Implementation of Full -Parallelism AES Encryption and Decryption

M.Anto Merline

M.E-Commuication Systems, ECE Department
K.Ramakrishnan College of Engineering-Samayapuram, Trichy.

Abstract-Advanced Encryption Standard (AES) is a symmetric key encryption algorithm used to encrypt block of data. AES is widely used encryption standard in today's world. In this paper inorder to increase the performance of the encryption algorithm, various architecture models is proposed with data-level parallelism and task-level parallelism. The proposed architecture is design in ASAP processor. The proposed architecture increases the throughput and speed of the encryption and decryption process by reducing the latency between the processes by using parallelism architecture. This paper proposed four different architecture models among that full parallelism architecture gives better performance among other models. The throughput of this model is 70 cycles per block. The encryption process uses online key expansion which increases the security of the data. The full parallelism gives higher throughput per chip area compare to other existing techniques.

Keywords- *symmetric key, data level parallelism, task level parallelism, online Key Expansion, ASAP.*

I. INTRODUCTION.

Cryptography is a technique in which the data is encrypted in the sender side before it transmits through the unsecure channel. At the receiver side the original message is obtain by decryption process. For encryption and decryption a key is commonly used. Based upon the key, the process is split into two process: Symmetric key encryption and asymmetric key encryption. Many algorithms are proposed to encrypt the original message in secure manner which prevent the message from the outside attacker. In cryptography the original message is known as Plain Text and the encrypted message is known as Cipher Text. In olden days the encryption technique mostly depends on two techniques. Those are: Substitution and Transposition techniques [1].In substitution technique each word in the plain text is replaced by some other word to form a cipher text . In transposition technique the words in the plain text is rearranged by or hidden within some other text to form a cipher text. In modern cryptography many algorithm such as Data Encryption Standard, blow Fish, RC5 are proposed. Among that till 2001 DES is most widely used data encryption standard. It is developed by IBM in 1960. It is used to encrypt the text by using 56 bit key length. It is based on Substitution and Permutation technique. Since it is less secure because only 56 bit key length is used [1].

In 2001 National Institute of Standards and Technology (NIST) introduced Advanced Encryption standard (AES) using Rinjdeal Algorithm as a secure encryption standard. It can use to encrypt the data of 128 bits with variable key length 128,192 and 256 bits. Advanced Encryption Standard is a symmetric key encryption process [2]. Symmetric key encryption process represent that the same key is used for both encryption and decryption process. AES process is a block cipher which denotes that encryption process is carried out for block of data. The input data is split into block of data with fixed size. The encryption and decryption process is carried out for each block. AES encryption process is most secure encryption algorithm than the DES process which is used before AES encryption [2]. AES encryption and decryption process contain two major processes. Those are: Encryption/Decryption process and Key expansion process.

The Rijndael algorithm is used in AES encryption. The mathematical basis necessary for understanding the specifications followed by the design rationale and the description itself [3]. The memory less pipelined architecture achieves a speed of 8 Gbps@ 250 MHz clock. The pipelined architecture can be made to toggle between the encryption and decryption modes without the presence of any dead cycle [4]. The architecture of a fully pipelined, loop unrolling and inner round and outer round pipeline maximize the throughput to 21.54 Gbits/s[5]. The area-throughput trade-off for an ASIC implementation of the Advanced Encryption Standard (AES). Different pipelined implementations of the AES algorithm as well as the design decisions and the area optimizations that lead to a low area and high throughput AES encryption processor are presented. With loop unrolling and outer-round pipelining techniques, throughputs of 30 Gbits/s to 70 Gbits/s are achievable in a 0.18**µm** CMOS technology [6]. Asynchronous Array of Simple Processor contains many cores in one processor which increases the throughput by performing many processes at the same time in parallel manner.

II. AES ENCRYPTION PROCESS

AES encryption part contains four step by step operations. In AES process the encryption/Decryption process undergoes many round according to key length of the key used to encrypt block of data. Each round having the four step of operation except the last round. The last round contain only three step of operation. Those four steps of operations used in encryption are: Subbyte, Shiftrow, Mixcolumns and Addround Key.

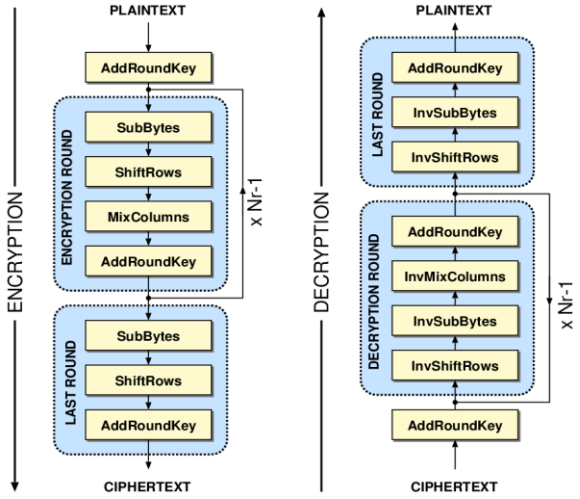

Fig.1 AES Encryption and Decryption Process

A. Sub Bytes:

Each byte from the input state is replaced by another byte according to the substitution box (called the S-box). The S Box denotes substitution Box. It can implemented using LUT or in manual calculation. Implementation of S-Box in manual calculation required two steps. Those are: multiplicative inverse followed by affine transform. It is calculated from finite field GF (2^8). But it will increase the complexity of the calculation. In order to reduce the complexity the calculation is split into smaller power. The GF (2^8) is calculated from Galois Field $(2^2)^2)^2$. This will reduce the complexity of the circuit. The implementation of the S-Box from the Lookup Table (LUT) is difficult than manual implementation of S-Box. By using this S-Box only the Subbyte operation replace each byte in the plain text by another byte. The S-Box contains rows and columns using this rows and columns only the correspondent byte for byte in the plaintext can find and replace.

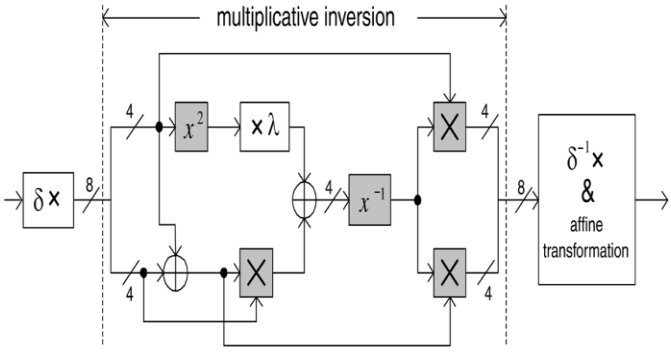

Fig.2 Subbyte Operation

B. Shift Rows:

In the Shift Rows transformation, the first row of the state array remains unchanged. The bytes in the second, third, and forth rows are cyclically shifted by one, two, and three bytes to the left, respectively.

C. Mix Columns:

During the Mix Columns process, each column of the state array is multiplied with the another one standard column matrix.

D. Add Round Key:

A round key is added to the state array using a bitwise exclusive-or (XOR) operation. Round keys are calculated in the key expansion process.

III. AES DECRYPTION PROCESS

The decryption process also contains four operation similar to encryption but the operations are in the inverse manner. The Add Roundkey operation is common to both encryption and decryption process.

A. Inverse Sub Bytes:

Each byte from the input state is replaced by another byte according to the inverse substitution box (called the S-box). The inverse S box is calculated by affine transform followed by multiplicative inverse operation. By using that inverse S-box only this step can be performed.

B. Inverse Shift Rows:

In the Inverse Shift Rows transformation, the first row of the state array remains unchanged. The bytes in the second, third, and forth rows are cyclically right shift by one, two, and three bytes respectively.

C. Inverse Mix Columns:

During the Inverse Mix Columns process, each column of the state array is multiplied with another one standard inverse column matrix.

D. Add Round Key:

A round key is added to the state array using a bitwise exclusive-or (XOR) operation. Round keys are calculated in the key expansion process. In decryption the key used in reverse manner that means key 10 used in the first round and key 1 is used in the last round.

IV. KEY EXPANSION

The key expansion process expands the key and produces new key for each round. For 128 bits 10 rounds are performing therefore 10 keys are used. These keys are formed from the key expansion process. The 128 bit keyword is split into 4*4 state array. The key expansion process contains three steps. Key Sub word, Key Rot word and XOR Operation.

A. Key SubWord:

In this step each byte in the state array of the key is substitute by another byte according to S-Box.

B.Key RotWord:

In this step a left cyclic permutation is carried out in the first column of the key matrix.

C. XOR Operation:

The XOR operation is carried out between W[i-1] and W[i - Nk]. Nk = 4/6/8 according to the key size 128,192 and 256 bits.

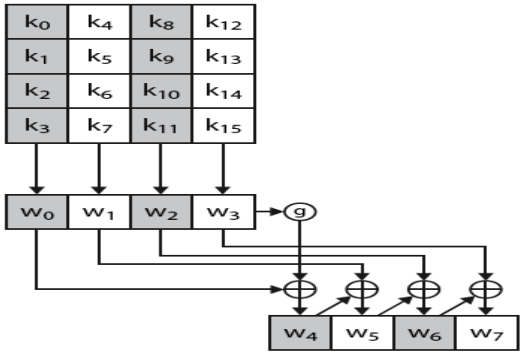

Fig.3 Key Expansion Process

V. PROPOSED WORK

AES encryption standard is widely used encryption standard for most of the applications in recent world. AES encryption with high throughput and high performance is needed to increase the security of the process. So various methods are proposed to increase the performance of the AES encryption. All the models uses ASAP processor.

A.Small Encryption:

The small encryption method is similar to common encryption method. This is used to implement the AES using some number of cores. It only required eight cores to implement AES encryption in online process. Each Block on FPGA has only a 128 X 32-bit instruction memory and a 128 X 16-bit data memory. Since it uses only eight cores the hardware required will be reduced.

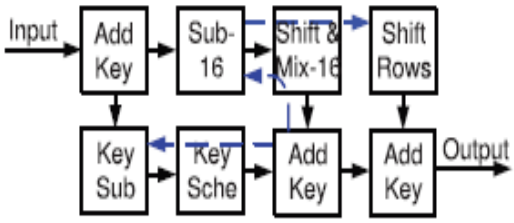

Fig.4 Small Encryption Model Data flow

B. One-Task One-Processor (OTOP):

In this one task one processor method each processor in a core in FPGA is assigned to one task. In this all the task of AES encryption is carried out parallel by each processor in a core at the output all the processed data are combined to get full output. This parallel processing of data will decrease the latency of the process and increases the throughput and the performance of the encryption process. The nine rounds of the AES process is carried out in loop and this loop is processed by a same processor in the core. The output of one loop become a input to the another loop. This process is known as loop rolling process. The key expansion process is also carried out parallel. The throughput of the OTOP implementation is determined by the nine (Nr_1 ¼ 9) loops in the algorithm.

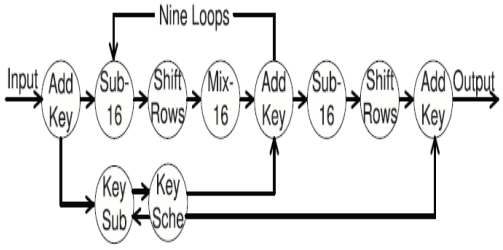

Fig.5 One-Task One-Processor Model Data Flow

C. Parallel-Mixcolumns:

This is one of the methods used to increase the throughput of AES encryption process. In this process the key expansion process carried out parallel. In single loop, the Mixcolumn-16 contributes 60 percentage of latency. In this each block of data, each column is processed independently, it will increase the latency. Inorder to decrease the latency the Mixcolumns-16 is split into four Mixcolumns- 4 block. Therefore each column in a state can be processed parallel. This will reduce the latency offered by the Mixcolumn block. After processing all the output from Mixcolumns-4 will combine to get a state array for next process. By using this method the latency will be reduced therefore the throughput will increases. The throughput of parallel Mixcolumns is higher than the previous method. The Mixcolumns block can be further divided into Mixcolumns-2 to reduce the delay. But it will increase the area without a sufficient increment in the throughput. Therefore only Mixcolumns-4 is used. In this loop unrolling mechanism is used. In this all the nine loops are processed parallel by separate processor in order to decrease the latency. This will also decreases the latency of the whole process and decreases the clock cycle of the process.

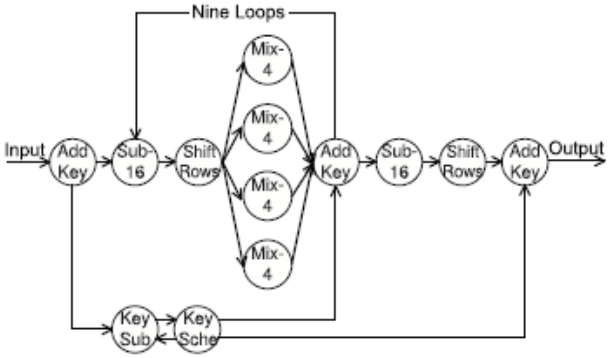

Fig.6 Parallel Mix Columns Model Data Flow

D. Full-Parallelism:

In the Full-parallelism model, inorder to increase the throughput of the AES encryption process compared to parallel Mixcolumns model it proposed a new architecture. In this architecture along with parallel Mixcolumns the subbyte-16 core of the process also split into four subbyte-4. The subbyte operation in AES is carried out independently for each row in the state array separately. Therefore by split the subbyte core each row in the state is processed parallel therefore the latency will further reduce compared to parallel Mixcolumns model. In this process also loop unrolling method is used. It can also reduce the delay and the latency of the process. Therefore among the entire proposed model for AES the full parallelism method is more efficient than another model. The latency and the clock cycle is reduced compare to others. The throughput is very high compare to other models due to its architecture. It required only 70 cycles per block. The decryption process is similar to the encryption process but the steps are inverse of encryption process. The full parallelism in the proposed system contains both encryption and decryption process and the implementation are carried out in the software platform.

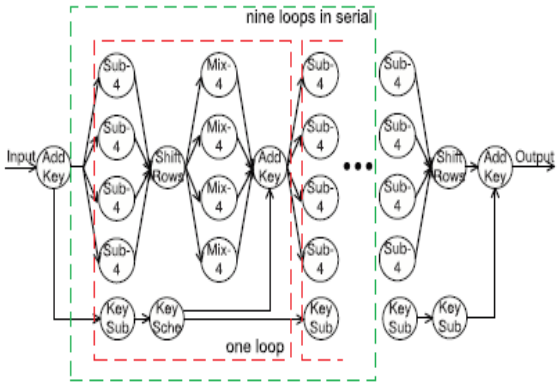

Fig.7 Full Parallelism Model Data Flow

VI. SIMULATION AND RESULTS

The coding for all the model of AES encryption and decryption is written in Verilog HDL language which is mostly preferred language to write the VLSI coding. The coding is simulated in XILINX 13.2 version to obtain a desired result.

The figure 8 below shows the simulated output of small encryption process which contain both encryption and decryption process.

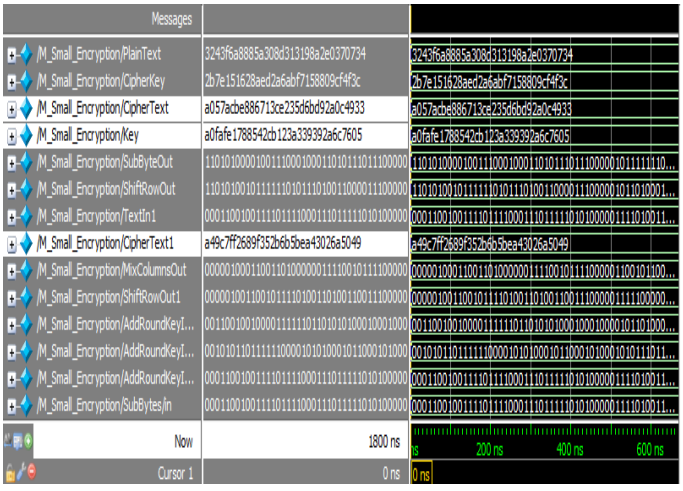

Fig.8 Small Encryption Simulated Output

The figure 9 below shows the simulated output for One Task One processor AES architecture model.

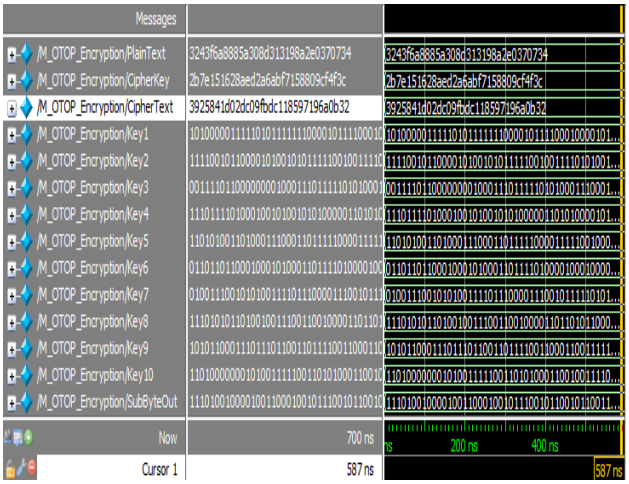

Fig.9 One Task One processor Simulated Output

The figure 10 below shows the simulated output for Parallel Mixcolumn AES architecture model.

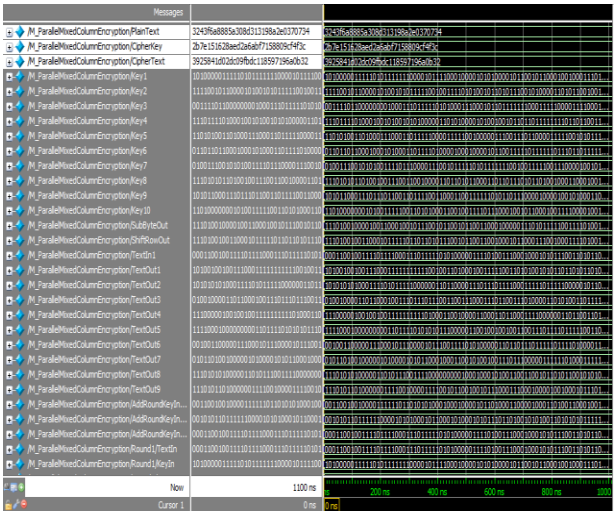

Fig.10 Parallel Mixcolumns Simulated Output

The figure 11 below shows the simulated output for Full Parallelism AES architecture model. The throughput and the efficiency of full parallelism AES architecture model is higher than the other three models discussed above. It reduce the latency by using task level parallelism in both encryption and decryption algorithm.

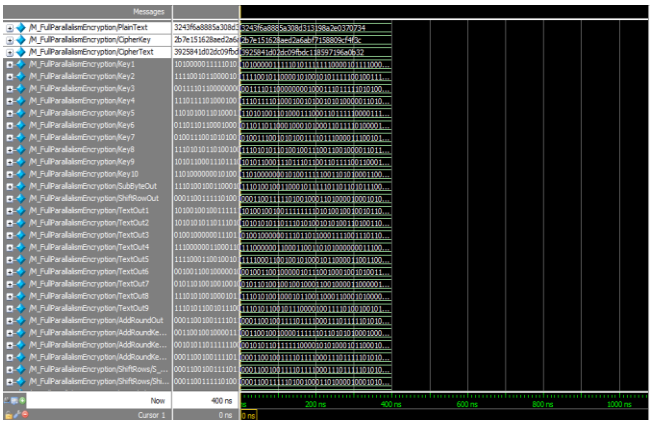

Fig.11 Full Parallelism Simulated Output

VII. CONCLUSION

AES is widely used encryption technique. In order to improve the efficiency of the encryption process different architecture model with task level parallelism and data level parallelism is used to reduce the latency and increase the throughput. Among four architecture model for AES proposed in the paper full parallelism is the better architecture which reduces the latency between process by using parallel process and increase the throughput. The throughput of the full parallelism model is 70 cycles per block. The efficiency of Full parallelism model is higher than the existing model for AES.

REFERENCES

[1] NIST, "Data Encryption Standard (DES)," http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf, Oct. 1999.

[2] NIST, "Advanced Encryption Standard (AES)," http://csrc.nist.-gov/publications/fips/fips197/fips-197.pdf, Nov. 2001.

[3] J. Daemen and V. Rijmen, "The Design of Rijndael". Springer-Verlag, 2002.

[4] D. Mukhopadhyay and D. RoyChowdhury, "An Efficient end to End Design of Rijndael Cryptosystem in 0:18_m CMOS," Proc. 18th Int'l Conf. VLSI Design, pp. 405-410, Jan. 2005.

[5] A. Hodjat and I. Verbauwhede, "A 21.54 gbits/s Fully Pipelined AES Processor on FPGA," Proc. IEEE 12th Ann. Symp. Field-Programmable Custom Computing Machines, pp. 308-309, Apr. 2004.

[6] A. Hodjat and I. Verbauwhede, "Area-Throughput Trade-Offs for Fully Pipelined 30 to 70 Gbits/s AES Processors," IEEE Trans. Computers, vol. 55, no. 4, pp. 366-372, Apr. 2006.