Volume 4, Issue 1, January 2017 ISSN: 2348-8565 (Online)
International Journal of Modern Engineering and Research Technology
Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

FPGA Implementation for Secure Data Encryption Using AES S-Box

Ankit Soni
Research Scholar, (M.Tech.)
Vindhya Institute of Technology And Science
Jabalpur (M.P.), [INDIA]
Email: ankitsoni.ec@gmail.com

Amit Sharma
Assistant Professor
Department of Electronics & Communication
Vindhya Institute of Technology And Science
Jabalpur (M.P.), [INDIA]
Email:amit_2440@yahoo.co.in

ABSTRACT

Data security has turned into an exceptionally basic part of present day processing frameworks. With the worldwide acknowledgment of web, for all intents and purposes each PC on the planet today is associated with each other. While this has made enormous profitability and extraordinary open doors on the planet we live in, it has additionally made new dangers for the clients of these PCs [2]. The clients, organizations and associations worldwide need to live with a consistent risk from programmers and aggressors, who utilize an assortment of methods and apparatuses so as to break into PC frameworks, take data, change information and cause havoc[2]. The printed material goes for planning and executing a safe information correspondence between any two clients in light of the acknowledgment of cutting edge Symmetric-key Cryptographic calculation called Advanced Encryption Standard (AES) on a FPGA based processor.

Keywords:— *AES (Encryption & Decryption) Algorithm, CPLD (Complex Programmable Logic Design), EDA (Electronic Device Automation), FPGA (Field Programmable Gate Array), ISE (Integrated Simulation Environment), IOB (Input Output Buffer), LUT (Look up table).*

I. INTRODUCTION

Cryptographic innovation is an important approach to ensure data security, and is the way to information wellbeing. In a wide range of cryptographic calculations, Advanced Encryption Standard Algorithm (AES) is exceedingly favored as it offers high security, adaptability, proficiency, advantageous utilization, and great execution [4]

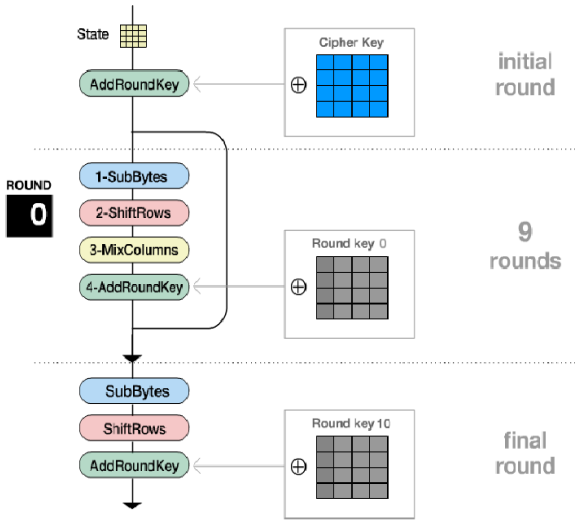


Figure 1. Encryption process Block Diagram

The AES algorithm is a symmetric Key block encryption that can cipher, and decipher, (decrypt), information. Encryption changes over information to a confused (bewildered) frame called Ciphered-content. Unscrambling of the figured content recovers the information once more into its unique shape, which is called plaintext (i.e. unique information). The

AES calculation can utilize cryptographic keys of 128, 192 and 256 bits to encode and unscramble information in the pieces of bits [7]. AES encryption is indicated as various reiterations of modification adjusts that change over the information plaintext into the last yield of figured content [5].

Each round comprises of various preparing steps, including one that relies on upon the figure key. An arrangement of turn around rounds are connected to AES utilize plan rule known as a Substitution stage organize [5]. Despite the fact that its ancestor, DES does not utilize a Feistel organize. AES works on a 4x4 cluster of bytes called state which is a network shape. The calculation comprises of performing four disconnected straightforward operations. These operations names as: Sub Bytes, Shift Rows, Mix Columns and Add Round Key [5].

II. Design Method

AES operates on a 4x4 array of bytes (referred to as "state"). The algorithm consists of performing 4 different operations[4].

2.1 SubBytes Transformation:

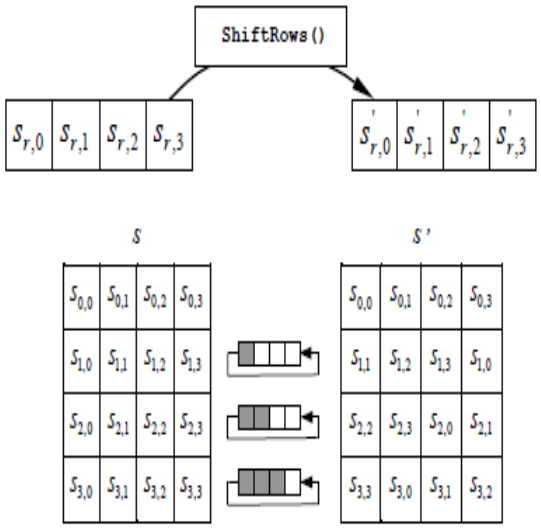
SubBytes Transformation is a non-linear byte substitution that operates independently on each byte (8 bit) of the State using a substitution table (S-box) [5]. AES use 8 bit input and 8 bit output for Substitution box.[1].

Table 1. S-Box: Substitution

x \ y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

2.2 ShiftRows Transformation:

The first row, r = 0, is not shifted. The shift value shift (r, Nb) depends on the row number, r, as follows (recall that Nb = 4) [7]:



shift(1,4) =1; shift(2,4) = 2 ; shift(3,4) = 3

Figure 2. ShiftRows ( ) cyclically shifts the last three rows in the state.

2.3 ShiftRows Transformation:

The Mix-Columns () transformation operates on the State column-by-column, treating each column as a four-term polynomial [5]. In the MixColumns step, each column of the state is multiplied with a fixed polynomial a(x).

In the MixColumns step, the four bytes of each column of the state are combined using an invertible linear transformation [5]. The MixColumns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes [5].

Columns are considered with fixed

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}\,. \quad (1)$$

Polynomial a(x), given by Let .....(2)

for 0 =c < Nb MixColumns( ) operates on the state column-by-column.

2.4 AddRoundKey Transformation:

A Round Key is added to the output of MixColumn operation (state) by a simple bitwise XOR operation. For each round of operation, separate key is generated using Key Expansion [5].

III. KEY EXPANSION

Round keys are derived from the cipher key using Rijndael's key schedule. The AES algorithm takes the Cipher Key, K, and performs a Key Expansion routine to generate a key schedule. The Key Expansion generates a total of Nb (Nr + 1) words. The expansion of the input key into the key schedule proceeds as per the functions Rotword(), Subword(), Rcon [i/Nk], Xor operations[1].

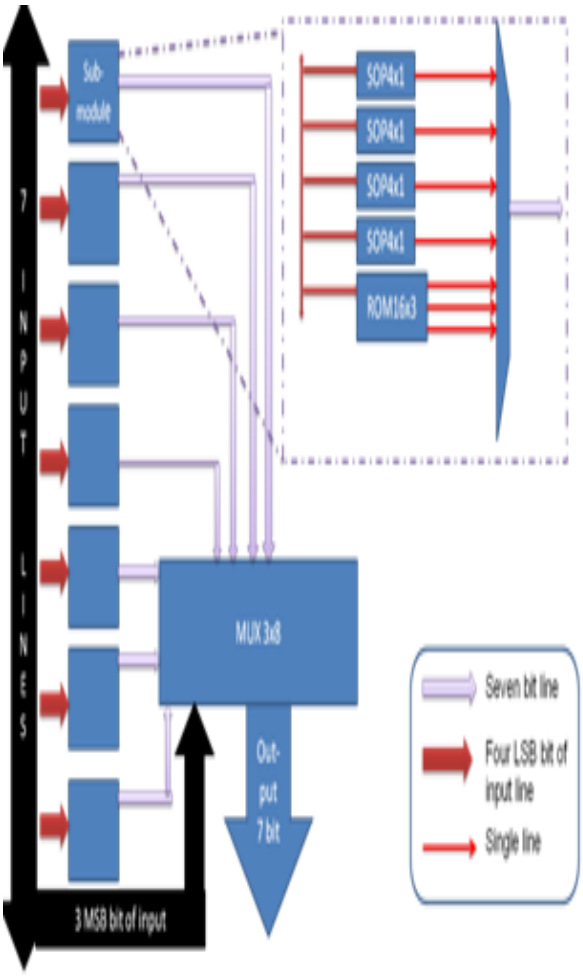


Figure 3. Proposed architecture of S-box

IV. TOOL PLATFORMS AND LANGUAGE USED

4.1 Tool: Xilinx ISE:

It is a software tool produced by Xilinx for synthesis and analysis of HDL designs. Language used: Verilog HDL: Verilog, standardized as1364, is a description language (HDL) used to model electronic systems. It is most commonly used in the design and verification of circuit's the-transfer level [5].

4.2 platform used:

FAMILY Vertex5, Device- XC4VLX80, Package-FF1148. Target FPGA is a Vertex FGPA because the same platform is been used by base papers [5].

V. RESULTS

From the simulation as shown in above slides

Key :A234567ba234a234a234567ba234a234
Result:-1

Output:	Cde5017b64cd7e93
Input:	A234567ab234a234
Output^Input:	6fd15700c6f9dac7
Avalanche:	41 bit change/64 bit

Result:-2

Output:	Df5ab6daed24e9c5
Input:	A234a234567ba234
Output^Input:	7d6e14bbee5f4bf1
Avalanche:	45 bit change/64 bit

VI. Comparative Results

Table 2. Comparative Results

Full AES design table for logical delay				
		[1] Dr. R.V. K. shirsagar, IEEE, 2012		
	OUR	General AES without pipelining	Fully pipelining AES	Fully pipelining with 10 sub pipelining AES
Logical Delay	8.439 ns	1,150.970 ns	116.867 ns	111.890 ns

Table 3. Comparative Results

Full AES design table for No. of LUT				
		[1] Dr. R.V. K. Shirsagar, IEEE, 2012		
	OUR	Gurmail Singh,2011, IJCTA	Sumanth Kumar Reddy S,2011, IJAEST	HadiSamiee, 2011, IEEE
No. Of LUT	6603	6352	8896	7865

VII. Conclusions

The substantial number of potential endorsers and the top of the line administrations to give may have incredible difficulties as far as ensured privacy and trustworthiness of both data and flagging. An improved and smaller equipment plan of the AES calculation has been depicted in this work, and with the aftereffects of its execution in FPGA innovation. These proposed S8-box strategy may be use to outline superior conservative executions of Feistel-like piece figures (AES, IDEA and so forth.). Not exclusively does this proposition accomplish a superior, yet is a standout amongst the most cost productive outlines as far as range and speed.

It can be finished up as examine that S8-box is an imperative prerequisite in AES figure era and it get utilize 38 times for creating 64 bit figure content from plaintext. proposed S8-box 7.741 ns time delay and just 64 cuts, which is less as contrast with every current work which are been talk about in section 3. The near outcomes is in section 6 additionally demonstrates that proposed work is an advanced as far as range and speed as contrast with the current work.

References

[1] Mr. Atul M. Borkar, Dr. R. V. Kshirsagar, Mrs. M. V. Vyawahare, FPGA Implementation of AES Algorithm, 978-1-4244-8679-3/11, 2011 IEEE

[2] Hassen Mestiri, Noura Benhadjyoussef, Mohsen Machhout and Rached Tourki, An FPGA Implementation of the AES with Fault Detection Countermeasure, CoDIT'13, 978-1-4673-5549-0/13/, 2013 IEEE

[3] Dr. R. V. Kshirsagar, M. V. Vyawahare, FPGA Implementation of High speed VLSI Architectures for AES Algorithm, 2012 Fifth International Conference on Emerging Trends in Engineering and Technology, 978-0-7695-4884-5/12, 2012 IEEE, DOI 10.1109/ ICETET.2012.53

[4] Shylashree. N, Nagarjun Bhat and V. Shridhar, FPGA Implementations of Advanced Encryption Standard: A Survey, International Journal of Advances in Engineering & Technology, May 2012. ISSN: 2231-1963

[5] http://www.xilinx.com/support.htm

[6] Wisniewski, Remigiusz (2009). Synthesis of compositional microprogram control units for programmable device. Zielona Góra: University of Zielona Góra. p. 153. ISB 978-83-7481-293-

[7] Thomas Jakobsen and Lars Knudsen. The interpolation attack on block ciphers. In Fast Software Encryption '97, volume 1267 of LNCS, pages 28 –40. Springer-Verlag, 1997.

[8] Eli Biham and Adi Shamir. Differential Cryptanalysis of the Data Encryption Standard. Springer-Verlag, 1993.

[9] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Advances in Cryptology—Eurocrypt '93, volume 765 of LNCS, pages 386–397. Springer-Verlag, 1993.

[10] http://www.xilinx.com/products/silicon-devices/fpga/vertex-5.html

[11] Kaisa Nyberg. Linear approximation of block ciphers. In Advances in Cryptology—Eurocrypt '94, volume 950 of LNCS, pages 439–444. Springer-Verlag, 1995.

[12] Marko Mali, Franc Novak and Anton Biasizzo "Hardware Implementation of AES Algorithm" –Journal of ELECTRICAL ENGINEERING, Vol. 56, No. 9-10, 2005, 265- 269.

[13] Kaisa Nyberg and Lars Knudsen. Provable security against a differential attack. Journal of Cryptology, 8(1):27–37, 1995.

* * * * *