Journal of Medical Systems (2019) 43: 31
https://doi.org/10.1007/s10916-018-1145-9

IMAGE & SIGNAL PROCESSING

An Optimized S-Box Circuit for High Speed AES Design with Enhanced PPRM Architecture to Secure Mammographic Images

T. Manojkumar [1] · P. Karthigaikumar [1] · Varatharajan Ramachandran [2]

Received: 13 July 2018 / Accepted: 19 December 2018 / Published online: 5 January 2019

Abstract

In AES, the total time taken by the architecture while implementing in low power and high-speed circuit is the most important thing that to be considered. Also in AES, implementing S-Boxes consumes the major part of the total time consumed by the entire architecture. In this research paper, we propose a very low-power and high efficient S-Box circuit architecture: a multi-stage modified version of PPRM architecture over composite fields. In this modified S box design, only AND and Hazard transparent XOR gates are used. Because of this architecture dynamic hazards which form the main aspect of power consumption in S box gets eliminated. A low propagation delay of 4.58 ns and occupies 120 slices in the xilinx FPGA device xc6vlx75t-3ff784, while the low propagation delay and slice area is 5.552 ns and 120 respectively for the conventional PPRM architecture. This new proposed architecture is used to protect the mammographic images from being unauthorized access.

Keywords Advanced encryption standard · Positive polarity reed muller · S-Box · Hazard · Field programmable gate array · Application specific integrated circuits · BDD · SOP · Data encryption standard · Pipeline architecture · Mammographic images

Introduction

Medical images must be protected from the security breaches, if the patient information is to be confident. Mammography is one of the most sensitive techniques to identify the breast cancer of a women by examining the breasts. In mammography, X-ray imaging is used in the detection and diagnoses of breast diseases in women. There are high chances for these information to be hacked and data may be altered by the third party. So this images which are obtained from mammogram must be encrypted and stored in the secure manner. For this security purposes, cryptographic algorithm can be employed. Even the recent advanced breast tomosynthesis images can also be encrypted and decrypted using the different cryptographic algorithms.

For more than 2 decades DES (Data Encryption Standard) is the most standard algorithm used for the secure communication between end users. But because of some shortcomings in DES, such as key length, in 2001, NIST (National Institute of Standards and Technology) made Rijndael the new standard cipher AES [1]. This new AES architecture was jointly invented and processed by Belgian researchers Joan Daemen and Vincent Rijment.

With embedded systems, faster execution of the AES (Advanced Encryption Standard) algorithm is a critical problem. Therefore, many alternative possible ways are to be developed in implementing AES with minimum propagation delay. Over the past two decades, many embedded architectures have been proposed and their performances are evaluated using Field Programmable Gate Arrays and Application Specific Integrated Circuits. Most of these new architectures are based on the variations in S-Box architecture to provide faster execution and less power consumption. Area occupied by AES encryption algorithm is also a constraint on implanting the algorithm in ASIC's and FPGA's.

In this research paper, we investigated and proposed a new design methodology for a better AES with faster execution and better area management. In this paper, the new design

This article is part of the Topical Collection on *Image & Signal Processing*

✉ Varatharajan Ramachandran
varathu21@yahoo.com

P. Karthigaikumar
p.karthigaikumar@gmail.com

[1] Department of Electronics and Communication Engineering, Karpagam College of Engineering, Coimbatore, Tamil Nadu 641 032, India

[2] Department of Electronics and Communication Engineering, Sri Ramanujar Engineering College, Chennai, Kancheepuram, Tamil Nadu 600 126, India

has been focused on the logic or gate level, rather than looking at the transistor level. Because of this consideration, this new architecture can be applied to numerous applications for the secure transmission and reception. Cryptographic AES algorithm can be implemented in both hardware and software. But thehardware implementation is a good choice in terms of speedof execution and security. Hardware implementation strategy varies based on our necessities. Our new algorithm can be implemented in the e-commerce servers since higher execution speed is a predominant factor in those servers.

After deep investigation it is found that the total time, area occupied and power consumption, it is clear that the S-Box operation in the SubBytes process consumes major part of the total time taken, area occupied and power requirement of the entire AES architecture. Therefore, a change in the S-Box architecture affects the whole AES Encryption and Decryption resource utilization.

There are three different types of architecture for AES hardware implementation.

(i). Looping architecture
(ii). Fully unrolled pipelined architecture
(iii). Deep sub pipelined fully unrolled architecture.

The looping architecture implemented by Gaj [2] uses feedback of data for each round,the fully unrolled pipelined architecture implemented by Saggese et al. [8] uses pipeline registers at the time unrolled stages of AES, Whereas the Deep sub pipelined fully unrolled architecture implemented by I Hammad et al. [10] is done by further dividing each stage by pipeline registers.

Many researchers have contributed significantly in developing area efficient and faster AES architectures. The innovative work offered by K. Gaj [2], Good and Benaissa [3] andRouvroy et al. [4] clearly depicts the efficient utilization on area (slice utilization). Grando et al. [5] design based on composite field arithmetic $GF(2^2)^4$ a using sub-pipelined loop unrolled architecture provides higher throughput. A design that provides good speed with sacrifice in the slice usage is proposed by Saggase et al. [6] with unrolling, tiling and pipelining transformations. With deep sub pipelining technique Hammad et al. [7] achieved better throughput and also reduced area. Different ideas of efficient Virtex-E FPGA implementations of the AES Algorithm with higher throughput and efficient area utilization were proposed by Standaert et al. [8].

Here in this paper, the analysis of Positive Polarity Reed Muller form S-Box for eliminating larger power consumption due to dynamic hazards is done and modifications in the PPRM structure is introduced to achieve better speed and make the AES algorithm to be area efficient. The new idea is targeted to achieve high throughput and area efficient while implementing the architecture in FPGA.

Rest of the paper is organized as follows Section II briefs about the structure and processes involved in AES algorithm for the 128-bit key. Section III details about the various S-Box architectures. Modified PPRM structure, its Performances and its effect on mammographic images are explained in Section IV. Conclusion is given in the last section.

AES algorithm

Advanced Encryption Standard algorithm is a symmetric key cryptographic algorithm which is widely used in the world for secure communication. In case of AES single algorithm can be used for both encryption and decryption operations. Also same is used in both encryption and decryption operation. AES has a fixed block size of 128 bits, whereas the secret key length varies [9–12]. But the data block length is fixed as 128 bits for all key lengths. Based on the different key length of 128, 192, 256 bits, AES has three different versions as AES-128, AES-192, AES-256. According to key length, number of encryption and decryption operation takes place on a single data block of 128 bits also differs. These relations are given in Table 1. For each round of encryption and decryption operation, a separate key called subkeys is needed. These subkeys are derived from the single main key. This process of creating subkeys from main key is called key expansion or key scheduling process. From Table 1 it is clear that if Nr is the number of rounds for a given version of AES, then Nr + 1 is the required number of subkeys.

In all the versions of AES, pre encryption round and last rounds are different and all the other rounds are same. Input data block of 128 bits are arranged as [4*4] matrix which is often called as state. Sate acts as an input or output for a particular operation in AES encryption and decryption. Each entry in the state matrix has a byte of data [13–15].

AES encryption and decryption operation

Both Encryption operation and decryption involves four steps and these four steps are repeated in each round of operation except last round. Encryption and decryption has the following operations in each repetitive round.

1. Byte Substitution / Inverse Byte Substitution
2. Shift Rows / Inverse Shift Rows

Table 1 Relation between Key length and Rounds of operation

AES Version	Key length	No of rounds	No of Subkeys
AES-128	128	10	11
AES-192	192	12	13
AES-256	256	14	15

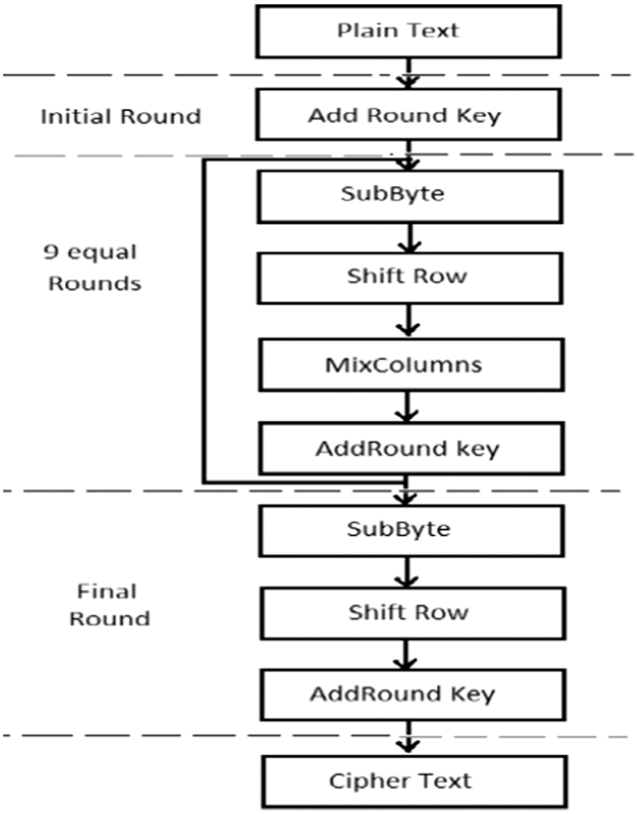


Fig. 1 AES Process

3. Mix Columns / Inverse Mix Columns
4. Add Round Key

In last round of encryption and in first round of decryption operations, mix column step is not included since it can be easily invertible. So its inclusion can be avoided. First two process, byte substitution and shift rows can be interchanged without affecting the encryption operation.

In Byte Substitution and Inverse Byte Substitution, each entry in the state matrix is replaced by another entry. This is a highly non-linear process in the AES algorithm. Byte Substitution can be implemented in several ways. Shift rows shifts the elements in the second, third and fourth row towards left by 1, 2, 3 positions respectively during encryption. It shits towards right in decryption operation. During mix column, bytes are multiplied with a predefined modular matrix. At last subkeys are combined with the output of mix column operation by performing EXOR operation. It is to be noted that Byte Substitution is the only non-linear process in the AES algorithm. All the other processes are reversible. Figure 1 describes the process of AES-128 Encryption [16, 18, 19].

S-Box architectures

Byte Substitution produces a one-byte nonlinear transformation using 128-bit S-Boxes. Each entry in the S-Box is a multiplicative inversion on a Galois field GF(2^8) and then an affine transformation. Galois Field uses an irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$ [20, 21]. Since the S-Box transformations are the only non-linear function, security of the AES is majorly dependent on the S-Box implementation second after the secret key information. S-Box architecture can be realized in hardware and software methods.

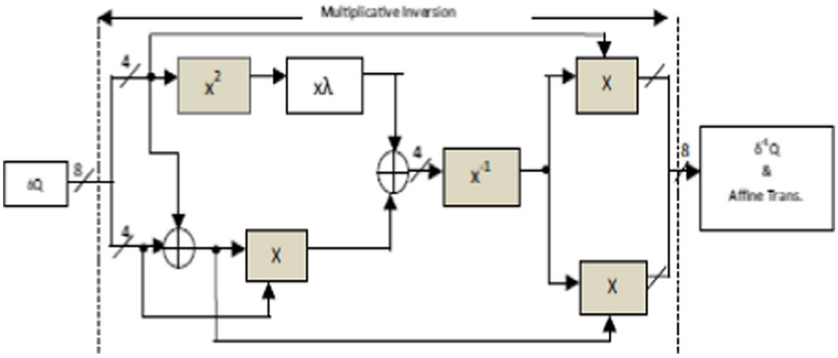
Hardware realization

In Hardware realization, substitution of values is calculated directly. No memory is required for hardware techniques. But calculating the S-Box value needs additional hardware circuits which occupy more chip area. In case of hardware approach of synthesizing S-Box values involves two steps. Firstly, multiplicative inverse of the given data is to be found and followed by an affine transformation leads to the required value, which is slower and area ineffective [23–25].

Software realization

In case of software realization of S-Box structure, Look Up Tables (LUTs) are used. Substitute values are substituted directly from the predefined LUTs. In this technique a dedicated RAM is used for storing the LUT values. LUT values can be obtained with the help of SOP, POS expressions. BDD method can also be implemented with the help of SOP and POS

Fig. 2 CFA combinational circuit for calculating multiplicative inverse



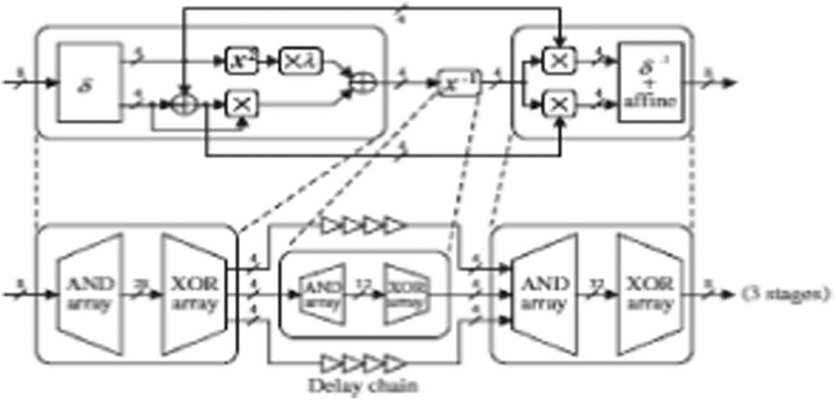


Fig. 3 Composite Field S-Box and Multistage PPRM

expressions for obtaining S-Box value very efficiently. Both hardware and software approaches can be combined and derive an efficient S-Box architecture [26–30].

Here in this research paper, an efficient software approach is discussed. With software implementations, we can achieve higher speed. But LUT occupies large area and nearly about 200 times LUT must be referred by the algorithm to encrypt a data block of 128 bits. IN this case LUT occupies larger chip area and has higher propagation delay. Therefore in order to reduce area and propagation delay an efficient method alternative to LUT based algorithm must be devised [31].

CFA structure in S-Box

Composite Field Arithmetic (CFA) calculates values of S-Box at the required time. There is no necessity to store the value in the predefined memory location. Implementing GF [28] is highly difficult since general implementation consumes more power and high propagation delay. In Composite Field Arithmetic structure instead of calculating GF [28] directly, GF $[22]^2$ is calculated [17]. Here calculations arebeing made in the lowerorder field. Multiplicative Inverse can be found in the lower order field with help of following polynomial

$$(bx+c)^{-1}=b\left(b^2+\lambda+c(c+b)\right)^{-1}x+(c+b)\left(b^2+\lambda+c(c+b)\right)^{-1} \tag{1}$$

With the help of above expression (1) combinational circuit can be derived as shown in Fig. 2 and Multiplicative Inverse can be found.

Steps involved in Byte substitution process with CFA structure

1) By isomorphism function, map all elements of the field A to a composite field B.
2) Compute the multiplicative inverses over the field B.
3) Re-map the computation results to A, using inverse isomorphism function [32].

S-Box realization by PPRM architecture

Main disadvantage of Composite Field Arithmetic architecture is its complicated signal paths. Because of this

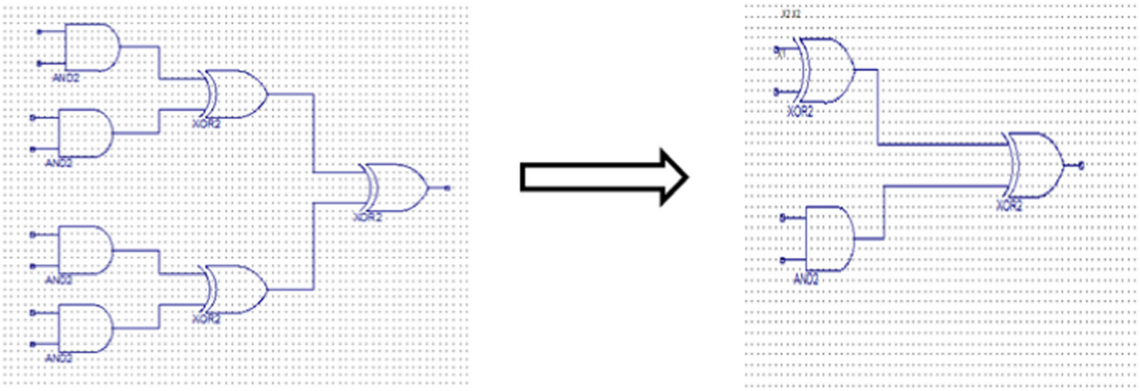


Fig. 4 Illustration of obtaining MPPRM from PPRM

Table 2 Comparison of different S-Box Architectures in 3s500efg320–5 (Spartan 3E FPGA)

Method	Propagation delay (ns)	No of Slices	Power Consumption (mW)
CF [23]	20.861	36	0.103
TBDD [33]	13.659	108	0.368
PPRM [34]	17.879	53	0.081
MPPRM	16.547	51	0.081

Table 4 Comparison of different S-Box architectures in 6slx75tfgg676–3 (Spartan 6 FPGA)

Method	Propagation delay (ns)	No of Slices	Power Consumption (mW)
CF [23]	15.998	45	0.16
TBDD [33]	11.863	194	0.518
PPRM [34]	14.953	119	0.064
MPPRM	13.189	73	0.064

complicated signal paths power consumption and signal delay is increased. This disadvantage can be eliminated by converting some of S-Box into two level logics. Much better S-Box can be derived by converting carefully selected subcomponents into two-level logic, if an appropriate partitioning of the S-Box into sub-components is done. Positive Polarity Reed Muller (PPRM) is an approach for finding Multiplicative Inverse using array of EXOR and AND gates alone. Main advantage of this process is that dynamic hazards occurring in the combinational circuits get reduced. As a result, total power consumption of the entire S-Box architecture can also be reduced since dynamic hazards consume much of the power in S-Box realization [33, 34].

PPRM architecture can be realized in three stages. These three stages of operation take place at the three different stages of Composite Field Arithmetic Architecture i.e. Pre-inversion, Inversion and Post-inversion stages. As Shown in Fig. 3, all the three stages can be realized by using AND array and EXOR array. Figure 3 also clearly explains that two signals which are not passing through Multiplicative Inversion process is delayed using some delay circuits thus avoiding the unwanted hazards. This delay circuit does not perform any operation on the given signal data. It just delays the signal transmission as close to the time taken by Inversion block.

Table 3 Comparison of different S-Box Architectures in 6vlx75tff784–3 (Virtex 6 FPGA)

Method	Propagation delay (ns)	No of Slices	Power Consumption (mW)
CF [23]	6.035	47	3.004
TBDD [33]	5.048	197	5.87
PPRM [34]	5.552	120	1.570
MPPRM	4.581	76	1.568

Modified PPRM architectures

The above multistage PPRM architecture can be further reduced without affecting the total power consumption but achieving greater speed and better area efficiency. With the help of simple optimization given in the combinational circuit relation between EXOR and AND gate, this new improved version of Positive Polarity Reed Muller S-Box architecture can be achieved. This process is illustrated in the Fig. 4.

First picture in Fig. 4 is a part of stage 2 of 3 stage PPRM realization. In that circuit realization, it is necessary to have 4 AND gates and three EXOR gates. But this circuit can be simplified as the second picture in Fig. 4. Both the circuit consumes the same amount of power except that second circuit has only one AND gate and two EXOR gates.

Dynamic hazards can be eliminated by adding some delay circuits in the path of AND gate result. By this new circuit design we can save three AND gate and one EXOR gate area with the same amount of power consumption. Main aim of having PPRM circuit is that, PPRM architecture consumes lesser power than that of other types of S-Box architectures. Since there is no variation in the power consumption between PPRM and Modified PPRM architecture, Modified PPRM can be preferred over conventional PPRM with added advantage having lower propagation delay and chip area.

Both PPRM and MPPRM S-Box realizations have been simulated and synthesized over three different types of FPGAs and the results are tabulated. In this research paper, Power consumption, Propagation delay and chip area of the S-Box architecture are the major constraint. Tabulations 2,3 and 4 clearly shows the result of synthesis over Spartan 3E, Virtex 6 and Spartan 6 respectively. Composite Field architecture (CF), Twisted Binary decision diagram (TBDD), Positive polarity Reed Muller architecture's performances in terms of power consumption is measured by implementing the above architectures in Application Specific Integrated Circuits (ASIC).

In this research paper, the architectures are implemented and performances are measured in the Field Programmable Gate Arrays (FPGA). Since the implementation in FPGA is

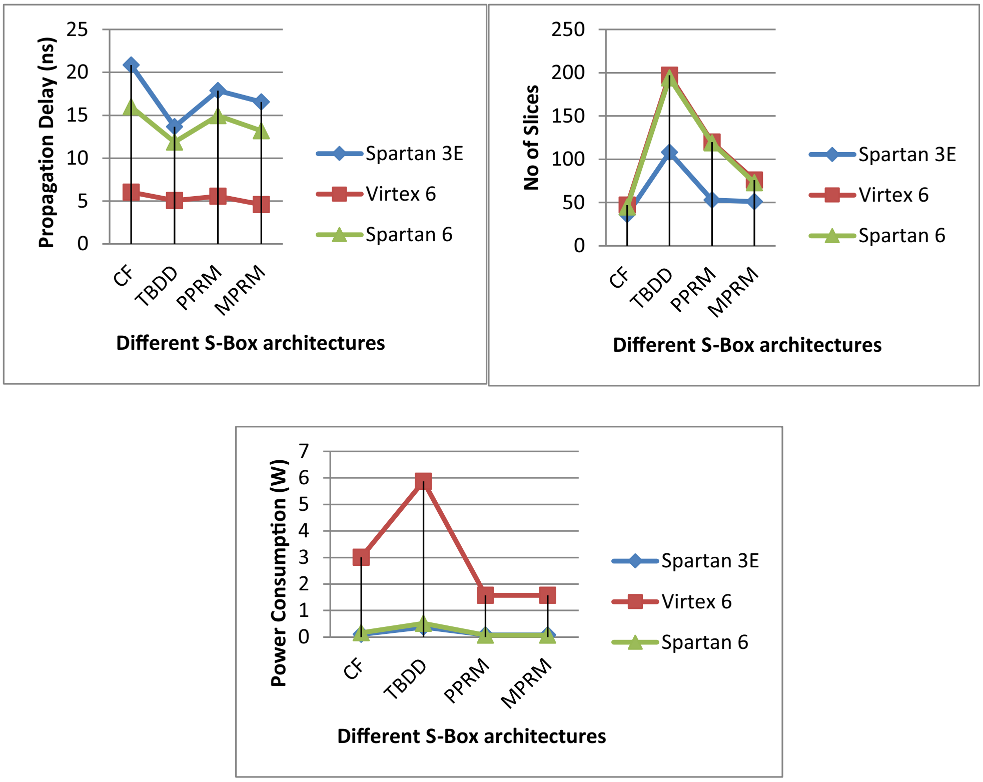


Fig. 5 Performance comparisons of various methods of S-Box Architectures

simpler than that of ASIC implementation [22]. As shown in Tables 2, 3 and 4 in Section III, the PPRM and MPPRM S-Box architecture achieves the lowest power consumption among all other S-Box architectures. Modified Positive Polarity Reed Muller architecture shows a great variant in terms area and the speed of the operation when compared to that of convention 3 stage PPRM S-Box structures. In all the devices, MPPRM structure occupies about less than 62% of area occupied by the PPRM structure. Also this new algorithm implements the cryptographic process 12% faster than that of convention PPRM architecture.

Though there is no much change in the power consumption between these two architectures, MPPRM holds the advantage of having faster execution and lesser area. Figure 5 shows performance comparisons of different S-Box architectures in the line chart. In power consumption chart of Fig. 5, it clear that both the architecture PPRM and MPPRM consumes the same amount of power irrespective of the FPGAs used. Improved MPPRM architecture shows the improvement in the execution speed and total area usage. Overall execution speed of the AES algorithm is improved since much of the AES execution time is allocated to S-Box operation. As it is mentioned earlier that, for the AES cryptographic process, all operations are repeated over sometimes for both encryption and decryption process.

This modified and improved S-Box is used for building AES algorithm and this improved version of AES is implemented for securing the information that has to be obtained from the mammographic images. Pixel values in the mammogram images acts as the data input the AES and the pixel values of the mammographic images can be obtained with the help of Matlab [22]. Input data to the AES is encrypted and the encrypted images looks like the noisy form of original images which are shown in Fig. 6. The encrypted pixel values can be

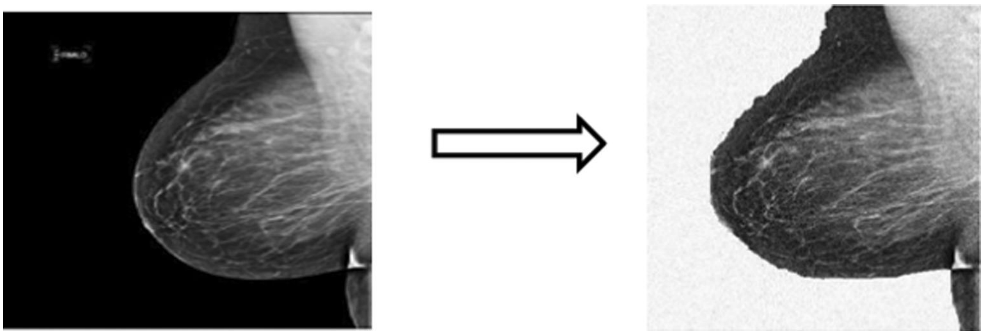


Fig. 6 Original and encrypted mammographic images

combined to form a mammographic image. The resultant image will not reveal any information regarding the diseases in breast of women. Encrypted images can be transferred to the destination and in the receiver side the original mammographic images can be obtained by decrypting the received images. Here also Matlab can be used to convert images into data and data to image after decryption. This modified AES algorithm works better in securing data in the mammographic images.

Conclusion

In this research work, we have developed a Modified multi-stage PPRM architecture for High speed S-Box circuit to secure the data in the mammographic images. In general S-Box of the AES architecture takes much of the total execution time for encryption and decryption operations. Total propagation delay of PPRM S-Box circuits can be significantly reduced by providing a simple modification in the combinational circuit realization in second stage of three stage PPRM architecture. A total propagation delay of only 4.9 ns in Virtex 6 FPGA was achieved, while the propagation delay of the conventional PPRM, TBDD and composite field S-Boxes are 5.552, 5.048 and 6.035 respectively. With MPPRM design the output of the S-Box is achieved 12% faster than conventional PPRM. Area minimization of about 60% can be achieved with this new proposed architecture. MPPRM also has lower power consumption when compared to Composite Field Arithmetic and Twisted BDD architecture. The resultant AES algorithm is efficient in encrypting and decrypting the mammographic images. This AES algorithm can be extended to securing the data of other healthcare applications like ECG, Endoscopy, Hysteroscopy etc.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. National Institute of Standards and Technology (NIST), Advanced Encryption Standard (AES). FIPS Publication 197, 2001. http://csrc.nist.gov/encryption/aes/index.html
2. Gaj, K., Very Compact FPGA Implementation of the AES Algorithm. In Proc. Int. Work. CHES2003, No. 2779, pp. 319–333, 2003.
3. Good, T. and Benaissa, M., Very small FPGA application-specific instruction processor for AES, IEEE Trans. Circ. Sys. I, 53(7), 2006.
4. Rouvroy, G., Standaert, F. X., Quisquater, J. J., and Legat, J., Compact and Efficient Encryption/Decryption Module for FPGA Implementation of the AES Rijndael Very Well Suited for Small Embedded Applications. Proc. Int. Conf. Cod. Comp. 2:583–587, 2004.
5. Granado-Criado, J. M., Vega-Rodriguez, M. A., Sanchez-Perez, J. M., and Gómez-Pulido, J. A., A new methodology to implement the AES algorithm using partial and dynamic reconfiguration. Integr. VLSI J. 43:72–80, 2010.
6. Saggese, G. P., Mazzeo, A., Mazzocca, N., and Strollo, A. G. M., An FPGA-based performance analysis of the unrolling, tiling, and pipelining of the AES algorithm. In: Proc. FPL, pp. 292–302, 2003.
7. Hammad, I., Sankary, K. E., and Masry, E. E., High Speed AES Encryptor with Efficient Merging Techniques. IEEE Embed. Syst. Lett. 2(3):67–71, 2010.
8. Lin, T. F., Huang, C. T. S., and Wu, C. W., A high-throughput low-cost AES cipher chip‖, IEEE Asia-Pacific Conf. ASIC, pp. 85–88, 2002.
9. Mayer, U., Oelsner, C., and Kohler, T., Evaluation of different Rijndael implementations for high end servers, IEEE Int. Symp. Circ. Syst. ISCAS, pp. 348–351, 2002.
10. Naveen Jarold, K., Karthigaikumar, P., Sivamangai, N.M., Sandhya, R, Asok, S. B., Hardware implementation of DNA based cryptography. 2013 IEEE Conference on Information & Communication Technologies (ICT), 696–700, 2013.
11. Mali, M., Novak, F., and Biasizzo, A., Hardware implementation of AES algorithm. J. Electr. Eng. 56(9–10):265–269, 2005.
12. Karthigaikumar, P., Anitha Christy, N., Siva Mangai, N. M., PSP CO2: An efficient hardware architecture for AES algorithm for high throughput. Wireless Personal Communications, 305–323, 2015.
13. Rahimunnisa, K., Karthigaikumar, P., Christy, N., Kumar, S., Jayakumar, J., PSP: Parallel sub-pipelined architecture for high throughput AES on FPGA and ASIC", Europian Journal of Computer Science, 173–186, 2013.
14. Mangard, S., Aigner, M., and Moninikus, S., A highly regular and scalable AES hardware architecture. IEEE Trans. Comput. 52(4): 483–491, 2003.
15. Bryant, R. E., Graph-Based Algorithms for Boolean Function Manipulation. IEEE Trans. on Computers C-35(8):677–691, 1986.
16. Fischer, V. et al., Two methods of Rijndael implementation in reconfigurable hardware. Proc. CHES2001, LNCS 2162:81–96, 2001.
17. Itoh, T., and Tsujii, S., A Fast Algorithm for Computing Multiplicative Inverses in GF(2^m) using Normal Bases. Inf. Comput. 78(3):171–177, 1988.
18. Abhitjith, et al., High performance hardware implementation of AES using minimal resources. In Proc. IEEE Int. Conf. Intelligent Syst. Sig. Processing, pp. 338–343, 2013.
19. Tillich, S., Feldhofer, M., and Großschädl, J., Area, delay, and power characteristics of standard-cell implementations of the AES S-box. In Embedded Computer Systems: Architectures, Modeling, and Simulation, Springer Verlag, pp. 457–466, 2006.
20. Bertoni, G., Macchetti, M., Negri, L., and Frangneto, P., Power-efficient ASIC synthesis of cryptographic S-boxes. In: Proc. the 14th ACM Great Lakes symposium on VLSI (GLSVLSI 2004), pp. 277–281, 2004.
21. Aziz, A., and Ikram, N., Memory efficient implementation of AESS-boxes on FPGA. J. Circuits, Systems, and Computers 16(4):603–611, 2007.
22. Manoj kumar, T., et al., FPGA implementation of an optimized key expansion module of AES algorithm for secure transmission of personal ECG signals. in Design Automation for Embedded Systems, 2017.
23. Satoh, A., Morioka, S., Takano, K., and Munetoh, S., A compact rijndael hardware architecture with S-box optimization. Advances in Cryptology – ASIACRYPT 2248:239–254, 2001.
24. Badillo, I. A., Uribe, C. F., and Para, R. C., Design and implementation of an FPGA-based 1.452 Gbps nonpipelined AES architecture. In Proc. of the International Conference on Computational Science and its applications, Lecture Notes in Computer Science, Springer-Verlag, 3982, pp. 446–455, 2006.
25. Abuelyman, E. S., and Alsehibani, A. A. S., An optimizedimplementation of the S-Box using residue of prime numbers. International Journal of Computer Science and Network Security 8(4):304–309, 2008.

26. Trejo, E. L., Henriquez, F. R., and Perez, A. D., An efficient FPGA implementation of CCM using AES. In Proc. of the 8th International Conference on Information Security and Cryptology, Lecture Notes in Computer Science, Springer-Verlag, pp. 208–215, 2005.

27. Kuo, H. et al., Architectural Optimization for a 1.82 Gbits/sec VLSI Implementation of the AES Rijndael Algorithm. Proc. CHES2001, LNCS 2162:53–67, 2001.

28. Reddy, S. K., Saktivel, R., and Paneeth, P., VLSI implementation of AES crypto processor for high throughput. International Journal of Advanced Engineering Sciences and Technologies 6:22–26, 2011.

29. Chen, C.-N., and Yen, S.-M., Differential Fault Analysis on AES Key Schedule and Some Countermeasures. Information Security and Privacy, 118–129, 2003.

30. Yoo, S. M., Kotturi, D., Pan, D. W., and Blizzard, J., An AES crypto chip using a high speed parallel pipelined architecture. Microprocess. Microsyst. 29:317–326, 2005.

31. Fischer, V., and Drutarovsk, M., Two methods of Rijndael implementation in Reconfigurable Hardware, Cryptographic Hardware and Embedded Systems, CHES, pp. 77–92, 2001.

32. Henriquez, F. R., Saqib, N. A., and Perez, A. D., 4.2 Gbits/s single chipFPGA implementation of AES algorithm. Electron. Lett. 39(15):1115–1116, 2003.

33. Morioka, S. and Satoh, A., A 10 Gbps Full-AES Crypto Design with a Twisted-BDD SBoxArchitecture. 2002 IEEE Intl. Conf. on Computer Design, 2002.

34. Morioka, S., and Satoh, A., An Optimized S-Box Circuit Architecture for Low Power AES Design. CHES 1:172–186, 2002.