A REVIEW ON VARIOUS SECURED DATA ENCRYPTION MODELS BASED ON AES STANDARD

Ashish Dwivedi[1]

*[1]Department of Electronics Engineering, Institute of Engineering and Technology, Lucknow
Dr APJ Abdul Kalam Technical University Lucknow India [226021]*

Abstract - *In today's era there is a thing which is most important for transferring data that is encrypting the data for secure transmission. The main aim is to achieve a secured data with achievable optimized power, delay, throughput and latency. In order to achieve this thing AES standard is redesigned using a proper optimising over architecture using the help of Pipelining and over various rounds of encryption as discussed in AES standard FIPS 197 [6]. This paper shows the review of various models and their proper performance comparison over various parameters on Xilinx ISE. Various models are also given according to respected FPGA board with their maximum frequency.*

Key Words: **Rijndael, AES, FIPS, FPGA, cipher text, DES, Encryption, Throughput**

1. INTRODUCTION

Encryption represents the conversion of data into secrets code. This is the most effective way to achieve data security. In order to achieve your data file you have to access the key which is your password known as cipher key. Unprocessed data is called plain text, encrypted data is known as cipher text. There are various attacks to hack these cipher texts such as Brute force attack, Known plaintext attack, Chosen cipher text attack, Cipher text attack. The Brute force attack when done AES-128 bit up to 5th round further analysis get terminated and data is secured. There are various algorithms available in cryptography like MARS, RSA, TWOFISH, SERPENT and RIJNDAEL. Advanced Encryption Standard selected a Rijndael Cryptography algorithm for its encryption. AES is more secured from its previous version that is DES[6] which is AS short key algorithm and hence less secured than that if AES algorithm. The AES algorithm achieved its standardization by National Institute of Standards and Technology (NIST) [6]of the United States of America and ready to displace DES as encryption standard.

Cryptographic AES is currently being used in a various scenarios. The common examples are e-commerce and financial transactions, having strong security requirements. The Advance Encryption Standard (AES) is a standard acts as encryption of data. It consists of 128 bit plain test which is converted into cipher text using a key of 128bit, 192bit and 256 bit sizes having 10, 12 and 14 rounds respectively. The pipelining technology is used in various round for reducing the overall throughput of core. The size of key defines the overall performance parameter as they varies according to given key size also depends on the maximum frequency of used FPGA kit.

2. Related Work

AES implementations area unit classified into software system and hardware implementations[1]. Hardware implementation offers quicker speed, additional security Associate in Nursing and consumes less power and therefore is an attractive alternative as compared to software system implementation[12] (Karthigai Kumar and Baskaran, 2010). Implementation of algorithms on hardware may be achieved exploitation either Application Specific microcircuit (ASIC) or Field Programmable Gate Array (FPGA) devices[13] (Gaj and Chodowiec, 2009). The authors conferred ASIC implementation of the science formula (Chih-Pin et al., 2003; Liu and Luke, 2003). ASIC is Associate in nursing microcircuit designed for specific application and lacks flexibility. Moreover, increased Non continual Engineering price makes tiny volume production unaffordable. As compared to ASIC, Associate in Nursing FPGA device is reconfigurable, efficient, offers additional flexibility and needs less time to plug and thus FPGA[14] could be a in style alternative for hardware implementation (Alexandru and Fratila, 2011; Elbirt et al., 2001). FPGAs conjointly succeed far better performance than multi-core CPUs for mathematical computations. In loop unrolled structures[5], all the cipher rounds square measure unrolled and information is coiled through the cipher rounds consecutive till the complete encryption/decryption is completed (Ali et al., 2011; Standaert et al., 2003; Zhang and Parhi, 2004). The loop unrolled structure implementation, results in increased speed and accrued space utilization. The AES rule can be enforced exploitation non-pipelined and pipelined or sub pipelined techniques. The non-pipelined implementations result in optimisation of space at the value of speed (Hussain and Jamal, 2012; Rais and Qasim, 2009a, 2009b). Whereas pipelined or sub pipelined implementations end in increased outturn with increased space utilization as pipelining permits process of multiple blocks at the same time (Hammad et al., 2010; Tibeto-Burman language et al.,2015; Qu et al., 2009; Wang and HA, 2013).

3. Overview of Algorithm

The AES process block diagram is shown in figure 1 which gives us the proper understanding of the algorithm. The process is divided into these parts:

- Sub Bytes
- Shift Rows
- Mixed Column
- Add Round key

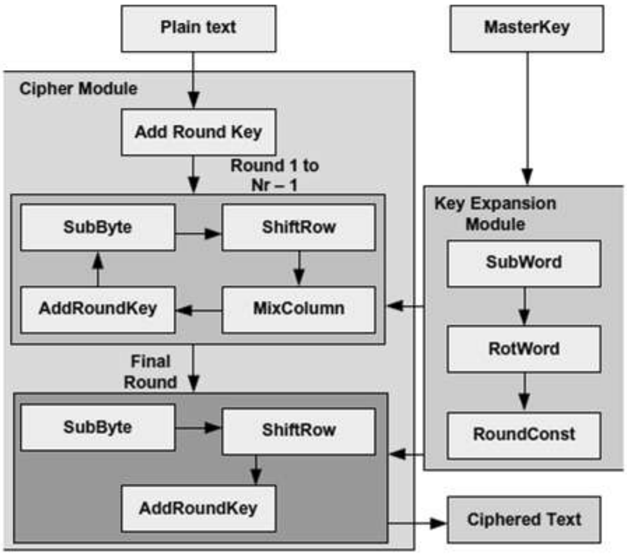
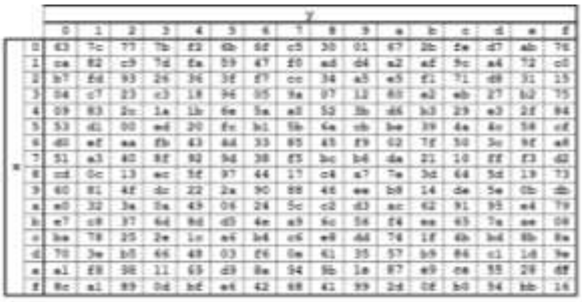


Figure 1. AES Algorithm for Encryption

Now we will discuss about the each sub steps taken in order to complete the given rounds.

3.1 Sub Bytes

This is the process in which a S-Box matrix is created using VHDL/Verilog codes. This is predefined matrix generated by Galios Field conversion[x]. SubBytes gives step by step byte substitution using Rijndael's S.Box lookup table[x]. The table has 16x16 dimension containing hexadecimal format. The hexadecimal is replaced by look up table of S-box matrix. Where XY is used to give the location of that code on S-box for substitution. Substituted matrix is given by X'Y'(figure 2).



		y														
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	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

Figure 2: Sbo'x (16 x 16 byte)

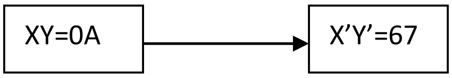


Figure 3: Sub-Byte process

3.2 Shift Rows

This process consists of circular transformation of matrix from row(0) to row(3).The shift length is different for each row. For first row no shifting is done. For second row shifting is done once, for next row twice and for last thrice. The process is shown for shift rows (figure3).

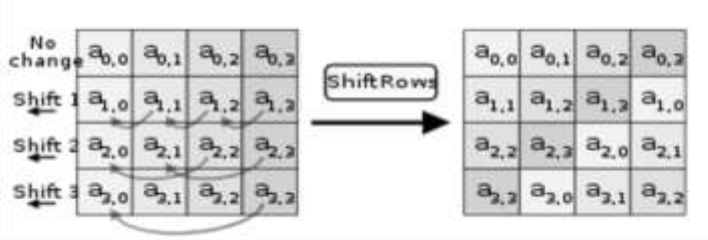

No change
Shift 1
Shift 2
Shift 3
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
ShiftRows
$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,1}$ $a_{1,2}$ $a_{1,3}$ $a_{1,0}$
$a_{2,2}$ $a_{2,3}$ $a_{2,0}$ $a_{2,1}$
$a_{3,3}$ $a_{3,0}$ $a_{3,1}$ $a_{3,2}$

Figure 4:Shift Row transformation

3.3 Mix Column

This transformation is done over each 4 byte column separately while omitted in 10th round. Columns are considered as polynomials over Galios Field (2^8) and are multiplied by a fixed polynomial c(x) modulo (x^4+1) .(figure 4)

$$C[x] = [03]x^3 + [01]x^2 + [01]x + [02] \quad ...1$$

The matrix form of mixed column is given by:

$$\begin{bmatrix} b0 \\ b1 \\ b2 \\ b3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} a0 \\ a1 \\ a2 \\ a3 \end{bmatrix} \quad ....2$$

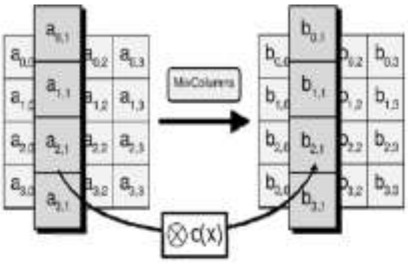

$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
MixColumns
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
⊗c(x)

Figure 5: Mixed Column transformation

3.4 Add Round Key

It is the process the converted plain text matrix is XOR-ed with the key matrix and final output obtained is cipher text. Different round key is added to each round as key transformation operation is also done at each round. Each byte of key is XOR-ed with each byte of text (figure5).

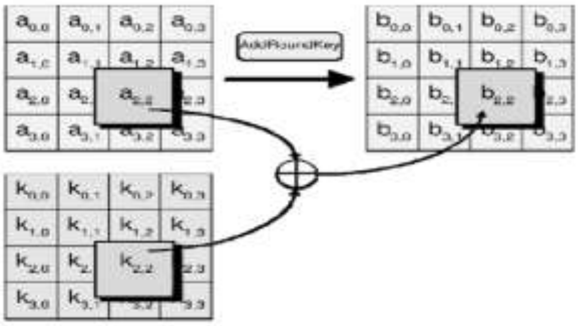


Figure 6: Add Round Key process for final output.

4. COMPARISON

Various modules are created and categorised in given table 1.Models are based on optimizing the area speed throughput latency power etc. All the models are designed on reconfigurable FPGA models and software simulation was done on XILINX ISE 14.7, I-SIM.

Further we can also provide the various FPGA models on the basis of given data of clock frequency of each devices based on previous researches. A comparison table is given on next page for that factors comparisons.

Table 1: Comparison table for performance parameter

Designs	Throughput (Gbps)	Area(Slices)	Mbps/Slice	Fmax (Mhz)	Bitwidth(bits)	Pipelined stages
Harshali Zodpe(2018)[1]	5.93	4095	1.44	463.4	128	0
Qiang et al. (2015)[2]	3.45	335	10.29	323.7	128	0
Wang and Ha(2013)[9]	40.8	5927	6.90	319.2	128	6
Henezen and Fichter	119.30	1499	8.06	233.0	128	2
Reddy and Parneeth (2011)[8]	25.89	8896	2.91	202.2	128	3

5. CONCLUSION

Encryption using reconfigurable FPGA equipment platforms is widely used to secure data and improve throughput. The Rijndael cipher[6] style is well matched for hardware use. This implementation is meted out through different trade-offs between space and speed. The trade-offs is that AES needs extra power and should not be supported by hardware, conjointly there's wide selection of equipment used for coding that is required for authentication and security. AES is programmed in software or engineered with pure hardware. The AES is the latest standard for cryptography and has been taken wide support to secure digital information.

References

1. Zodpe, H., Sapkal, A. An efficient AES implementation using FPGA with enhanced security features. Journal of King Saud University – Engineering Sciences (2018).

2. Pritamkumar N. Khose, Prof. Vrushali G. Raut, "Implementation of AES Algorithm on FPGA for Low Area Consumption",2015 International Conference on Pervasive Computing (JCPC).

3. Qiang, L., Zhenyu, X., Yuan, Y., 2015. High throughput and secure advanced encryption standard on field programmable gate array with finepipelining and enhanced key expansion. IET Comput. Digit. Tech. 9, 175–184.

4. Wenfeng Zhao, Yajun HaandMassimoAiioto,"Novel Self-BodyBiasing and Statistical Design for Near-Threshold Circuits With Ultra Energy-Efficient AES as Case Study",IEEE transactions on very large scale integration (VLSI) systems, vol. 23, no. 8, august 2015.

5. Mohammed, A., Mousa, F., Radwan, T., Odeh, M., 2011. Survey paper: cryptography is the science of information security. Int. J. Comput. Sci. Secur. 5, 298–309.

6. Fips-197, 2001. Advanced encryption standard (AES). Natl. Inst. Stand. Technol. 8–12.

7. Hussain, U., Jamal, H., 2012. An efficient high throughput FPGA implementation of AES for multi-gigabit protocols. Proc. 10th Int Conf. Front. Inf. Technol. FIT 2012, 215–218

8. Reddy, R.S.S.K., Praneeth, P., 2011. VLSI implementation of AES crypto processor for high throughput. Int. J. Adv. Eng. Sci. Technol. 6, 22–26.

9. Wang, Y., Ha, Y., 2013. FPGA-based 40.9-gbits/s masked AES with area optimization for storage

area network. IEEE Trans. Circuits Syst. II Express Briefs 60, 36–40.

10. Zhang, X., Parhi, K.K., 2004. High-speed VLSI architectures for the AES algorithm. IEEE Trans. Very Large Scale Integr. Syst. 12, 957–967.

11. Zhang, Y., Wang, X., 2010. Pipelined implementation of AES encryption based on FPGA. IEEE Int. Conf. Inf. Theory Inf. Secur, 170–173.

12. Karthigai Kumar, P., Baskaran, K., 2010. An ASIC implementation of low power and high throughput blowfish crypto algorithm. Microelectron. J. 41, 347–355.

13. Gaj, K., Chodowiec, P., 2009. FPGA and ASIC implementations of AES. Cryptogr. Eng. 235–294.

14. Alexandru, C., Fratila, R., 2011. Cryptographic Applications using FPGA Technology. J. Mobile Embedded Distrib. Syst. 3, 10–16.

BIOGRAPHIES

ASHISH DWIVEDI received the B.Tech degree in Electronics and Communication Engineering from Shri Ramswaroop Memorial College of Engineering and Management, Abdul Kalam Technical University Lucknow, India and is currently working towards his M.Tech degree in Microelectronics with the research interest in Digital VLSI and the enhancing the performance of Cryptographic Circuit from Institute of Engineering and Technology, Lucknow, Uttar Pradesh.