A New Approach for Detecting and Correcting Errors in the Satellite Communications Based on Hamming Error Correcting Code

Ashkan Masoomi and Roozbeh Hamzehiyan

Abstract—**This paper presents a novel model to detect and correct Single Event Upsets in on-board implementations of the AES algorithm, which is based on Hamming error correcting code. The encrypted satellite data can get corrupted before reaching the ground station due to various faults. One major source of faults is the harsh radiation environment, Therefore any electronic systems used on-board satellites such as processors, memories etc. are very susceptible to faults induced by radiation. Single Even Upset (SEU) faults can occur on-board during encryption due to radiation. A detailed analysis of the effect of SEUs on the imaging data during on-board encryption using the modes of AES is carried out. Faults in the data can also occur during transmission to the ground station due to noisy transmission channels. In this paper the impact of these faults on the data is discussed and compared for all the five modes of AES. From five modes of AES, CRT mode is selected to encrypt satellite video and image links. A detailed analysis of the effect of SEUs on the imaging data during on-board encryption using the modes of AES is carried out.**

Index Terms—**Encryption, single even upset, hamming code, fault tolerant.**

I. Introduction

Earth Observation (EO) satellites observe the Earth by taking images with smart imaging sensors (cameras) on-board to be used in monitoring the environment, disasters, vegetation, map marking, urban planning etc. The on-board imaging cameras are becoming more and more sophisticated with a wide range of spectral band cameras to observe vegetation, thermal variations of Earth's atmosphere etc and sub-meter resolution cameras to capture the scenery with more details [1]. As the sensitive and valuable information available through satellite images is growing the security concern to these images is also growing [1], [2]. Latest unauthorized intrusions into satellite networks to access satellite data have proved that satellite data is vulnerable to hacking threats. Satellite data can be secured by applying cryptographic protection means to the data on board. Usually, satellite links model with an AWGN channel [3], [4].

Encryption, by far the most widely adopted cryptographic protection in terrestrial networks, is the conventional solution to protect the satellite data from unauthorized users. At present, more and more EO satellites are equipped with on-board encryption to protect the data transmitted to the ground station [5]. The encryption algorithms used in present satellite missions are typically proprietary or outdated algorithms like DES rather than algorithms based on the latest encryption standards [6].

The Rijndael algorithm approved as the Advanced Encryption Standard (AES) by the US National Institute of Standards and Technology (NIST) in October 2000 is being adopted by many organizations across the world [6]. It is used across a wide range of platforms ranging from smart cards to big servers because of its simplicity, flexibility, easiness of implementation and high throughput. Therefore, the AES is well suited for resource constraint platforms like on-board small satellites [7], [8].

In most EO satellites high throughput encryption processing is required to cover high data rate transmissions. This requirement can be easily met by present implementations of AES [9], [10], which have achieved a throughput ranging from few Mbps to Gbps. However in addition to high throughput, fault detection and tolerance is very important particularly in satellites. Because if faulty data is transmitted to the ground station, the user's request for data re-transmission has to wait until the next satellite revisit period, with revisit time varying from a couple of hours to weeks.

The satellite data can get corrupted due to various faults. The two major sources of faults are: (1) faults that occur during the encryption and (2) faults during the transmission. Satellites operate in harsh radiation environment consequently any electronic system used on-board including the encryption processor is susceptible to radiation-induced faults. Most of the faults that occur in satellite on-board electronic devices are radiation induced single bit flips called single event upsets (SEU) [9]. SEUs can corrupt the data during on-board encryption. The other source of faults is noise in the transmission channel. Satellite data can get corrupted during transmission to ground due to this noise.

Satellites operate in a harsh radiation environment and the interaction of such radiation with electronic systems used on-board, such as processors, memories etc., can cause failure, degradation or malfunctioning in their performance. Even a single fault during the encryption process can propagate as many faults in the final encrypted data and can corrupt the whole data from the point where fault has occurred [8]. Various methods have been proposed for fault detection of AES, which are mainly aimed at avoiding cryptanalysis of AES by injection of faults [6], [10].

Only fault detection is not enough for space applications

Manuscript received September 6, 2012; revised November 20, 2012.

The authors are with the Department of Communication Engineering, Islamic Azad University, Ahram Branch, Iran (e-mail: ashkan.masoomi@yahoo.com, r_hamzehyan@yahoo.co.uk).

DOI: 10.7763/IJCTE.2013.V5.683

but fault correction is equally important. There is no exception for an AES encryption processor used on-board. So the encryption processor should be robust enough to faults in order to avoid corruption of valuable data and subsequent transmission to ground.

Most of the faults that occur in satellite on-board electronic devices are radiation induced single bit flips called Single Event Upsets (SEU) [8]. SEUs are soft temporary faults and correcting them can restore the normal operation.

SEUs must be detected and corrected on board before sending the data to ground to avoid redundant transmission and use of erroneous data. The Triple modular redundancy (TMR) technique is one of the most widely used redundancy based SEU mitigation technique in satellites. A TMR design consists of three identical modules, which are connected by a majority voting circuit to determine the output [11]. However, with the TMR technique the area and power overheads triplicate in comparison with the original module. This paper presents a novel fault tolerant model for the AES algorithm aimed at mitigation of radiation induced SEUs on board satellites, which features a reduced hardware overhead.

In this paper we at first describe encryption by CRT mode of AES and then the proposed fault tolerant model of AES and its implementation is discussed. The advantages of FPGAs such as flexibility of design, shorter time-to-market, lower cost, remote configurability etc., make them suitable for use in small satellite on-board systems. The overhead caused by the error detection and correction model is also calculated and discussed.

II. ENCRYPT SATELLITE LINKS BY CRT MODE OF AES

The AES is a symmetric key algorithm, in which both the sender and the receiver use a single key for encryption and decryption. The standard defines the block length to 128 bits, and supports key lengths of 128, 192 or 256 bits [5]. The AES is an iterative algorithm and each iteration is called a round. The total number of rounds (*Nr*) is 10, 12, or 14 when the key length is 128,192 or 256 bits, respectively. Each round in AES except the final round consists of four transformations: Sub Bytes, Shift Rows, Mix Columns and Add Round Key. The final round does not have the Mix Columns transformation as shown in Fig. 1.

The round transformation of AES and its steps operate on some intermediate result, called state. The state can be visualized as a rectangular matrix with four rows. The number of columns in the state is denoted by N_b and is equal to the block length in bits divided by 32. For 128 bit data block the value of N_b is 4.

The Sub Bytes transformation is a non-linear byte substitution, operating on each byte of the state matrix independently. This transformation can be calculated on the fly for each state byte. Alternatively, the Sub Bytes transformation is computed in advance and the results are stored in a look-up table (LUT) of 28 (= 256) elements called S-Box (*SRD*) table. Shift Rows cyclically left shifts the last three rows of the state by 1, 2 and 3 bytes respectively.

Mix Columns transforms every column in the state by multiplying it with a predefined polynomial [2], [3], [11]. Finally, the Add Round Key transformation adds the expanded round key to the state by an XOR operation [5]. The transformations involved in the key expansion use operations like substitution (Sub Bytes), shift and XOR, which are similar to that of AES transformations and therefore they are not discussed in detail.

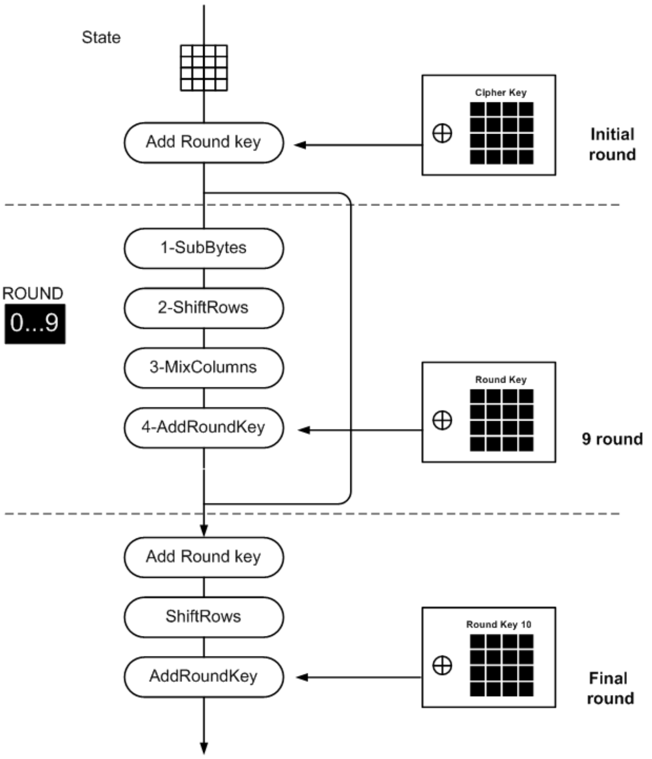

Fig. 1. AES algorithm block diagram.

To apply the AES block cipher to encrypt data of more than one block (128-bits) modes of operation, have been defined. So, implementation of AES just won't make any sense unless the mode of implementation is mentioned. The most commonly used modes with AES are Electronic Code Book (ECB) mode, Cipher Block Chaining (CBC) mode, Output Feed Back (OFB) mode, Cipher Feed Back (CFB) mode and Counter (CTR) mode. ECB and CTR are known as non-feedback modes whereas CBC, CFB and OFB modes are known as feedback modes. In addition ECB and CBC are referred to as block cipher modes as they require the whole block before encryption and OFB, CFB and CTR are referred to as stream cipher modes as they don't require the whole block before encryption.

The encryption and decryption of satellite multispectral images have been implemented using a purpose-built software program written using the Java programming language. The AES encryption implementation is divided into core modules and feedback modules. The feedback modules consist of encryption and decryption routines for ECB, CBC, CFB, OFB and CTR modes. The Sun's Java API for JPEG images [10] is used for image encoding and decoding during the encryption and decryption process. The software is also designed in such a way that the fault can be randomly injected at any round, transformation, byte and bit level to simulate the SEU [1].

Fig. 2 shows a new mode, the Counter mode, which came into effect after the AES has been made as a standard. In CTR mode, a counter is encrypted to generate a key stream, which is then XO Red with the plain data to generate the cipher data. A property of CTR mode, which is different from the CBC, CFB and OFB modes, is that there is no feedback or chaining; therefore one can perform several encryptions in parallel. This is a significant advantage in high-performance applications [12].

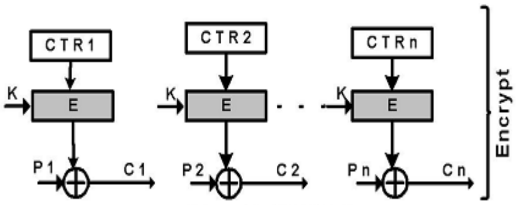

Fig. 2. CTR mode.

It has been observed that SEU can propagate faults from one block to multiple blocks depending on the mode of operation. In case of ECB, CBC, CFB and CTR modes SEU corrupts one block of data whereas in case of the OFB mode it can propagate the faults to the whole data starting from the point where the SEU has occurred. Hence the CBC, CFB and CTR modes are suitable for use on board as they propagate faults to just one block.

It has also been observed that the faults occurring during transmission can propagate from one block to multiple blocks depending on the mode of operation. In case of ECB the faults propagate to one block, whereas in CBC and CFB modes the faults can propagate to two blocks. In contrast, in OFB and CTR modes, only a single bit in the plain data is affected and the error does not propagate to other parts of the message. So the transmission fault is not propagated. Based on this analysis, we conclude that the OFB and CTR modes are more suitable for noisy channels, because unlike other modes, cipher data bit transmission errors are not expanded in the received plain data. Table I summarizes the amount of data corrupted due to single bit faults during encryption and transmission.

III. A Novel Model to Detect and Correct Single Event Upsets

Bit flip faults can occur during encryption as satellites operate in a harsh radiation environment and therefore any electronic systems used on-board, such as processors, memories etc., are very susceptible to faults induced by radiation. There is no exception for an encryption processor used on-board, which should be robust enough to faults in order to avoid transmission of corrupted data to ground. Most common and frequent radiation faults in satellite on-board electronics are single bit flips called SEUs [9]. SEUs are soft or temporary faults and correcting them can restore the normal operation of the device.

TABLE I: Fault Propagation Due to Single Bit Errors During Encryption and Transmission.

Amount of Data Corrupted Due to	ECB	CBC	OFB	CFB	CRT
Encryption	One block	One block	Complete data	One block	One block
transmission	One block	Two blocks	No fault	Two blocks	No fault

A study measuring the fault propagation in one block of AES has reported that even a single fault during the encryption process can result in many faults in the final encrypted data and on average 50 of the bits in the final encrypted data block will be corrupted [8]. In this paper, we have extended the study in [8] a step further from fault propagation within a block to within multiple blocks as modes of operation involve multiple blocks during encryption.

As it can be seen from Table 1 the CTR mode prove to be the best choice for satellite applications as it achieves minimum fault propagation in both cases - during on-board encryption and during data transmission.

It has also been observed that the faults occurring during transmission can propagate from one block to multiple blocks depending on the mode of operation. In case of ECB the faults propagate to one block, whereas in CBC and CFB modes the faults can propagate to two blocks. In contrast, in OFB and CTR modes, only a single bit in the plain data is affected and the error does not propagate to other parts of the message. So the transmission fault is not propagated. Based on this analysis, we conclude that the OFB and CTR modes are more suitable for noisy channels, because unlike other modes, cipher data bit transmission errors are not expanded in the received plain data. The amount of data corrupted due to single bit faults during encryption and transmission for CRT mode is low.

Bit flip faults can occur in the satellite channels due to noise during transmission of encrypted data to ground. There are techniques like Forward Error Correction (FEC) in place to detect these faults and correct them. Using FEC technique extra bits are added to the data to allow the receiver to correct some errors without having to request a retransmission of data. The maximum fraction of errors that can be corrected is determined in advance by the design of the code, so different forward error correcting codes are suitable for different conditions [10].

In the feedback modes the faults in one block can propagate to other blocks because of the feedback. We have investigated how a single bit fault occurring during encryption and during transmission can propagate to subsequent blocks. An elaborate study has been carried out to measure the fault propagation in the feedback modes in order to propose a suitable mode of encryption for satellite on-board use.

This section describes a new approach for error detection and correction (EDAC) for the AES algorithm. The proposed fault tolerant model is based on the Hamming Error correcting codes. The model provides fault detection and correction functionality in the data path of AES using the parity based Hamming error correcting code at byte level in each transformation of each round.

The fault tolerant model presented here is based on the Hamming code [12], [8], the simplest of the available correcting codes. The Hamming code [12], [8]. detects and corrects a single-bit fault in a byte and it is a good choice for satellite applications, as most frequently occurring faults in on-board electronics of the faults are single bit flips induced by radiation. However, the correction model can be extended to correct multiple bit faults by using sophisticated codes like

Hamming code [12], Read-Solomon codes etc.

The proposed new approach to fault detection and correction is based on predicting the Hamming code at the end of each transformation from the pre-calculated Hamming code tables. The Hamming code bits of each byte of the S-Box look-up table (S_{RD}), (S_{RD} @{02}) and(S_{RD} @{03})) are pre-calculated and stored in the form of a memory table, which is referred to as Hamming code memory. The symbol 0 represents multiplication in Galois filed. The pre-calculated Hamming code tables, which are referred to as h_{RD},h_{2RD},h_{3RD}. As it can be seen from the expressions below h_{RD} is the Hamming code of the S-Box look-up table (S_{RD}), h_{2RD} is the Hamming code of (S_{RD} @{02}) and h_{3Ro} is the Hamming code of (S_{RO} @{03}).

$$\begin{aligned} h(S_{RD}[a]) &\rightarrow h_{RD}[a] \\ h(S_{RD}[a]\otimes\{02\}) &\rightarrow h_{2RD}[a] \qquad (1) \\ h(S_{RD}[a]\otimes\{03\}) &\rightarrow h_{3RD}[a] \end{aligned}$$

where [a] represents the state byte. The procedure to derive the h_{RDt} h_{2RD}, h_{3Ro} tables is described by taking one data byte as an example. Let a is a state byte represented by bits (b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_o). The Hamming code of the state byte a is a four bit party code, represented by bits (P_3, P_2, P_1, P_0), which are derived as follows:

$$\begin{aligned} &p_3 \rightarrow is\ parity\ of\ bit\ groups\ b_7, b_6, b_4, b_3, b_1 \\ &p_2 \rightarrow is\ parity\ of\ bit\ groups\ b_7, b_5, b_4, b_2, b_1 \\ &p_1 \rightarrow is\ parity\ of\ bit\ groups\ \ b_6, b_5, b_4, b_0 \qquad (2) \\ &p_0 \rightarrow is\quad arit\quad o\quad bit\quad rou\ \ s\ b\ \ , b\ \ , b\ \ , b \end{aligned}$$

The Hamming code matrix of SubBytes transformation is predicted by referring to the h_{RD} table. The Hamming code matrix prediction for ShiftRows involves simple cyclic rotation of SubByte Hamming code bits. The Hamming code matrix for MixColumns is predicted with the help of h_{RD},h_{2RD}, and h_{3Ro} parity tables and it can be expressed by the equations below:

$$\begin{aligned} &h_{0,j} \\ &= h_{2RD}[a_{0,j}] \oplus h_{3RD}[a_{1,j}] \oplus h_{RD}[a_{2,j}] \oplus h_{RD}[a_{3,j}] \\ &h_{1,j} \\ &= h_{RD}[a_{0,j}] \oplus h_{2RD}[a_{1,j}] \oplus h_{3RD}[a_{2,j}] \oplus h_{RD}[a_{3,j}] \\ &h_{2,j} \\ &= h_{RD}[a_{0,j}] \oplus h_{2RD}[a_{1,j}] \oplus h_{3RD}[a_{1,j}] \oplus h_{3RD}[a_{3,j}] \\ &h_{3,j} \\ &= h_{3RD}[a_{0,j}] \oplus h_{RD}[a_{1,j}] \oplus h_{RD}[a_{2,j}] \oplus h_{2RD}[a_{3,j}] \\ &0 \le j < 4 \end{aligned} \qquad (3)$$

As shown in Fig. 3, for each transformation, the Hamming code is predicted using the input state to the transformation by referring to Hamming code memory and also the Hamming code is calculated from the output of the transformation. The predicted and calculated Hamming codes are compared to detect and correct the fault as discussed below [12].

Let the predicted Hamming code bits of transformation input be represented by (x_3, x_2, x_1, x_0) and the calculated Hamming code of transformation output be represented by(y_3, y_2, y_1, y_0) The location of the faulty bit is detected by comparing the predicted and calculated Hamming codes following the bit match patterns in Table II. Once the faulty bit position is identified the fault correction is performed by simply flipping that bit. The encryption is then continued without any interruption to the encryption process.

TABLE II: HAMMING CODE BIT MATCH TABLE TO LOCATE A FAULTY BIT

Hamming code Bit Match	Faulty Bit Position
(X_3, Y_3) & (X_2, Y_2)	0
(X_3, Y_3) & (X_1, Y_1)	2
(X_3, Y_3) & (X_0, Y_0)	5
(X_2, Y_2) & (X_1, Y_1)	3
(X_2, Y_2) & (X_0, Y_0)	6
(X_1, Y_1) & (X_0, Y_0)	7
(X_1, Y_1)	1
(X_0, Y_0)	4

The AES fault tolerant model was verified using a purpose-built software simulator written in the JAVA programming language. The model is tested through injecting faults randomly at different round, transformation, byte and bit levels [7]. The model was tested extensively using the Known Answer Test (KAT) and Monte Carlo Test (MCT) vectors described by NIST [8], [12]. The testing has shown that the software simulator is able to detect and correct all the faults up to bit level as expected using the Hamming codes [8], [12].

In CTR mode either the SEU fault or the transmission fault propagates to only one block as in ECB mode as there is no feedback here to propagate the faults. The SEU during encryption corrupts one complete block whereas transmission fault corrupts the corresponding single bit in the data.

The architecture of the AES data path is a round implementation with look-up tables, where each round is computed within a single clock cycle. Currently, the system computes only AES with 128-bits of key, mainly to keep the key expansion schedule simple. A multiplexer controls the final output, setting the output to a null value until the final round is completed. The FPGA utilization, power and maximum frequency of operation are measured and tabulated in Table III. The encryption of 128-bit data block is computed in 12 clock cycles.

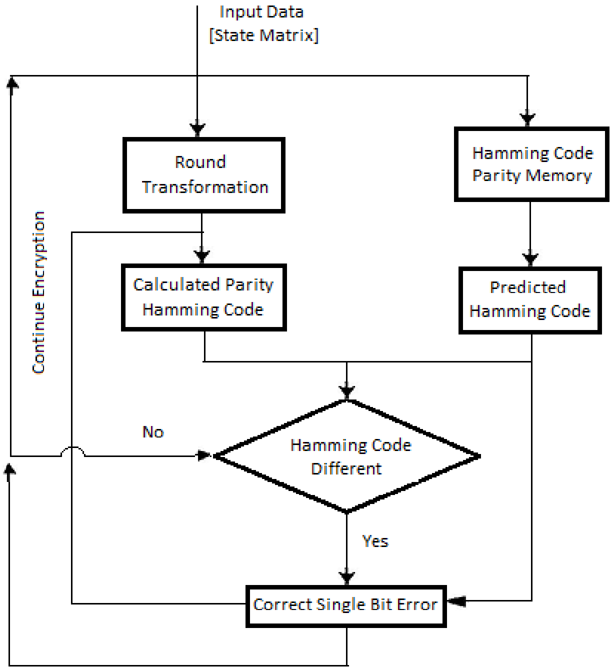

Fig. 3. Fault correction flow chart.

IV. Conclusions

Advantages and disadvantages of Popular modes of AES such as ECB, CBC, OFB, CFB and CTR have been been compared. SEUs are the most common faults that occur on-board due to radiation. The impact of SEU faults occurring during on-board encryption has been analyzed. In addition, an analysis of faults that occur during transmission due to noise has been carried out, as satellite channels are very noisy. The CTR mode has been recommended as the optimum choice for satellite applications. Since none of the AES modes are free from faults, error detection and correction is very important in satellites in order to prevent faulty data transmissions.

In this paper our proposed model for fault detection and correction of a single bit fault in AES is illustrated using the Hamming code. Also FPGA implementation is carried out to calculate the area overhead of the proposed fault correction model. The FPGA hardware overhead is 38.25% and power overhead is 136%. Although high, this overhead is far less than the triple modular redundancy technique. The model can be extended for detection and correction of multiple bit faults by using sophisticated error correction codes such as modified Hamming code, Reed-Solomon codes etc. The fault detection and correction model targets the satellite application domain, however it can also be used in other areas of industry that deal with harsh radiation environments such as unmanned aerial vehicles and aeronautical, medical, military, underwater and offshore, nuclear industry etc.

TABLE III: FPGA Area Overhead of AES Fault Correction Implementation.

	FPGA	power (mW)	f_{max}
AES	1226 slices	1130	80 MHz
Fault correction	1695 slices	2670	72 MHz
Overhead	38.25 %	136 %	10 %

Acknowledgements

This work was supported in part by Aerospace Research Institute depends on Islamic Azad University, Boushehr Branch.

References

[1] T. Vladimirova, R. Banu, and M. N. Sweeting, “On-board encryption in satellites,” *Proceedings of the 8th Military and Aerospace Applications of Programmable Logic Devices and Technologies International Conference (MAPLD'2005)*, Washington DC, US, NASA, vol. 184, September 2005.

[2] J. Daemen and R. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. 1St Edition, Spriger-Verlag publication, 2002.

[3] W. Sun, P. Stephens, and M. N. Sweeting, “Micro-minisatellites for affordable EO constellations – rapid eye and DMC,” *Proceedings of the IAA Symposium on Small Satellites for Earth Observation*, Berlin, IAA-B3-0603, April 2001

[4] W. E. Burr, “ Selecting the advanced encryption standard,” *Security and Privacy Magazine, IEEE*, vol. 1, no. 2, pp. 43 - 52, Mar-Apr 2003.

[5] K. Sweet, “The increasing threat to satellite communications,” *Online Journal of Space Communication*, vol. 6, November 2003

[6] G. Bertoni, L. Breveglieri, I. Koren, P. Maistri, and V. Piuri, “Error analysis and detection procedures for a hardware implementation of the AES,” *IEEE Transactions on Computers*, April 2003.

[7] K. Sweet, “The lncreasing threat to satellite communications,” *Online Joumal of Space Communication*, vol. 6, November 2003

[8] W. Sun, P. Stephens, and M. N. Sweeting, “Micro-minisatellites for affordable EO constellations –rapid eye and DMC,” *Proceedings of the IAA Symposium on Small Satellites for Earth Observation*, erlin, IAA-B3-0603, April 2001

[9] J. Daemen and R. Rijmen, *The Design of Rijndael: AES, The Advanced Encryption Standard*, 1st Edition, Spriger-Verlag publication, 2002.

[10] P. Zhang, “High-Speed VLSI architecture for the AES algorithm,” *IEEE Transaction on VLSI*, September 2004.

[11] C. H. Yen and B. F. Wu, “Simple error detection methods for hardware implementation of advanced encryption standard,” *IEEE Transactions on Computers*, vol. 55, no.6, June 2006.