Design of low power S-Box in Architecture Level using GF

N.Shanthini[1], P.Rajasekar[1], Dr. H.Mangalam[1]

[1]Asst. prof, Department of ECE, Kathir college of Engg, Coimbatore

E-mail – rajasekarkpr@gmail.com

Abstract - Information security has become an important issue in the modern world and also the technology is going to increase very fast.Data encryption & decryption method were more popular for real-time security communication application used in nowadays. For that purpose AES has to be proposed. One of the most critical problem in AES is the power consumption. In this paper presents an optimized composite field arithmetic based S-Box implemented in four stage pipeline.Here we mainly concentrate the power consumption of S box which is the most power consuming block in AES. The construction procedure for implementing Galois Field (GF) combinational logic based S-Box is presented here. S Box operation is divided in to the GF based multiplication and inverse operation and illustrated in a step-by-step manner. The XC2VP30 device of Xilinx FPGA is used to validate the power with VHDL code for the proposed architecture. Power consumption has been measured by Xpower analyser tool in ISE 14.7 design suite.

Keywords - AES,S-Box, composite field arithmetic, GF, Pipelining, FPGA,VHDL.

INTRODUCTION

One of the most important think in the modern world is the information because without information we cannot doing anything. The evolution of information technology and in particular the increase in the speed of processing and power consumption devices has necessitated the need to reconsider the cryptographic algorithms used. So it's necessary to encrypt and decrypt our information. Encryption normally hides our original message into unreadable form for anyone at the same way decryption change the unreadable form into readable form for the respective person. Cipher system is one of the security mechanism to protect any information from unauthorized person or any public person. Cipher systems are usually subdivided into block ciphers and stream ciphers. Block ciphers operates on simultaneously encrypts the groups of characters, and also the stream ciphers usually operate on the individual characters of a plain text message one at a time, cryptographic algorithms used. There are two types of encryption algorithm Private(symmetric key)& Public where as Private uses only one key for both encryption & decryption, Public uses two key one for encryption & another one for decryption. Substitution-permutation networks (SPNs) are natural constructions for symmetric key Cryptosystems that realize confusion and diffusion through substitution and permutation operations, respectively.In SPNs the only non-linear operation is substitution step, and it can commonly referred to as an S(ubstitution)-box,the construction of S-BOX is very difficult and it's important in AES.

cryptographically strong block ciphers that are resilient to common attacks, including both the linear and differential cryptanalysis, and also the algebraic attacks. The two Claude Shannon's properties of confusion and diffusion are strengthening the symmetric key cryptosystem where Confusion can be defined as the complexity of the relationship between the secret key and cipher text, and diffusion can be defined as the degree to which the influence of single input plaintext bit is spread throughout the resulting cipher text. The National Institute of Standards and Technology of the United States (NIST) in cooperation with industry and cryptographic communities have worked together to create a new cryptographic standard. The symmetric block cipher Rijndael was standardized by the NIST as the AES in November of 2001.AES is an Advanced Encryption Standard provides high security as compared to other encryption techniques along with RSA model. At the time of introducing AES the NIST publicly calls for nominees for the new AES. Totally 15 algorithm has to be applied in that 5 finalists were chosen based on the Presentation ,analysis & testing. From that 5 the one algorithm will be chosen as the successful one that one is Rijndael. Finally Rijndael AES cipher is adapted which is a Symmetric key encryption standard. This algorithm is proposed by the two Belgian cryptographers Vincent Rijmen and Joan Doemen. In Advanced Encryption Standard (AES) symmetric-key blockcipher, the construction of cryptographically strong S-boxes with efficient hardware and software implementations in these cryptosystems has become a topic of critical research. The basic difference between the normal AES & Rijndael AES is that in the Normal AES fixes block length to 128 bits & support key length of 128,192,256 were as the Rijndael AES block & key length can be independently fixed to any multiple of 32,ranging from 128 to 256 bits.In this paper we investigate a design methodology for low power S-Box because the S-Box is one of the non-linear operation in AES. The FPGA implementation of the architecture is done along with comparison of some exsisting system.

The remaining part of this paper as follow: Section II describe the AES operation .The S-Box construction method was described in Section III. Section IVcontain the Proposed S-Box architecture. The simulation result & conclusion are drawn from Section V, Section VI respectively.

AES Encryption algorithm

In previous the DES was used but it can support only 56 bit key. The AES is a symmetric block cipher, which uses the same key for both encryption and decryption. It has been broadly used for different applications, like smart cards, cellular phones, website servers, automated teller machines etc. The process of generating cipher is Similar to other symmetric ciphers, the AES applies round operations iteratively to the plaintext to generate the cipher text. There are four transformations in a round operation: SubBytes, ShiftRow, MixColumn and AddRoundKey. The subbyte is a non-linear operation where one byte is substituted for another based on the algorithm we have to use. In the shiftrow operation data is shifted with in row. Row 0 is not shifted, Row 1 is shifted 1 byte like wise. The mixcolumn operation has perform mixing of data within columns. The actual encryption is performed in the add round key function, when each byte in the state perform xor operation with the subkey.

The AES process can be defined in three types based on length of the key used for the generating the cipher text which are AES 128, AES192, AES256. In this operation, the AES cipher maintains an internally 4 by 4 matrix of bytes called states. The state consists of four rows of bytes, each row containing Nb bytes, where N is the number of byte and b is the block length divided by 32 (4 for 128-bit key, 6 for 192-bit key, 8 for 256-bit key). At the same time key length and number of rounds differ from key to key, i.e we have to use 10 round for 128-bit key,12 round for 192-bit key,14 round for 256-bit key. The last round operation is different from the Previous other rounds as there is no mixcolumn transformation. The AES encryption & decryption operation is shown in Fig1.

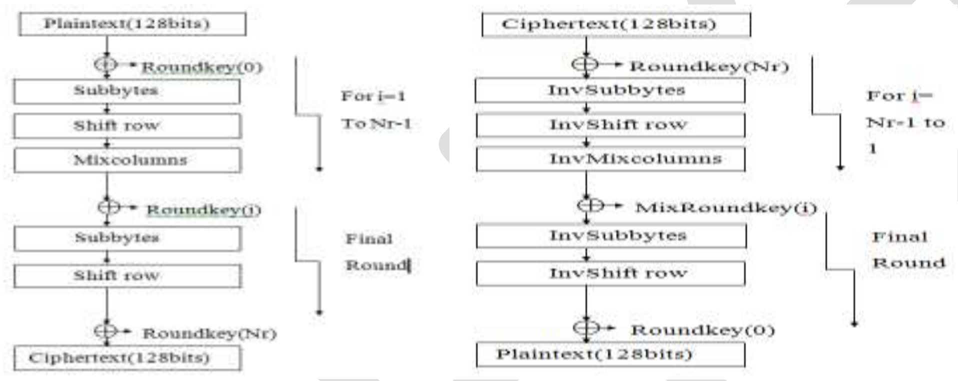


Fig 1: AES encryption and decryption algorithm

S-Box Transformation

The Sub Bytes transformation is a nonlinear byte substitution that operates independently on each byte of the State using a substitution table (S-box). ThisS-box, was usually invertible, and it can construted using two method :

1.Look up table

2.Composite field arithmetic

In that Look up table all the values are predefined based on the ROM so the area and memory access & latency is high. So our method is based on the composite field arithmetic it contain two main operation as follows:

(1) Perform the multiplicative inverse in GF(2^8).

(2)Perform the affine transformation over GF(2).

The GF stands for Galois Field. The Arithmetic in a finite field(Galois Field) is usually different from the standard integer arithmetic. The finite field should contain the limited number of elements. The finite field with (p^n)element is denoted GF(p^n), wherepis a prime numbercalled the characteristic of the field andnis a positive integer. Aparticular case is GF(2) which has only two elements (1 and 0), where addition is exclusive OR (XOR) and multiplication is AND. The element “0” is never invertible, the element “1" is always invertible and inverse to itself. Therefore, the only invertible element in GF(2) is “1". Since the only invertible element is “1" and the multiplicative inverse of “1" is also “1", division is an identity function.

The individual bits in a byte representing a GF(2^8) element can be viewed as coefficients to each power term in the GF(2^8) polynomial. For instance, {10001011}2 is representing the polynomial $q^7+ q^3+ q + 1$in GF(2^8). From [2], it is stated that any arbitrary polynomial can be represented as bx + c, given an irreducible polynomial of $x^2+ Ax + B$.

Thus, element in GF(2^8)may be represented as bx + c in that b is the most significant nibble while c is the least significant nibble. So the multiplicative inverse can be construted using the equation below,

$$(bx + c)^{-1} = b(b^2\text{B+bcA+}c^2\text{)-1 x+(c+bA)}(b^2 B + bcA + c2) - 1$$

where A=1, B=λ so that the equation become

$$(bx + c)^{-1} = b(b^2\lambda\text{+bc+}c^2\text{)-1 x+(c+b)}(b^2\lambda + bc + c2) - 1 \longrightarrow \quad (1)$$

Proposed S-Box Design Method

This section says that the multiplicative inverse computation will first be covered and the affine transformation will then follow to complete the methodology involved for constructing the S-BOX for the subbyte operation.For the invsubbyte operation,that can reuse multiplicative inversion module and combine it with the inverse affine transformation.So the multiplicative inverse can be constructed using the equation **1**,

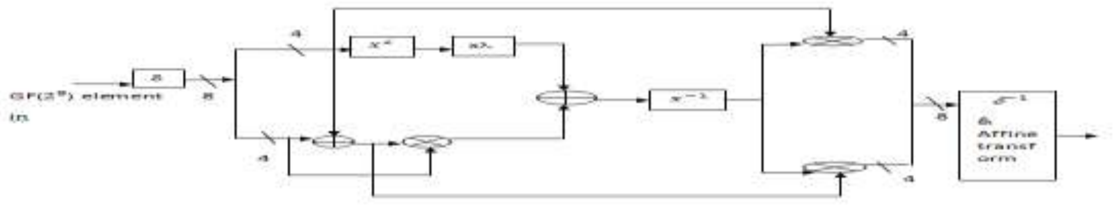


Fig 2 : Show the block diagram for S-box

Show the Description for the building blocks of the S-Box

δ	=Isomorphic mapping to composite field
x^2	=Squarer in GF(2^4)
$x\lambda$	=Multiplication with constant,λ in GF(2^4)
$\oplus$	= Addition operation in GF(2^4)
x^{-1}	= Multiplicative inversion in GF(2^4)
X	= Multiplication operation in GF(2^4)
δ^{-1}	= Inverse isomorphic mapping to GF(2^8)

Affine Transform

The affine transform is normally should improve our result. It’s the second building for the composite field arithmetic based S-Box. Our proposed affine transform & Inverse affine transform as follows:

$$\delta = b_i \oplus b_{((i+4) mod\ 8)} \oplus b_{((i+5) mod\ 8)} \oplus \quad b_{((i+6) mod\ 8)} \oplus b_{((i+7) mod\ 8)} \oplus d_i \quad \longrightarrow (2)$$

Where d = {01100011},I = 0 to 7

$$\delta^{-1} = b_{((i+2) mod\ 8)} \oplus b_{((i+5) mod\ 8)} \oplus b_{((i+7) mod\ 8} \oplus d_i \quad \longrightarrow \quad (3)$$

Where d={00000101},I = 0 to 7

Isomorphic and Inverse Isomorphic mapping

Computation of the multiplicative inverse in composite fields cannot be directly applied to an element which is based on GF(2^8) .So for that we have to decomposing the more complex GF(2^8) to lower order fields of GF(2^1), GF(2^2), GF($(2^2)^2$).To accomplish this ,the following irreducible polynomials are used .

$GF(2^2) \Rightarrow$ GF(2) $:x^2 + x + 1$

GF($(2^2)^2$) $\Rightarrow$ GF(2^2) $:x^2 + x + \phi$

GF($((2^2)^2)^2$) $\Rightarrow$ GF($(2^2)^2$) $:x^2 + x + \lambda$

where ϕ = {10}2 and λ = {1100}2.

The element in GF(2^8) has to be mapped to its composite field representation via an isomorphic function,δ.After performing the multiplicative inversion, the result will also have to be mapped to its equivalent in GF(2^8) via the inverse isomorphic function δ^{-1}.Let q be the element in GF(2^8) ,in that δ&δ^{-1} can be represented as 8x8 matrix, where q7 is the most significant bit,q0 is the least significant bit. The equation is given as below,

$$\delta X q = \begin{pmatrix} q_7 \oplus q_5 \\ q7 \oplus q6 \oplus q4 \oplus q3 \oplus q2 \oplus q1 \\ q7 \oplus q5 \oplus q3 \oplus q2 \\ q7 \oplus q5 \oplus q3 \oplus q2 \oplus q1 \\ q7 \oplus q6 \oplus q2 \oplus q1 \\ q7 \oplus q4 \oplus q3 \oplus q2 \oplus q1 \\ q6 \oplus q4 \oplus q1 \\ q6 \oplus q1 \oplus q0 \end{pmatrix}$$

$$\delta^{-1} \ \mathrm{X\ q} = \begin{pmatrix} q7 \oplus q6 \oplus q5 \oplus q1 \\ q6 \oplus q2 \\ q6 \oplus q5 \oplus q1 \\ q6 \oplus q5 \oplus q4 \oplus q2 \oplus q1 \\ q5 \oplus q4 \oplus q3 \oplus q2 \oplus q1 \\ q7 \oplus q4 \oplus q3 \oplus q2 \oplus q1 \\ q5 \oplus q4 \\ q6 \oplus q5 \oplus q4 \oplus q2 \oplus q0 \end{pmatrix}$$

Arithmetic operation in composite Field

In Galois Field the element q can be split into qHx+qL i.e the higher & lower order term.

Addition in GF(2^4)

Addition of two elements in Galois Field can be translated to simple bitwise XOR operation between the two elements.

Squaring in GF(2^4)

We have to take k=q^2,where k and q is an element in GF(2^4) ,represented by the binary number of {k3 k2 k1 k0}2 and {q3 q2 q1 q0}2 respectively, From that,

k3 k2 = kH , k1 k0 = kL, q3q2 =qH

q1 q0= qL So,

kH x+kL = (qH x+qL)^2

Using the irreducible polynomial x^2+x+1,and setting it to x^2 =x+1,so the higher and lower order term is given by,

kH = q3(x+1)+q2

i.e k3x+k2 = q3x+(q2+q3) $\longrightarrow$ (4)

kL = q3(1)+q2x+q1(x+1)+q0

i.e k1x+k0 = (q2+q1)x+(q3+q1+q0)(5)$\longrightarrow$

From the equation 2 & 3 the formula for computing the squaring operation in GF(2^4) is shown below.

k3 = q3

k2 = q3⊕q2

k1 = q2⊕q1

k0 = q3⊕q1⊕q0

Multiplication with constant λ

In that we take k=qλ where k={k3 k2 k1 k0}2,q={q3 q2 q1 q0}2 and λ={1100}2 are element in GF(2^4) we proceed the same procedure as seen in Addition we get,

k3 = q3

k2= q3⊕q2

k1=q2⊕q1

k0=q3⊕q1⊕q0

GF(2^4) Multiplication

Let k=qw where k={k3k2k1k0}2, q={q3q2q1q0}2 & w={w3w2w1w0}2 are element of GF(2^4)

k = kHx+kL = (qHwH+qHwL+qLwH) x +qHwHϕ+qLwL

GF(2^2) Multiplication

k=qw, where k={k1 k0}2,q={q1q0}2 & w={w1 w0} are element of GF(2^2) we get

k1=q1w1⊕ q0w1⊕q1w0

k0=q1w1⊕q0w0

Multiplication with constant ϕ

Let k=qϕ, where k={k1k0}2,q={q1q0}2 and ϕ={10}2 are element of$GF(2^2)$

k1=q1⊕q0,k0=q1.

Multiplicative Inversion in $GF(2^4)$

q is an element of $GF(2^4)$ such that $q^{-1}=\{q3^{-1},q2^{-1},q1^{-1},q0^{-1}\}$,the inverse of the individual bits can be computed as below,

$q3^{-1}$=q3⊕q3q2q1⊕q3q0⊕q2

$q2^{-1}$=q3q2q1⊕q3q2q0⊕q3q0⊕q2⊕q2q1

$q1^{-1}$=q3⊕q3q2q1⊕q3q1q0⊕q2⊕q2q0⊕q1

$q0^{-1}$=q3q2q1⊕q3q2q0⊕q3q1⊕q3q1q0⊕q3q0⊕q2⊕q2q1⊕q2q1q0⊕q1⊕q0

From the above discussion is the operation for the composite field arithmetic based S-Box .Our proposed method is the implementation of this S-Box in the four stage pipeline. So that the area, delay, power will be reduced. The diagram will shown below,

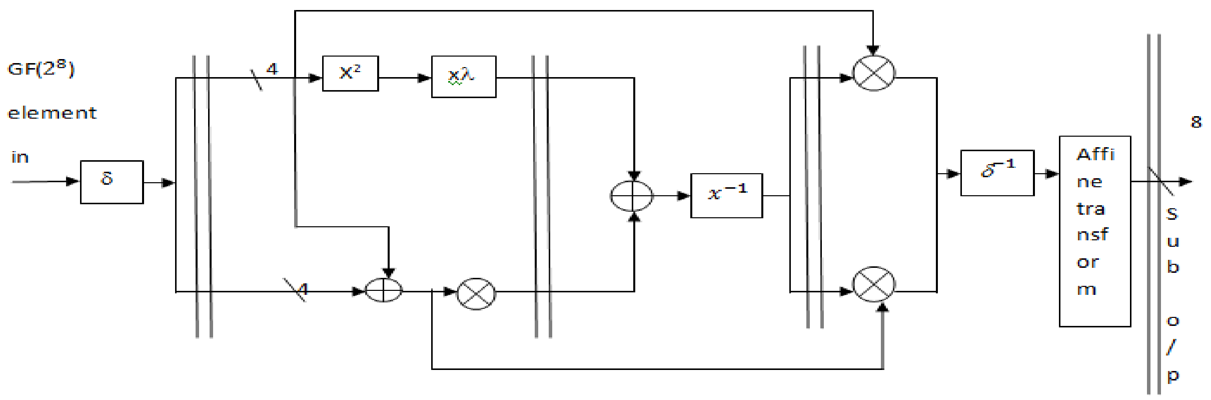


Fig 3: Proposed Pipelined implemented S-Box

Comparison Result

We design the S-Box is based on composite field arithmetic method. In this paper proposed method coding can be written using VHDL hardware description language. The XC2VP30 device of xilinx FPGA is used to validate the power with VHDL code for the proposed architecture also the power is analysed using Xilinx ISE 14.7 Xpower analyzer. Table 1 show the comparison of power ,delay and slices for conventional & proposed method.fig 3 show power report for the proposed method,

Table 1:Comparison Result for conventional&Proposed architecture, Simulation

Implementation	**No.of 4 LUT'S**	**No of Occupied slices**	**Dynamic power(W)**	**Delay (ns)**
Conventional structure(C.S)	76	40	8.278	19.866
C.S in 2 stage pipelined	83	43	5.072	15.76
C.S inv replace equ	74	40	8.278	18.986
C.S inv rep equ in 2(pipe)	81	43	5.076	14.412
C.S inv rep equ in 4(pipe)	82	43	5.064	6.275
C.S inv replace mux	76	39	8.277	18.863
C.S inv rep mux in 2(pipe)	83	44	5.16	14.627
C.S inv rep mux in 4(pipe)	88	49	8.36	6.275
Operand based S-Box(OP)	75	39	8.278	18.366
OP in 2(pipe)	86	45	5.012	18.608
OP in 4(pipe)	77	40	5.061	6.318
OP inv replace equ	75	39	8.277	18.318
OP inv rep equ in 4(pipe)	79	40	5.098	6.318
OP inv rep mux	74	39	8.278	18.318
OP inv rep mux in 2(pipe)	79	40	5.066	16.869
OP inv rep mux in 4(pipe)	76	40	5.78	6.318
Proposed architecture	85	44	5.053	6.275

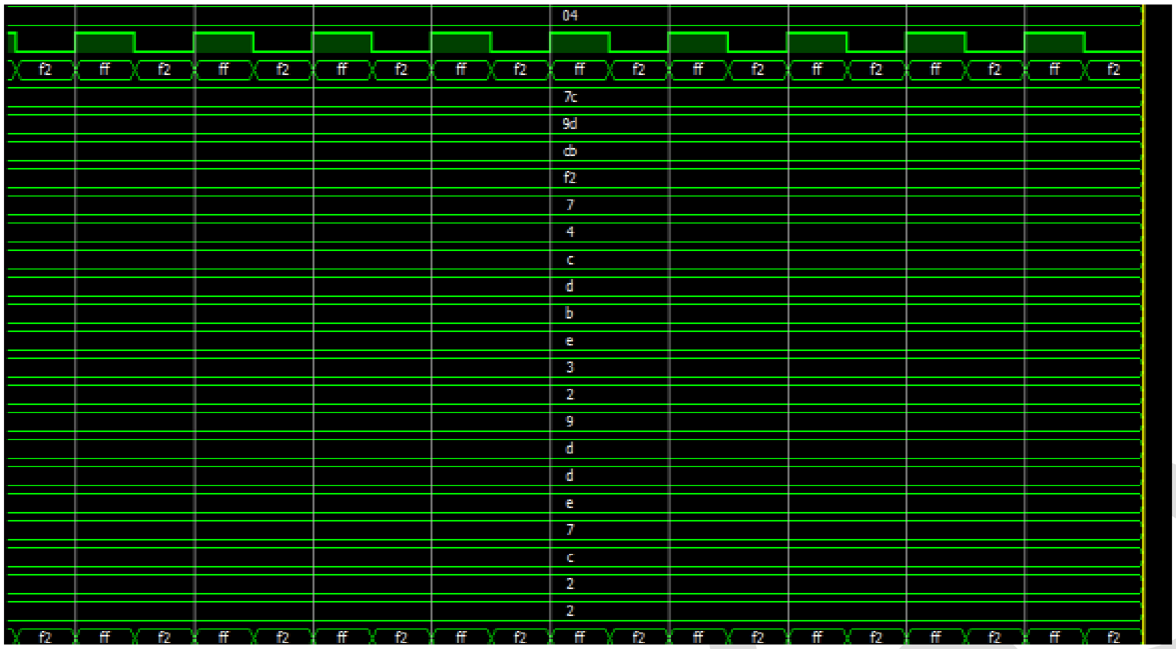
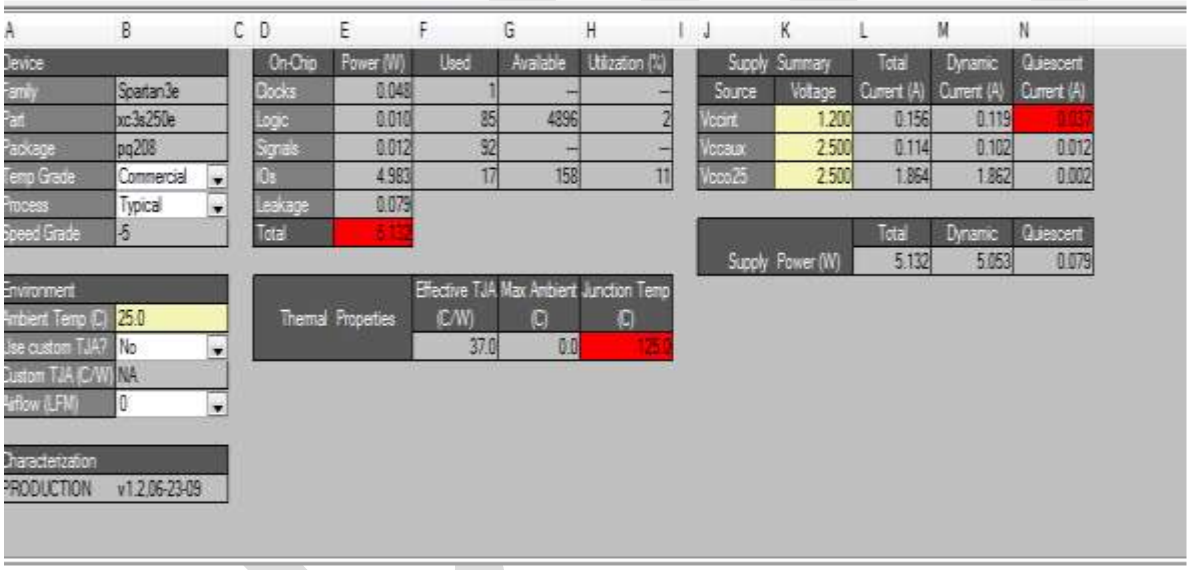


Fig 4: Simulation Result for Proposed structure



A	B	C	D	E	F	G	H	I	J	K	L	M	N
Device			On-Chip	Power (W)	Used	Available	Utilization (%)		Supply	Summary	Total	Dynamic	Quiescent
Family	Spartan3e								Source	Voltage	Current (A)	Current (A)	Current (A)
Part	xc3s250e		Clocks	0.048	1	---	---		Vccint	1.200	0.156	0.119	0.037
Package	pq208		Logic	0.010	85	4896	2		Vccaux	2.500	0.114	0.102	0.012
Temp Grade	Commercial		Signals	0.012	92	---	---		Vcco25	2.500	1.864	1.862	0.002
Process	Typical		IOs	4.983	17	158	11						
Speed Grade	-5		Leakage	0.079							Total	Dynamic	Quiescent
			Total	5.132					Supply	Power (W)	5.132	5.053	0.079
Environment			Thermal Properties		Effective TJA (C/W)	Max Ambient (C)	Junction Temp (C)						
Ambient Temp (C)	25.0				37.0	0.0	125.0						
Use custom TJA?	No												
Custom TJA (C/W)	NA												
Airflow (LFM)	0												
Characterization													
PRODUCTION	v1.2,06-23-09												

Fig 5: Power report for the proposed architecture

Conclusion

The main aim of this paper is to design and implementation of the composite field arithmetic method based S-Box. Proposed method is based on combinational logic , thus it's Power & delay is very low. The proposed approach is based on pipelining technique. In this paper we have to use four stage pipelining in S-Box design. The proposed S-Box design is only based on XOR, AND, NOT, OR logic gates. The pipelined based S-Box has low power & high speed than the conventional structure.

Acknowledgment

The author's would like to thank Kathir College of Engineering to utilize the lab facility, network resources to complete this paper in time. The suggestion & comments of anonymous reviewers, Which have greatly helped to improve the quality of this paper & knowledge.

REFERENCES:

[1]. Sumio moroika, Akashi Satoh, "An optimized S-Box circuit architecture for low power AES design", Springer – Verilog Berlin Heidelberg 2003.

[2]. Joon-HoHwang,"Efficient Hardware Architecture of SEED S-Box for the application of smart cards", journal December 2004.

[3]. P.Noo-intara, S. Chantarawong, S.Choomchaay, "Architecture for mixcolumn transform for the AES",ICEP 2004.

[4]. GeorgeN.Selimis,Athanasios P.Kakarountas,ApostolosP.Fournaris,Odysseas Koufopaylou, "A Low Power design for S-box cryptographic Primitive of AES for the Mobile end user",Journal 2007.

[5]. Xing Ji-Peng,Zou Xue-cheng,Guo Xu,"Ultra-Low power S-Boxes architecture in the AES method",journal march 2008.

[6]. L.Thulasimani,M.Madheswaran,"A Single chip design & Implementation of AES - 128/192/256encryptionalgorithm",International journal 2010.

[7]. MohammadAminAmiri,Sattar Mirzakuchaki, Mojdeh Mahdavi,"LUT based QCA realization of a 4x4 S-Box in the AES method", Journal April 2010.

[8]. Yong-sung Jeon,Young-Jin Kim,Dong-Ho Lee,"A Compact Memory-Free architecture for the AES algorithm using RS methods",Journal 2010.

[9]. MuhammadH.Rais,Mohammad H.Al.Mijalli,"ReconfigurableImplementation of S-Box using Virtex-5,Virtex-6,Virtex-7 based reduced residue of Prime number".

[10]. TomoyasuSuzaki,KazuhikoMinematsu,Sumio moroika Eita Kobayashi, "TWINE : A light weight block cipher for multiple Platforms".

[11]. Vincent Rijmen "Efficient Implementation of the Rijndael S-Box" Katholieke Universiteit Leuven,Dept,ESAT Belgium.

[12]. Akashi Satoh, Sumio Morioka,Kohji Takano and Seiji Munetoh, "A Compact Rijndael Hardware Architecture with Optimization", Springer-Verlag Berlin Heidelberg.

[13]. Saurabh kumar, V.K. Sharma, K.K.Mahapatra "Low latency VLSI architecture of S-Box for AES encryption".

[14]. Saurabh Kumar, V.K. Sharma, K.K. Mahapatra "An improved VLSI architecture of S-Box for AES encryption".

[15]. S.Limanarrag,Abdellatifhamdown, Abderrahimtragha, Sulaheddinekhamilich, "Implementation of stronger AES by using dynamic S-Box dependent of master key", journal of theroretical and applied information technology, 20th july 2013, vol.53 no.2.

[16]. Cheng wang, "Performance characterization of pipelined S-Box implementation for the AES", Journal January2014.