DOI: 10.5370/JEET.2011.6.3.418

Compact Design of the Advanced Encryption Standard Algorithm for IEEE 802.15.4 Devices

Ohyoung Song† and Jiho Kim*

Abstract – For low-power sensor networks, a compact design of advanced encryption standard (AES) algorithm is needed. A very small AES core for ZigBee devices that accelerates computation in AES algorithms is proposed in this paper. The proposed AES core requires only one S-Box, which plays a major role in the optimization. It consumes less power than other block-wide and folded architectures because it uses fewer logic gates. The results show that the proposed design significantly decreases power dissipation; however, the resulting increased clock cycles for 128-bit block data processing are reasonable for IEEE 802.15.4 standard throughputs.

Keywords: Advanced Encryption Standard (AES), Secure sensor networks, Zigbee security

1. Introduction

Cryptographic operation in wireless devices using little memory and a low-power processor causes system overhead; therefore, implementing security hardware dedicated to cryptographic operation is necessary [1]. Many implementation methods have been proposed for advanced encryption standard (AES) design using field programmable gate array (FPGA) or application-specific integrated circuit. A typical AES structure, known as “block-wide,” reveals a great deal of parallelism in a 128-bit block [2]. Because it focuses on high performance, this manner of implementation uses numerous logic gates and consumes a large amount of power. For better resource utilization, previous studies include folded AES designs [3-5] in which the 128-bit data block is divided into four 32-bit data blocks or sixteen 8-bit data blocks, and in which each block is processed independently. The folded AES requires more clock cycles for processing a 128-bit block because of the reuse of the hardware resources. A more compact AES design is needed for wireless sensor networks (WSNs) that operate using resource-constrained WSN devices.

In this paper, the main focus is creating the smallest possible design for the AES core that consumes less power due to the use of fewer logic gates and satisfies the throughput requirements of the IEEE 802.15.4 standard [6]. In AES design, the substitution table (S-Box) plays a major role in optimization. A very compact AES core using only one S-Box is therefore proposed.

In the results, the total logic element usage of the proposed design is significantly reduced by 18.76% for the block-wide design and 59.71% for the folded design. Therefore, the total power consumption of our design is also significantly reduced by 16.9% for the block-wide designs and 59.5% for the folded design.

2. FPGA Design Platform

The proposed design is targeted to the Altera Stratix FPGA device family [7]. The design tool used in this work is Altera Quartus. To determine the hardware complexity of the various designs, the final number of Altera Stratix resources, such as logic elements and memory bits, is considered. Altera Quartus was used for all stages of the computer-aided design flow. Analysis and synthesis were configured to perform an optimization of speed for critical portions of the design and the area for the remainder of the design. When memories were required in some modules, they were coded by explicit instantiation in the register transfer level using an appropriate configuration of Altera’s design library function, called *altsyncram.*

The power consumed by the FPGA device can be divided into two components: (1) static power consumption caused by the leakage current and static current due to the stable input voltage and (2) dynamic power consumption caused by the charge and discharge of the total output capacitance and the short-circuit current during the switching transient. The Quartus reads the signal activity file and calculates the static and dynamic power consumption. The static power consumption is scaled by the fraction of the core FPGA area used by the circuit. The static power consumption is proportional to the logic usage. The dynamic power measurements by Quartus can be directly compared to check the more efficient design methodology in power consumption.

† Corresponding author: School of Electrical and Electronic Engineering, Chung-Ang University, Seoul, Republic of Korea (song@cau.ac.kr)

* School of Electrical and Electronic Engineering, Chung-Ang University, Seoul, Republic of Korea (jihokim@wm.cau.ac.kr)

Received: August 16, 2010; Accepted: December 30, 2010

3. Design Criteria: Power vs. Area

Two implementation results for previous AES designs on an FPGA are shown: the block-wide design and the modified folded design. The features of the block-wide AES design targeted to the Altera Stratix device are summarized in Table 1. The block-wide structure consists of several logic gates, and consumes more power than desirable for a mobile wireless application operating under resource constraints. The block-wide AES encryption unit is designed in VHDL and synthesized in the Altera Quartus. The area comprises 84.4% of all the hardware [8], and the power consumption in S-Boxes is at least 75% of the total [9]. In the simulation results, the area and the power consumption of S-Boxes are 90% and 85.8% of the total, respectively.

The dynamic power consumption in the block-wide AES according to the operating clock frequency and data processing rate is shown in Fig. 1. The dynamic power consumption in the AES S-Box used for the SubByte transformation is proportional to the clock frequency. However, in the other logic, the dynamic power consumption does not depend on the clock frequency, only on the data processing rate, as shown in Fig. 1. The other logic that excludes the S-Box does not increase the dynamic power even if the operating clock frequency increases. When the AES module processes incoming data at 2 Mbps, the dynamic power consumption increases nearly 0.22 mW whenever the clock frequency increases by 1 MHz.

In the folded architecture [10], four AES S-Boxes are used in the SubBytes transformation and another S-Box is used in the key scheduling process. The architecture is improved by only using four total S-Boxes in both the SubBytes transformation and the key scheduling process, which is called the modified folded AES architecture. By doing the four AES S-Boxes required in the key scheduling process of the block-wide can be reduced. However, one round of operation takes five clock cycles instead of four clock cycles in the folded AES [10]. Table 1 summarizes the features of the modified folded AES targeted to Altera Stratix devices. Memory bit usage in the modified folded design is shortened by 80% compared with that in the block-wide design.

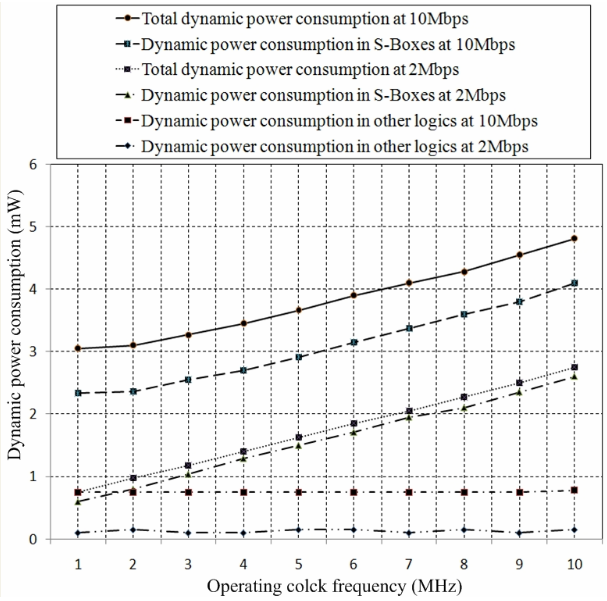

Fig. 1. Dynamic power consumption in block-wide AES

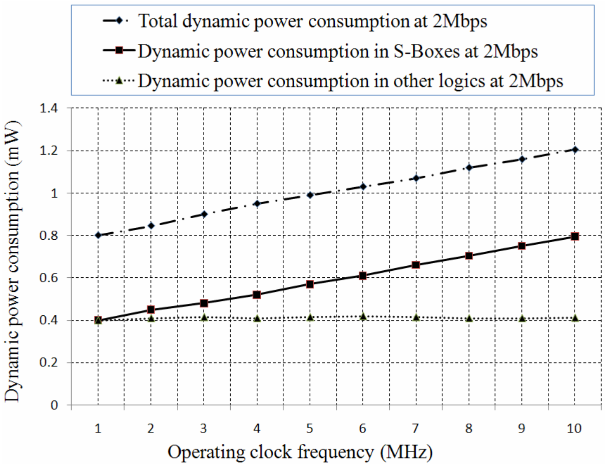

Fig. 2. Dynamic power consumption in the modified folded AES

Fig. 2 shows the variance in the dynamic power consumption in the modified folded design for a 2 Mbps data processing rate accordingly with the increase of clock frequency. Because of the block-wide design, the dynamic power consumption in AES S-Boxes increases with the clock frequency. The total dynamic power consumption in the modified folded design is nearly the same as in the block-wide design with the same data processing rate. However, the dynamic power consumption that excludes AES S-Boxes in the folded design is larger than that in the block-wide design because the modified folded design uses more sequential logic, such as registers and control logics.

Table 1 shows the total power consumption of the block-wide and folded designs, respectively, for 1 Mbps data processing at optimal clock frequencies. The dynamic power consumption in the block-wide design is lower than that in the folded design because the clock frequency in the block-wide is five times lower than that in the folded. However, the total power consumption in the folded design is lower than that in the block-wide design because the static power consumption gap between the folded and the block-wide is larger than the dynamic power consumption gap. At the same data throughput, the static power consumption in the folded design is significantly reduced (i.e., by 68.6%) compared with that in the block-wide design because the static power consumption is proportional to the logic usage. Although the dynamic power consumption in the folded design is 0.11 mW more than that in the block-wide design, the total power consumption is lowered by 65.4% of the block-wide.

Because the modified folded design uses four S-Boxes, the number of S-Boxes was reduced from 20 in the block-wide by 16, and the static power reduction was 6.3 mW, as shown in Table 1. One S-Box consumes approximately 0.4 mW static power. The dynamic power consumption increased by 0.1 mW. Therefore, if the four S-Boxes of the modified folded design are reduced by three S-Boxes, using only one S-Box should decrease the static power consumption by 1.2 mW and increase the dynamic power consumption by approximately 0.1 mW. The AES module using only one S-Box should use an operating clock frequency four times higher than that in the folded design, while the folded design should have a frequency five times higher than the block-wide design for the same data throughput. Therefore, the total power consumption of the proposed AES design that uses only one S-Box is expected to be considerably lower than the block-wide and folded designs.

The proposed AES design should operate at 20 (or four) times faster clock frequency than the block-wide (or the folded) design to support the same data throughput. The data throughput in the proposed AES is 1/20 of the block-wide and 1/4 of the folded; however, this uses fewer gates and consumes less static power than others. The use of faster clock frequency in the proposed AES design does not require changes to the design library or basic function blocks such as S-Box, ShiftRows, MixColumns, and AddRoundKey in the block-wide and the folded. Therefore, faster clock frequency in the proposed AES design does not result in an increase in cost.

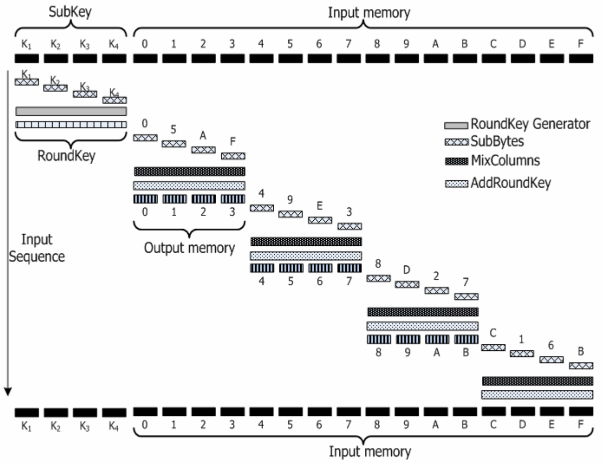

Fig. 3. The data flow in the proposed AES architecture

4. Small Design of the AES Algorithm

4.1 Behavioral Operation

For the AES algorithm [11], the length of the input block, the output block, and the cipher keys is 128 bits. The state array is the internal matrix upon which the data are manipulated and which consists of four rows of four bytes each. The AES algorithm has four basic transformations: SubBytes, ShiftRows, MixColumns, and AddRoundKey. The operations are performed in 10 rounds. The SubBytes transformation is a nonlinear byte substitution that operates independently on each byte of the state using an S-Box. The ShiftRows transformation is a cyclic shift operation with constant offsets applied to the rows of the matrix. The MixColumns transformation operates on the State column by column, treating each column as a four-term polynomial. The AddRoundKey transformation performs an XOR operation on the round key obtained from the initial key via a key expansion procedure. The AES algorithm takes the seed key and performs a key expansion routine to generate the round keys in the AddRoundKey transformation.

In the proposed design, the entire round operation consists of five phases. Each phase executes the four basic transformations in sequence, as shown in Fig. 3. The first phase is executed to obtain the round key. Each of the remaining four phases performs the four basic transformations for the input block. Therefore, one round operation uses 20 steps, which requires 20 clock cycles with each step using a single clock cycle. The following procedure, shown in Fig. 3, can then be executed in phases.

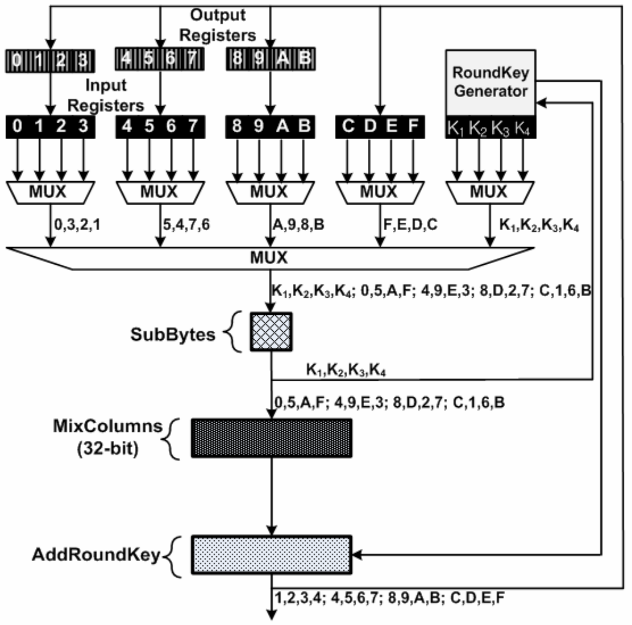

Fig. 4. Architecture of the AES using one S-Box

- First phase (Steps 1-4): Read K1 from the Subkey, execute SubBytes, and write the result to the corresponding locations in the RoundKey Generator. Repeat for K2, K3, and K4. Then, generate the round key with the result found in the RoundKey Generator.
- Second phase (Steps 5-8): Read input byte 0 from the input memory. Execute SubBytes and write the result to the corresponding location in MixColumns. Repeat for 5, A, and F, then run MixColumns for the written 4

bytes, and write the result to the output memory addresses 0, 1, 2, and 3, respectively.

- Third phase (Steps 9-12): Repeat the second phase for input bytes 4, 9, E, and 3 in the same way. Write the result to the output memory addresses 4, 5, 6, and 7, respectively.
- Fourth phase (Steps 13-16): Repeat the second phase for input bytes 8, D, 2, and 7 in the same way. Write the result to the output memory addresses 8, 9, A, and B, respectively.
- Fifth phase (Steps 17-20): Repeat the second phase for input bytes C, 1, 6, and B in the same way. Write the result to the input memory addresses C, D, E, and F, respectively. The input memory C, D, E, and F are reused as the corresponding outputs for the next round.

4.2 AES Architecture

A compact AES design using only one S-Box is presented. Figs. 4 illustrates the architecture of the proposed AES design. This design consists of twelve 8-bit output registers for output memory, sixteen 8-bit input registers for input memory, five 4-to-1 multiplexers, one 5-to-1 multiplexer, a RoundKey Generator, one S-Box for SubBytes, one 32-bit MixColumns, and an AddRoundKey unit. During one AES round operation, 20 SubBytes transformations are required: four for key scheduling and 16 for the AES encryption. One round operation that takes 20 clock cycles by reusing one AES S-Box is described as follows:

- Key scheduling (four clock cycles): Read the K1 register in RoundKey Generator using two multiplexers that select an appropriate input, execute SubBytes, and write the result to the K1 register in RoundKey Generator. The operation thus far takes one clock cycle. Repeat for K2 and K3 during two clock cycles. Read the K4 register, execute SubBytes, and then generate the round key with the result and the 3-byte registers for K1, K2, and K3 in RoundKey Generator during the fourth clock cycle. This precomputed RoundKey can be used for AddRoundKey operation.
- AES encryption (16 clock cycles): Read an input register addressed in 0, execute SubBytes, and write the result to the corresponding register in MixColumns during one clock cycle. Repeat for 5 and A during two clock cycles. In the fourth clock cycle, read the F input register, execute SubBytes, and then run MixColumns for the result and the 3-byte registers written during the previous clock cycles, run AddRoundKey, and write the 4-byte result to the output memory, addressed in 0, 1, 2, and 3, respectively. These operations take four clock cycles. Repeat the above operations for input registers 4, 9, E, and 3 in the same way. Write the result to the output registers addressed in 4, 5, 6, and 7, respectively. This takes four clock cycles. Next, repeat for input registers 8, D, 2, and 7 in the same way, and write the result to the output registers, addressed in 8, 9, A, and B, respectively. This also takes four clock cycles. Finally, repeat for input registers C, 1, 6, and 8 in the same way and write the result to the input registers, addressed in C, D, E, and F, respectively. The output registers are stored in input registers in corresponding locations simultaneously. This takes four clock cycles for the repetition of input registers C, 1, 6, and 8. In all, one round takes sixteen clock cycles.

The data path of the AES round for encryption is shown in Fig. 4. The ShiftRows is the only operation that mixes throughout the entire 16-byte block. In the implementation, the ShiftRows are performed using several multiplexers that reorder the input bytes. The SubBytes block substitutes for the State Array using the S-Box. The SubBytes block is instrumented by using the block of RAM (BRAM) embedded in the target FPGA device. One 8×256-bit BRAM is configured to implement the AES S-Box as a look-up table (LUT) to compute the 128-bit data substitutions. The 32-bit MixColumns can be implemented simply by using XOR gates. The MixColumns block consists of four 8-bit resisters and XOR gates that operate the 32-bit MixColumns operation. The RoundKey Generator block generates round keys that are used in the AddRoundKey transformation. The RoundKey Generator block uses the SubBytes block that is used in the encryption and consists of four 8-bit resisters and some latches. An AES Control block synchronizes the whole process and controls the information flow. This generates the signals to control the multiplexers and latches that are used in the AES components. The logic usage and the power consumption can be significantly reduced by eliminating multiple S-Boxes that occupy a large area at the same throughput (1 Mbps) using an optimized operating clock frequency for each design.

Table 1. Synthesis results of various AES designs at the optimized operating clock to satisfy 1 Mbps throughput

Measure	Block-wide	Modified folded	proposed design
Device	Stratix EP1S10F484C5		
Logic elements (A)	691	692	702
No. of S-Box, Memory bits (converted logic elements: B)	20 40960 (4160)	4 8192 (832)	1 2048 (208)
Total logic elements (A+B)	4851	1524	910
Clock cycles for a 128-bit block data processing	11	55	220
Operating clock frequency (MHz)	0.1	0.5	2
throughput (Mbps)	1	1	1
Dynamic power dissipation (mW)	0.28	0.39	0.47
Static power dissipation (mW)	9.19	2.89	1.72
Total power dissipation (mW)	9.47	3.28	2.19

4.3 AES Results

Table 1 summarizes the features of the proposed AES design. The total logic element usage of the AES design is 18.76% of the block-wide and 59.71% of the folded. The optimized operating clock frequencies are selected differently under the same throughput (1 Mbps) reasonable for the maximum data rate (250 kbps) of IEEE 802.15.4: 0.1 MHz for block-wide, 0.5 MHz for folded, and 2 MHz for the proposed design. The static power consumption in the proposed AES core for IEEE 802.15.4 is 16.9% of the block-wide and 59.5% of the folded design. The dynamic power consumption of the proposed AES core is larger than the others because it operates at a clock frequency 20 times (or 4 times) greater than the block-wide (or the folded). However, the difference in the dynamic power consumption among the three designs is less than 0.11 mW, which is trivial. The total power consumption of the proposed design is the smallest of the three. The total power consumption in the proposed design is reduced to approximately 21% of the block-wide and 62% of the folded.

5. Conclusion

An optimal AES algorithm for low-power WSN nodes is designed. In the proposed design, only one S-Box was used, which reduced the logic usage and the power consumption compared with the block-wide and folded designs at the same throughput (1 Mbps), satisfying the data rate requirements of the IEEE 802.15.4 standard.

Acknowledgements

This research was supported by a grant (CR070019) from Seoul R&BD Program funded by the Seoul Development Institute of the Korean Government.

References

[1] Y. Xiao, et al., "MAC Security and Security Overhead Analysis in the IEEE 802.15.4 Wireless Sensor Networks," *EURASIP Journal on Wireless Communications and Networking*, 2006, p. 1-12

[2] A. Dandalis, et al.: 'A Comparative Study of Performance of AES Final Candidates Using FPGAs', Lecture Notes in Computer Science, 2000, 1965, p. 133-153

[3] P. Chodowiec, et al., "Very compact FPGA implementation of the AES algorithm," *Lecture Notes in Computer Science*, 2003, 2779, p. 319-333

[4] M. Feldhofer, et al., "AES implementation on a grain of sand," *IEE Proc. Inf. Secur.*, 2005, 152, (1), p. 13-20

[5] P. Hämäläinen, et al., "Design and implementation of low-area and low-power AES encryption hardware core," *9th Euromicro Conf. on Digital System Design*, 2006, p. 577-583

[6] Wireless Medium Access Control and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LRWPAN), IEEE Std. 802.15.4, 2006.

[7] Altera website, http://www.altera.com, 2008.

[8] S.J. Park, "Analysis of AES Hardware Implementations", Department of Electrical and Computer Engineering, Oregon State University, 2003

[9] A. Satoh, et al., " A Compact Rijndael Hardware Architecture with S-Box Optimization", *Theory and Application of Cryptology and Information Security (ASIACRYPT 2001)*, Gold Coast, Australia, 2001

[10] S. Morioka, et al., " An Optimized S-Box Circuit *Architecture* for Low Power AES Design", *Cryptographic Hardware and Embedded Systems (CHES 2002)*, San Francisco Bay, CA, 2002.

[11] Advanced Encryption Standard (AES), FIPS Std. 197, 2001.

Ohyoung Song received his B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1980; an M.S. degree in electronics and electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1982; and a Ph.D. degree in computer engineering from the University of Massachusetts at Amherst in 1992. From 1992 to 1993, he was a member of the chief research staff of IBM, USA. He is presently a Professor at the School of Electrical and Electronic Engineering, Chung-Ang University, Seoul, Korea. His research interests include mobile and ubiquitous computing and smart grids.

Jiho Kim received his B.S., M.S., and Ph.D. degrees from the School of Electrical and Electronic Engineering from Chung-Ang University, Seoul, Korea, in 2000, 2002, and 2007, respectively. His major research interests include ubiquitous computing, WPAN, WLAN, and mobile network security.