Volume 6, Issue 5, May 2016 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Software Implementation of AES Encryption Algorithm

Abhilasha CP
Electronics and Communication Under VTU,
Bangalore, Karnataka, India

Nataraj KR
Professor and Head of ECE Department,
Bangalore, Karnataka, India

Abstract: AES represents an algorithm for advanced encryption standard consist of different operations required in the steps of encryption and decryption. The proposed architecture is based on optimizing area in terms of reducing no of slices required for design of AES algorithm .This paper produces 3 step designs. AES (TOP), AES (1- 9ROUNDS), AES (LAST ROUND) in which code is divided in to three parts instead of 4 groups in single round. This paper presents AES-128 bit algorithm design consist of 128 bit symmetric key & Xilinx ISE 14.1 Project Navigator used for synthesis and simulation of this proposed architecture purpose.

Key words: symmetric, key length, cipher text, plain text and ISE14.1

I. INTRODUCTION

Each day millions of users generate and interchange large volumes of information in various fields, such as financial and legal files, medical reports, and bank services via Internet. These and other examples of applications deserve a special treatment from the security point of view, not only in the transport of such information but also in its storage. In this sense, cryptography techniques are especially applicable. For a long time, theData Encryption Standard (DES) was considered as a standard for the symmetric key encryption. DES has a key length of56 bits. However, this key length is currently considered small and can easily be broken. For this reason, the National Institute of Standards and Technology (NIST) opened a formal call for algorithms in September 1997. A group of fifteen

AES candidate algorithms were announced in August 1998. Next, algorithms were subject to assessment process performed by various groups of cryptographic researchers all over the world. In August 2000, NIST selected five algorithms: Mars, RC6, Rijndael, Serpent and Twofish as the final competitors. These algorithms were subject to further analysis prior to the selection of the best algorithm for the AES. Finally, on October 2, 2000, NIST announced that the Rijndael algorithm was the winner. Field Programmable Gate Arrays (FPGAs) are hardware devices whose function is not fixed which can be programmed in system.

Cryptographic algorithm during operation. Algorithm upload- It is perceivable that fielded devices upgraded with new encryption algorithm which did not exist at design time.

II. RELATED WORK

Next design, 8-bit embedded microcontroller similarity with the Xilinx PicoBlaze soft core. The main objective of the research is to find out the design area for AES algorithm in FPGA. AES also known as Rijndael, because it was developed by Rijmen and daemen. This algorithm is recognized in the freely available US government newspaper, FIPS-197.

AES is taken by most them as research paper to find out the correct architecture to implement on the hardware. Speed and resource requirement are the mainly considered to select the architecture for the application. In this area and speed considered but other important parameter is delay. There are number of operation methods were there for FIPS-197. One of the method is ECB. Additional flexibility to attack can be obtained by using one of the modes. Next Output Feed Back (OFB) mode appropriately such mode, also restricts the efficiency of pipelining. FPGA role is improving from prototyping level to main stream production level. This drastic change is due to the high pressure to decrease design cost, less time to market and risk. Due to growth in technology, results in various version of FPGA by the leading manufacturer. In some of the application made the user to move from FPGA to ASIC (application specific integrated circuits), it gives low cost for user.

In this found another way to specific FPGA, which gives output in optimal level, results in high performance. New design reported have output at the rate of 25 Gbps on a Spartan-III FPGA Xilinx, having application in hardware acceleration for servers. In this paper, they presented an extra inventiveness of the fresh high speed design without any loss of key may change and between encryption and decryption it may change.

This makes the design to support different mode of operation where blocks batch can be encrypted or decrypted for concurrent channels with different keys without losing any output. 32 bit of data path was there for low area architecture. Same way 32bit of mix-column and key-expansions are the only way for AES encryption. Next felhofer al used ASIC design with 8bit data path married to 32bit mix-column.

Still, even Mix-Columns can rewrite in form of 8bit, uncomplaining a higher control overhead and decreased output. From the author’s literature knowledge, no 8 bit ASIP (application specific integrated processor) was developed for AES. From such processor design, which is supposed to be the smallest, are mentioned in this paper. For about 60% were occupied by the smallest available Spartan-II Xilinx device and attains an output of 2.2 Mbps which is related to numerous requests in the mobile and home communications. Latest 32 bit FPGA design was compare with 8bit design. This Comparison may include cost, area of design.

Two different FPGA designs were announced for the Advanced Encryption Standard (AES). The first is supposed to be the fastest, accomplishing 25 Gbps output using a Xilinx Spartan-III (XC3S2000) device. Next one is assumed to be the slightest and fits into a Xilinx Spartan-II (XC2S15) device, only lacking between two is block memories and slices-124 to achieve an amount of 2.2 Mbps. These proposals show the extravagances of what is possible and application with radically different from e-commerce IPs high speed servers to low power home application.

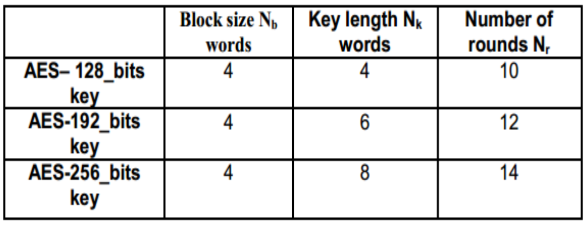
Design with high speed here covers support for output during change in key for decryption and encryption; they were removing the older pipeline design. Objective of the research id to find the design with AES (advanced encryption standard) algorithm with FPGA (field programmable gate array) for the hardware implementation. AES also known as Rijndael, because it was developed by Rijmen and daemen. This algorithm is recognized in the freely available US government newspaper. The Rijndael, in 2000 tis selected as new commercial cryptographic algorithm and was rendered the compliment the Advanced Encryption Standard (AES). AES is taken by most them as research paper to find out the correct architecture to implement on the hardware. Speed and resource requirement are the mainly considered to select the architecture for the application. In this area and speed considered but other important parameter is delay. There are number of operation methods were there for FIPS-197. One of the methods is ECB. Additional flexibility to attack can be obtained by using one of the modes. Next Output Feed Back (OFB) mode appropriately such mode, also restricts the efficiency of pipelining. FPGA role is improving from prototyping level to main stream production level. This drastic change is due to the high pressure to decrease design cost, less time to market and risk. Due to growth in technology, results in various version of FPGA by the leading manufacturer. In some of the application made the user to move from FPGA to ASIC (application specific integrated circuits), it gives low cost for user.

III. INPUTS, OUTPUTS AND THE STATE

The plaintext input and cipher text output for the AES algorithms are blocks of 128 bits. The cipher key input is a sequence of 128, 192 or 256 bits. In other words the length of the cipher key, Nk, is 4, 6 or 8 words which represent the number of columns in the cipher key.

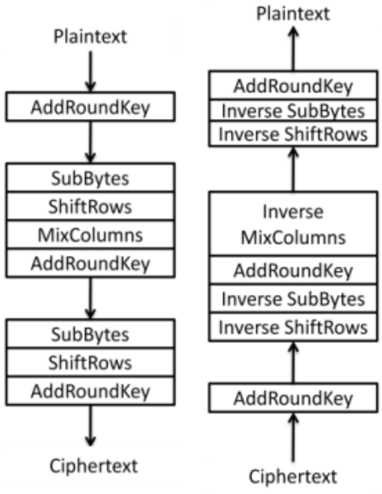
The AES algorithm is categorized into three versions based on the cipher key length. The number of rounds of encryption for each AES version depends on the cipher key size. In the AES algorithm, the number of rounds is represented by *Nr*, where *Nr* = 10 when *Nk* = 4, *Nr* = 12 when *Nk* = 6, and *Nr* = 14 when *Nk* = 8. The following table

FEATURES OF AES FOR DIFFERENT KEY LENGTHS



	Block size N_b words	Key length N_k words	Number of rounds N_r
AES– 128_bits key	4	4	10
AES-192_bits key	4	6	12
AES-256_bits key	4	8	14

The table gives the description about the different key sizes AES algorithm. As the key size increases the security for the data increases.



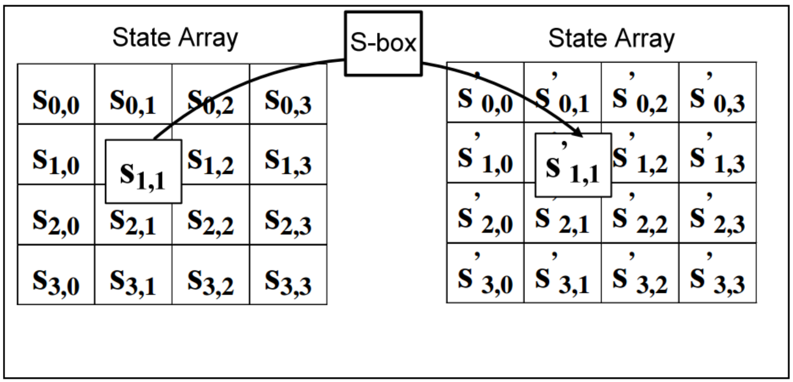
(a) Encryption process (b) Decryption process

Encryption and the decryption flow chart shown above. This gives cipher text at encryption output and plain text back at the decryption.

IV. ADD ROUND KEY AND SUB BYTES TRANSFORMATION

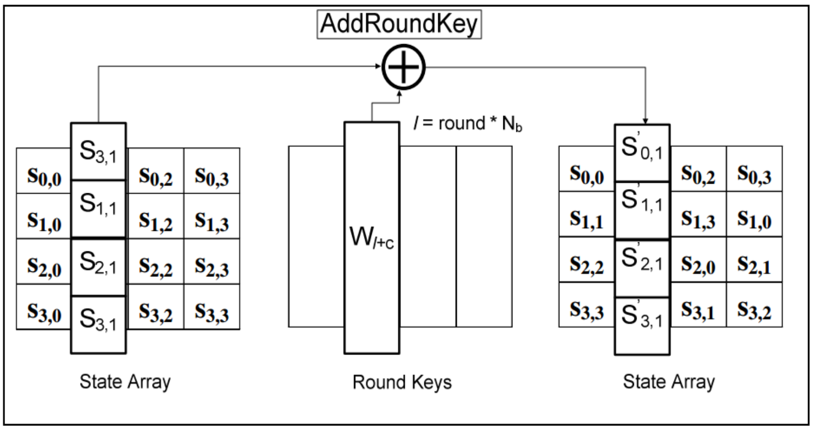
The *Sub Bytes* is a byte substitution operation performed on individual bytes of the *State*, as shown in Figure 3, using a substitution table called S-box.

Transformations during the Add Round Key transformation, the round key values are added to the *State* by means of a simple *Exclusive or*(XOR) operation. Each round key consists of Nb words that are generated from the Key Expansion routine. The round key values are added to the columns of the state in the following way:


State Array
S-box
State Array
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s'_{0,0}$ $s'_{0,1}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{1,1}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,1}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,1}$ $s'_{3,2}$ $s'_{3,3}$

SubBytes Transformation

Above figure shows how the sub byte operation performed on the given data. AES have sequence of data transformation. One of that is sub byte transformation.

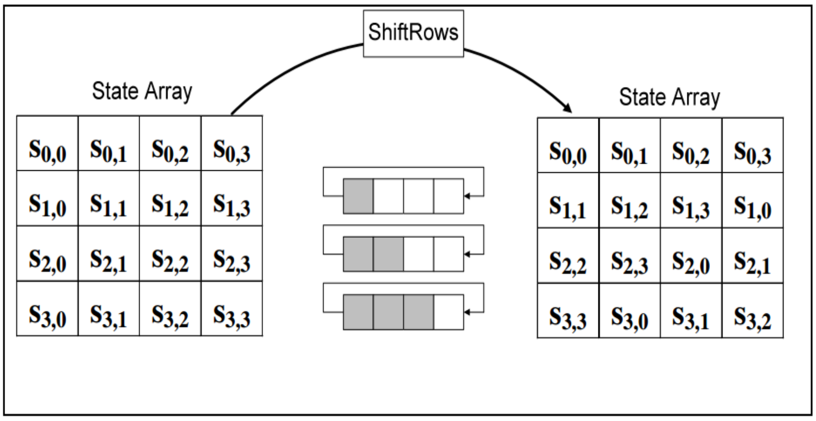

AddRoundKey
l = round * N_b
$s_{0,0}$ $S_{3,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $S_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $S_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $S_{3,1}$ $s_{3,2}$ $s_{3,3}$
W_{l+c}
$s_{0,0}$ $S'_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,1}$ $S'_{1,1}$ $s_{1,3}$ $s_{1,0}$
$s_{2,2}$ $S'_{2,1}$ $s_{2,0}$ $s_{2,1}$
$s_{3,3}$ $S'_{3,1}$ $s_{3,1}$ $s_{3,2}$
State Array
Round Keys
State Array

AddRoundKey Transformation

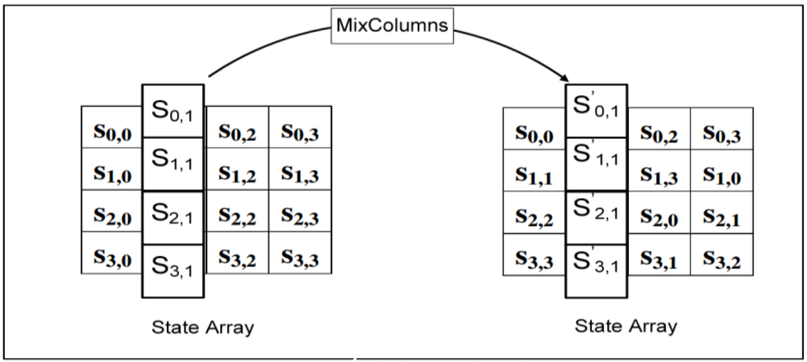
Above figure shows how the add round key operation performed on the given data. AES have sequence of data transformation. One of that is add round key transformation.

V. SHIFT ROWS AND MIX COLUMN TRANSFORMATION

The *Shift Rows* transformation cyclically shifts the last three rows of the state by different offsets. The first row is left unchanged in this transformation. Each byte of the second row is shifted one position to the left. The third and fourth rows are shifted left by two and three positions, respectively. The *Shift Rows* transformation is illustrated in Figure


ShiftRows
State Array
State Array
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,1}$ $s_{1,2}$ $s_{1,3}$ $s_{1,0}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,3}$ $s_{3,0}$ $s_{3,1}$ $s_{3,2}$

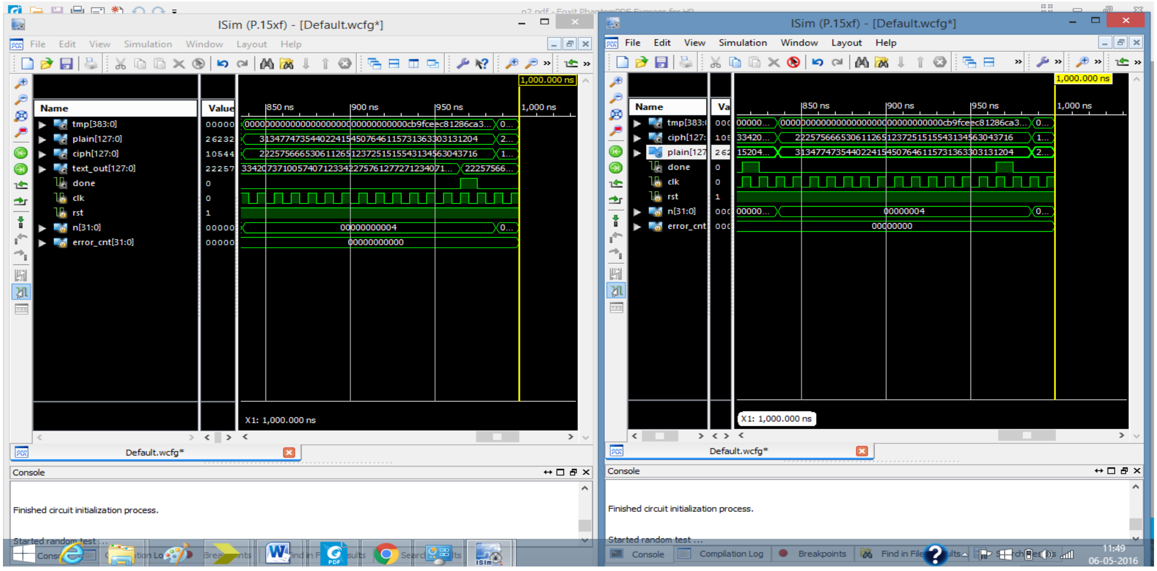
Above figure shows how the shift row operation performed on the given data. AES have sequence of data transformation. One of that is shift row transformation.


MixColumns
$S_{0,0}$ $S_{0,1}$ $S_{0,2}$ $S_{0,3}$
$S_{1,0}$ $S_{1,1}$ $S_{1,2}$ $S_{1,3}$
$S_{2,0}$ $S_{2,1}$ $S_{2,2}$ $S_{2,3}$
$S_{3,0}$ $S_{3,1}$ $S_{3,2}$ $S_{3,3}$
State Array
$S_{0,0}$ $S'_{0,1}$ $S_{0,2}$ $S_{0,3}$
$S_{1,1}$ $S'_{1,1}$ $S_{1,3}$ $S_{1,0}$
$S_{2,2}$ $S'_{2,1}$ $S_{2,0}$ $S_{2,1}$
$S_{3,3}$ $S'_{3,1}$ $S_{3,1}$ $S_{3,2}$
State Array

MixColumns Transformation

Above figure shows how the mix column operation performed on the given data. AES have sequence of data transformation. One of that is mix column transformation.

VI. RESULT


ISim (P.15xf) - [Default.wcfg*]
File Edit View Simulation Window Layout Help
1,000.000 ns
Name
Value
850 ns
900 ns
950 ns
1,000 ns
tmp[383:0] 00000 000000000000000000000000000000000cb9fceec81286ca3... 0...
plain[127:0] 26232 313477473544022415450764611573136303131204 2...
ciph[127:0] 10544 2225756665306112651237251515543134563043716 1...
text_out[127:0] 22257 3342073710057407123342275761277271234071... 22257566...
done 0
clk 0
rst 1
n[31:0] 00000 00000000004 0...
error_cnt[31:0] 00000 00000000000
X1: 1,000.000 ns
Default.wcfg*
Console
ISim (P.15xf) - [Default.wcfg*]
File Edit View Simulation Window Layout Help
1,000.000 ns
Name
Va
tmp[383: 000 00000... 000000000000000000000000000000000cb9fceec81286ca3... 0...
ciph[127: 105 33420... 2225756665306112651237251515543134563043716 1...
plain[127 262 15204... 313477473544022415450764611573136303131204 2...
done 0
clk 0
rst 1
n[31:0] 000 00000... 00000004 0...
error_cnt 000 00000000
X1: 1,000.000 ns
Default.wcfg*
Console
Console
Compilation Log
Breakpoints
Find in Files
11:49
06-05-2016

```
Finished circuit initialization process.
Started random test ...
```

Above simulation shows the output for both encryption and the decryption out put..

VII. FUTURE WORK

The objective of this paper was to present the hardware implementation of Advanced Encryption Standard (AES) algorithm. The importance of the Advanced Encryption Standard and the significance of area-throughput balanced implementations of the Rijindael have examined. We have worked with an iterative structure and modifications such as merging of Subbytes and ShiftRows, Look Up tables for decryption, and optimization of each clock cycle to incorporate maximum number of operations etc. have been successfully implemented. The encryption and decryption process of Rijndael algorithm was captured in VHDL language and corresponding FPGA implementation resulted in reduced number of slices and achieved a data throughput of 1.4 Gbit/sec. The combination of security, and high-speed implementation and marginal silicon area makes it a very good choice for wireless systems.

VIII. CONCLUSION

The paper discussed the studies on the advancement of cryptanalysis research on AES. It aimed at identifying specific vulnerabilities and threats against the communication application in the sensitive domain. The threat model of the sensitive presents quite highly equipped opponent and lot more critical conditions faced against the opponent in comparison to any commercial domain. We demonstrated that that progress is being in the field of cryptanalysis research against AES and it requires a great deal of caution as the major work is being carried out in the public domain. Vulnerabilities of AES against the different side channel attack were also discussed. Yet, if the available countermeasures are applied properly then the weaknesses can be negated at the hardware level. Steady progress is also seen in the alternate techniques like hybrid attack and algebraic attack etc., yet no reported breakthroughs are available. On the basis of the above we can say that the AES will not have the standard life span, which is expected out of an algorithmic suit that has obtained approval for applications of the classified domain. As aresult it can be stated that AES is not appropriate for beingused in the strategic applications that have been classified. But Programmed cryptography is employed at the hardware in the strategic communication equipment's. If there is a breakthrough in the public domain, a secure algorithm can be developed relatively faster and the length of vulnerability timeframe would be more dependent on logistic aspects rather than the technical aspects. However, the plan to handle such an inevitable situation is required.

REFERENCE

[1] “Real-time Efficient FPGA Implementation of AES Algorithm”, by El Maraghy M, Hesham S and Abd El Ghany M.A, IEEE International SOC Conference Sept 2013.

[2] “Evaluation Of Microblaze and Implementation Of AES Algorithm using Spartan-3E”, byM.Sambasiva Reddy and Mr.Y.Amar Babu, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 2, July 2013.

[3] “An Efficient FPGA Implementation of The Advanced Encryption Standard algorithm”, by Hoang Trang and Nguyen Van Loi, IEEE International Conference on Computing and Communication Technology, page 1 -4, Ho Chi Minh city, 2012.

[4] “A new moodier version of Advanced Encryption Standard based algorithm for image encryption”, by Kamali S.H, Shakerian R, Hedayati M and Rahmani M (ICEIE) International Conference on Electronics and Information Engineering, volume 1, Aug 2010.

[5] IEEE Symposium on Industrial Electronics & Applications, Oct 2010. Ahmad N, Hasan R and Jubadi W.M, “Design of AES Sbox using combinational logic optimization”.

[6] M. Zeghid, M. Machhout, L. Khriji, A. Baganne, and R. Tourki, International Journal of Computer, Information, Systems and Control Engineering, 2007 “A Modified AES Based Algorithm for Image Encryption”. .

[7] FIPS 197, "Advanced Encryption Standard (AES)," November 26, 2001.

[8] "Data Encryption Standard (DES)," National Technical Information Service VA 22161, 1999.National Institute of Standards and Technology (NIST).