HAL
open science

Protecting FPGA Bitstreams Using Authenticated Encryption

Karim Moussa Ali Abdellatif, Roselyne Chotin-Avot, Habib Mehrez

To cite this version:

Karim Moussa Ali Abdellatif, Roselyne Chotin-Avot, Habib Mehrez. Protecting FPGA Bitstreams Using Authenticated Encryption. 11th IEEE International Conference of New Circuits and Systems (NEWCAS), Jun 2013, Paris, France. pp.1-4, 10.1109/NEWCAS.2013.6573635 . hal-01017823v2

HAL Id: hal-01017823

https://hal.sorbonne-universite.fr/hal-01017823v2

Submitted on 8 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Protecting FPGA Bitstreams Using Authenticated Encryption

Karim M. Abdellatif, R. Chotin-Avot, and H. Mehrez
LIP6-SoC Laboratory, University of Paris VI, France
{karim.abdellatif, roselyne.chotin-avot, habib.mehrez}@lip6.fr

***Abstract*—This paper describes low cost solution for bitstream security by adding authentication and encryption to the reconfiguration process using Authenticated Encryption (AE). Compact ASIC architecture for AE is presented: Counter with Cipher Block Chaining-Message Authentication Code (CCM). Proposed architecture utilizes Advanced Encryption Standard (AES) in Counter mode (CTR) for encryption. For authentication, AES in Cipher Block Chaining (CBC) is used. Therefore, one architecture of AES for both encryption and authentication decreases the consumed area. In addition, using AES in 32-bit enhances the compact architecture. Our design was evaluated by using a 90 nm CMOS standard cell library. The proposed architecture of CCM requires 0.045 mm^2. In term of speed, CCM works with 407 Mbps. Our proposed architecture can be used efficiently for secure configuration of FPGAs.**

***Keywords*-Authenticated Encryption; Compact architecture; ASIC; FPGA bitstream.**

I. Introduction

FPGAs are now being used in consumer products where attacking is more common. In order to redefine their functionality, a bitstream configuration file is sent to the FPGA, this is known as a reconfiguration process. The bitstream is processed by the configuration logic- a part of the FPGA that is not programmable in order to establish routing to and from instantiated elements by setting the state of memory cells, pass gates, and routing switches. The user logic is the FPGA's reconfigurable part and where the user-defined application operates. Reconfiguration of FPGAs is becoming increasingly popular particularly in networking applications and it is vital to provide security against malicious parties interfering with equipment functionality through this mechanism. Also, remote reconfiguration is attractive in such systems to offer new multimedia features or to repair eventual security vulnerabilities.

However, remote update requires transmitting the hardware Intellectual Property (IP) over insecure communication channels and thus introduces new security issues. IP designers are mainly interested in the protection of the confidentiality of their IPs. Current FPGA vendors provide bitstream encryption for this need [1], [2]. Encryption protects the bitstream content independently of the device (from cloning, reverse engineering, etc.) while authentication ensures the correct and intended operation of the FPGA. Therefore, encryption and authentication must be computed.

Encryption algorithms are used to ensure confidentiality using a secret key between the sender and receiver. Message Authentication Codes (MACs) when used along with hash functions (also called as HMACs) can provide complete integrity.

Two algorithms, one for encryption like AES [3], and another one for authentication, like HMAC, consume more area in term of hardware implementations [4]. Techniques have been invented to combine encryption and authentication into a single algorithm which is called Authenticated Encryption (AE) algorithm.

AE is a block cipher mode of operation which simultaneously provides confidentiality, integrity and authenticity assurances on the data. The advantages of using one algorithm for both encryption and authentication are: smaller area, less power, and one key is used for encryption and authentication.

CCM mode (Counter with CBC-MAC) is a mode of operation for cryptographic block ciphers. It is an authenticated encryption algorithm designed to provide both authentication and confidentiality. In CCM, the confidentiality is achieved through encryption using AES [3] in CTR mode and authentication through Cipher Block Chaining (CBC) mode.

Our contribution: We present compact hardware implementation for CCM in order to be used efficiently for secure reconfiguration of FPGAs. One 32-bit AES architecture is used for both encryption and authentication for obtaining low cost solution to protect the FPGA bitstream.

An introduction for AES-CCM is presented Section II. Compact architecture for AES-CCM is presented in Section III. Implementation details and performance comparison are discussed in Section IV. Bitstream security using compact CCM is discussed in Section V. Section VI concludes this work.

II. CCM architecture

In order to discuss the architecture of CCM clearly, we have to first present Rijndael algorithm (AES).

AES was adopted as the Advanced Encryption Standard (AES) [3] in 2001. Fig. 1 shows the AES algorithm. The encryption process starts with the first key addition, followed by a number of round functions which depends on the key size and the total number of rounds, N_r, is 10, 12, or 14, when the key length is 128, 192, or 256 bits, respectively.

©2013 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. DOI: 10.1109/NEWCAS.2013.6573635

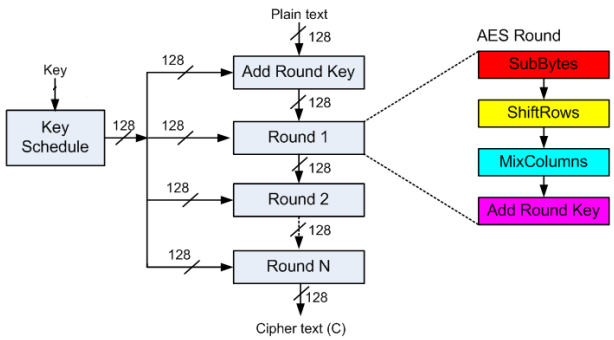

Figure 1. The Advanced Encryption Standard (AES)

The round function consists of four transformations applied to 16 bytes. ShiftRows stage cyclically shifts to the left the bytes in the last three rows of the state, using different offsets; SubBytes function is a non-linear byte substitution and operates independently on each byte of the state; MixColumns transformation operates on the State column by column, treating each column as a four-term polynomial. The columns are considered as polynomials over GF(2^8) and multiplied by a fixed polynomial a(x) modulo $x^4 + 1$ given by $a(x) = (03)x^3 + (01)x^2 + (01)x^2 + (02)$.

There are several techniques to implement AES on hardware. The method of implementation depends on the application requirements. Iterative design uses one round of AES but the speed of the design is limited as the output is ready after 11 clock cycles in case of using AES-128 [5]. 32-bit AES is used for applications which require small area where (1/4) round is used, the output is ready after 55 clock cycles [6]. For high speed applications, pipelining concept is used to reduce the critical path and enhances the speed of the overall deign [7]. Much effort must be directed to SubByte stage, it is the most area consuming part of AES.

In case of using FPGAs, there is an advantage to store the tables of SubByte stage in BRAM as most of modern FPGAs contain BRAM [8]. If the target is ASIC, the mathematical model of the inverse multiplication over GF(2^8) is used for area optimization goal but sub-pipelining must be used to decrease the critical path resulting from multiplication over GF(2^8)[7].

CCM [9] can be used in conjunction with any approved 128-bit block cipher like AES. It is designed for packet environment, where all the necessary data is available in storage before CCM processing. This implies that it is not online. CCM has been specified in the draft IEEE 802.11i standard for wireless networks. Fig. 2 shows the block diagram of CCM. Firstly, the message *M* is stored in a memory. Secondly, *Y* is generated using CBC mode, this value is used for authentication. Finally, CTR mode is used to generate ciphered text *C*. CCM is not suitable for on line applications as all data must be stored in memory before

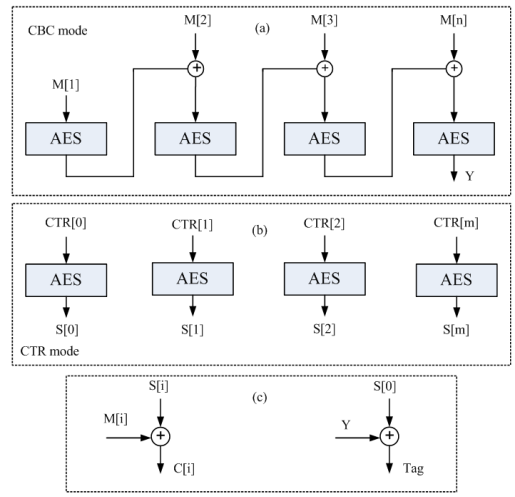

Figure 2. CCM mode

CCM processing. In [4], iterative AES (one round) was used to implement the architecture of CCM on FPGA using BRAM for SubByte stage. Also, two components of AES were implemented for both encryption and authentication in [10].

III. Compact Authenticated Encryption

For low cost implementations which usually have throughput lower than 1Gbps like wireless communications and embedded hardware applications, Compact authenticated encryption is very suitable and compatible. In the proposed CCM, we use 32-bit (1/4 round) AES that has an advantage of reducing the consumed area with a suitable throughput that is able to support applications lower than 1Gbps. Fig. 3 shows the architecture of 32-bit AES. The key schedule shares the SubByte stage with the data bus. As a result, only four s-boxes are used. Therefore, we can avoid the long data path resulting from using composite field approach by implementing a ROM to store the values of s-boxes. Moreover, only one MixColumn stage is used.

Our CCM architecture uses one 32-bit (1/4 round) AES for both encryption and authentication as shown in Fig. 4. All data must be stored in a memory. Firstly, authentication process is accomplished using CBC mode. Secondly, encryption process is performed using CTR mode. A 128-bit frame takes 55 clock cycles to be encrypted or added to the authentication tag. The achieved throughput of our presented CCM is calculated as follow:

$$Throughput(Mbps) = \frac{128 \times F_{max}}{55 \times 2} \quad (1)$$

Table I
HARDWARE COMPARISON

Design	architecture	Technology	Area mm^2	Frequency MHz	Throughput Mbps
This work	AES-CCM	90 nm	0.045	350	407.2
parelkar et al.[4]	AES-CCM	90 nm	0.057	148	434
Parelkar et al.[4]	AES+HMAC	90 nm	0.183	101.2	1293

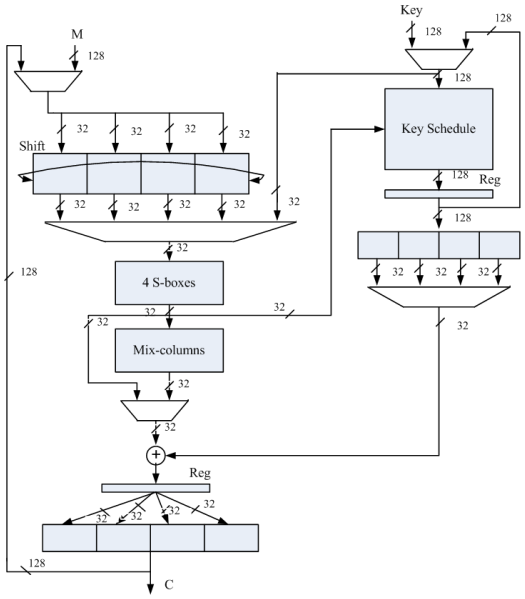

Figure 3. 32-bit AES

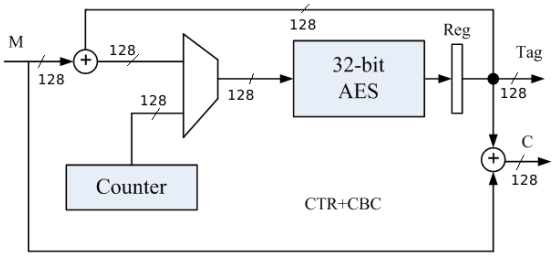

Figure 4. Proposed CCM

IV. PERFORMANCE COMPARISON IN ASICS

This section compares our presented architecture with the previous work. Presented architecture has been implemented using a 90 nm CMOS standard cell library and its performances are compared with the prior art in Table II.

To the best of our knowledge, there are no much work on how compact CCM is implemented in ASIC. Most of previous publications used FPGAs for implementation [11], [12], [13]. As a result, we selected only ASIC implementations for comparison (compact architectures).

The proposed CCM occupies 0.045 mm^2 with 350.8 MHz as a maximum frequency. However Parelkar et al. [4] presented 128-bit AES for both encryption and authentication, the performance of our CCM is better because we used 32-bit AES with ROM for SubByte stage as there are only four s-boxes are used. Therefore, operating frequency of our CCM is better than [4]. Also, [4] presented AES with HMAC but this method consumed more area compared to our GCM and CCM.

V. COMPACT ARCHITECTURES FOR BITSTREAM SECURITY

Bitstream encryption was introduced in Virtex-2 FPGAs to protect the intellectual property of the bitstream. The bitstream is encrypted with a user-selectable key. The same key is pre-loaded in the FPGA. When the FPGA is configured, it loads encrypted data and decrypts it internally. This behavior is not enough to prevent attackers from destroying the FPGA remotely using certain malicious bit-stream combinations. Therefore, the FPGA should accept only bitstreams from an authenticated source.

This section describes how our compact architecture can be used to provide encryption and authentication to the reconfiguration process. CCM is used in the static part of the FPGA as shown in fig. 5. The encrypted bitstream is decrypted using CCM. Also, CCM is used to compute the MAC and compare it with the bitstream's MAC. If they are equal, the FPGA will continue to the startup sequence. Otherwise, configuration will abort and the cells be cleared.

The adopted solution must meet the current configuration throughput and not adversely affect total configuration times. Table II shows the maximum throughput of the largest family members of recent FPGAs.

Unlike current FPGAs [1], [2] which support only encryption for bitstream security, our solution adds encryption with authentication in order to enhance the security of the configuration process.

Table II
CONFIGURATION THROUGHPUT OF SOME FPGA FAMILY MEMBERS

FPGA	device	Technology	Throughput
Virtex-5[14]	LX330T	65-nm	800 Mbits/s
Spartan-3 [15]	5000	90-nm	400 Mbits/s

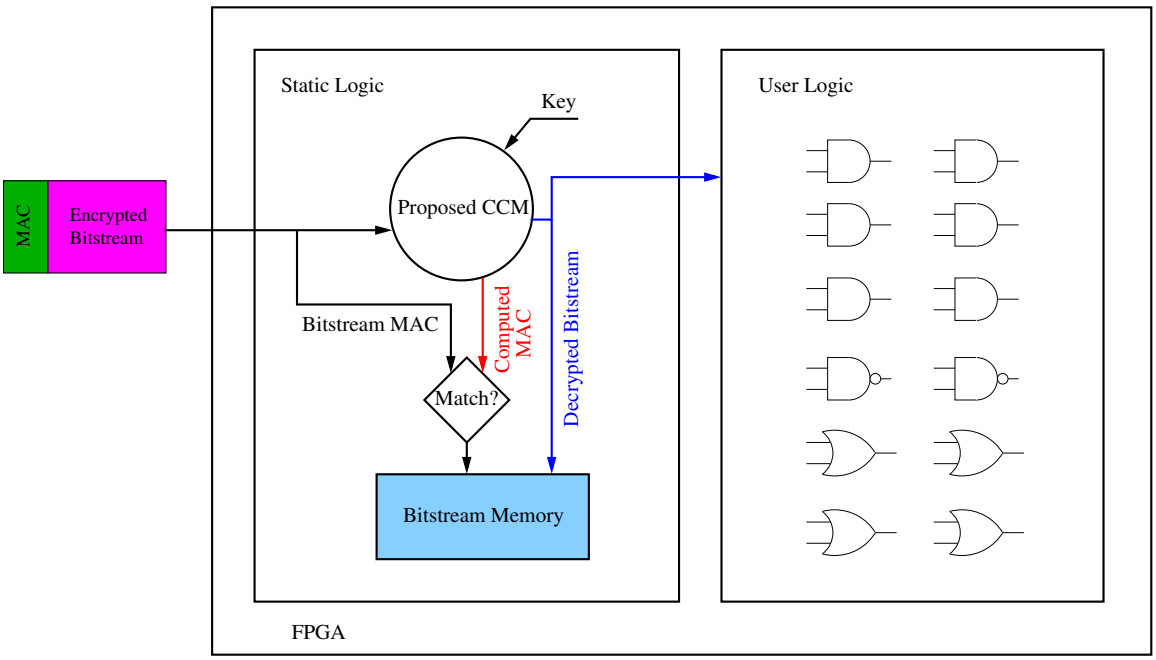

Figure 5. Bitstream security using compact AE (CCM)

VI. Conclusion and Future Work

This paper proposes low cost solution for bitstream security. This is achieved by compact architecture for authenticated encryption (AES-CCM). Presented architecture was evaluated through ASIC implementation. In order to minimize the hardware size of CCM mode, one 32-bit AES is used for both encryption and authentication. Future work includes Protocols which use the proposed architecture against attacks.

Acknowledgment

This work is a part of the project Robust FPGA ANR 11 INS-02, funded by The French National Research Agency, The Pole Systematic and The Pole Minalogic.

References

[1] Altera whitepaper, "Design Security in Stratix III Devices." [Online]. Available: http://www.altera.com/literature/wp/wp-01010.pdf

[2] Xilinx commercial brochure. Lock Your Designs with the Virtex-4 Security Solution. [Online]. Available: http://www.xilinx.com/publications/xcellonline/xcell$_$52/xc$_$pdf/xc$_$\v4security52.pdf.

[3] N. Pub, "197: Advanced Encryption Standard (AES)," *Federal Information Processing Standards Publication*, vol. 197, pp. 441–0311, 2001.

[4] M. Parelkar, "Authenticated Encryption in Hardware," Ph.D. dissertation, George Mason University, 2005.

[5] P. Bulens, F. Standaert, J. Quisquater, P. Pellegrin, and G. Rouvroy, "Implementation of the AES-128 on Virtex-5 FPGAs," *Progress in Cryptology–AFRICACRYPT*, pp. 16–26, 2008.

[6] A. Satoh, S. Morioka, K. Takano, and S. Munetoh, "A Compact Rijndael Hardware Architecture with S-box Optimization," *Advances in CryptologyASIACRYPT 2001*, pp. 239–254, 2001.

[7] X. Zhang and K. Parhi, "High-speed VLSI Architectures for the AES Algorithm," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 9, pp. 957–967, 2004.

[8] R. Chaves, G. Kuzmanov, S. Vassiliadis, and L. Sousa, "Reconfigurable Memory Based AES Co-processor," pp. 8–pp, 2006.

[9] M. J. Dworkin, "SP 800-38C. Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality," 2004.

[10] E. López-Trejo, F. Henrıquez, and A. Dıaz-Pérez, "An Efficient FPGA Implementation of CCM Mode Using AES," vol. 3935, pp. 208–215, 2005.

[11] A. Aziz and N. Ikram, "An FPGA-based AES-CCM crypto core for IEEE 802.11 i Architecture," *International Journal of Networks Security*, vol. 5, no. 2, pp. 224–232, 2007.

[12] S. Drimer, T. Güneysu, and C. Paar, "DSPs, BRAMs, and a pinch of logic: Extended recipes for AES on FPGAs," *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, vol. 3, no. 1, p. 3, 2010.

[13] P. Chodowiec and K. Gaj, "Very compact FPGA implementation of the AES algorithm," *Cryptographic Hardware and Embedded Systems-CHES*, pp. 319–333, 2003.

[14] Xilinx. Virtex-5 FPGA Data Sheet:DC and Switching Characteristics. [Online]. Available: http://www.xilinx.com/support/documentation/data$_$sheets/ds202.pdf

[15] Xilinx1. Spartan-3 FPGA family:Complete data sheet. [Online]. Available: http://www.xilinx.com/support/documentation/data$_$sheets/ds099.pdf