International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 7, July 2014)

A Novel Design for a High Throughput S-Boxes Advanced With AES and SKPM Techniques

Abdul Mateen Ahmed[1], Md.Jayeed[3]
Md.Nizamuddin Salman[2] Hasan Abdullah Faiq[4]
HOD[1], Assistant Professor[2] Assistant Professor[3, 4]
ECE Department, ECE Department
Nizam institute of Engineering and technology, Hyderabad, India
Nizam_ece@yahoo.com[1], mhdzaid@gmail.com[2], salman.nizamuddiin@gmail.com[3],hasanfaiq@gmail.com[4]

***Abstract*- In this Paper we Design a Complex Field Architecture to get maximum Throughput approximately 3.49 Gbps for AES Synchronous Boxes(S-boxes) of the Glossial field, The best construction is obtained after a sequence of algorithmic and architectural process, the design results a minimal implementation area cost and critical paths. This design has a total of 36AND gates and 96 XOR gates with critical path of 4 AND gates and 20 XORs. This Optimization results in 20% reduction in S-boxes area and hence performance is enhanced.**

***Index Terms*—Complex Field Architecture, AES, S-boxes, Glossial Filed.**

I. INTRODUCTION

The Advanced Encryption Standard (AES) is an encryption standard chosen by the National Institute of Standards and Technology (NIST) in 2001, which has its origin in the Rijndael block cipher. Several studies in the area had identified the nonlinear Sub Bytes transformation as the major bottleneck in achieving both small area and high speed VLSI AES implementations. This brief presents a methodology that is based on a pure combinatorial circuitry. In which, the Galois inverse of elements in is computed prior using the composite field arithmetic (CFA) [1].

To date, there are several successful composite field constructions reported for AES S-box implementations. Summarizing from the previous works, the smallest composite field AES S-box is attributed to can right. However, the issue of critical path was not addressed in Can Right's work [3]. A short critical path is highly desirable in VLSI architectures, as it enables deep sub-pipelining for an increased performance in the clock frequency. On the other hand, the works of Zhang and Parham and contributed an AES S-box with the shortest critical path to date However, their work requires a larger area compared to Can Right's.

Here we explain our approach to minimize the area of the S-box and compare our new solution with the S-box of Satoh. The current work improves on the compact implementation of in the following ways. Many choices of representation (isomorphism's) were compared, and the most compact turns out to use a normal basis for each subfield (uses a polynomial basis for each subfield). And while used the "greedy algorithm" to reduce the number of gates in the bit matrices required in changing representations, here each bit matrix is fully optimized, resulting in the minimum number of gates. These various refinements result in an S-box circuit that is 20% smaller, a significant improvement [2].

The AES algorithm, also called the Rijndael algorithm, is a symmetric encryption algorithm; meaning encryption and decryption are performed by essentially the same steps. It is a block cipher, where the data is encrypted / decrypted in blocks of 128 bits. Each data block is modified by several "rounds" of processing, where each round involves four steps. Three different key sizes are allowed: 128 bits, 192 bits, or 256 bits, and the corresponding number of rounds for each is 10 rounds, 12 rounds, or 14 rounds, respectively. From the original key, a different "round key" is computed for each of these rounds. For simplicity, the discussion below will use a key length of 128 bits and hence 10 rounds. There are several different modes in which AES can be used [8].

For some of these, such as Cipher Block Chaining (CBC), the result of encrypting one block is used in encrypting the next. These are called feedback modes, and the feedback effectively precludes pipelining (simultaneous processing of several blocks in the "pipeline"). Other modes, such as the "Electronic Code Book" mode or "Counter" modes do not require feedback. These no feedback modes may be pipelined for greater throughput. But for hardware implementations of AES, there is one drawback of the table look-up approach to the S-box function: each copy of the table requires 256 bytes of storage, along with the circuitry to address the table and fetch the results. Each of the 16 bytes in a block can go through the S-box function independently, the byte substitution step. This then effectively requires 16 copies of the S-box table for one round. To fully pipeline the encryption would entail "unrolling" the loop of 10 rounds into 10 sequential copies of the round calculation. This would require 160 copies of the S-box table, a significant allocation of hardware resources [4].

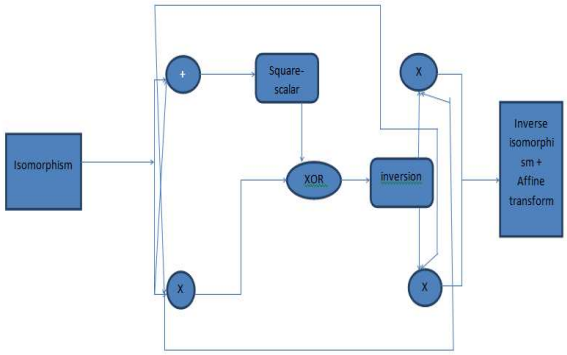
In contrast, this work describes a direct calculation of the S-box function using sub-field arithmetic, similar to while the calculation is complicated to describe, the advantage is that the circuitry required to implement this in hardware is relatively simple, in terms of the number of logic gates required. This type of S-box implementation is significantly smaller (less area) than the table it replaces, especially with

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 7, July 2014)

the optimizations in this work. Furthermore, when chip area is limited, this compact implementation may allow parallelism in each round and/or unrolling of the round loop, for a significant gain in speed. We have divided this paper in I-V sections, Section-I Introduction where the existing system and proposed one is explained, Section-II Proposed system & Algorithm, Section-III Isomorphic mapping and our proposed algorithm implementation, Section-IV Simulation and Results, Section-V Conclusion.

II. PROPOSED SYSTEM &ALGORITHM

A novel design for a high throughput s-boxes advanced with AES and SKPM techniques is nothing but Algebraic Normal Form-Complex Field Architecture advanced encryption S-boxes with fine grained pipelining, where Architecture optimization is achieved with 30 XOR gates in the total area and 3 XOR gates in critical path. Sub sharing is available in the subfield multiplier, 2-bit factor shared by two GF(2^2) multipliers saves one XOR addition while a 4-bit factor shared by two GF(2^4) multiplier saves 5 XORs. As such a common factor in multipliers save 18 XORs in total. We combine a GF(2^2) multiplier with scaler results in saving 3 XOR gates and one in critical path. Combining the higher and lower inputs and following square scaler saves 2 XORs [2].



Fig(1) ANF-CFA AES S-box fine-grained pipelining

II. IMPLEMENTATION OF THE PROPOSED ALGORITHM

The Rijndael algorithm, developed by Joan Daemen and Vincent Rijmen, was selected as the AES standard. The major focus of The Rijndael Algorithm includes designs seeking maximum throughput, minimum power consumption and minimization of circuitry. However, these approaches involve computations in Galois Field GF(2^8), which may involve high hardware complexity. The Rijndael Advanced Encryption Standard (AES) algorithm is a secret-key crypto-system recently approved as standard by NIST. AES is intended to replace the widely used DES and Triple-DES crypto-systems due to the last two's limited level of security [1].

AES is an evolution of DES and extends it with respect to three different sets of features: the mathematical structure—AES is more complex than DES, requiring a larger number and more powerful basic operations; the control-path—AES uses longer keys than DES does; and the data-path—AES operates on larger blocks of data than DES.

Implementations of DES, both in software and in hardware, have existed for quite some time. Several software and hardware implementations of AES have recently been proposed. The software solutions have targeted various platforms with the goal of reducing the number of clock cycles required to encrypt a data block [4].

Hardware solutions have been presented for field-programmable VLSI devices such as FPGA implementations. The objectives there were to increase the throughput while reducing the number of gates in the FPGA and to obtain reconfigurable devices able to cope with the different sizes allowed by AES for the keys and the data blocks. Custom devices, in contrast, are less flexible, but are more resistant against tampering or physical alteration than field-programmable ones. Some macro cell and coprocessor cryptographic architectures also known as crypto-processors of this kind have been proposed and evaluated at the simulation level.

A composite field is an object of study in field theory. Let L be a field, and let F, K be subfields of L. Then the (internal) composite of F and K is defined to be the intersection of all subfields of L containing both F and K. The composite is commonly denoted FK. The Sub Byte transformation is computed by taking the multiplicative inverse in GF(2^8) followed by an affine transformation. For its reverse, the InvSubByte transformation, the inverse affine transformation is applied first prior to computing the multiplicative inverse. The steps Involved for both transformations are shown below .

Sub Byte: 1. Multiplicative Inversion in GF(2^8)
2. Affine Transformation

InvSubByte: 1. Inverse Affine Transformation
2. Multiplicative Inversion in GF(2^8)

A more refined way of implementing the S-Box is to use combinational logic. This S-Box has the advantage of having small area occupancy, in addition to be capable of being pipelined for increased performance in clock frequency [5].

The multiplicative inverse computation will be done by decomposing the more complex GF(2^8) to lower order fields of GF(2^1), GF(2^2) and GF((2^2)^2).

GF (2^2) -> GF (2) : $x^2 + x + 1$
GF ((2^2)^2) ->GF (2^2) : $x^2 + x + \varphi$
GF (((2^2)^2)^2) ->GF ((2^2)^2): $x^2 + x + \lambda$

Where $\varphi = \{10\}_2$ and $\lambda = \{1100\}_2$

In the design GF(2^8) is decomposed into GF((2^4)^2) , and composite field arithmetic is applied to all the transformations in the AES algorithm. The optimum construction scheme for GF((2^4)^2) is selected based on minimizing the total gate count in the implementation of all transformations. However, it is more efficient to apply composite field arithmetic only in the computation of the

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 7, July 2014)

multiplicative inversion in the Sub Bytes and Inv Sub Bytes transformations. In this case, the construction scheme selected is no longer optimum. The schemes propose apply composite field arithmetic only to the multiplicative inversion. GF((2^8), is decomposed into GF((2^4)2), while GF((2^8) is decomposed into GF(((2^2)2)2 .

Nevertheless, each of them proposed only one possible way to construct the composite field. There exist other construction schemes with smaller gate counts and shorter critical paths. Different irreducible polynomials can be used to construct the composite fields of the same order. This brief presents 16 ways to construct GF(((2^2)2)2 Using composite field arithmetic, the complicated multiplicative inversion in GF((2^8) is mapped to operations in subfields. This brief provides the analytical results of how the coefficients in the irreducible polynomials affect the complexities of the subfield operations. In addition, for each construction scheme, there exist eight isomorphic mappings with various complexities to map the elements between GF ((2^8) and GF (((2^2)2)2[6].

An efficient algorithm is proposed in this brief to find all the isomorphic mappings. Moreover, the lowest mapping complexity is provided for each proposed composite field construction scheme. Based on the complexities of both the subfield operations and the isomorphic mappings, the optimum constructions of the composite field GF(((2^2)2)2 for the AES algorithm are proposed. Other composite field construction optimization approaches have been published recently. However, the approach is optimized based only on the complexity of isomorphic mappings. The approach optimizes for overall area requirement [8].

IV. SIMULATION AND RESULTS

ModelSim6.4c is a useful tool that allows you to stimulate the inputs of your modules and view both outputs and internal signals. The ModelSim6.4c advanced code coverage capabilities provide valuable metrics for systematic verification. Xilinx ISE 13.2 is a useful synthesis tool that allows us to generate the synthesis report following is the final synthesis report produced by Xilinx

Final Results:

RTL Top Level Output File Name : S_Mulit_inverse_case_3.ngr
Top Level Output File Name : S_Mulit_inverse_case_3
Output Format : NGC
Optimization Goal : Speed
Design Statistics
IOs : 16

Cell Usage:

BELS : 40
LUT2 : 4
LUT3 : 4
LUT4 : 32
IO Buffers : 16
IBUF : 8
OBUF : 8

Device utilization summary:

Selected Device: 3s100evq100-5

Number of Slices: 23 out of 960 2%
Number of 4 input LUTs: 40 out of 1920 2%
Number of IOs: 16
Number of bonded IOBs: 16 out of 662 4%

S_Mulit_inverse_case_3 Project Status			
Project File:	reports.xise	Parser Errors:	No Errors
Module Name:	S_Mulit_inverse_case_3	Implementation State:	Synthesized
Target Device:	xc3s100e-5vq100	• Errors:	No Errors
Product Version:	ISE 14.4	• Warnings:	4 Warnings (1 new)
Design Goal:	Balanced	• Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	

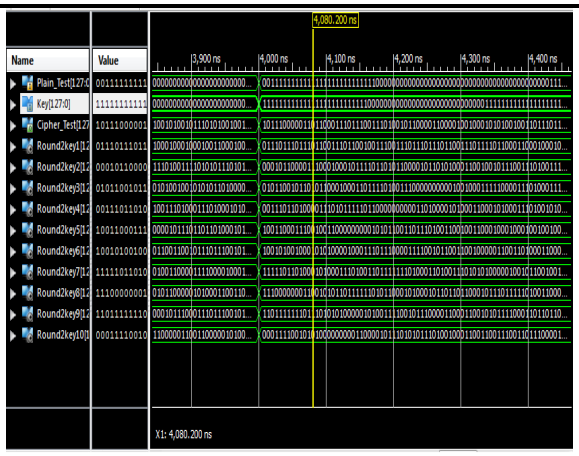
Topmodule_multiplicative_inverse project status

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	23	960	2%
Number of 4 input LUTs	40	1920	2%
Number of bonded IOBs	16	66	24%

Device utilization summary

Top module output screenshot:

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 7, July 2014)



AES encryption output screen shot:

V. CONCLUSION

The detailed study on composite field construction for the S-box function in AES was presented. The major contribution of our work was the derivation of a new composite field AES S-box that achieves an optimally balanced construction in terms of area of implementation and critical path, compared to the previous studies. Furthermore, we had explored all of the possible isomorphic mapping for each of the composite field construction and employed a new CSE algorithm to derive the most optimum isomorphic and inverse isomorphic mapping with affine transformation.

The best architecture obtained (i.e., Case III) possesses a total of 36 AND gates and 96 XOR gates with critical path of 4 AND gates and 20 XORs. Furthermore, we have found that there is a substantial gain in our CFAAES S-box in achieving a high throughput FPGA implementation.

REFERENCE

[1]. A. Rudra, P. K. Dubey, C. S. Jutla, V. Kumar, J. R. Rao, and P. Rohatgi, “Efficient Rijndael encryption implementation with composite field arithmetic,” in Proc. CHES, 2001, pp. 171–184.
[2]. N. Men tens, L. Bateman, B. Greenland, and I. Verbauwhede, “A systematic evaluation of compact hardware implementations for the Rijndael S-box,” in Proc. Topics Cryptology (CT-RSA), 2005, vol. 3376/ 2005, pp. 323–333.
[3]. D. Canright, “A very compact Rijndael S-box,” Naval Postgraduate School, Monterey, CA, Tech. Rep. NPS-MA-04-001, 2005.
[4]. X. Zhang and K. K. Parhi, “On the optimum constructions of composite field for the AES algorithm,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 10, pp. 1153–1157, Oct. 2006.
[5]. J. L. Fan and C. Paar, “On efficient inversion in tower fields of characteristic two,” in Proc. IEEE ISIT, 1997, p. 20.
[6]. D. R.Wilkins, “Part III: Introduction to Galois Theory,” 2000.
[7]. M. M. Wong and M. L. D. Wong, “A new common sub expression elimination algorithm with application in composite field AES S-box,” in Proc. 10th Int. Conf. Inf. Sci. Signal Process. Their Appl. (ISSPA), 2010, pp. 452–455.
[8]. M. Chen, “In Greedy Algorithms,” in A Greedy Algorithm with Look Forward Strategy. Vienna, Austria: IN-TECH, 1998, pp. 1–16.