DESIGN OF AES ON FPGA HARDWARE

[1]SHRUTI SHRIVASTAVA, [2]A.Y.KAZI

[1]P.G. Student, Department of Electronics Engineering, AISSMS'S COE, Pune Univeristy, Pune, Maharashtra (India)
[2]Department of Electronics Engineering, AISSMS'S COE, Pune Univeristy, Pune, Maharashtra (India)

Abstract— Advanced Encryption Standard (AES) is the contemporary encryption standard. The Advanced Encryption standard approved by federal information processing standard (FIPS) defines the cryptographic algorithm that can be used to protect electronic data. The AES algorithm encrypt (encipher) and decrypt(decipher) information in a symmetric manner. Encryption converts data to an indistinct form called cipher text; and again decrypting the cipher text converts the data back into its readable form, called plaintext. The AES algorithm can use cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

Keywords— AES Encrption, FPGA, Algorithm.

I. INTRODUCTION

Now days embedded cryptographic hardware are costly as well as not safe. With the growth and widespread use of information and technology, the internet security is very important. Therefore for more security purpose, field programmable gate array gives the most efficient source communication in the wireless networks. The RIJNDAL algorithm is a block cipher that can symmetrically process data blocks of 128 bit using cipher keys with key length of 128,192,256 bits. The algorithm will be known as "AES algorithm."
DES (Data Encryption Standard) whose security is failed because of its short length key (56 bit), afterwards AES algorithm chosen by NIST (National Institute of Standard and Technology) which becomes popular in the symmetric cryptographic algorithm.
Since AES algorithm is symmetric, reverse transformation for decryption can also be execute of the same key. The pipelining technology was utilized to achieve the AES 128 encryption circuit. To minimize operations which are done manually a verilog code is developed. Synthesis and code simulation is done by using Xilinx. Since it was easy to implement, it gone successful and could run in moderate amount of time on a regular computer.

II. AES ALGORITHM

This AES encryption algorithm converts a plain text message into coded text message called cipher text which can only be retrieved by authorized receiver using a decryption technique. The Rijndael algorithm is an iterative private key symmetric block cipher. The number of encryption or decryption round is processed on different key length. This relationship is given in the below table1.

TABLE1. AES ALGORITHM ROUND AND KEY LENGTH

AES	Key length	Group size	No of rounds
AES-128	128	128	10
AES-192	192	128	12
AES-256	256	128	14

This algorithm involves four module in which three module shift rows, mix column and add round key are all linear transformation except the byte substitution

- Add round key: in this round, every byte in matrix is simply XOR with the round key which is taken from key generation algorithm.

- Sub byte: in this round, every byte in matrix is simply replaces each byte of 128 bit input plain text with the corresponding inverse element of Galois field GF (2^8). These steps introduce non-linearity in the cipher. There are two methods to obtain the s-box look up table; the first one is to define each element in the table separately, also called static table. The second one is by generating the look up table with respect to modulo polynomial which involves finding the multiplicative inverse and affine transformation, so called dynamic table.

- Shift rows steps- this operation just takes cyclic shift of the state matrix. Each row is rotated left by a different number of bytes; the first row is not rotated but the second, third and fourth rows are rotated by one, two and three bytes respectively

- Mix column step- this step uses linear transformation to fully mix each row in the matrix by taking multiplication and addition operation of previous step result with the irreducible polynomial $x^8+x^4+x^3+x+1$.

A. AES Encryption process

The Encryption and decryption process consists of different transformations applied constantly over the data block bits, in a fixed number of iterations, called rounds. The total number of rounds depends on the length of the key used for the encryption process. For 128 bits key length, the number of round required is 10. (Nr = 10). As shown in Fig. 1, each of the first Nr-1 rounds consists of 4 transformations.

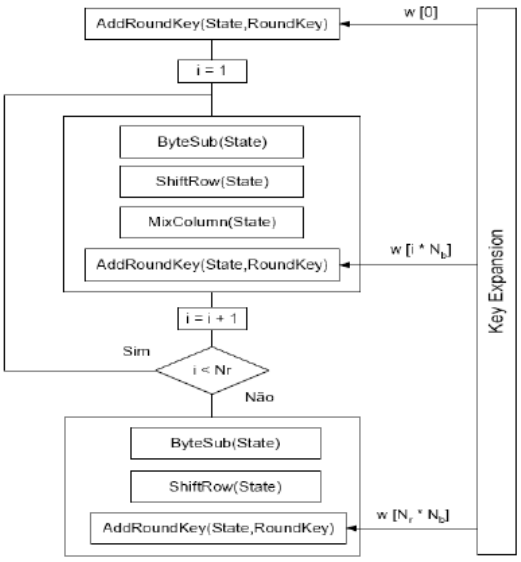

FIGURE 1: ENCRYPTION PROCESS

B. AES Decryption process.

The decryption process is straight inverse of the encryption process. All the modification applied in encryption process are inversely applied to decryption process. Therefore the values of last round of both the data and key will became the first round inputs for the decryption process and accompany in decreasing order. The decryption process is carryout using the following functions: InvSubBytes(), Inv.ShiftRow(), AddRoundKey( ), and InvMixColumn( ) respectively.

III. DESIGN METHODOLOGY AND FSM

To create a novel architecture for AES we used FSM to execute the encryption and decryption using a common block. This resource sharing reduced the area effectively. A finite-state machine (FSM) is a computation mathematical model that is used for designing computer programs and sequential logic circuits. It is frame as an abstract machine that can be in one of a finite number of states defined for that machine. The machine is in only one state at a time. It can also transit from one state to another state when initiated by a triggering event which is called a transition. A particular FSM is defined by its states, and the triggering condition for each transition. Finite-state machines can model number of problems, among which is automation electronic design, parsing language, communication design protocol and other applications associated in engineering.

IV. ANALYSIS TOOLS FOR FPGA

For implementing a design in a FPGA, different analysis tools are required. Synthesis and Placement are the fundamental design tools required to compile FPGA design and simulation tools are very useful to verify that a design meets its specification. Synthesis tools are used to interpret the HDL (Source Hardware Description Language) code usually VHDL (or Verilog).This tools typically convert the HDL code to Register Transfer Level (RTL) which defines the logical function.

Synthesis tools are commonly provided by FPGA manufacturers such as Altera or Xilinx.

V. RTL SCHEMATIC AND SIMULATION RESULT

The synthesis result of AES is as shown in fig2.

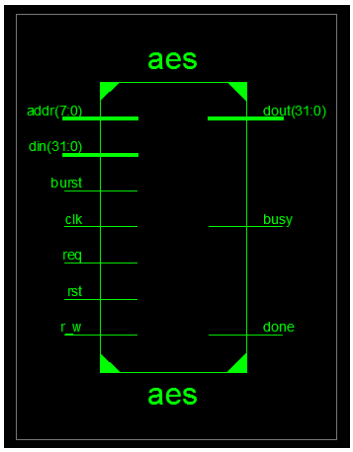

Figure2: RTL of AES.

The input and output port is shared between encrypted data and decrypted data Burst-mode communication varies from traditional steady mode communication in that data is transmitted in bursts or packets of 32 bit each rather than a continuous data stream. As all the inputs and outputs are 128-bit wide so a burst operation for both writing and reading of data on 32-bit port is utilized. Represents the black box view of the AES module.

Table II lists the total input/output signals of the module with their functional description.

TABLE II: SIGNALS OF THE MODULE AES

Name	Description	Direction
clk	Clock signal to the aes.	Input
rst	Reset signal to the aes.	Input
addr	8-bit wide signal used for different modes of FSM.	Input
burst	Signal used to describe burst read and burst write operation.	Input
req	Signal used to acknowledge the presence of input data and registering it.	Input
r_w	Read or Write signal. High for reading and Low for writing.	Input
din	32- bit wide signal used for providing data to the aes.	Input
dout	32- bit wide signal used for providing encrypted or decrypted data.	Output
done	Signal to indicate the presence of encrypted or decrypted data at the output port.	Output
busy	Signal to indicate the initialization of rounds of the aes.	Output

A. Simulation Result

The test vectors provided by NIST are:
key [127:0] 128'h000102030405060708090a0b0c0d0e0f
plaintext [127:0] =128'h00112233445566778899aabbccddeeff
.After ten rounds of the AES the cipher text will appear as
Cipher text= 128'h69c4e0d86a7b0430d8cdb78070b4c55a
Cipher key= 128'h13111d7fe3944a17f307a78b4d2b30c5.

As shown in Figure.3.11 and 3.12. For decryption we use cipher text as input and use the same cipher key for decryption algorithm and get back the original plaintext.

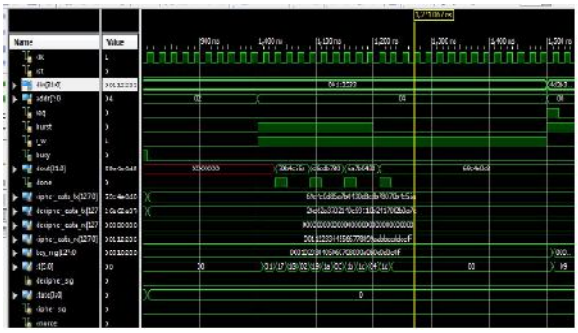

Figure 3.11. Encrypted output at dout

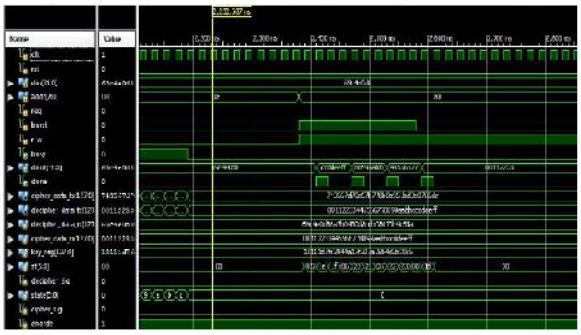

Figure 3.12. Decrypted Output at dout

CONCLUSION

The AES-Rijndael algorithm was selected as the current Advanced Encryption Standard (AES) for several reasons. The objective was to generate an algorithm that was resistant against known attacks, simple, and rapid to code. Choosing to use field GF (2)8 was a very good decision. The b size of the block and key size can differentiate making the algorithm versatile. Optimized and Synthesizable VHDL code is developed for the implementation of encryption process.

NIST provide a part of sample vectors through which each program is tested and output results are perfect with minimal delay. Therefore, AES can significantly be implemented with reasonable efficiency on an FPGA.

REFERENCES

[1] Yuwen Zhu, Hongqi Zhang, Yibao Bao, “Study Of AES Realization Method On The Reconfigurable Hardware”, 2013 International Conferences On Computer Science And Application.

[2] NIST, “Advanced Encryption Standard (AES)”,NIST,FIPS-197,2001

[3] AI-Wen Luo, Qing-Ming Yi, Min Shi. “Design and Implementation of Area-optimized AES on FPGA, IEEE Inter.conf.chalsci com engin.,978-1-61284-109-0/2011.

[4] J.Yang, J.Ding, N.Li and Y.X.Guo,“FPGA-based design and implementation of reduced AES algorithm” IEEE Inter.Conf. ChalEnvirSci Com Engin(CESCE).,Vol.02, Issue.5-6, pp.67-70, Jun 2010.

[5] Fedaral Information Processing Standards publication 197 November 26,2001”ADVANCED ENCRYPTION STANDARD (AES)” 2001.

[6] http : // Advanced Encryption Standard-wikipedia , the free encyclopedia.html.

★★★