HIGH PERFORMANCE AES DESIGN USING PIPELINING STRUCTURE OVER GF($(2^4)^2$)

Saleh Abdel-hafeez,
Jordan University of Science and Technology
sabdel@just.edu.jo

Ahmed Sawalmeh
General Organization for Technical Education and Vocational Training, Saudi Arabia
sawalmeh@gotevot.edu.sa

Sameer Bataineh
Jordan University of Science and technology, Jordan
samir@just.edu.jo

ABSTRACT

High data throughput AES hardware architecture is proposed by partitioning the ten rounds into sub-blocks of repeated AES modules. The blocks are separated by intermediate buffers providing a complete ten stages of AES pipeline structure. Furthermore, the AES is internally evenly divided to ten pipeline stages; with the addition feature that the shift rows block (*ShiftRow)* is structured to operate before the byte substitute (ByteSubstitute) block. The use of this swapping operation has no effect on the AES encryption algorithm; however, it streamlines the process of four blocks of data in parallel rather than 16 blocks which is considered the key advantage for area saving. We evaluate the performance of our new implementation and current implementations in terms of throughput rate and hardware area for ALTERA MAX3000A family FPGA EMP3128ATC100-5. The simulation results show that the proposed AES has higher throughput rate of about 16% than the general AES pipeline structure with a saving hardware area of 36%.

Index Terms— *AES pipeline, FPGA, GF($(2^4)^2$), Rijndael, S-BOX*

1. INTRODUCTION

The National Institute of Standards and Technology (NIST) have initiated a process to develop a Federal Information Processing Standard (FIPS) [1] for the Advanced Encryption Standard (AES) [2]. In October of 2000, NIST announced Rijndael as the winner of Advanced Encryption Standard (AES) contest in effort to address threatened key size of Data Encryption Standard (DES). Initially, most AES algorithms are implemented in software [3]; thereby, the secret key is vulnerable to attacks and relies on underlying programs. In addition, achievable speed by software implementation is not acceptable for internet applications, such as routers [4]. This leads to hardware design of AES where parallel processing and pipelining is possible [5]-[11]. Thus, hardware systems offer superior throughput performance, and are considered physically secured and preclude tempering by an outside attacker.

Consequently, early hardware implementations of AES began with various loop unrolled architectures [5], [7], where huge tables were generated with limited number of architecture optimizations. Thus, a pioneer attempts from V. Rijmen [9] proposed an AES s-box implementation based on composite field [2] which considered as the firs step in compaction the AES implementation [6],[10],[11].

Therefore, Galois fields (GF(2^8)) serves as a vehicle for several AES hardware implementation blocks. In general, most of the current hardware architectures were based on memory, shift register, or pipeline approaches [6]-[10], where these choices are limited to its applications. Besides, many of these attempts provide performance advantages tradeoff between area and speed; such as, iterative looping provides small silicon area and low data rate of throughput, while pipelining gives high data rate and large area overhead and might not be attractive for modes with feedback operations [6],[9],[10],[11].

In this brief, we proposed a new pipeline FPGA implementation of AES algorithm based on the composite field of GF($(2^4)^2$) that provides high data throughput and small hardware area in comparing to current existing pipeline designs. The proposed design reorders the ByteSub and ShiftRow operations based on [10]; such that, 4-byte instead of 16-byte are processed in parallel, yielding a small hardware area. Furthermore, the design adhere deep pipelining of 10 stages for one AES block,

1-4244-1236-6/07/$25.00 © 2007 IEEE

resulting in 110 pipeline stages for complete 10 rounds AES blocks, which is the key advantage for high throughput design.

Analysis and performance evaluations were adequately performed by having the general pipeline AES algorithm on the same FPGA device and compared thoroughly with our proposed design in terms of area and throughput. In addition, all simulations were conducted for one block and 16 blocks of input data, where each block consists of 16 bytes.

The paper is organized as follows; in section 2 we present the proposed designs structure and mathematical comparative analysis, FPGA hardware design is the subject of section 3. Then, section 4 presents experimental results and discussion. Finally, we present the conclusion in section 5.

2. DESIGN STRUCTURE AND MATHEMATICAL COMPARATIVE ANALYSIS

In this section, we evaluate two general designs for AES algorithm against our proposed design, where all of them are based on GF($((2)^4)^2$). The first design does not have the advantage of pipeline and strictly use iterative looping approach, in contrast, to second design which uses pipeline buffers between ten AES stages. On the other hand, our design uses the pipeline architecture of ten AES pipeline stages as well as it divides the AES into ten stages yielding an over all of 110 pipeline stages.
The resulting speed in terms of throughput rate and implementation area is evaluated and compared with existing design implementations in terms of the same gate technology factor and pipeline architectural measure in [12]. We began the design of the AES by analyzing the basic architecture as introduced in [9], which shows structure of the AES round for encryption. In addition, let's define the following notations:

Ngate:	*Number of cascaded gates*
Gate_delay:	*ALTERA MAX3000A 2-input AND gate delay, 1.636 nano second (nsec)*
Number_of_rounds:	*Number of AES rounds*
Number_of_blocks:	*Number of raw data coming blocks*
N_longest_pipe:	*Longest cascaded number of gates between two pipelines, 9 gates*
N_BS:	*Cascaded number of gates for ByteSubstitution, 33 gates*
N_SR:	*Cascaded number of gates for ShiftRowTransformations, 5 gates*
N_MC:	Cascaded number of gates for MixColumns Transformations, 3 gates
N_ARK:	Cascaded number of gates for AddRoundKey Transformations, 3 gates
K:	(N_SR + N_MC + N_ARK) x Gate_delay
M:	Number of pipeline stages
N:	Number of data blocks, where each block is 16 bytes of data

2.1. General AES structure without pipelining

In this architecture we implement only one round with Iterative looping structure; such that, the data-path consists of Key Addition, Mix Column, Byte-Substitution, and Row Shift as clearly demonstrated in [7].

2.1.1. Time analysis
The amount of time required to encrypt N blocks for 10 rounds can be evaluated using the following:

$$T = ((Number\_of\_blocks \times Ngate) + (N\_SR + N\_MC + N\_ARK))\ Gate\_delay \times Number\_of\_rounds, \qquad (1)$$

where $N\_SR + N\_MC + N\_ARK = K$. Thus, the required time for one block of input data equals to 16 byte of information is:

$$T = (1 x\ 33 + K)\ x\ 1.636\ x\ 10 = (540 + 10 x\ 11 x\ 1.636) = 719.9\ ns \qquad (2)$$

Consequently, the required time for 1600 blocks of input data which is 25600 byte of information:

$$T = 1600\ x\ 33\ x\ 10\ x\ 1.636 + (N\_SR + N\_MC + N\_ARK)\ x\ 1600\ \ x\ 10\ \ x\ 1.636 = 1,151,744\ ns \qquad (3)$$

2.2. General AES structure with 10 pipeline stages

In this structure, the implementation of a fully pipelined AES using deep pipeline structure is used. The AES Iterative Looping architecture must be converted into a suitable form for deep pipelining as in Figure 1. This is achieved by removing all the loops to form a loop-unrolled design, and by replicating the round function hardware and registering the intermediate data between rounds, where the data is moved through the round execution resources. On each clock cycle, a new item of data (1 block) is input and progress over 10 cycles through the pipeline resulting in the output of data at each cycle.

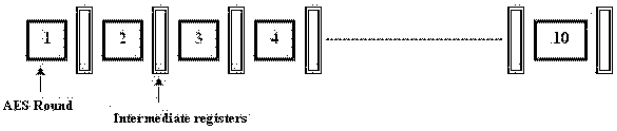


Figure 1. AES with 10 pipeline stages

2.2.1. Time analysis
The amount of time required to encrypt N blocks for 10 rounds (M pipeline stages) can be evaluated using the following:

$$T = [(N\text{-}1) + M]\ x\ (Ngate\ x\ Gate\_delay + K) \qquad (4)$$

For one block of input data which is equal to 16 bytes of information:

$$T = [(1-1) + 10] \times (33 + K) \times 1.636$$
$$= 719.9 \text{ ns} \quad (5)$$

For 1600 blocks of input data which is equal to 25600 byte of information:

$$T = [(1600 - 1) + 10] \times (33 + K) \times 1.636$$
$$= 115{,}822.3 \text{ ns} \quad (6)$$

2.3. Proposed AES pipeline structure

In this section, the architecture evaluation of the proposed design is achieved, where the AES is divided equally into ten pipeline stages with equal gates delay of about nine gates per stage. In addition, 10 AES blocks are used which are separated by pipeline stages resulting in total eleven stages per each round as depicted in Figure 2. In order to reduce the area and alleviate the processing of all 16-byte (one block) at a single AES, the ByteSub and ShiftRow operation are reordered as clearly shown in the figure 2.

2.3.1. Time analysis

The amount of time required to encrypt N blocks for 10 rounds can be evaluated using the following:

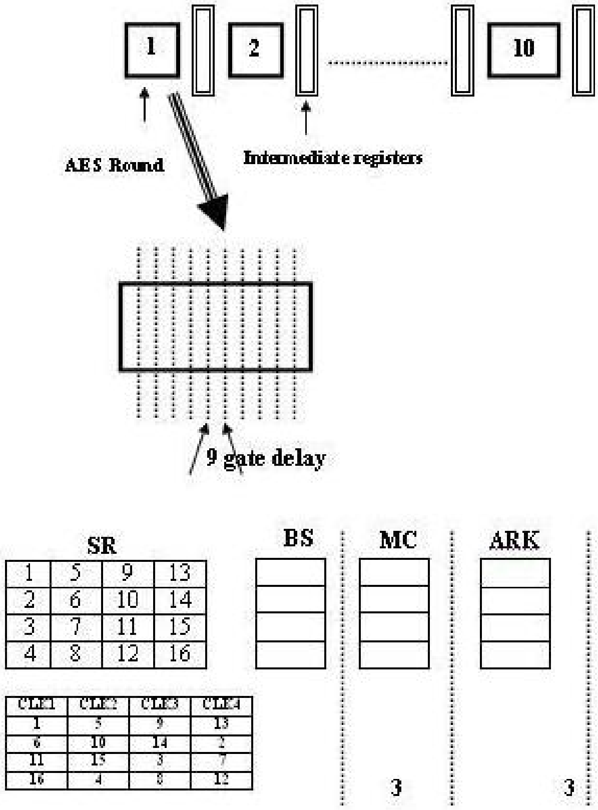
$$T = [((N \times 16/4) - 1) + M] \times [N\_logest\_pipe \times Gate\_delay] \quad (7)$$

For one block of input data which is equal to 16-byte of information:

$$T = [(16/4 - 1) + 110] \times [9 \times 1.636]$$
$$= 113 \times 15 = 1{,}695 \text{ ns} \quad (8)$$

For 1600 blocks

$$T = [((1600 \times 16/4) - 1) + 110] \times [9 \times 1.636]$$
$$= 6399 \times 15 = 95{,}985 \text{ ns} \quad (9)$$



1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

CLK1	CLK2	CLK3	CLK4
1	5	9	13
6	10	14	2
11	15	3	7
16	4	8	12

Figure 2. Proposed AES pipeline design

3. FPGA HARDWARE DESIGN

The S-box is usually implemented as a look-up table consisting of 256 entries; each entry is 8 bits wide. It is fast and inexpensive in terms of power consumption, as the study in [6]-[8] indicates. The most compact AES S-box implementations are the one based on inversion in the composite field over $GF((2^4)^2)$, which is considered as the basis of our design. Thus, we fully used combinational logic implementation of inversion in composite field as described in [11]. Consequently, the S-BOX components are implemented using HDL Verilog, which includes square unit, multiplication unit, inversion unit, XOR unit, Isomorphic δ, Isomorphic δ inverse, A ∗ Isomorphic δ, A ∗ Isomorphic δ inverse, and MUX unit.

In this architecture each round contains 10 stages of pipeline with 9-gate delay per stage; such that, each gate delay provides 1.636 *ns for ALTERA MAX3000A device technology*. The overhead area of this topology is the large number of logic cells due to register type which serves as an intermediate buffers between stages. However, the actual AES is 25% smaller than the general AES (i.e. case 1 & 2).

4. EXPERIMENTAL RESULTS AND DISCUSSION

FPGA simulation results demonstrate similarities between mathematical analysis and actual hardware representations. Many synthesized optimizations techniques were used and floor plan manual arrangement of slices was follow in order to utilize the device cells and minimize the routing. There was no global routing and components were locally interconnects. Small variations of timing were expected due to the device capacity and cells structure which were summarized in Table 1. The proposed design has throughput enhancement over AES with no pipeline and AES with 10 pipeline stages of 91.6% and 16.75, respectively. This enhancement is implemented at the expense of area cost; such that, the AES with no pipeline is 86% of saving logic cells. Furthermore, while the proposed design is supposed to save about 75% of logic cells in comparing to AES with 10 pipelines, it is only saving 36%. This due to large number of pipeline stages (110) which occupies substantial number of registers; even though, each stage is processing only four blocks in contrast to 16 blocks.

Table 1. FPGA comparison results of AES encryptions based on $GF(((2)^4)^2)$ and 1600 blocks of incoming data

Case	Design	Time (ns)	Logic Cells
1	No pipelining AES	1,181,344	23,517
2	10 pipelining AES	117,847.3	273,570
3	Proposed design	98,107	173,992

5. CONCLUSION

New hardware architecture is proposed for AES algorithm over $GF((2^4)^2)$ *and compared against two AES hardware structures which are iterative looping and ten rounds pipeline approach. The physical implementations of the three structures were conducted through FPGA ALTER MAX3000A device. The comparison analysis show that our proposed design has the advantage of saving logic cells of about 36% over AES with ten pipeline structure, and provides a higher throughput of about 16%. Furthermore, it gives a superior throughput advantage over iterative looping AES structure of about 91.6% rate; however, the area is 7.4 times larger than the iterative looping structure.*

The main key advantage for saving area is the rewording of ByteSub and ShiftRow which streamlines the process of evaluating four bytes of data in parallel rather than having 16 bytes in parallel with all required hardware. Another key feature is having high throughput by partitioning the AES into ten sub-blocks with global intermediate buffers between them for ten rounds; thus, creating a deep pipelining structure for 110 stages.

As a result, for applications that starve for increase instruction throughput and small area overhead, our proposed design is the solution to be adapted. One of the draw back feature of the design is larger power consumptions due to fully pipeline structure, and large time consuming for synthesization and floor plan for hardware design.

6. REFERENCES

[1] FIPS PUB 197, Advanced Encryption Standard (AES). National Institute of Standards and Technology, U.S. Department of Commerce, November 2001; http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[2] Kris Gaj, and Pawel Chodowiec, Hardware performance of the AES finalist's survey and analysis of results. George Mason University. Proc. 3rd Advanced Encryption Standard (AES) Candidate Conference, New York, 2000:1-5.

[3] A. Elbirt, Recon_gurable Computing for Symmetric-Key Al-gorithms, Ph.D. thesis, Department of Electrical Engineering, Worcester Polytechnic Institute, 2002.

[4] Elena Trichina, and Tymur Kokishko, Secure AES Hardware Module for Resource Constrained Devices. Security in Ad-hoc and Sensor Networks. First European Workshop. ESAS 2004, Heidelberg ,Germany, August 6, 2004; (3313):215-229.

[5] Sumio Morioka, and Akashi Satoh, An Optimized S-BOX Circuit Architecture for low power AES Design. IBM Research, Tokyo Research laboratories, CHES 2002; (2523):172-186.

[6] 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.

[7] Akashi Satoh, and Sumio Morioka, Unified Hardware Architecture for 128-Bits Block Ciphers AES and Camellia. IBM Research, Tokyo Research laboratories, CHES 2003; (2779):304-318.

[8] Wolfram Drescher, Kay Bachmann 1, and Gerhard Fettweis, VLSI Architecture for Non-Sequential Inversion Over GF(2^m) Using the Euclidian Algorithm. Mobile Communications Systems, Dresden University of Technology, D - 01062 Dresden. sponsored in part by the Deutsche Forschungsgemeinschaft within the Sonder for schung sbereich SFB 358.

[9] Joan Daemen and Vincent Rijmen, The Design of Rijndael, AES - The Advanced Encryption Standard, Springer-Verlag 2002; 238.

[10] Pawel Chodowiec, and Kris Gaj, Very Compact FPGA Implementation of the AES Algorithm. George Mason University, CHES 2003; (2779):319-333.

[11] Tim Good, and Mohammad Benaissa, AES on FPGA From the Fastest to the Smallest. University of Sheffield, Department of Electrical and computer Engineering, CHES 2005; (3659):427-440.

[12] Amos R. Omondi, The Microarchitecture of Pipelined and Superscalar Computers, Kluwer Academic Publishers, 1999.