COLM, SCREAM, POET and Minalpher ciphers are depicted in Figure 3. Profoundly, Minalpher is the authenticated cipher with the greatest T/A ratio at 0,524; COLM, SCREAM and POET are following consequently, at 0,477, 0,459 and 0,366 accordingly.

Fig. 3. Statistics of Throughput to Area ratio

Another outstanding, deriving from the comparative table, fact is that SCREAM manages less than the half of the throughput of each other remaining ciphers except COLM. Remarks concerning area, specified in terms of LUTs, can be extracted; SCREAM is the cipher with the fewest LUTs. Namely, its total of LUTs is only 40, 43 and 49 percentages of the total of POET, COLM and Minalpher respectively.

VI. JUXTAPOSITION WITH AES, IDEA, DES AND RC5

There is an interesting perspective of putting CAESAR selected ciphers aside to other symmetric encryption algorithms. In [14], [15] and [16] four well known and widely used ciphers, are being analyzed and implemented on various FPGA devices; measurements are collected and analyzed for AES (both AKE and ARKL architetcures), IDEA, DES and RC5 (Carry Look Ahead design). All ciphers were functionally examined to ensure a flawless process for encryption and decryption.

	AES (AKE)	***AES (ARKL)***	***IDEA***	***DES***	***RC5***
T_{CLK} (ns)	45,45	35,09	20	11,77	23,81
f_{CLK} (MHz)	22	28,5	50	85	42
CLB Slices	2358	17314	1852	341	437
Throughput (Mbps)	259	3650	711	245	5376
Throughput to Area (T/A)	0,110	0,211	0,384	0,718	1,23

Table 2. Implementation Synthesis Results Comparisons

As depicted in Table 2, AES with feedback logic (AKE) can reach a throughput up to 259 Mbps whereas AES with pipelining technique (ARKL) can significantly achieve throughput up to 3,65 Gbps. As far as IDEA and DES are concerned, they can range up to 711 and 245 Mbps respectively. Based on the extremelly high throughput measurments and its relatively good frequency, RC5 manages to achieve the best performance among all ciphers mentioned.

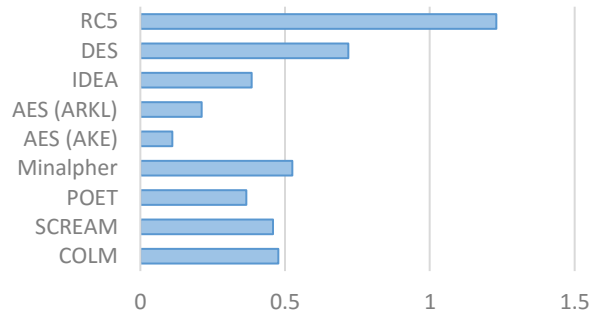

Fig. 4. Throughput to Area for other ciphers

VII. CONCLUSIONS AND OUTLOOK

In this work, we examined four authenticated ciphers of CAEASAR competition, that passed successfully to Round Two. Candidates were urged to submit their encryption algorithms updated with any applicable tweaks for Round Two, all described in a hardware description language (HDL). The implementation was developed using RTL design on a shared application interface, particularly designed for the purpose of the competition, known as CAESAR Hardware API for symmetric encryption algorithms. Once implementation was complete, in order to verify the gathered measurements, Xilinx tools were used. Comparison of ciphers and results were assembled for a Xilinx Virtex FPGA device.

Although using one of the highest rates of LUTs, Minalpher had the greatest Throughput -to-Area (T/A) rate in the Virtex FPGA; SCREAM, POET and COLM were following respectively. Finally, a comparison of CAESAR contestants and other ciphers was attempted in Section V.

ACKNOWLEDGMENT

This work is supported under the framework of EU COST IC 1204: TRUDEVICE (Trustworthy Manufacturing and Utilization of Secure Devices) Project. The authors would like to thank Miss Maria Katsaiti, for her valuable comments and support, regarding this publication.

REFERENCES

[1] "CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness", June, 2014.

[2] A. Moradi, "A hardware implementation of POET", University of Bochum, Germany, February, 2014.

[3] M. Liskov, R. Rivest, and D. Wagner, "Tweakable Block Ciphers," Journal of Cryptology, Vol. 24, No. 3, Jul. 2011, pp. 588-613.

[4] E. Homsirikamol, W. Diehl, F. Farahmand, A. Ferozpuri, M. U. Sharif, K. Gaj, "GMU Hardware API for Authenticated Ciphers", Cryptology ePrint Archive, Report 2015/669, Dec. 6, 2015.

[5] E. Homsirikamol, W. Diehl, A. Ferozpuri, F. Farahmand, P. Yalla, J. P. Kaps, K. Gaj, "CAESAR Hardware API", Cryptology ePrint Archive, Report 2016/626.

[6] E. Homsirikamol, W. Diehl, A. Ferozpuri, F. Farahmand, K. Gaj, "Implementer's Guide to the CAESAR Hardware API v1.1".

[7] E.Andreeva, A. Bogdanov, N. Datta, A. Luykx, B. Mennink, M. Nandi, E. Tischhauser, K. Yasuda, "COLM v1".

[8] S. Kerckhof, L. Gaspar, SCREAM Version 3, Université Catholique de Louvain, Dec. 13, 2015.

[9] A. Moradi, "A Hardware Implementation of POET 2", Ruhr-Universität Bochum, Germany.

[10] M. Kosug, M. Yasuda, A. Satoh, "FPGA implementation of authenticated encryption algorithm Minalpher", 2015 IEEE 4th Global Conference on Consumer Electronics (GCCE), pp. 572-576, 20-30 Oct. 2015, Osaka, Japan.

[11] V. Grosso, G. Leurent, F. Standaert, K. Varici, A. Journault, F. Durvaux, L. Gaspar, S. Kerckhof, "SCREAM, Side-Channel Resistant Authenticated Encryption with Masking", Version 3 (Second Round Specifications), Aug. 2015.

[12] F. Abed, S. Fluhrer, J. Foley, C. Forler, E. List, S. Lucks, D. McGrew, J. Wenzel, "The POET Family of On-Line Authenticated Encryption Schemes", Version 2.01, Sep. 15, 2015.

[13] Y. Sasaki, Y. Todo, K. Aoki, Y. Naito, T. Sugawara, Y. Murakami, M. Matsui, S. Hirose, "Minalpher V1.1".

[14] N. Sklavos, O. Koufopavlou, "Architectures and VLSI Implementations of the AES-Proposal Rijndael", IEEE Transactions on Computers, Vol. 51, Issue 12, pp. 1454-1459, 2002.

[15] N. Sklavos, P. Kitsos, K. Papadopoulos, O. Koufopavlou, "Design, Architecture and Performance Evaluation of the Wireless Transport Layer Security", Journal of Supercomputing, Springer-Verlag, Vol. 36, No 4, pp. 33-50, 2006.

[16] N. Sklavos, A.P. Fournaris, O. Koufopavlou, "WAP Security: Implementation Cost and Performance Evaluation of a Scalable Architecture for RC5 Parameterized Block Cipher", proceedings of IEEE Mediterranean Electrotechnical Conference (IEEE MELECON'04), Dubrovnik, Croatia, May 12-15, 2004.