IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 4, Issue 2, Ver. IV (Mar-Apr. 2014), PP 54-64
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

An Efficient VLSI Implementation of Low Power AES – CTR

[1]Podila Sushma, [2]J.Selva Kumar

Abstract: *This paper delineates an efficient VLSI architecture implementation in order to increase the throughput and security using Advanced Encryption standard (AES) algorithm. The existing architecture depicts the blocks like Sub Bytes, Shift Rows, Mix Column, and AddRoundkey which are used in AES algorithm. Meliorating the design, a new technique named AES-CTR was introduced which is an iterative algorithm. It resulted in the transformation of the stream cipher, which is generated by performing the xor operation between pseudorandom bits & plaintext. These Pseudo random bits are resulted due to the encryption of data. Performance metrics of VLSI such as power and area of AES-CTR architecture are evaluated using a gpdk of CMOS 180nm. The AES & AES-CTR design were modelled and synthesized used TSMC'S 180nm standard cell library using RTL complier & physical design implementation usingSOC Encounter Digital. Drastic improvement of power and area are abided along with the improvement of security of the entire system.*
Index Terms: *AES, AES-CTR, (NIST) National Institute of Standard and technology*

I. Introduction

Security of a system plays a major role in transmitting and storing the information. Many Cryptography techniques like DES algorithm provide a mean for security, DES is 64-bit cryptosystem, here 64-bit plain text and 64-bit cipher text for the encryption and Decryption process. There is 56-bit same key has been used for both encryption and decryption and round-key generator generates the different round key for each round. The linear cryptanalysis attack could break the DES algorithm and made it unconfident algorithm. Several published brute force attacks started to fail DES algorithm. The NIST started looking for replacement of DES algorithm because of its failure but, the disadvantage being that it has only 56 key lengths which could be easily broken.

In order to increase the reliability, National Institute of Standards and Technology (NIST) proposed 15 highly secured algorithms by which the security of transmitting data is increased. Cryptography is a form of security in which the input data is converted to encrypted data and is transmitted in the Encryption module and in the decryption module; the encrypted data is converted again to decrypted data which is same as the input data. Several cryptographic algorithms have been proposed in the past few years. Some of the cryptographic algorithms are Blow fish, DES, Triple DES, SAFER, IDEA, RC4, etc. The Advanced Encryption Standard (AES) algorithm was selected as the winner algorithm by NIST [1] (National Institute of Standards andTechnology), specifications required 128 bits block size and three different key sizes of 128, 192 and 256 bits, should be an open algorithm. The NIST declared that Rajndael cipher was selected as Advanced Encryption Standard (AES). This is the federal standard to protect the sensitive information

AES has already received widespread use because of its high security, high performance in software implementations. AES is a 128 symmetric data block cipher with128, 192 or 256 bits key. The data block is described in a 4x4 array known as state array [2]. The data block is sent through four basic functions: Substitute bytes, Shift Rows, Mix Column and Add Round Key. These four steps make one round of the AES. The number of rounds depends upon the Key length (N_k) words. The key length (N_k), Block Size (N_b) and the Number of rounds (Nr) combination for AES-128, AES-192 and AES-256.

The Mix Column round is excluded for the last round. The decryption is the reverse order of the ciphering process. Operations are just similar and inverse of the encryption process. Many implementations are done in software but it seems to be too slow for fast applications such as routers and wireless communication systems [3]. The implementations are physically secure since attacking from outside is very difficult. Reduction in the hardware resources to gain a compact and efficient implementation circuit is ever increasing in demand [4].Hence, the less area implementation of AES - CTR architectures may be suitable for some low end embedded applications.

AES algorithm is an iterative algorithm, which requires many computation cycles. A software platform can provide the high speed encryption of data, specially used for real-time applications. Audio/video content encryption is required in real-time for the business deals via video conferencing. Therefore, dedicated Software implementation is inevitable in such applications. Software implementation can be done through different architectures trading with area and power consumption. At any time, designing best architecture for a particular design with low area and low latency is a challenge. Software implementations of particular design with low area and low latency are a challenge because AES algorithm implementation vary according to the application.

Some applications like e-commerce servers require very high security but others require a medium range for the design of cell phones. Some others require very low area and power implementations to be used for the application like RFID cards.

The rest of the paper has been organized as follows. Section II renders the basic AES algorithm. Section III describes proposed architecture; Section IV provides the results and compares with the existing work done. Section V concludes with some final comments.

II. Ixias Algorithm

The National Institute of Standards and Technology (NIST) announced that Rajndael planned by two Belgium researchers Joan Daemen and Vincent Rijment was adopted as Advanced Encryption Standard (AES) for encryption and decryption of blocks of data. The draft is published in December 2001, under the name as FIPS-197 (Federal Information Processing Standard number 197).The criteria defined by selecting AES fall into three areas Security, Implementation and cost of the algorithm

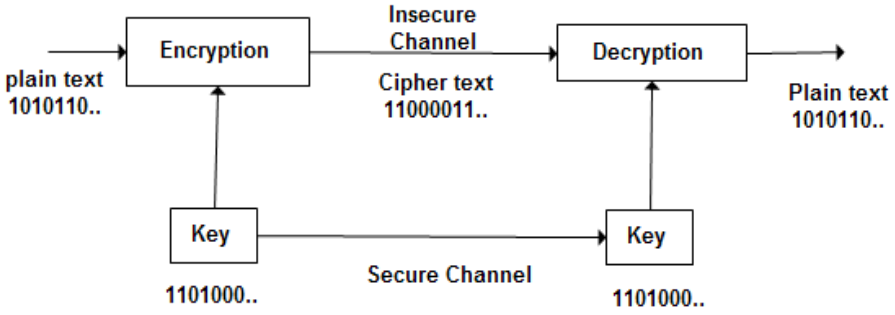


Fig.1 Symmetric Key Cryptography

The main emphasis was the security of the algorithm to focus on resistance of cryptanalysis attacks, implementation, cost should be less so it can be used for small devices like smart cards. The AES algorithm is a private key block cipher. It encrypts data of block size 128 bits. Each key may be of size either 128 bits, 192 bits or 256 bits. AES uses three different types of round operations. Table I shows the number of rounds in three versions of AES. But, in each version final round key is 128 bits.

Table I:Round key size and number of rounds in three versions of AES.

Cipher Key Size	No. Of Rounds (Nr)	Round Key size
128 bits	10	128 bits
192 bits	12	128 bits
256 bits	14	128 bits

The initialization is done by adding first round key (128 bits) with 128 bits plain text. In subsequent steps, the following transformations are done**:** Sub Bytes, Shift Rows, Mix Columns and Add Round Key**.**

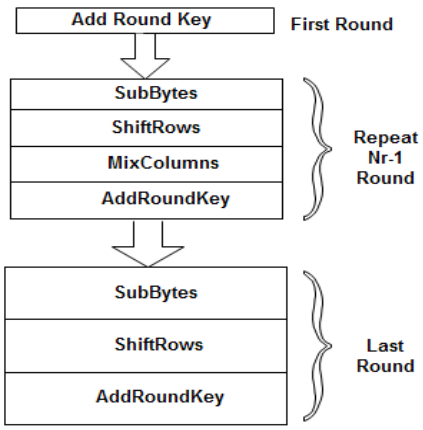


Fig.2 AES Structure

A. Subbytes/inverse subbytes transformation

The first transformation, Sub Bytes, is used for encryption and inverse SubBytes used for decryption. The SubBytes substitution is a nonlinear byte substitution that operates independently on each byte of the State using a substitution table (S-box) as shown in fig.3. Take the multiplicative inverse in the finite field GF (2^8) and affine transform to do the SubBytes transformation as shown in fig. 4. Inverse affine transform have to find for inverse SubBytes transformation then multiplicative inverse of that byte. The SubBytes transformation is done through S-box.

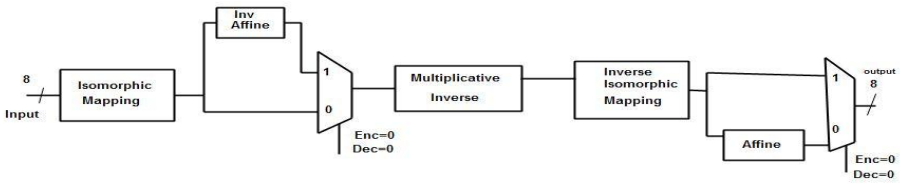


Fig.3 SubByte/Inverse SubByte Implementation

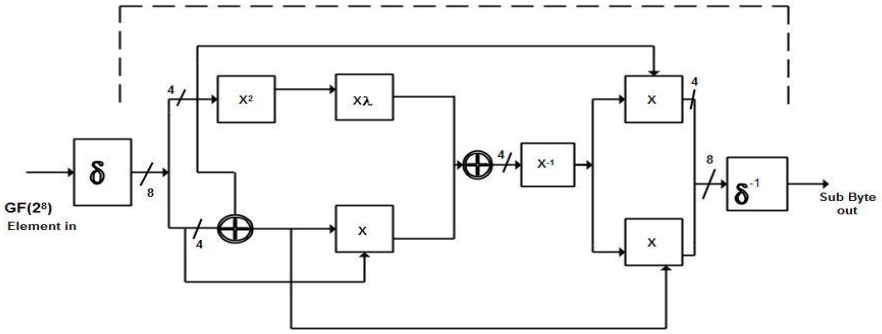


Fig.4 Implementation of Sub Byte Transformation

B. Shift Rows/Inverse shift Rows transformation

The transformation is called ShiftRows performs in encryption, in which rows are cyclic shifting to the left. The number of shifting depends upon the row number of the state matrix as shown in fig. 5. First row no shifting, second row one byte, third row two bytes and fourth row three byte shifting left. In the decryption, InvShiftRows transformation performs the right cyclic shifting operation inverse of ShiftRows; number of shifting depends on number of row number.

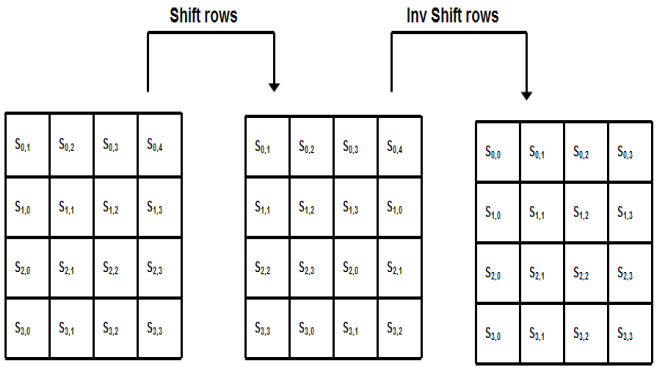


Fig.5 ShiftRows/Inverse ShiftRows transformation

C. MixColumns/ InvMixColumns transformation

The MixColumns transformation functions after the ShiftRows on the State column-by-column, considering each column as a four-term polynomial. Inverse MixColumns are the inverse process of MixColumns which is used in the decryption of cipher text. The columns are considered as polynomials over GF (2^8) and multiplied modulo x^4+1 with a fixed polynomial A (x), given in equation 2.1.

$$A(x) = \{03\}\, x^3 + \{01\}\, x^2 + \{01\}\, x + \{02\}.$$

The algorithm for MixColumns and Inverse MixColumns involves multiplication and addition in GF (2^8). The MixColumns multiplies the rows of the constant matrix by a column in the state.

D. AddRoundKey transformation

The AddRoundKey adds the round key word with each column of state matrix. It is similar to MixColumns; the AddRoundKeyproceeds one column at a time. The most important in this Transformation, that it includes the cipher key. The state column will get XOR with key which is generated by key generator and create another state.

E. Key Expansion Logic

The initial RoundKey will be the same as the initial key in encryption whereas in decryption it will be the last RoundKey. The round keys will be generated using a unit called the key generation unit. This unit will be generating 176, 208 or 240 bytes of round keys depending on the size of the used key. The RoundKey for all other rounds are generated from the Key Expansion logic.

III. Proposed Architecture

To encrypt data using AES with counter mode as shown in fig. 6 (AES-CTR), the plaintext is divided into 16-byte blocks M_1,M2…..Mn; then, AES ciphering is performed on series of blocks called counters xi (see Fig. 1) to generate corresponding blocks $E_k(x_i)$of pseudorandom numbers [2]. The plaintext blocks M1...Mn... are combined by XOR operation with the $E_k(x_i)$ blocks to produce the cipher text (C_1,...C_n) given by $C_i = M_i \oplus E_k(x_i)$.If the same counter x_i was used for two different messages M_i and M_j, the XOR of the cipher texts will be $[M_i \oplus E_k(x_i)] \oplus [M_j \oplus E_k(x_i)] = Mi \oplus Mj$. In this case, when the message M_i contains a series of zero, $M_i \oplus M_j = M_j$, and the transmitted message loses its encryption. Thus, the uniqueness of the counter x_i is extremely important to provide a high degree of security.

This makes it a nonce, because it guarantees that two identical packets sent, from the same sender, with the same key and belonging to the same block, does not ever give the same results $E_k(x_i)$ [2].Each 16-byte block M_i uses its own varying counter x_i. To decrypt the received data and retrieve the original plaintext, the receiver computes $M_i = C_i \oplus E_k(x_i)$. Clearly, the receiver needs the counter value x_i to get M_i. The x_i counter (16 bytes) is composed [2] of:

1) headers (2 bytes: 1 byte options and 1 byte Priority);
2) MAC address of the transmitter (6 bytes);
3) Packet number, PN (6 bytes);
4) Key counter (2 bytes).

The transmitter increments PN for each encrypted packet. The key counter is incremented when PN ever reaches its maximum value. The **nonce** must never be repeated within the lifetime of a used key, and the role of the packet and key counters is to prevent its reuse, thus providing a high degree of security. Also, the MACaddress is to make sure that two stations, sending at the same time, will never have the same counter; the sender does not need to include it with the packet since the receiver can infer its value for each block. A simplified AES (S-AES) algorithm was developed to reduce execution time [6]. The structure of S-AES (see Fig. 2) is exactly the same as AES. The differences are in the key size (16 bits), the block size (16 bits), and the number of rounds (2 rounds).

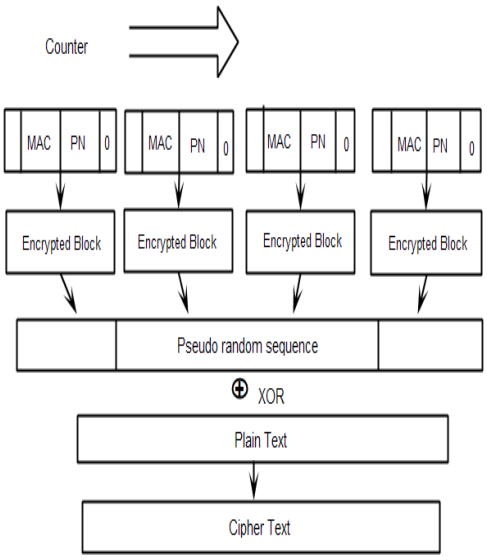


Fig.6 AES Counter Mode

B. Analysis and Proposition

The AES algorithm is not applied directly to the data, but rather, to the counters (CTR mode). The resulting random sequences are then used to encrypt the plaintext. This mode is more secure than the electronic codebook (ECB) mode, where the encryption algorithm is applied directly to the plaintext, i.e., $C_i = E_k (M_i)$. The ECB mode presents serious problems and is not recommended at all. The disadvantage of this mode, and contrary to the CTR mode, that identical plaintext blocks Mi are encrypted into identical cipher text blocks C_i; thus, it does not hide data patterns well. It does not provide serious message confidentiality. Among all the existing modes of operation in the literature (ECB, CBC, OFB, CTR, and CFB), CTR mode is widely used and it is well suited to operate on a multiprocessor machine where blocks can be encrypted in parallel. The CTR mode transforms the encryption algorithm to a stream cipher whose role is only to generate random sequence number. Consequently, the AES-CTR becomes a stream cipher, and its role is simply to produce pseudorandom sequence number (which is combined with the plaintext).For ZigBee network, the AES-CTR is very secure, but it is complex and heavy (computational and memory requirements). This makes it too slow to meet the real-time requirement of most applications. In addition, it needs complicated Software configuration, while the ZigBee sensors are small and cheap devices characterized by small memory and Designed for limited power consumption.

On the other hand, the simplified version S-AES is quick and light, but it is robust enough. For these reasons, we propose to replace the AES-CTR by astream cipher. The latter can be significantly lighter and faster than the AES-CTR, and is able to perform the same function.

IV. Implementation Results And Discussion:

Fig.7& Fig. 8 represents Simulation waveform 128 bit S-Box&MixColumn.

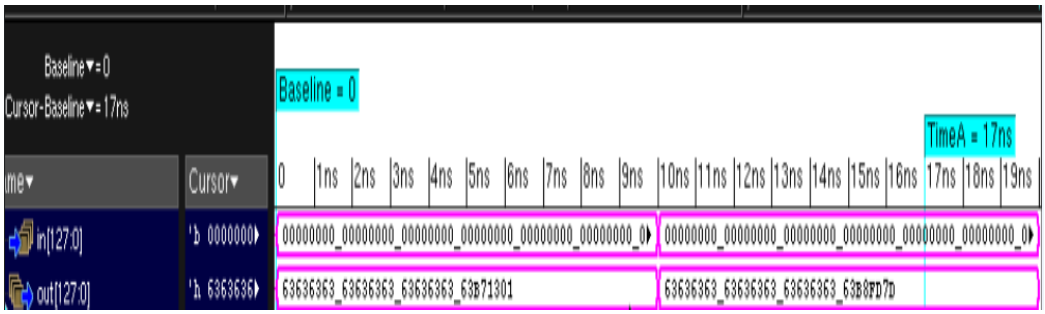


Fig.7Simulation Waveform 128 bit S-Box

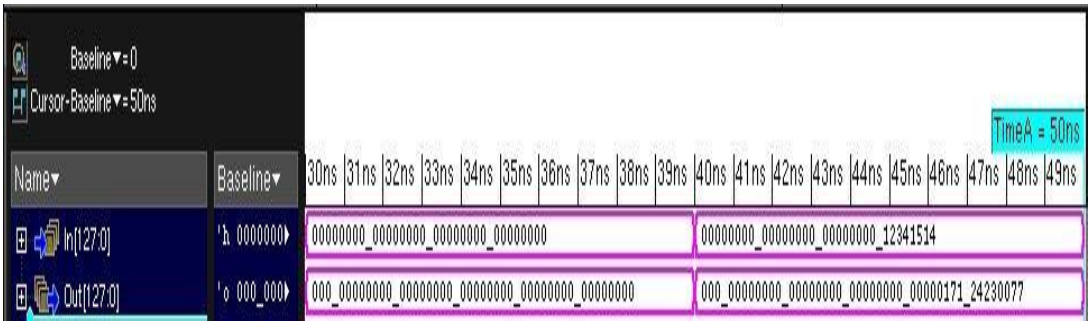


Fig.8 Simulation Waveform 128 bit MixColumn

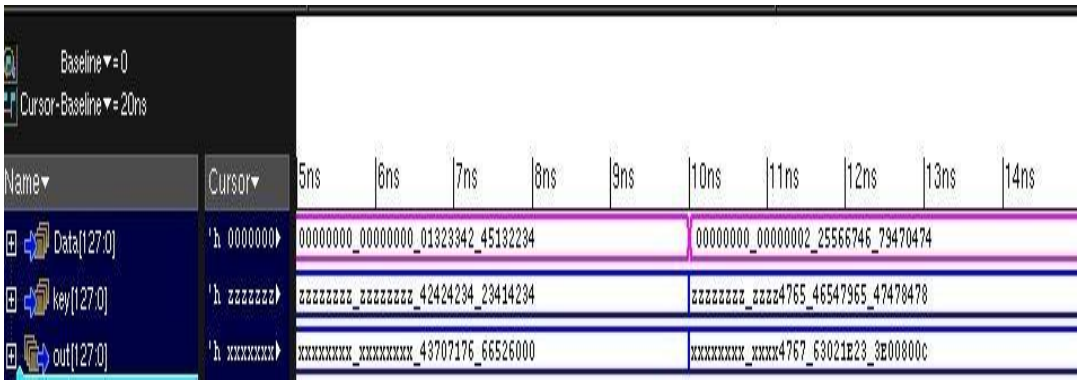


Fig.9 Simulation Waveform 128 bit AddroundKey

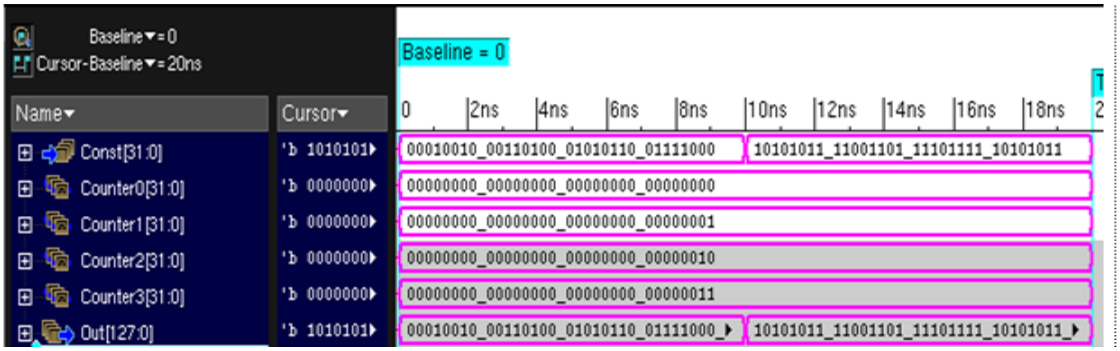


Fig.10 Simulation Waveform 128 bit Counter Mode

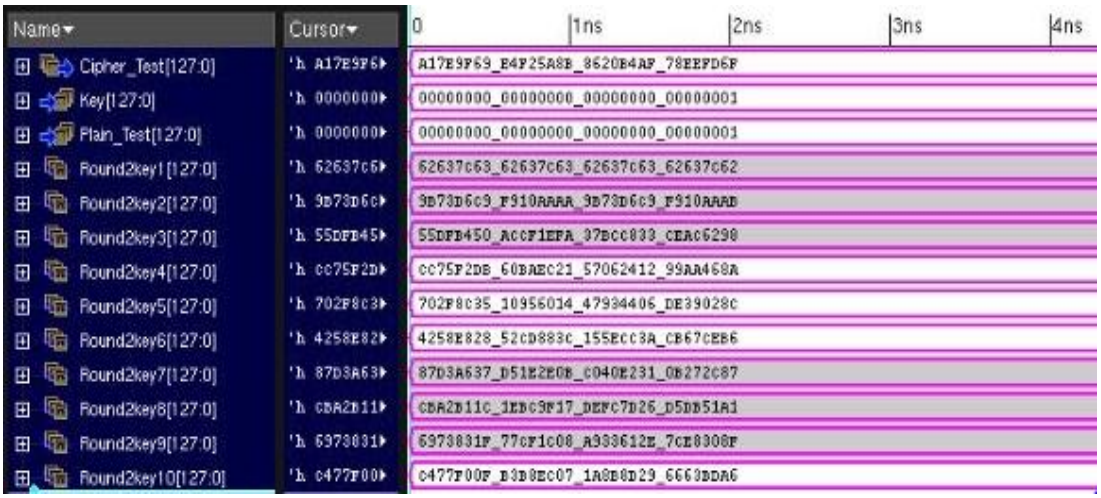


Fig.11 Simulation Waveform of the Basic AES Encryption

Fig.9,10 & 11 represents simulation waveform 128 bit AddroundKey, Counter Mode, Basic AES Encryption.

Fig.12 shows thesimulation waveform of the Basic AES Decryption& fig. 13 representsAES_CTR simulation waveform.

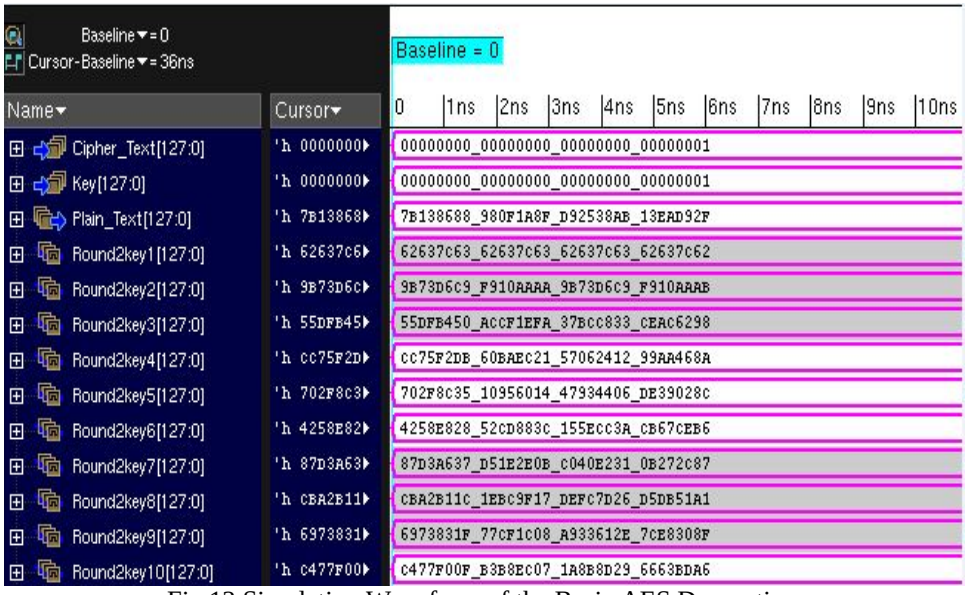


Fig.12 Simulation Waveform of the Basic AES Decryption

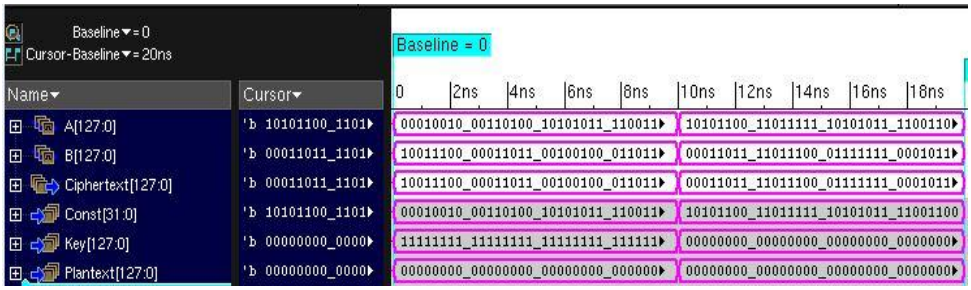


Fig. 13 Simulation Waveform of the AES_CTR

The AES & AES-CTR design were modelled and synthesized used TSMC'S 180nm standard cell library using RTL complier & physical design implementation uisng SOC Encounter Digital. Drastic improvement of power and area are abided along with the improvement of security of the entire system.

VOverall AES/AES_CTR Implementation

The AES algorithm is implemented using a single Substitute Byte and on the fly key generation is used in this implementation because the pre-computed key generation takes extra memory to store the keys for all rounds of operation. And it is used for key generation and Mixcolumn implementation. A single S-Box is used to implement the AES Algorithm. The Substitute byte uses the S-Box 16 times to transform the input 128 bit data. Similarly the Key Schedule block repetitively used the S-Box 4 times. The ShiftRows operation is included in the Substitute Byte. So there is no need of extra registers to store the values. The individual blocks in the S-Box are grouped together so that the number of transitions as well as the gates is reduced. The S-Box and the Mixcolumn are implemented with minimum number of XOR gates so as to reduce the internal transitions which consumes less power. By making use of AES_CTR mode and minimization of number of operations in the AES algorithm results in achieving low power in this implementation. The implementation results of AES encryption and decryption in 0.18 µm is shown in Table II, III,IV& V.

Table.II: Power Evaluation of AES/AES_CTR

Pre Layout (AES)

INTERNAL (mW)	SWITCHING (mW)	LEAKAGE (mW)	NET (mW)	TOTAL (mW)
624	1019.4	.045	395.39	1019.435

INTERNAL (mW)	SWITCHING (mW)	LEAKAGE (mW)	NET (mW)	TOTAL (mW)
566.87	919.21	.045	352.33	919.245

Pre Layout (AES_CTR)

Table.III: Power Evaluation of AES/AES_CTR

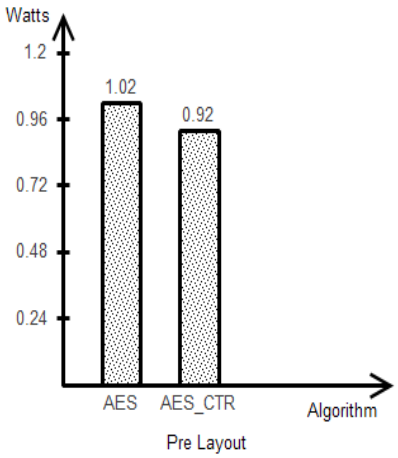
Post layout (AES)

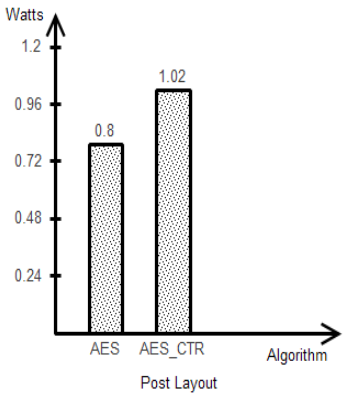
INTERNAL (mW)	SWITCHING (mW)	LEAKAGE (mW)	TOTAL (mW)
497.6	299.1	.1152	1020.435

Post Layout (AES_CTR)

INTERNAL (mW)	SWITCHING (mW)	LEAKAGE (mW)	TOTAL (mW)
516.7	507.1	.08616	1023.8

Graphical Representation





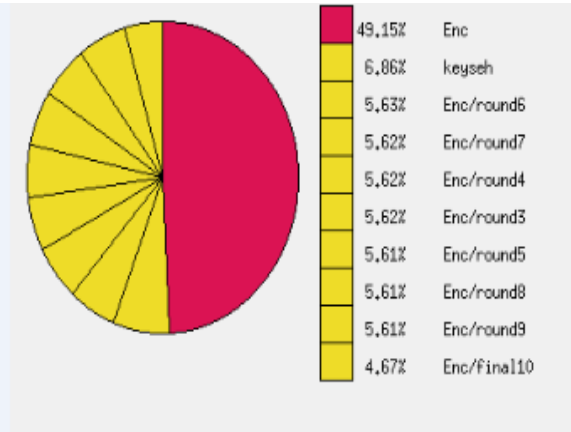


Fig.14 Instance Power Usage (AES)

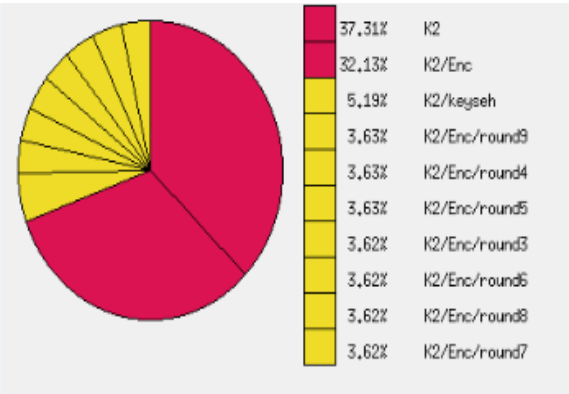


Fig.15 Instance Power Usage(AES_CTR)

Table.IV Area Evaluation of AES/AES_CTR
Pre Layout (AES)

TOTAL CELLS	TOTAL AREA (mm^2)
51740	1140.928

TOTAL CELLS	TOTAL AREA (mm^2)
51932	1146.8163

Pre Layout (AES_CTR)

Table.V Area Evaluation of AES/AES_CTR
Post Layout (AES**)**

TOTAL CELLS	No. of GATES	GATE AREA (mm^2)	TOTAL AREA (mm^2)
48336	115212	9.9792	1149.723

TOTAL CELLS	No. Of GATES	GATE AREA (mm^2)	TOTAL AREA (mm^2)
51932	114920	9.9792	1146.8163

Post Layout (AES_CTR)

The power consumption is summarized in Table II & III. This is evident from the table that static power is negligible that is desirable. Total power consumption is only 0.29%.The area consumed is found to be 0.25% in 180nm technology.

Fig, 15 & 16 represents Instance power usage of AES & AES-CTR. The complete chip layout after placement and routing in 180nm technology is rendered in fig16,17.. The colored area in the centre is the core area containing placement of standard cells. Boundary corner cells are used to provide power and ground connectivity. On all boundaries input-output pads are shown in fig. 16 & 17. Routing wires are also shown reed colored. Connectivity to power and ground nets that are VDD and VSS pads is also shown in fig.15, 16.

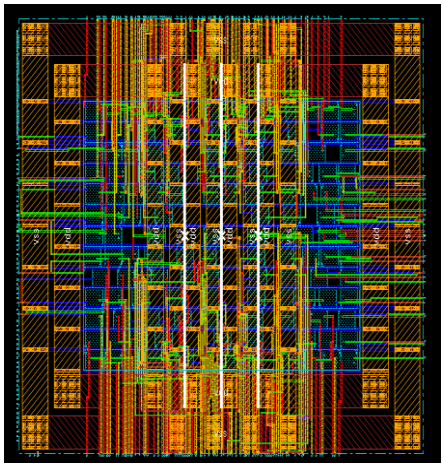


Fig.16 Complete ASIC Chip Layout (AES)

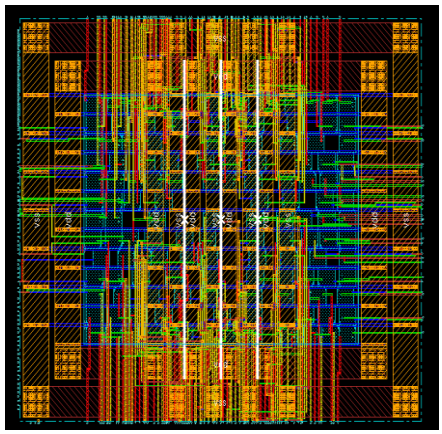


Fig.17 Complete ASIC Chip Layout (AES_CTR)

V. Conclusion

In this paper an efficient architecture of the AES algorithm is implemented in order to reduce the area and power when compared with the previous algorithms. But, AES – CTR provides better performance when compared in terms of area & power. Advanced Encryption Standard & AES – CTR architecture for the 128 bit data length and 128 bit key length was designed using Verilog and synthesized with RTL complier, physically design implementation using SOC Encounter. ASIC implementation using 180nm Technology depicts thatdecrease in overallarea and power. This design has a scope of using it in portable devices, where bulk transmission of data is required with high security.

References

[1]. National Inst. of Standards and Technology, "Federal Information Processing Standard Publication 197, the Advanced Encryption Standard(AES),"Nov.2001.

[2]. Chang.C..I, Hu.C.W, Chang.K.H, Cheng Chen.Y.C and Hsieh.C.C,"High Throughput32-bit AES Implementation in FPGA",pp1806 - 1809,2008.

[3]. Samiee.H,Atani.R.E,"A Novel Area-Throughput Optimizes Architecture for the AES International Conference on Electronic Devices,Systems and Applications,pp 29-32,2011.

[4]. Luo.A.W, Qing Ming Yi and Min Shi, "Design and Implementation of Area-optimized AES based on FPGA",pp 743-746,2011.

[5]. Morioka, S., Satoh, A."An Optimized S-Box Circuit Architecture for Low Power AES Design". In: Kaliski Jr., B.S., Koç, Ç.K., Paar, C. (eds.) CHES 2002. LNCS, vol. 2523, pp. 172–186. Springer, Heidelberg (2003).

[6]. Mestiri, H., Machhout, M., Tourki, R., "Performances of the AES design in 0.18µm CMOS technology," Design & Technology of Integrated Systems in Nanoscale Era (DTIS), 2012 7th International Conference on, vol., no., pp.1-6, 16-18 May 2012.

[7]. Yunping Liang; Ye Li, "A low-power and cost-effective AES chip design for healthcare devices," Biomedical and Health Informatics (BHI),2012,IEEE-EMBSInternational Conference on, pp.795-798, 5-7 Jan. 2012.

[8]. Zhen-rong LI, Yi-qi ZHUANG, Chao ZHANG, Gang JIN "Low-power and area-optimized VLSI implementation of AES coprocessor for Zigbee system" The Journal of China Universities of Posts andTelecommunications, Volume 16, Issue 3, June 2009, Pages 89–94.

P.Sushma received the B.Tech in Electronics & Communication Engineeringfrom SreeDhatha Engineering college, Andhra Pradesh, India in 2011. She is currentlypursuing the M.Tech. in VLSI Design from SRM .University. Her current research interests include low-powerhigh-performance digital CMOS circuits.

Dr. J SelvaKumar,received the B.E in Electronics & Communication Engineering from MADRAS University, India in 1999. He received his M.Tech from Anna University, 2003 &Ph.D. degreefrom SRM University, 2013 respectively. His current research interests are Low power& Reconfigurable VLSI architecture Design.