LOW POWER FAULT TOLERANT S-BOX DESIGN FOR XTS-AES ENCRYPTION

Arun Kumar.P[1], Pandian.P[2], Raja Paul Peringham[3]
[1] Assistant Professor, SENSE School , VIT University, Vellore-632014
[2] Senior Professor, SAS, VIT University, Vellore-632014
[3]Professor, ECE Department, Karpaga Vinayagar College Of Engineering, Chennai

***Abstract* : This paper discuss a low power fault tolerant S-Box design for XTS- AES Algorithm, also called as P1619 Crypto Core was developed by SISWG (Security In storage Work Group) a Hard Disk Encryption standard algorithm. The faults are injected by Fault Injection Circuits which are considered in terms of Hardware Failures for the S-Box Transformation in every round during the circular shift operation for the block size of 128 bits and the technique applied to correct the fault is component reusability which never uses extra overhead components or spare circuits. The design has been synthesized in Cadence 90nm Technology with clock frequency about 1700MHz and the cell area obtained is 256980µm^2 and the power consumption is 20198.53µW**

Keywords: Fault Tolerant, Low Power, S-Box, XTS-AES algorithm.

I.INTRODUCTION

In recent technology as digitalization is the main concept for transmitting the data through a secure medium, there are chances for the data to be corrupted by hackers. Communicating between the two ends through any medium, data protection is a major factor so that the receiver receives the data accurately in an encrypted format and decrypts it and sends an acknowledgement to the sender so that no data loss occurs. Many algorithms have been developed by the designers to protect the data against inevitable security threats [22]. Data integrity faces a major concept in today's technology which has to maintain a safe path during transmission which may be error free but the attacks to the transmission are not. It may be due to natural attack or manual attack [13]. In a limited bandwidth of transmission several security measures have to be developed which maintains the effective protocol transmission with high speed architectures for effective data packet transmission [19].

Now-a-days, hardware cryptography plays a vital role as because of reconfigurability in its architecture for its high speed applications and low power consumption. In general a cryptography algorithm proposed should provide greater resistance to attacks either in hardware or software implementation where the attack is on the Block cipher where the information related security key is obtained and the data is tampered by the third party in the middle during transmission and the receiver gets a wrong data which was not send by the sender. Authentication plays a vital role in security systems which establish the relationship between the two ends but at the same time it should be resistant to the attacks so that the information shared between the two parties is secure either in terms of Private or Public Key cryptographic method [14]. Therefore the authenticated Key obtained during the encryption process should be modified from time to time in order to maintain secrecy. Mostly faults are injected in the algorithm applied to retrieve the key and corrupt the system so that loss occurs in the transmission. Therefore apart from secure data transmission, fault tolerant architectures and error detection schemes also have to be developed so that no data loss occurs. The encryption algorithm applied adds extra confidential data so that the originality of the data remains the same but in encrypted format which has to be transmitted over networks. The algorithm applied should be strong such that the cryptanalyst should not be able to find the weakness of it. Once the fault has been detected, the sender has to immediately apply the corrective measures so that the secrecy of the message is not lost and the strength of the algorithm is retained.

Now-a-days Hardware Fault Tolerant plays a major role in Computing Systems and the techniques applied to rectify the fault are Voting Systems and TMR [17] but in this paper we deal with component reusability, a new technique in Fault Tolerance where the fault is rectified by using the previous components. Therefore the security level applied to the algorithm should be difficult to find the Key even when the faults are injected by several methods by the attacker by using several modern techniques where the information is retrieved by breaking the strength of the algorithm by repeating the hacking algorithm several times.

II. XTS-AES ALGORITHM

XTS-AES algorithm also called as P1619 Crypto Core was developed by SISWG (Security In storage Work Group) mainly focuses on encryption on Hard Disk Drives. The algorithm is mainly based on AES which

works on fixed block size of 128 but the key size is given as 256,384 and 512 respectively and the number of rounds is 10, 12 and 14 respectively. In this paper we consider the block size of 128 bits and the block diagram for encryption is given below and the steps are as follows [23]. The main difference between AES and XTS AES is that Tweak Key which is employed in XTS. Its main property is that it must not reveal the Master Key if exposed to the attacker or it must not be equal to the Master Key or invertible function of it and also reduces the weakness of side channel attacks when the hacker tries to modify the data which takes into the accountability of data protection.

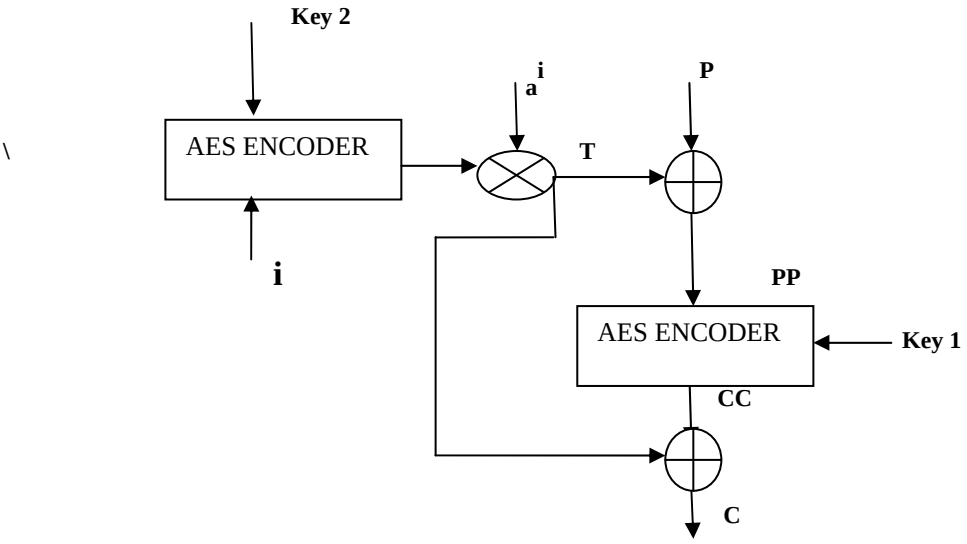

Figure 1: XTS AES Encryption

where Key = Key1 | Key 2 = 128 + 128 = 256 bits , 192 + 192 = 384 bits and 256 + 256 = 512 bits

P = Plain Text = 128 bit

C is the resultant Cipher Text resulting in 128 bits

i is the value of the 128-bit tweak

j is the sequential number of the 128-bit block inside the data unit

The cipher text shall then be computed by the following sequence of steps:

1) T ← AES - enc (key2; i) $\otimes$ a^j

2) PP ← P $\oplus$ T

3) CC ← AES- enc (key1; PP)

4) C← CC $\oplus$ T

The algorithm steps are as follows.

1) **Sub Bytes Transformation:** It is a nonlinear substitution operation in which each byte of the input state is replaced by the multiplicative inverse which follows Galois's field of GF (2^8) and the resulting field is replaced by the new byte using the function called S-Box which is formed of 4 X 4 Matrix.
2) **Shift Rows Transformation:** It is a Cyclic Rotation in which the rows of the S-Box are cyclically rotated Left by using multiplexers in Figure 10 where each input is of 32 bit length and depending upon the conditions of select signals as shown below depending on the loop number and the faults are induced which are detected by using concurrent fault detection technique [5]. It is generally represented by N – 1 where N is the loop number.

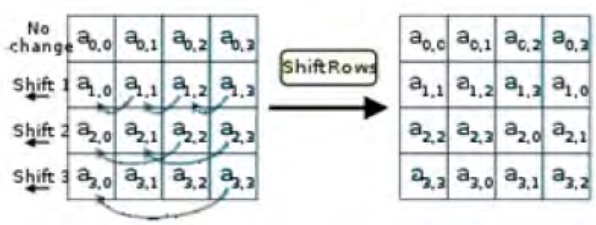

Figure 2: Shift row transformation

3) Mix Column: In this each column in the matrix which is of 128 bit is multiplied by a fixed polynomial by using Galois Field [5]. It transforms group of 4 bytes together forming 4 byte polynomials and multiplies the polynomials with a fixed polynomial mod (x ^ 4 + 1).

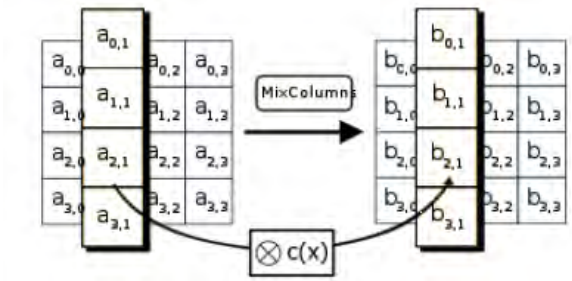

Figure 3: Mix Column transformation

4) Add Round Key: In this each byte is combined with the round key where the key is derived from the Key Schedule Operation [5].

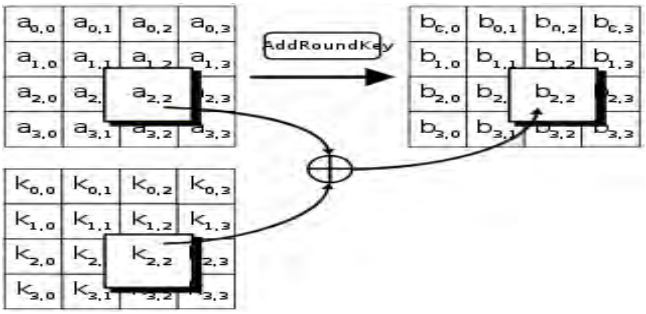

Figure 4: Add Round Key transformation

III. LOW POWER AND HARDWARE MINIMIZATION ARCHITECTURES

Today as the technology scales down, area and power are the major criteria to be taken into account. As the technology scales down, improvement in hardware complexity leads to energy efficient arithmetic operators and algorithms Low Power analysis has been done with high clock frequency about 800MHz. Low Power has emerged a goal to extend the battery powered devices and in many electronic devices to consume less power during peak operations. But the low power mainly arises due to improved architectures i.e. mainly due to Parallel Processing technique at both high and low frequency. The major factor to be taken in encryption process is the area overhead which means the storage should not depend on additional factors which may play a vital role in the encryption process. Arithmetic operations play a vital role which decides both the Low Power and Area of the device as they play a major role in designing the latest processors which should satisfy less clock cycle operations and also less power consumption which in other case leads to adverse effects. Therefore optimization effects have to be taken into account before designing the arithmetic modules. Several architectures have been proposed for high speed with parallel processing techniques for low power consumption by reducing the partial products of the multipliers of fixed width which reduces the hardware complexity [8,21,24] as shown in Fig 5 for 4 bit Baugh-Wooley multiplier. In this paper, the design has been implemented on Virtex-5 FPGA and the adder used is in terms of Multiplexer based adders which reduces the Partial Products generated and also the Hardware Complexity , delay and also consumes less power [3] as shown in Fig 6.

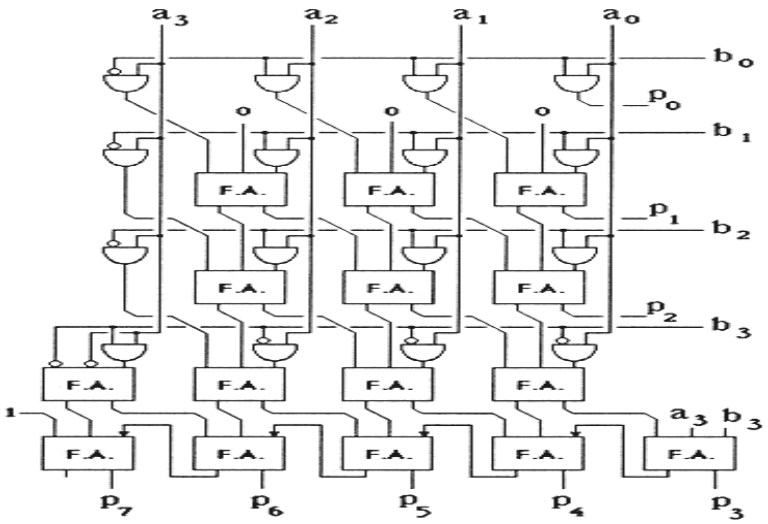

Figure 5: 4 bit Baugh-Wooley multiplier

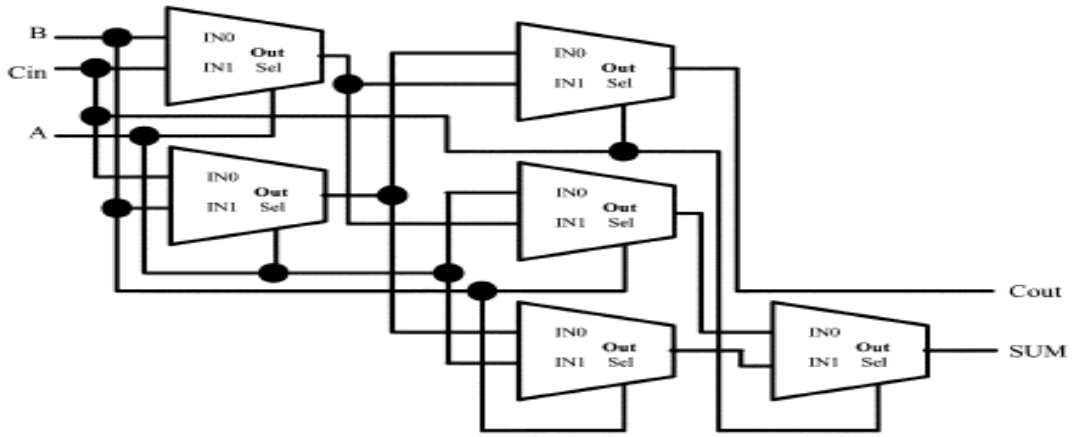

Figure 6: Xilinx Virtex-5 carry chain adder

Here we implement the RCA adder in the form of multiplexer based CSA which is a category of Fast Adders which uses $\frac{n}{2}$ multiplexers to select a carry signal. Figure 6 shows the internal structure of the F.A. and its implemented example is shown in Example 1 below for 4×4 multiplier. From the example it is clear that instead of using many adder chains as shown in Figure 5, we can use only Seven 2 to 1 Multiplexers as shown in Figure 6 above where the F.As gets reduced by $\frac{1}{2}$ thus reducing the hardware complexity and also this parallel architecture reduces the delay. Block shown in Figure 5 where $S_{\circ},\ldots,S_n$ denotes the Partial Products generated namely $P_{\circ},\ldots,P_n$ of the multiplier which uses column by-pass and carry save technique. Consider the example shown below for 4 bit multiplier.

```
                    A3      A2      A1      A0
                              X
                    B3      B2      B1      B0
  ------------------------------------------------------
                  A3B0    A2B0    A1B0    A0B0
                                                        +
          A3B1    A2B1    A1B1    A0B1
    A3B2  A2B2    A1B2    A0B2
A3B3 A2B3 A1B3    A0B3
```

Example 1: Implementation of 4 X 4 Multiplier

The Low Power Design for 128 Bit AES-Enc is shown below in Figure 7, its Key Schedule Unit is shown in Figure 8 and its Mix Column Unit is shown in Figure 9.

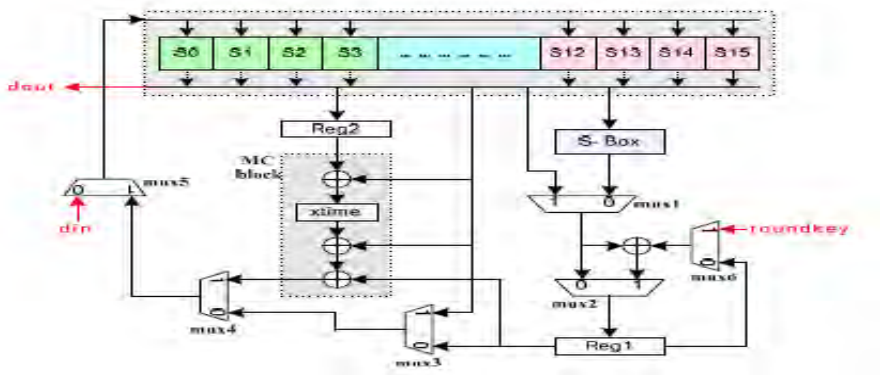

Figure 7: Low Power AES Module

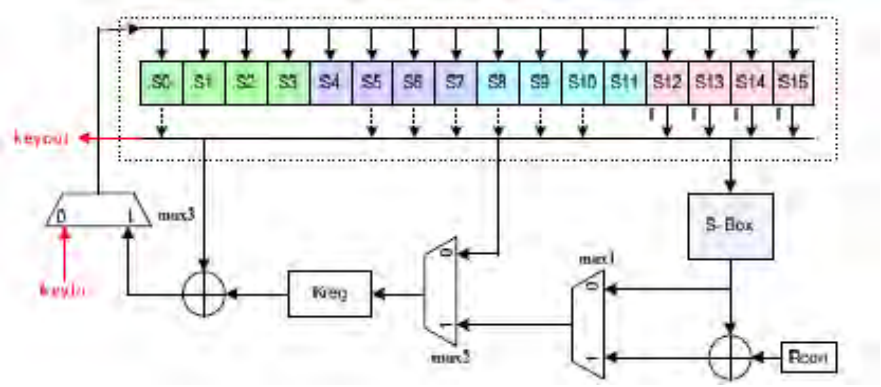

Figure 7: Low Power Key Schedule Unit for AES Module

To reduce the hardware complexity of the S-Box and also to improve parallelism, it is designed by using sub pipelining modes, Key Expansion Architectures and RAM for Key Storage [16, 18]. Furthermore, this algorithm which uses two AES blocks as shown in Fig.1 which leads to more complexity and in order to reduce it, a more sophisticated hardware has been developed which employs selection logic of the keys to be generated and the algorithmic operation either for encryption or decryption takes place in parallel depending upon the selection signal [7] and this core reusability technique helps us to achieve in reduction of area and also our implementation helps us to increase the throughput by 50% for this continuous operation of both encryption or decryption [4]. In RTL Design, the main objective in Low Power Design is to reduce the Switching Activity between the modules i.e. at a time only one module should be in active state and others in idle state and also reducing the sequential blocks as it may lead to Flip Flops / Latches. To perform the operation of S-Box at a higher speed, the combinational design has been employed which reduces the delay by using combinational field arithmetic [6] and it is mainly designed by using Boolean Functions.

IV. FAULT INJECTION, DETECTION AND RECTIFICATION

From the previous implementations, the Fault Detection scheme is mainly based on Parity Checking method where the actual parity is compared with predicted parity with OR Gates which is used for error detection [15]. The actual drawback in this condition is the OR Gate generated erroneous output which reduced the Fault Coverage to 97%.In Reconfigurable Architecture [1], the cells have been portioned into 4 x 4 matrix with the parity bits and the output parity is compared with the input parity and the faulty cell is being replaced with the neighbor cell with routing scheme which offers the reconfigurable architecture and increases the Fault coverage to 98% by increasing the area. In Low Overhead Parity Scheme, the S-Box is divided into 5 blocks and this offers 97% of the Fault coverage approach for multibit parity prediction. [12].In parallel AES technique, the on-line fault detection scheme is based on functional redundancy [9] where the sub-blocks which are repeated are partioned and compared with on line fault detection scheme where the faults are injected by side channel attacks and the fault detection probability is found to be 99.9%. In single fault masking technique of a Dual Port Ram, a part of the hardware is used for Parity Checking where the basic advantage of this method is it requires no extra clock and it requires no extra FPGA RAM blocks and the fault coverage is found to be 100% [20]. Here we consider the Differential Faults in terms of Hardware Fault Analysis and introduce a new concept of Component Reusability which never uses the spare of the original. Reusability means one faulty behavior of one circuit exactly matches the behavior of a faultless circuit the first faulty circuit can replace the second circuit whenever the second circuit becomes faulty without the need to go for a spare circuit. It is a method of Testing and the best applications of Fault Tolerant Design. By this method, we can save the hardware resources and make the system to shoot up immediately in case if a Fault is detected. Here we consider the Fault in the S-Box during the cyclic shift operation where each cell in the S-Box contains 8 bits and is designed with the help of multiplexers which performs the Left Rotate operation. The Differential Fault is injected using Fault Injection Circuits which leads to Component Malfunction so that the S-Box becomes faulty as shown below in TABLE 1where the Faulty Operations are indicated by RED Color. In this method, we are specifying the fault at a particular location by the user defined module to the inputs. Consider the 8 To 1 Multiplexer shown below in Figure10 which is used for the design of rotator.

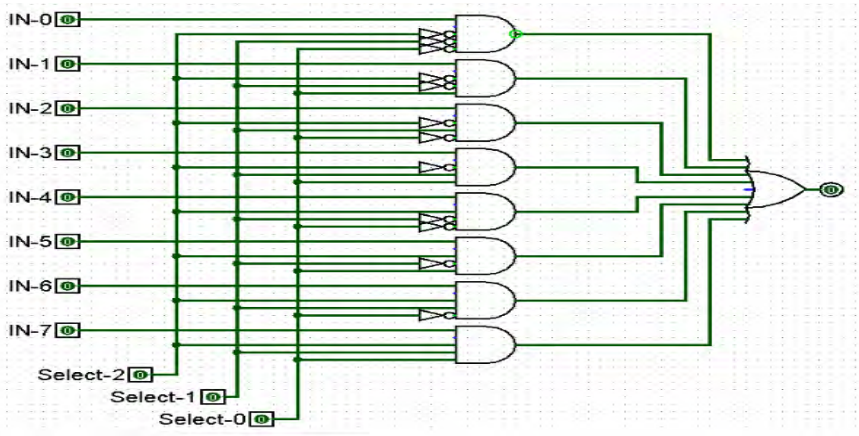

Figure 10: Multiplexer Unit used for Rotator of AES Module

Suppose if the AND Gate connected to the IN-0 input fails, then the following modules will be malfunctioning based on the Control Signals shown below in RED Color in TABLE 1.

Control Signal	Operation	S-Box Operation			
000	No Change	a0,0	a0,1	a0,2	a0,3
001	Left Rotate 1	a1,1	a1,2	a1,3	a1,0
010	Left Rotate 2	a2,2	a2,3	a2,0	a2,1
111	Left Rotate 3	a3,3	a3,0	a3,1	a3,2

TABLE 1: Faulty S-BOX Unit for AES Module

Once the Fault has been indicated during the normal rotate operation, we have to use the shortest routing path to connect to the alternate hardware which has the same operation of the faulty one so that this will overcome the faulty circuit to reduce the delay in the operation which enhances the speed of the circuit operation. For example

as indicated if the AND Gate connected to the IN-0 input fails, we can use the AND Gate connected to the IN-1 to provide the exact output for IN-0 as due to similarity of the components connected. By this method, we can use the component reusability technique, a new method in VLSI Testing and save the hardware resources. In this case once the fault is identified, the particular hardware is identified and instead of creating a duplicate / spare of it, we can use the other working component whose function is similar to the faulty one and make the resource to work perfectly thus saving time.

V. IMPLEMENTATION RESULTS, CONCLUSION AND FUTURE IMPLEMENTATION

When analyzed with the previous implementations, they use Parity Bit scheme which verifies the parities at both the ends and requires more complexity in designing the scheme and the Fault coverage Efficiency is found to be 97% - 98%. Our implementation shows 100% Fault correction Capability where the given hardware itself acts as an auto corrector which uses the principle of Fault Tolerant Routing and also overcomes the previous methods of adding Parity Bits and never use spare components. The mean time required to resume a component takes 3 clock cycles as to input the test vectors to the nearest component which has the same operation of the failed component and to make the component to start exactly from the last input as when the fault occurred. The design has been taken high consideration in terms of low power, area and throughput for high frequency and to obtain high Fault coverage and has been implemented on Cadence 90nm Technology and the following results obtained for clock frequency of 1700MHz has been shown in TABLE below.

Fault Detection Approaches	Fault Coverage for Multiple Faults
GF_1 and GF_2 [15]	97%
Reconfigurable Cell Array [1]	98%
Fault Detection by Composite Fields [12]	97%
On-Line Self-Test Architecture [9]	99.99%
LUT Faults (AES Combinational Logic) [20]	88.4%
Our Implementation	100%

TABLE 2: Implementation results of XTS-AES for fault coverage

Family	Area (μm^2)
ASIC 180nm and 130nm [7]	1162489 and 542648 respectively
Our Implementation ASIC 90nm	256980

TABLE 3: Implementation results of XTS-AES for area

Technology	Throughput (Gb/s)
ASIC 180nm and 130nm [7]	27.4 and 37.3 respectively
ASIC 90nm [5]	3.7
ASIC TSMC 0.09 µ LV [5]	7
ASIC 90nm [5]	3
ASIC 90nm [5]	2-16
Our Implementation ASIC 90nm	42.5

TABLE 4: Throughput results of XTS-AES

Technology	Power Analysis (mW)
0.18 µm [11]	54 mW
0.18 µm [10]	20.35mW
0.18 µm [2]	110mW
Our Implementation	20.19853mW

TABLE 5: Low-power results of XTS-AES

From the above results, we conclude that our design takes the maximum throughput and offers 100% FC when compared to the previous methods and the reliability of the component is also high w.r.t reusability technique. The drawback of this method is it takes 3 clock cycles to reshoot the system to normal operation. The future implementation of it may be FPGA implementation of Fault Tolerant XTS-AES and image encryptions and also deriving the same for other keys namely 384 and 512 and also for XTS -AES Decryption.

REFERENCES

[1] Alireza Hodjat, David D. Hwang, Bocheng Lai, Kris Tiri and Ingrid Verbauwhede, "A 3.84 Gbits/s AES Crypto Coprocessor with Modes of Operation in a 0.18-µm CMOS Technology", GLSVLSI '05 Proceedings of the 15th ACM Great Lakes symposium on VLSI,, pp.60-63, 2005.

[2] Athanasios P. Kakarountas, Epameinontas Hatzidimitriou, and Athanasios Milidonis High-throughput ASIC implementation of an encryption core for securing shared storage media, IEEE International Conference on Digital Signal Processing, pp. 1 – 5, 2011.

[3] G. Di Natale, M. Doulcier, M. L. Flottes and B. Rouzeyre, "A Reliable Architecture for Parallel Implementations of the Advanced Encryption Standard", Springer Journals on Electronic Test, Vol. 25, pp.269-278,2009.

[4] Epameinontas Hatzidimitriou and Athanasios P. Kakarountas, "Implementation of a P1619 Crypto-Core for Shared Storage Media", 15th IEEE Mediterranean Electrotechnical Conference, MELECON 2010, pp.597-601, 2010.

[5] Hongge Li, Jinpeng Ding and Yongjun Pan, "Cell array reconfigurable architecture for high-efficiency AES system", Elsevier Journals on Microelectronics Reliability, Vol. 52, pp.2829–2836, 2012.

[6] Y.J. Huang, Y.S. Lin, K.Y. Hung and K.C. Lin, "Efficient implementation of AES IP", Circuits and Systems, APCCAS 2006- IEEE Asia Pacific Conference , pp.1418-1421. 2006.

[7] Israel Koren and C.Mani Krishna,"Fault Tolerant Systems" Morgan Kaufman Publishers, San Francisco, CA.

[8] Jin-Hao Tu and Lan-Da Van," Power-efficient pipelined reconfigurable fixed-width Baugh-Wooley multipliers", IEEE Transactions on Computers, Vol. 58, pp.1346 – 1355,2009.

[9] L. Liu and D. Luke, "Implementation of AES as a CMOS core," Electrical and Computer Engineering, IEEE CCECE 2003-,Canadian Conference , Vol. 1, pp.53-56, 2003.

[10] Mao-Yin Wang, Chih-Pin Su, Chia-Lung Horng, Cheng-Wen Wu and Chih-Tsun Huang, "Single and Multi Core Configurable AES architectures for flexible security", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 18, pp.541 – 552, 2010

[11] Mehran Mozaffari-Kermani and Arash Reyhani-Masoleh, "Fault Detection Structures of the S-boxes and the Inverse S-boxes for the Advanced Encryption Standard", Springer Journals on Electronic Test, Vol. 25, pp.225–245, 2009.

[12] Mehran Mozaffari-Kermani and Arash Reyhani-Masoleh," A Lightweight High-Performance Fault Detection Scheme for the Advanced Encryption Standard Using Composite Fields", IEEE Transactions on (VLSI) Systems, Vol. 19, pp 85-91, 2011.

[13] Mehran Mozaffari-Kermani and Arash Reyhani-Masoleh, "A Low-Power High-Performance Concurrent Fault Detection Approach for the Composite Field S-Box and Inverse S-Box", IEEE Transactions on Computers Vol.60, pp.1327 – 1340, 2011.

[14] Ming Li, Wenjing Lou and Kui Ren, "Data security and privacy in wireless body area networks", Wireless Communications, IEEE, Vol.17, pp.51 – 58, 2010.

[15] Mohamed Elmoghany, Mohamed Diab, Moustafa Kassem, Mustafa Khairallah, Omar El Shahat and Wael Sharkasy," FPGA Implementation of High Speed XTS-AES for Data Storage Devices" IEEE International Conference Internet Technology and Secured Transactions (ICITST) , pp.25 – 28,2011

[16] Nabihah Ahmad, Rezaul Hasan and Warsuzarina Mat Jubadi," Design of AES S-Box using combinational logic Optimization", IEEE Symposium on Industrial Electronics And Applications (ISIEA), pp.696-699, 2010.

[17] Qian Wang, Cong Wang, Kui Ren and Wenjing Lou, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", IEEE Transactions on Parallel and Distributed Systems, Vol. 22, pp. 847 –859,2011.

[18] N. Sklavos and O. Koufopavlou," Architectures and VLSI Implementations of the AES-Proposal Rijndael", IEEE Transaction on Computers, Vol. 51, pp.1454 – 1459, 2002.

[19] Sophie Belloeil-Dupuis, Roselyne Chotin-Avot and Habib Mehrez,"Exploring redundant arithmetics in computer-aided design of arithmetic data paths", INTEGRATION, the VLSI journal, Vol. 46, pp.104 – 118, 2013.

[20] Uroš Legat , Anton Biasizzo and Franc Novak, "A compact AES core with on-line error-detection for FPGA applications with modest hardware resources" Elsevier Journals on Microprocessor and Microsystems, Vol. 35, pp..405–416, 2011.

[21] H.T. Vergos and D. Bakalis, "Area-time efficient multi-modulus adders and their applications", Elsevier Journals on Microprocessor and Microsystems, Vol. 36, pp. 409 – 419, 2012.

[22] Xinmiao Zhang, and Keshab K. Parhi," High-Speed VLSI Architectures for the AES Algorithm", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol.12, pp. 957 – 967, 2004.

[23] Yingtao Jiang, Abdulkarim Al-Sheraidah, Yuke Wang, Edwin Sha, and Jin-Gyun Chung, "A Novel Multiplexer-Based Low-Power Full Adder", IEEE Transactions on Circuits and Systems-II: Express Briefs, Vol. 51, pp. 345 – 348, 2004.

[24] Zhuo Hao, Sheng Zhong and Nenghai Yu, "A Privacy-Preserving Remote Data Integrity Checking Protocol with Data Dynamics and Public Verifiability", IEEE Transactions on Knowledge and Data Mining, Vol. 23, pp.1432 –1437, 2011.