ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

An FPGA Implementation of Fault Diagnosis Architecture of S - Box For Cryptographic Application

Freeda Jancy. L

PG Student, Dept. of VLSI Design, MAM College of Engineering and Technology, Tiruchirappalli, Tamil Nadu, India

ABSTRACT: Efficient cryptographic architectures are used extensively in sensitive smart infrastructures. Nevertheless, usual defects call for defence through design for fault detection and reliability. In this paper, we present implications of fault detection cryptographic architectures for smart infrastructures. In addition, we present new architectures for its nine-to-seven uneven substitution box. Through error simulations, we assess resiliency against false-alarms which might not be tolerated in sensitive intelligent infrastructures as one of our contributions. We further benchmark the feasibility of the proposed approaches through Field Programmable Gate Array realizations. Based on the reliability objectives, the proposed architectures are a stepforward toward reaching the desired objective metrics suitable for intelligent, emerging, and sensitive applications. The Proposed nine-to-seven uneven substitution box is done by Verilog HDL and Simulated by Modelsim 6.4 c and Synthesized by Xilinx tool and proposed system implemented in FPGA Spartan 3 XC3S 200 TQ-144.

KEYWORDS: Application-specific integrated circuit (ASIC), reliability, smart infrastructures**.**

I. INTRODUCTION

Cryptography, often called encryption, is the practice of creating and using a cryptosystem or cipher to prevent all but the intended recipient(s) from reading or using the information or application encrypted. A cryptosystem is a technique used to encode a message. The recipient can view the encrypted message only by decoding it with the correct algorithm and keys. Cryptography is used primarily for communicating sensitive material across computer networks. The process of encryption takes a clear-text document and applies a key and a mathematical algorithm to it, converting it into crypto-text. In crypto-text, the document is unreadable unless the reader possesses the key that can undo the encryption. In 1997 the National Institute of Standards and Technology (NIST), a branch of the US government, started a process to identify a replacement for the Data Encryption Standard (DES). It was generally recognized that DES was not secure because of advances in computer processing power. The goal of NIST was to define a replacement for DES that could be used for non-military information security applications by US government agencies.

Of course, it was recognized that commercial and other non-government users would benefit from the work of NIST and that the work would be generally adopted as a commercial standard. The NIST invited cryptography and data security specialists from around the world to participate in the discussion and selection process. Five encryption algorithms were adopted for study. Through a process of consensus the encryption algorithm proposed by the Belgium cryptographers Joan Daeman and Vincent Rijmen was selected. Prior to selection Daeman and Rijmen used the name Rijndael (derived from their names) for the algorithm.

AES ALGORITHM

The AES encryption algorithm is a block cipher that uses an encryption key and a several rounds of encryption. A block cipher is an encryption algorithm that works on a single block of data at a time. In the case of standard AES encryption the block is 128 bits, or 16 bytes, in length. The term ―rounds‖ refers to the way in which the encryption algorithm mixes the data re-encrypting it ten to fourteen times depending on the length of the key. This is described in the Wikipedia article on AES encryption. The AES algorithm itself is not a computer program or computer

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

source code. It is a mathematical description of a process of obscuring data. A number of people have created source code implementations of AES encryption, including the original authors.

AES encryption uses a single key as a part of the encryption process. The key can be 128 bits (16 bytes), 192 bits (24 bytes), or 256 bits (32 bytes) in length. The term 128-bit encryption refers to the use of a 128-bit encryption key. With AES both the encryption and the decryption are performed using the same key. This is called a symmetric encryption algorithm. Encryption algorithms that use two different keys, a public and a private key, are called asymmetric encryption algorithms. An encryption key is simply a binary string of data used in the encryption process. Because the same encryption key is used to encrypt and decrypt data, it is important to keep the encryption key a secret and to use keys that are hard to guess. Some keys are generated by software used for this specific task. Another method is to derive a key from a pass phrase. Good encryption systems never use a pass phrase alone as an encryption key.
Side channel Attacks are attacks on the implementation of AES, not on the input or the AES cipher text.

It attempts to correlate various measurements of the encrypting tool with time in an attempt to guess the key. A professor at MIT,9 encoded an AES algorithm on his computer, an 850MHz, Pentium III running FreeBSD 4.8 and by measuring time delays between the CPU and memory was able to successfully guess the key in under 100 minutes. There is a correlation between the index of an array and the time it takes to get the results back. This is due to the physical location of the data in the memory device. Data closer to the output lead will not take as much time to be retrieved as data further away, because it will not take as long for the signal to propagate its way out of the chip.

He feels he can improve on this time. After running a few thousand encryptions he spent about an hour studying the results of his measurements. After studying the data, there were many repetitions to avoid noise, he concluded the key was one of several possibilities. By trying each one, he was able to find the key. He believes this analysis process can be programmed, cutting the time down to just a few minutes. The method of measuring time delays in memory requests are called timing attacks. Power attacks attempt to measure power consumption by the CPU. It takes more power to switch 8 bits than it takes to switch 1 bit. Some are also now measuring radiation levels from CPU's and gaining knowledge of its inner workings.

There are several techniques which can greatly frustrate side channel attacks. 1) Avoid use of arrays. Compute values of SBOX and RCon to avoid timing attacks. 2) Design algorithms and devices to work with constant time intervals. (independent of key and plaintext.) Study your device spec sheets, and insist on accurate data. For example you should know which takes longer, XOR or shift operations. 3) Use same memory throughout, remember, cache is faster than DRAM. 4) Compute Key Expansion on the fly. Don't compute the Key Expansion and then reference it from memory. 5) Utilize pipelining to stabilize CPU power consumption. 6) Use specialized chips whenever possible, right now they are significantly faster than CPU's and require extremely expensive equipment for side channel attack measurements. Rijindael was designed to have the following characteristics:

- Resistance against all known attacks.
- Speed and code compactness on a wide range of platforms.

DESIGN SIMPLICITY

The algorithm begins with an Add round key stage followed by 9 rounds of four stages and a tenth round of three stages. This applies for both encryption and decryption with the exception that each stage of a round the decryption algorithm is the inverse of it's counterpart in the encryption algorithm.

The four stages are as follows:

1. Substitute bytes
2. Shift rows
3. Mix Columns
4. Add Round Key

The tenth round simply leaves out the Mix Columns stage. The first nine rounds of the decryption algorithm consist of the following:

1. Inverse Shift rows

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

2. Inverse Substitute bytes
3. Inverse Add Round Key
4. Inverse Mix Columns

The Rijndael algorithm is a symmetric iterated block cipher. The block and key lengths can be 128, 192, or 256 bits. The NIST requested that the AES must implement a symmetric block cipher with a block size of 128 bits. Due to this requirement, variations of Rijndael that can operate on larger block sizes will not be included in the actual standard. Rijndael also has a variable number of iterations or rounds: 10, 12, and 14 when the key lengths are 128, 192, and 256 respectively. The transformations in Rijndael consider the data block as a four-column rectangular array of 4-byte vectors. The key is also considered to be a rectangular array of 4-byte vectors—the number of columns is dependent on key length.

II. RELATED WORK

In 2013, M. Mozaffari-Kermani and R. Azarderakhsh ,"Efficient fault diagnosis schemes for reliable lightweight cryptographic ISO/IEC standard CLEFIA benchmarked on ASIC and FPGA", Lightweight block ciphers are essential for providing low-cost confidentiality to sensitive constrained applications. Nonetheless, this confidentiality does not guarantee their reliability in the presence of natural and malicious faults. In this paper, fault diagnosis schemes for the lightweight internationally standardized block cipher CLEFIA are proposed. This symmetric key cipher is compatible with yet lighter in hardware than the Advanced Encryption Standard and enables the implementation of cryptographic functionality with low complexity and power consumption. To the best of the authors' knowledge, there has been no fault diagnosis scheme presented in the literature for the CLEFIA to date. Disadvandage is lookup table realizations are very difficult to analysis for Area and Delay Calculation.

In 2011, M. Mozaffari-Kermani and A. Reyhani-Masoleh. "A lightweight high performance fault detection scheme for the Advanced Encryption Standard using composite fields", lightweight concurrent fault detection scheme for the AES. In the proposed approach, the composite field S-box and inverse S-box are divided into blocks and the predicted parities of these blocks are obtained. Through exhaustive searches among all available composite fields, we have found the optimum solutions for the least overhead parity based fault detection structures. Disadvandage is Have more hardware and time complexities compared to their counterparts.

In 2011, M. Mozaffari-Kermani and A. Reyhani-Masoleh. "A low-power high performance concurrent fault detection approach for the composite field Sbox and inverse S-box", a concurrent fault detection scheme for the S-box and the inverse S-box as the only two nonlinear operations within the Advanced Encryption Standard. The proposed parity-based fault detection approach is based on the lowcost composite field implementations of the S-box and the inverse S-box. We divide the structures of these operations into three blocks and find the predicted parities of these blocks. Disadvantages are here dividing the structures of these operations into three blocks and find the predicted parities of these blocks. So the Delay will be increased.

In 2010, M. Mozaffari-Kermani and A. Reyhani-Masoleh. "Concurrent structure independent fault detection schemes for the Advanced Encryption Standard" , the study concurrent fault detection schemes for reaching a reliable AES architecture. Specifically, we propose low-cost structure-independent fault detection schemes for the AES encryption and decryption. We have obtained new formulations for the fault detection of Sub Bytes and inverse Sub Bytes using the relation between the input and the output of the S-box and the inverse S-box. The proposed schemes are independent of the way the S-box and the inverse S-box are constructed. Therefore, they can be used for both the S-boxes and the inverse S-boxes using lookup tables and those utilizing logic gates based on composite fields. Disadvandage is the error coverage of lesser than the proposed schemes, Their area and delay overheads have been high.

In 2008, A. Satoh, T. Sugawara, N. Homma, and T. Aoki. "Highperformance concurrent error detection scheme for AES hardware", efficient concurrent error detection scheme for hardware implementation of the block cipher AES. The proposed scheme does not require an additional arithmetic unit, but simply divides the round function block into two sub-blocks and uses the sub blocks alternately for encryption (or decryption) and error detection. The number of clock cycles is doubled, but the maximum operating frequency is increased owing to the shortened critical path of the sub-

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

block. Therefore, the proposed scheme has a limited impact on hardware performance with respect to size and speed. AES hardware with the proposed scheme was designed and synthesized using a 90-nm CMOS standard cell library with size and speed optimization options. Disadvandage is Pipeline operation is allowed so speed and was estimated to maximum.

In 2005, C. J. A. Jansen, T. Helleseth, and A. Kholosha. "Cascade jump controlled sequence generator (CJCSG)", Jumping LFSRs have recently been proposed as building blocks for stream ciphers. The proposed encryption primitive is the key stream generator part of a synchronous stream cipher accommodating a key of 128 bits and an IV of 64 up till 112 bits. A number of cryptanalytic attacks are considered, leading to the conclusion that the proposed primitive has a high resistance against those attacks. Disadvandage is the key stream generator is particularly designed to resist side-channel attacks but The Design is affected, if the Continues Side Channel attacks occurred. Less efficiency for hardware and software implementation.

III. PROPOSED SCHEME

Low-power architectures for the substitution box of the Pomaranch stream cipher. The proposed structures are based on tower field architectures of this substitution box. Specifically, we present low-power restructured architectures for this uneven substitution box useful for emerging constrained and sensitive usage models. Fault diagnosis approaches for the lightweight and low-power architectures of the nine-to-seven substitution box of Pomaranch. The proposed work presents false-alarm sensitive fault detection schemes for crypto structures. Such false alarms could be exploited to induce distrust to the user. We present implications of fault detection cryptographic architectures for smart infrastructures. In addition, we present low-power architectures for its nine-to-seven uneven substitution box.
Through error simulations, we assess resiliency against false-alarms which might not be tolerated in sensitive intelligent infrastructures as one of our contributions. We further benchmark the feasibility of the proposed approaches through application-specific integrated circuit realizations.

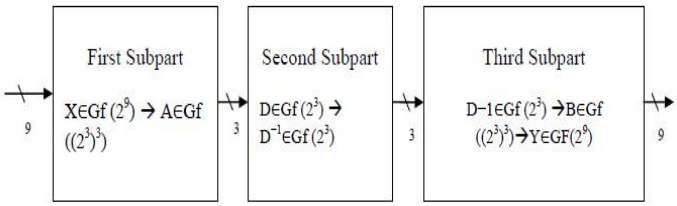

Fig.1 S-BOX Block Diagram

IV. EXPERIMENTAL RESULTS

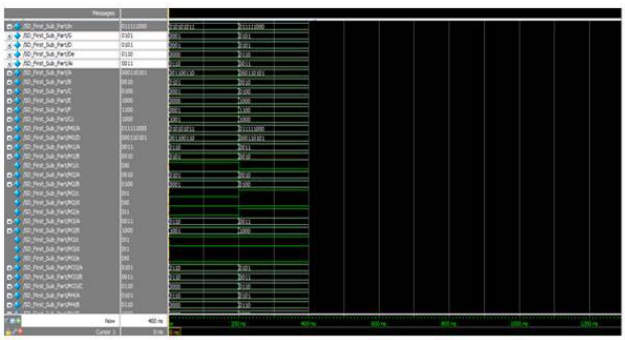

Fig .2. Output of this subpart, i.e., D∈GF (23)

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

Fig.3. Output of this subpart, i.e., D-1∈GF (23)

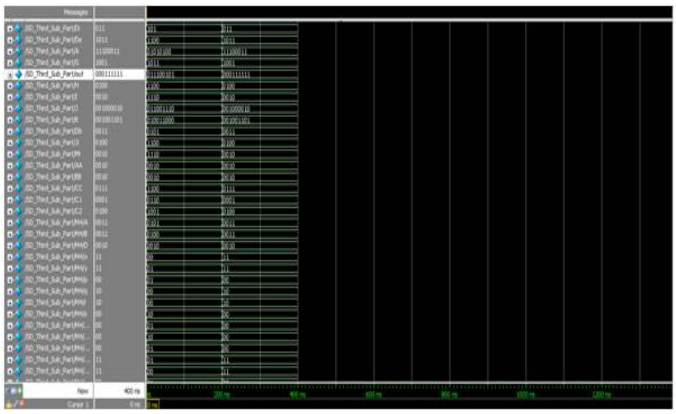

Fig.4. Output of this subpart, i.e., Y∈GF (29)

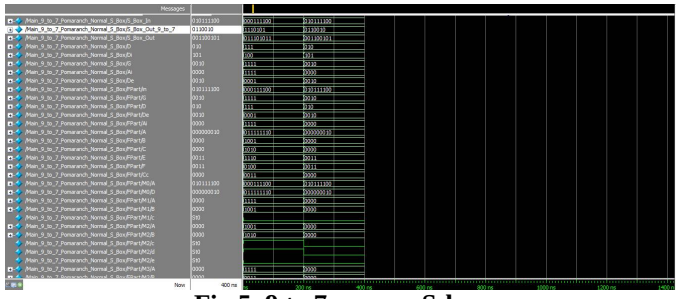

Fig.5. 9 to 7 uneven S-box

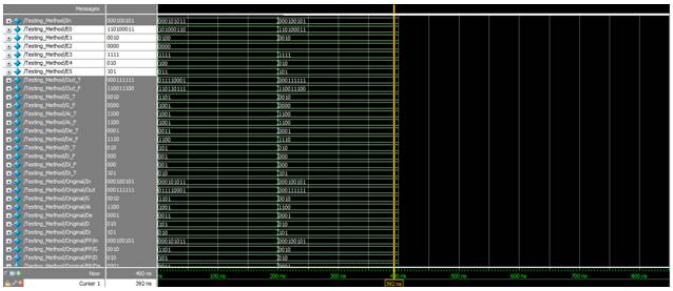

Fig.6. Fault Detection

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

V. CONCLUSION AND FUTURE WORK

Reliability of sensitive cryptographic application is benchmarked through a case study, i.e., the uneven substitution box of a stream cipher, to elaborate on the respective effects on smart infrastructures. We have presented low-power architectures for this stream cipher and then proposed a framework to provide fault immunity for infrastructures that need to deal with sensitive information and are smart and ubiquitous. The proposed architectures are benchmarked in terms of error coverage for different fault models and assessed for false-alarm immunity. Moreover, they have been synthesized on an ASIC platform and it is shown that with an acceptable overhead, high error coverage can be achieved for the proposed architectures. Furthermore, we have assessed the benefits and effects of such architectures for smart infrastructures. The benchmark details the smart infrastructure implications and elaborates on the fact that using the proposed framework, smart infrastructures can be more efficiently and reliably utilized. We can implement complete Encryption and Decryption process with using our proposed substitution box and inverse substitution box. And also we will encode the input based on the genetic algorithm.

REFERENCES

1. A.Satoh, T. Sugawara, N. Homma, and T. Aoki, ―High-performance concurrent error detection scheme for AES hardware,‖ in Proc. 10th Int. Workshop CHES, Aug. 2008, pp. 100–112.

2. C. J. A. Jansen, T. Helleseth, and A. Kholosha, ―Cascade jump controlled sequence generator and Pomaranch stream cipher (version 3),‖ Dept. Informat., Univ. Bergen, Bergen, Norway, Tech. Rep. 2006/006, 2006.

3. C. J. A. Jansen, T. Helleseth, and A. Kholosha, ―Cascade jump controlled sequence generator (CJCSG),‖ in Proc. Workshop Symmetric Key Encryption, 2005, pp.1–16.

4. D. Halperin, T. Kohno, T. S. Heydt-Benjamin, K. Fu, and W. H. Maisel, ―Security and privacy for implantable medical devices,‖ IEEE Pervasive Comput., vol. 7, no. 1, pp. 30–39, Jan./Mar. 2008.

5. H. Khurana, M. Hadley, N. Lu, and D. A. Frincke, ―Smart-grid security issues,‖ IEEE Security Privacy, vol. 8, no. 1, pp. 81–85, Jan./Feb. 2010.

6. K. Fu and J. Blum, ―Controlling for cybersecurity risks of medical device software,‖ Commun. ACM, vol. 56, no. 10, pp. 35–37, Oct. 2013.

7. M. Mozaffari-Kermani and R. Azarderakhsh, ―Efficient fault diagnosis schemes for reliable lightweight cryptographic ISO/IEC standard CLEFIA benchmarked on ASIC and FPGA,‖ IEEE Trans. Ind. Electron., vol. 60, no. 12, pp. 5925–5932, Dec. 2013.

8. M. Mozaffari-Kermani, M. Zhang, A. Raghunathan, and N. K. Jha, ―Emerging frontiers in embedded security,‖ in Proc. 26th Int. Conf. VLSI Design, Jan. 2013, pp. 203–208.

9. M. Mozaffari-Kermani and A. Reyhani-Masoleh, ―A low-power high performance concurrent fault detection approach for the composite field S-box and inverse S-box,‖ IEEE Trans. Comput., vol. 60, no. 9, pp. 132 , Sep. 2011.

10. M. Mozaffari-Kermani and A. Reyhani-Masoleh, ―A lightweight high performance fault detection scheme for the Advanced Encryption Standard using composite fields,‖ IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 1, pp. 85–91, Jan. 2011.