ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Implementation of Advanced Architecture Using DES over AES

G. Gowthami
M.Tech Student,
Nagole Institute of Technology & Science, Hyderabad, Telangana, India.

K.Srinivasa Reddy
Associate professor,
Department of ECE,
Nagole Institute of Technology & Science, Hyderabad, Telangana, India.

Evuri.Geetha Reddy
Assistant Professor,
Department of EEE,
Vignan's Institute of Technology & Aeronautical Engineering.

ABSTRACT:

A high throughput architecture is proposed for an efficient implementation of the Advanced Encryption Standard (AES) Algorithm. The presented architecture is adapted for AES encryptor-only as well as integrated AESencryptor/decryptor designs. The SubBytes/InvSubBytes operations are implemented using composite field arithmetic in order to exploit the sub-pipelining advantage within the loopunrolling methodology. The proposed architecture minimizes the critical path delay through the modification of the SubBytes/InvSubBytes as well as the KeyExpansion modules. Compared to previously reported AES encryptors andintegrated AES encryptors/decryptors designs, the proposed architecture provides an efficiency improvement of 61% and 29% respectively.

Keywords:

Advanced encryption standard, FPGA, subpipelining, composite field arithmetic, throughput.

I.INTRODUCTION:

The continuously growing number of internet and wireless communications users has marked security as a crucial designing factor for reliable communications. The Advanced Encryption Standard (AES) algorithm [1] was approved in October 2000 by the National Institute of Standards and Technologies to become the new encryptionstandard for its high flexibility and strong security. Several hardware implementations for the AES algorithm were previously presented targeting either ASIC as [2-4], or FPGA as .

The AES algorithm operates on 128-bit data blocks using a cipher key of possible lengths 128/192/256-bits throughout 10/12/14 iterative rounds respectively. Each round consists of a set of transformations namely: SubBytes, ShiftRows, MixColumns, AddRoundKey or their corresponding inverses during decryption. All operations are performed on the 128-bits arranged in a 4 × 4 matrix of bytes called State. In a parallel manner, the input cipher key is processed through a KeyExpansion module to produce 10/12/14 round keys used in each respective round The architecture of the AES round unit is the core which distinguishes each hardware design.

The implementation of the SubBytes/InvSubBytes and KeyExpansion modules is the design key for the AES round. The SubBytes/InvSubBytes modules are implemented either using storage memory units of size 256 bytes as [9-11] or using Composite Field Arithmetic (CFA) computations as. Similarly, the KeyExpansion module follows either the pre-computing and storage method as or the on-the-fly computation as In addition, the AES complete structure may adopt the loop-unrolling approach aiming for high throughput or the iterative looping approach aiming for area minimization.

II.RELATED WORKS:

The Proposed AES encryption round architecture is abstractly described by the block diagram shown in Fig. 1. Through the following subsections the implementation ofeach module is presented followed by the description of the sub-pipelining scheme utilized SubBytes Transformation using CFAThe SubBytes transformation processes the 128-bit state value on a byte level.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Each byte is substituted by its multiplicative inverse followed by an affine transformation The main complexity lies in computing the multiplicative inverse in GF(28). The CFA implementation guarantees a memory-free low-cost design, with optimum exploit of the subpipelining technique. In addition, the complexity of the multiplicative inverse computation is reduced from GF(28) to GF(24). Using CFA, any element A in GF(28) is mapped using a mapping matrix T into an element B in the composite field GF((24)2) with the form B = b1x + bo where b1,bo are in GF(24). The mapping matrix T is derived based on the defining irreducible polynomials of the fields GF(28) and GF((24)2) following the algorithm presented in .

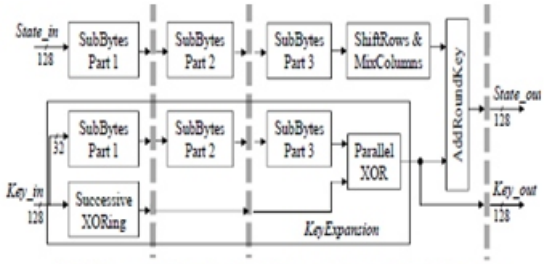

Fig. 1 Proposed AES sub-pipelined encryption round block diagram

III.PROPOSED SYSTEM:

The AES decryption process performs the inverse encryption transformations in the following reverse order: AddRoundKey, InMixColumns, InvShiftRows, InvSubBytes. Furthermore, the parallel KeyExpansion is executed in the reverse order starting from the final key value. In order to follow the same transformations order of the encryption procedure, the AES decryption is implemented using the equivalent Inverse cipher method . Accordingly, hardware sharing techniques between each transformation and its corresponding inverse can be easily applied. In the following subsections, the propos each two reciprocal operations is described. The transformation is the inverse of itself.

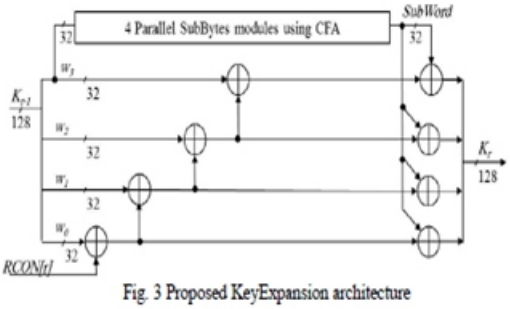

Fig. 3 Proposed KeyExpansion architecture

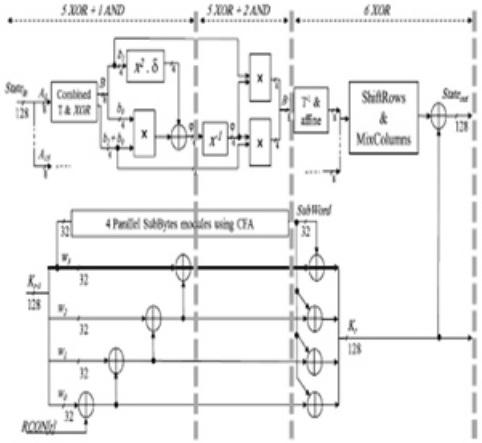

Fig. 4 Proposed AES encryption round architecture

Proposed Key Expansion:

The AES decryption proc values however in the revers is only initiated when the last key, is available. Consequen starts to generate the round ke reciprocal equations of (7) an round key K11-r-1 is generated each decryption round r from inverse cipher method im InvMixColumns on each ro AddRoundKey transformation proposed KeyExpansion archi is adapted, as shown in Fig. 5 as their reciprocals for encrypt The proposed parallelization SubWord computation impro XOR gates and 2 multiple conventional KeyExpansion .

Sub-pipelining Architectu:

The sub-pipelining technology integrated encryption/ decrypti three levels of pipelining reg stages division is similar to the encryption round. However, t stage 3 to stage 1 of next balancing between the stages d As illustrated in Fig. 5, stag which consists of 7 XOR gates blocks together.

Proposed SubBytes/InvSubBytes Implementation:

The SubBytes and InvSubBytes operartions the computation of the multiplicative inver However, they differ in the affine and inverse required at the end of the SubBytes and t InvSubBytes operations respectively. The archit is adapted for the integrated SubBytes/InvSubB Fig. 5. Two extra blocks are added: the pro block merged with the inverse affine transform in addition to the inverse mapping block at the two multiplexers are used to select between the decryption paths by setting the selection lin respectively.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

For the combined block with the affine transform, the bit level equations of the b1 + b0 are derived in a similar way .

Proposed ShiftRows/InvShiftRowsImplem:

The InvShiftRows transformation restores the their original places through a cyclic shift to same offset bytes of the ShiftRows operation. B location of each byte in the state matrix follow or an InShiftRows operation the following factsRow 1 is kept un-shifted in both cases, The offset shifting for row 3 is 2 bytes symmetric shift whether to the left (Shif right (InvShiftRows). Only rows 2 and 4 need multiplexin corresponding signal wiring for InvShiftRows based on the active mode decryption respectively.

Proposed MixColumns/InvMixColumns Implentation:

The InvMixColumns operation multiplies the state matrix with the fixed polynomial a- {0d}x2 + {09}x + {0e} modulo x4 + 1. The p can be decomposed into the MixColumns provided in Section II.C, in addition to the t m(x) = {08}(x3 + x2 + x + 1) and n(x) = {04} (a-1(x) = a(x) + m(x) + n(x). Consequently, the required in the InvMixColumns module are red multiplication by {02}, {08} or {04}. The are integrated modules, as shown in Fig. 5, MixColumns block parallel with the two block Fig. 4 Proposed AES encryption round arc sed integration of he AddRoundKey entation have in common rse using CFA. affine transforms the start of the tecture in Fig. 2 Bytes as shown in oposed combined at the beginning e end. Moreover, encryption or the es to '0' or '1' e merged inverse 4-bit output .

RESULTS AND COMPARISONS:

In this section, the perform is evaluated and compared designs. The presented results where the proposed round arc successive units referring to the Table I presents the results compared to the reported encr proposed architecture operate 453.45MHz providing a thro occupied slices on a Virtex-5 performance metric as through architecture provides an efficie to present a reliable compare implemented over similar devi LUT-based approach is adopt SubBytes, while the KeyExpan partial and dynamic reconfig round is divided into 4 sub exploit of the sub-pipelining te use of the storage LUTs w SubBytes occupying alone 1 BRAMs .

On the other design presented in while the KeyExpansion is on-the-fly structure. As architecture provides an efficiechitecture (x) respectively. A multiplexer is of the MixColumns block or the three blocks together. n for Integrated Desig acess utilizes the same round key order. Therefore, the decryption key value, K10 in case of 128-bit the KeyExpansion modul in a reverse manner using the The on-the-fly respective from the available K11-r duringm 1 to 10. The utilized equivalent implies the execution of the key before it enters then except for K10 and K0 [1]. The architecture described in Section II.C to implement (7) and (8) as well ption and decryption respectively. of the successive XORing with the ves the critical path delay by 3 exers delays compared to the techinque is applied on the AES ion round architecture by adding as shown in Fig. 5.

The one shown in Fig. 4 for the AES the AddRoundKey is moved from round in order to optimize the delays without extra registers cost. 3 represents the critical path s + 3 MUXs delays. mance of the proposed architectures with previously reported FPGAare for the loop-unrolled approach architecture is duplicated through 10 rounds of the AES algorithm. of the encryption architecture encryption designs in [9] and [5]. The maximum frequency of thoughput of 58Gbps with 5,337 FPGA.Defining the efficienc per unit slice, the proposed of 10.88Mbps/Slice. In order, the proposed architecture is for each reference. In [9], theted for the implementation of the module is implemented using guration. Furthermore, the AES pipelining stages. However, theechnique is not optimum due to the with unbreakable delays for the 10,240 slices, equivalent to 80 hand, the 3 sub-pipelining stages the SubBytes using CFA implemented using the conventional in Table I, the proposed improvement of 55%.

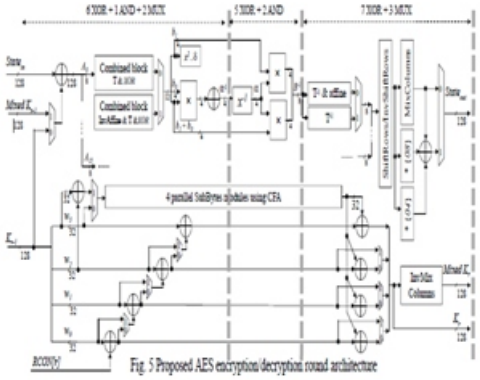

Fig. 5 Proposed AES encryption decryption round architecture

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

CONCLUSIONS:

A high throughput efficient architecture is proposed for AES encryptor round and adapted for integrated AES encryptor/decryptor. The SubBytes/InvSubBytes operations are implemented using composite field arithmetic in order to exploit the sub-pipelining advantage on the loop-unrollingmethodology. The proposed architecture allows the minimization of the critical path delay in both encryption and decryption processes. This is achieved through the proposed modifications in the SubBytes/InvSubBytes and KeyExpansion modules. The proposed encryption design provides a 58Gbps throughput with 5,337 slices on a Virtex- 5 FPGA. On the other hand, the proposed integrated encryption/decryption design provides a 37.7Gbps throughput with slice occupation of 6,741 on a Virtex-5 FPGA. Compared to previously reported FPGA designs, the proposed architectures for the encryption and the integrated encryption/decryption provide an efficiency improvement of 61% and 29% respectively.

REFERENCES:

[1] National Institute of Standards and Technology, "AdvancedEncryption Standard (AES)," 2001.

[2] S. K. Mathew, et al. "53 Gbps native GF(24)2 composite field AESencrypt/decrypt accelerator for content-protection in 45nm highperformancemicroprocessors," IEEE Journal of Solid-State Circuits,vol. 46, no. 4, pp. 767-776, April 2011.

[3] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "Efficient highperformance parallel hardware architectures for the AES-GCM," IEEETransactions on Computers, vol. 61, no. 8, pp. 1165-1178, August2012.

[4] S.-F. Hsiaso, M.-C. Chen and C.-S. Tu, "Memory-free low costdesigns of Advanced Encryption.

[5] X. Zhang and K. 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, September 2004.

Author's:

Ms. G. GOWTHAMI pursuing M.Tech in VLSI & ES from Nagole Institute of Technology and Science, Hyderabad. She completed B.Tech ECE from JNTUH affiliated engineering college ATRI, HYDERABAD.

Mr.K. Srinivasa Reddy is Associate Professor of the Electronics and Communication Engineering, Nagole Institute of Technology and Science, Hyderabad .He received his B.Tech degree in Electronics and Communication Engineering from JNT University, Hyderabad, and M.Tech degree in Embedded Systems from JNT University, Hyderabad.. He is a member of The International Association of Engineers (IAENG). He had ten publications in National and International Journals. He has written three text books in the field of wireless communications.

Mrs.Evuri.Geetha Reddy is Assistant Professor of the EEE, VITAE, Hyderabad .She received his B.Tech degree in EEE from JNT University, Hyderabad, and M.Tech degree PED from Vignan University,Guntur.. She is a member of The International Association of Engineers