International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

Improvement in the Advanced Encryption Standard Algorithm in Term of Low Area and Power Consumption by using FPGA

Umalaxmi Sawant[1], Venkat Ghodke[2]

[1]Savitribai Phule Pune University, ME VLSI and Embedded System, DPCOE, Wagholi, Pune

[2]Savitribai Phule Pune University, AISSMS Institute of information technology, Pune

Abstract: ***In all over the world communication of private and confidential data over the computer networks or the Internet, there is always a chance of the hacking the data from the wrong intention people. Data encryption maintains data privacy and authentication. Information has become of the most important thing in growing demand of have to store every single importance of events in everyday life. Messages have to be secured from unauthorized party. So that encrypts the data. There are two types of encryption algorithms, a private key (symmetric key) and public key. In terms of computational complexity, private key algorithm is not much as complex than a public key algorithm. In this paper implement the AES algorithm on FPGA using VHDL language with using software Xilinx ISE tool. The main target is by maintaining standard throughput of data to achieve low area as well as low power consumption. Also maintain high speed data processing and reduce time for key generating. For an instantaneous output in this paper use BRAM implementation which is alternative to conventional s-box combinational logic. It shows the performance better than other.***

Keywords: AES (Advanced Encryption Standard), DES (Data Encryption Standard), Encryption, Decryption, Cryptography, FPGA, cipher text.

1. Introduction

With worldwide communication security is very important issue for valuable, sensitive information in the communication technology. In order to gives the security or the protection to the important information transformation cryptography is used. Modem cryptography provides an efficient and secure approach in embedded systems [1].Cryptography is the science of the secret codes, enabling the confidentiality of communication of communication through an insecure channel. The National Institutes of Standards and Technology (NIST) recommended Rijndael Algorithm later it to be Advanced Encryption Standard (AES) in 2001[2]. By using hardware and software, advanced encryption algorithm has efficiently implemented. In this software implementation has less security as compared to the hardware security but software offers less cost. But for high speed application hardware implementation is extremely reliable, speedy and conveniently suitable [3]. It is widely used in embedded systems, automotive electronics, Personal Digital Assistant, mobile phone, smart cards, defenses, medical reports, bank services via Internet [4]. The original file or data converted into the coded form known as encrypted data whereas coded form of data converted into original form known as decrypted data.

Private Key or symmetric key and the public key are the two types of the encryption algorithm. Private Key has one key for encryption and decryption algorithm, and also it is simple in computation and suitable for faster implementation [5]. This algorithm has three different key length “AES-128”, “AES-192” and “AES-256” bits while block size must be 128 bit. For AES algorithm there are many architecture proposed, in this many of them poor in terms of speed, area, power consumption. This paper approaches to achieve a less area and low power consumption which maintain standard throughput of data, also achieve high speed data processing and decrease time for key generating [6].

2. Literature Review

Ashwini R. Tonde, Akshay P. Dhand proposed that Advanced Encryption Standard (AES) algorithm undergoes the process on FPGA kit Very high speed integrated circuit Hardware Description Language is used. For software purpose Xilinx ISE and Model Sim tool is used [1].

Hoang Trang, Nguyen Van Loi, Said that, an efficient FPGA implementation of 128 bit block and 128 bit key AES algorithm is proposed. In this the encryption /decryption algorithm is synthesized and implemented by Altera Tool and achieve Low Latency and the Throughput shows 1054Mbit/Sec for encryption and 615Mbit/Sec for Decryption [2]

Z. Yuan, Y. Wang, J. Li, R. Li and W. Zhao proposed that masking method are used to protect against power analysis attacks in embedded systems. In this various Masking techniques are used Such as Boolean masking, Additive masking, Multiplicative masking, mixed masking, Algorithmic level masking is used. To oppose against DPA (Differential Power Attack) is optimized AES implementation with 32-bits and 128-bits data path individually [3]

Saurabh Kumar, V. K. Sharma, K. K. Mahapatra paper presents and proposed that Low latency VLSI architecture of S-Box for AES Encryption.The conventional AES use

Licensed Under Creative Commons Attribution CC BY

BRAM architecture which suffers from unbreakable delay as fixed access time for read and write operation and low latency because of ROM access time. To increase throughput parallel ROMs are required expensive large size of chip area requires high amount of memory. Therefore S-box transformation for low latency with reduction in area against, composite field arithmetic is more suitable [4].

Saurabh Kumar, V. K. Sharma, K. K. Mahapatra paper presents the architecture is implemented in FPGA. By using unvarying area in terms of FPGA slices, delay is improved along with low power consumption. For showing the speed 0.6 ns along with low utilization of FPGA fabrics.In this paper architecture is implemented on FPGA and ASIC. ASIC is implemented using 0.18 µm standard cell technology library [5]

3. Proposed Work

AES algorithm is an iterative algorithm, which requires many computation cycles. A software platform cannot provide the high speed encryption of data, specially used for real-time applications. Therefore, dedicated hardware implementation is inevitable in such applications. In AES algorithm has mainly two stages Encryption and Decryption. At the encryption side covert the valuable information into coded form so that protect data from the unauthorized people and at decryption side coded form of data to the original form. By using this safely data obtain to the receiver.

A.AES Encryption

At the encryption side there are four main operations Sub byte, shift row, Mix column and add round key. In this each step has each function. In figure 1 shows plaintext data or the original data gives to the sub byte box which used nonlinear byte substitution. After that outputted value gives to the shift row operation, in these cyclically right shifted bytes in last three columns in state. Mix Column transformation in Cipher that takes all of the columns of State and mix their data separately of one another to produce new columns using GF (2^8) polynomial. And last operation bitwise exclusive-or (XOR) operation are performed in Add Round Key operation performing the XOR operation between outputs from mix column and round keys. Also Key Expansion module is used to perform key scheduling, which generate a series of Round Keys from cipher key. This encryption module connected as shown and convert original data into cipher text.

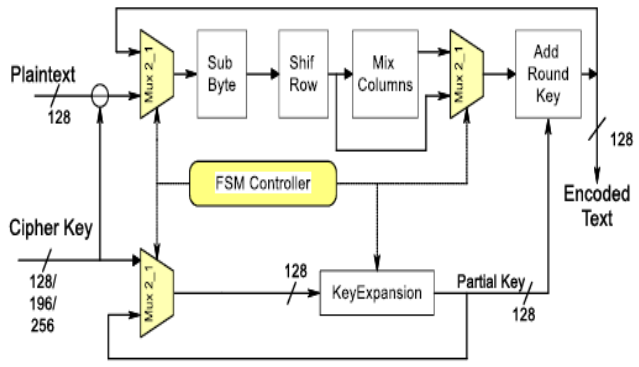

Figure 1: Block Diagram of AES Encryption

B. Sub Byte

Sub byte operation is the first operation of the AES encryption. In this nonlinear byte substitution is used, in term of cost and implementation substitution is the most complex step, which is operates independently on each byte of the State using a substitution table (S-box).Take the multiplicative inverse in the finite field GF (2^8) and affine transform to do the Sub Bytes transformation. The Sub Bytes transformation is done through S-box. BRAM Implementation and combinational logic this are two techniques to perform substitutions. Combinational logic architecture occupy huge area in AES algorithm, because it is used repetitively in this algorithm but there is some features like small area occupancy, and pipelined structure so that increases the performance in clock frequency. For low latency with reduction in area composite field arithmetic is more suitable.

In this paper another technique is used known as BRAM implementation. BRAM technique is more suitable and simple method for the AES algorithm. BRAM method is nothing but all pre-computed 256 value stored in RAM based look up table and input weird to the RAM address bus. Combinational S-box architecture require large no of LUT's for increase the throughput and in this BRAM available on FPGA. As compared to the ROM, BRAM has advantage like fixed access time for read write operation, low latency, and unbreakable delay

C.Key Expansion

In key expansion series of round keys generated when Key Expansion routine is used to perform key scheduling known as parallelism. Because of this parallelism it raises the speed and reduces the time of key generation. In Key Expansion routine sub word, rot word, rcon is present. 128bit cipher key divided into 32bit after that takes a 4-byte input word applies to S-box each of 4-bytes to produce an output word. Then this output undergoes the process of cyclic permutation where 4 byte word cyclic right shifted with 1 bytes boost. Rcon is array of bytes in a word having permanent logical value having size of 128 bit. For the temporal storage key registers are used. Here input to key expansion module is 128 bit cipher generates 10 number of 128 bit. It produces the Partial key for each round of operation. Repetitively used pipeline structure computation become lower down speed up to nine times as well as data rate a combinational logic of Key Expansion decrease period by nine time for key generating

D.AES Decryption

In this decryption side operation is nothing but just inverting of the encryption side operation. In this has four main operations InvSubByte, InvShiftRows, InvMixcolumn and add round key. In shown in figure 2 obtaining Encoded text or cipher text gives to the InvSubByte box which performances inverse of byte substitution transformation, applying inverse affine transformation followed by taking the multiplicative inverse in GF (2^8).

Licensed Under Creative Commons Attribution CC BY

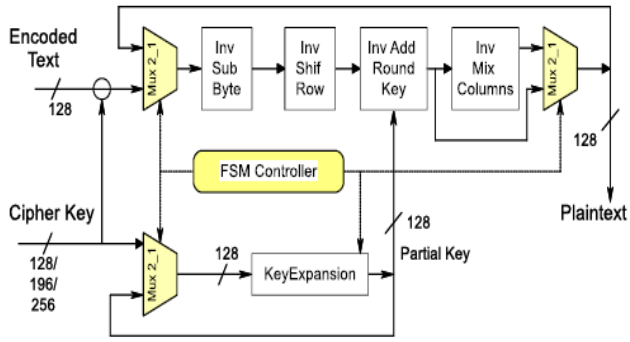

Figure 2: Block Diagram of AES Decryption

After that output given to the InvShiftRow, and in this cyclically left shifted last three row. In InvMixColoum operates on the State column-by-column, treating each column as a four-term polynomial and multiply modulo (x4+1) with fixed polynomial. InvAddRoundkey transformation its own inverse adds round key XOR operation.

4. Result

For the hardware part of implementation in this paper use FPGA kit Xilinx Spartan 6 and for the software part Xilinx ISE tool and Model SIM tool used for the synthesis for testing and verification purpose respectively. Here present Standard AES result and implemented AES result also shows the comparably study between various AES cipher keys.

Table 1: Design utilization of AES encryption

Parameter	*AES 128 (Standard)*	*AES 128*	*AES 192*	*AES 256*
Data path (bit)	128	128	192	256
No of Round	10	10	12	14
LUT FF	40	459	414	412
Slice LUT	24999	3559	3705	4274
Block RAM	11	4	8	8
Slice registers	4800	564	573	566
No. Of clock cycle	10	41	49	57
Max operating Frequency	102.990	273.997	263.742	267.408
Combinational Delay	11.992	0	0	0
Area constraints	>100%	50%	52%	66%
Throughput (Mbps)	13183.6	855.61	688.99	600.47

Table 2: Design utilization of AES Decryption

Parameter	*AES128 (Standard)*	*AES 128*	*AES 192*	*AES 256*
Data path (bit)	128	128	192	256
No of Round	10	10	12	14
LUT FF	3	426	472	531
Slice LUT	39712	3531	3703	4397
Block RAM	10	20	20	20
Slice registers	7120	607	598	611
No. Of clock cycle	10	41	49	57
Max operating Frequency	72.489	223.157	225.466	213.258
Combinational Delay(ns)	8.392	0	0	0
Area constraints Ratio	49%	49%	52%	66%
Throughput (Mbps)	9278.7	696.712	589.00	479.04

Table 3: Comparisons of existing design with proposed design

Parameter	[2]	[3]	[6]	Proposed Design
Datapath (bit)	128	32	128	128
Platform	Altera APEX20K-C	Xilinx Virtex-5 XC5VL50	Xilinx Virtex-2 XC2VP20	Xilinx Sparton-6 XC6SLX9
Throuput (Mbps)	1188	73.3	28250	200
Area	40960 slice / 895 LUT	769 slices/ 2350 LUT	9028 slices	554 slices/ 3531 LUT
Freqency (MHz)	-	100.8	220.7	277.369

Proposed design results of the AES-128(standard), AES-128, AES-192, AES-256 has shown in table 1 result for encryption side and table 2 shows results for decryption side. In this use seven segment display for result display purpose. For example display shows
Encryption side input (Plaintext)
"00112233445566778899AABBCCDDEEFF"
Encryption side Key
" 000203040506070809 0A0B0C0D0E0F"
Cipher output and Decryption side input
" 69C4E0D86A7B0430D8CDB78070B4C55A"
Decryption side Key
" 000203040506070809 0A0B0C0D0E0F"
Decryption side output
"00112233445566778899AABBCCDDEEFF". In this way Encryption and decryption are occurs in the AES algorithm.

5. Conclusion

We have proposed Effective Advanced Encryption Standard Algorithm which is based on software as well as hardware for using both it increases the secured level. AES Hardware implementation support high speed data processing using pipeline & parallel processing approach. In this main give the attention towards the low area as well as power consumption. Comparing both Standard AES implementation and proposed AES implementation, it shows an improvement, dropping 89% in LUT and slice register. A Sbox is implement using parallel processing approach so that speed of computation time is reduce down by one clock compare to BRAM Sbox architecture. The implemented design of AES algorithm uses pipeline structure for SubByte, ShiftRows, MixColumns and AddRoundKey transformation. The pipeline structure for repeated computation it become lower down speed up to nine times as well as data rate. But its capable follows as per standard AES.

References

[1] Ashwini R. Tonde, Akshay P. Dhand, "Review Paper On FPGA Based Implementation Of Advanced Encryption Standard Algorithm", International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 1, January 2014
[2] Hoang Trang, Nguyen Van Loi, "An efficient FPGA implementation of the Advanced Encryption Standard

Licensed Under Creative Commons Attribution CC BY

algorithm", Proc. Computing and Communication Technologies RIVF International Conference, pp. 1-4, 2012

[3] Z. Yuan, Y. Wang, J. Li, R. Li and W. Zhao, "FPGA based optimization for masked AES implementation", Proc. IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), pp.1-4, 2013. .

[4] Saurabh Kumar, V.K. Sharma, K. K. Mahapatra, "Low Latency VLSI Architecture of S-box for AES Encryption", Proc. International Conference on Circuits, Power and Computing Technologies, pp. 694-698, 2013.

[5] Saurabh Kumar, V.K. Sharma, K. K. Mahapatra, "An Improved VLSI Architecture of S-box for AES Encryption", International Conference on Communication Systems & Network Technologies, 2013

[6] Issam Hammad, Kamal El-Sankary, Ezz El-Masry, "High-Speed AES Encryptor with Efficient Merging Techniques" Proc. IEEE Embedded Systems, vol. 2, no. 3, Sept 2010.

[7] Abhijith.P.S, Mallika Srivastava, Aparna Mishra, Manish Goswami, B.R.Singh "High Performance Hardware Implementation of AES Using Minimal Resources" International Conference on Intelligent Systems and Signal Processing, 2013

Licensed Under Creative Commons Attribution CC BY