Vol 06, Article 01531; February 2015
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

REVIEW PAPER ON AN IMPLEMENTATION OF RIJNDAEL ALGORITHM ON FPGA

M K MANKAR[1], M V VYAWAHARE[2]
[1]II M.Tech VLSI Student, Priyadarshini College Of Engineering, Nagpur, Maharashtra, INDIA
[2]Department of Electronics Engineering, Priyadarshini College Of Engineering, Nagpur, Maharashtra, INDIA
[1]*manshree379*@gmail.com

ABSTRACT

Rijndael algorithm is accepted as a symmetric cryptography standard for transferring block of data securely. The available Rijndael algorithm is used for text data and it is also suitable foe image encryption and decryption to protect confidential image data from an unauthorized access .The design uses an iterative looping approach with block size of 128 bits and key size of 192 bits lookup table implementation of S-Box. This gives Low complexity architecture and easily achieves low latency as well as high throughput. A proposed FPGA implementation of Rijndael algorithm is presented in this project which offers high performance and flexibility to incorporate any protocol changes and gives quick solution. Xilinx software is used for simulation and optimization of the synthesizable VHDL code.

Keywords: *Rijndael algorithm (encryption, decryption),VHDL, Key expansion.*

1. INTRODUCTION

Rijndael algorithm was developed by John Daeman and Vincent Rijnmen. It was announced by National Institute Of Standards and Technology (NIST) in 1997 as a criteria of security, performance efficiency, flexibility and implementability and published the specifications of standard in Federal Information Processing Standard (FIPS) publication 197.Before long time, Data Encryption Standard (DES) was considered as a standard for symmetric key encryption. Data Encryption standard has key length of 56 bits which is considered small and can be easily broken. Rijndael can be specified with key and block sizes in any multiple of 32 bits. It has fixed block size of 128 bits and key size of 128, 192 , 256 bits.This paper deals with FPGA implementation of Rijndael encryptor / decryptor using an iterative looping approach with block size 128 bits and key size of 192 bits in 12 rounds.

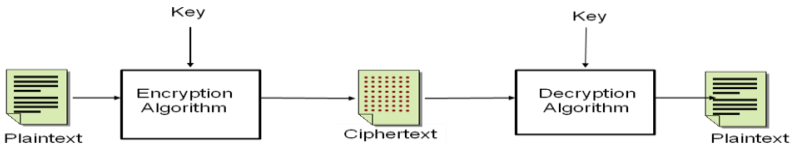


Fig.1. Basic Block Diagram of Rijndael Algorithm.

The encryption process is achieved by processing plaintext. Key expansion generates a key schedule that is used in cipher and inverse Cipher procedure and is composed of specific number of rounds. Ciphertext is a scrambled message produced as output. Decryption process is same but in reverse manner. The number of rounds to be performed during the execution of algorithm is dependent on the key length.

Table 1 Key-Block-Round comparison.

Type	Key length Nk words	Block Size Nb words	Number of Rounds Nr
Rijndael-128 bits key	4	4	10
Rijndael-192 bits key	6	4	12
Rijndael-256 bits key	8	4	14

The algorithm consists of performing four different simple operations. These operations are:

- Sub Bytes
- Shift Rows
- Mix Columns
- Add Round Key

Sub Bytes operation is a non-linear byte substitution that operates on each byte of the State using a substitution table. Shift Rows operation the bytes in the last three rows of the State are cyclically shifted over different numbers of bytes. Mix Columns step transforms each column using an invertible linear transformation. Add Round key is added to the State by a simple bitwise XOR operation. Rijndael is composed of four high-level steps. These are:

Key Expansion
Initial Round
Rounds
Final Round

Vol 06, Article 01531; February 2015
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

Key expansion is performed using key schedule. Initial round consists only an AddRound Key operation. The round step consists of a Subbytes, Shiftrows, Mix Columns and an Add Round key operation. The number of rounds in the rounds step varies from 10 to 14 depending on the key size. Finally, the Final Round performs a SubBytes, ShiftRows & an AddRoundKey operation.Decryption in Rijndael algorithm is done by performing the inverse operation of the simple operations in reverse order.

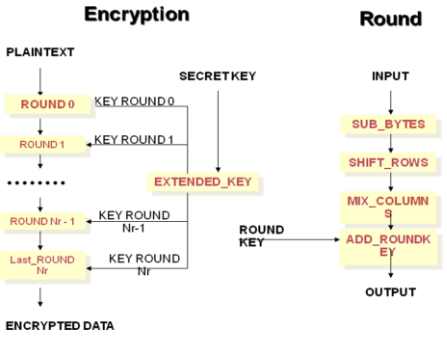


Fig.2 Basic Concept of Rijndael Algorithm

2. RIJNDAEL ALGORITHM

The Rijndael encryption and decryption processes for 128 bit plaintext shown in fig 3.

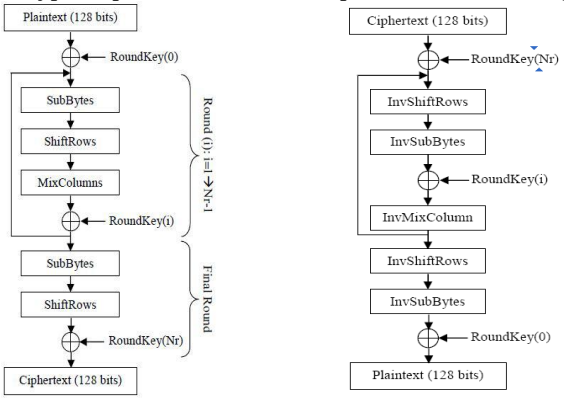


Fig.3:Encryption Structure Decryption Structure

TheRijndael algorithm is a round based algorithm and number of rounds for key length 128 bits is 10, for key length 192 bits is 12 rounds, and for 256 bits 14 rounds.

2.1. Encryption Process

In the Zencryption of the Rijndael algorithm, each round performs four transformations namely SubBytes, ShiftRows, MixColumns and AddRoundKey, while the final round does not perform the MixColumns transformation. The key used in each round which is called the round key, this is generated from theinitial key by a separate key scheduling module of Rijndael . These can be proposed as:

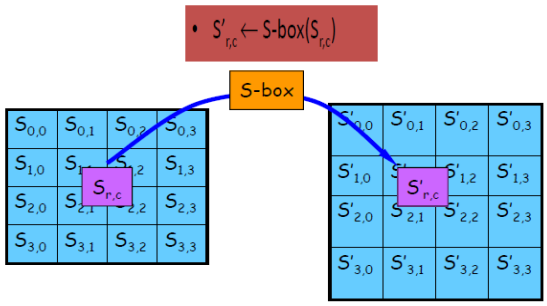
- Substitution of Bytes using S-box
- Row shifting operation using different offset
- Mixing of data within each column of state array
- Adding a round key with State
- Sub Byte Transformation : It is a non-linear byte substitution which operates independently on each byte of the state using the S-box table which contains 256 numbers.

	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

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

Fig.4: Rijndael encryption S-BOX Rijndael decryption S-BOX

Vol 06, Article 01531; February 2015
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556



- ShiftRows Transformation

In ShiftRows transformation , the rows of the state are cyclically left shifted over different offsets. Row 0 is not shifted; row 1 is shifted one byte to the left; row 2 is shifted two bytes to the left and row 3 is shifted three bytes to the left.

- Mixcolumns Transformation

In MixColumns transformation, the columns of the state are considered as polynomials over GF (2*) and multiplied by modulo $x^4 + 1$ with a fixed polynomial c(x), given by: $c(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$

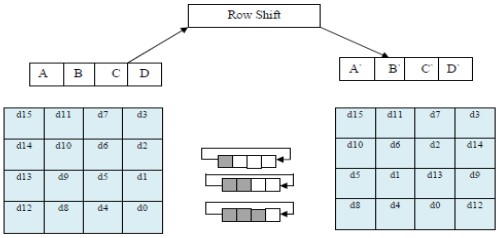


Fig 5: Shift Rows Operation

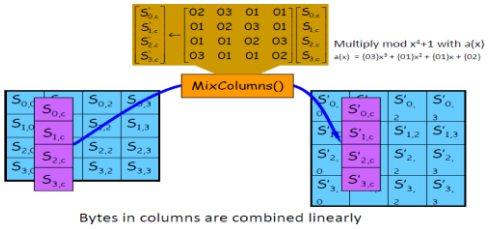
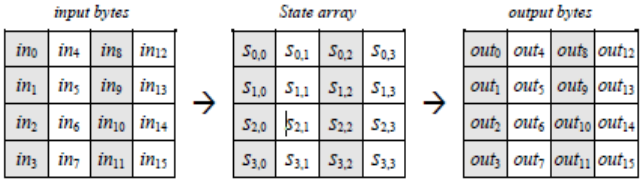


Fig 5: Mix Column Operation

- AddRoundKey Transformation



In AddRoundKey transformation, a Round Key is added to the state by a simple bitwise XOR operation. Each round Key consists of Nb words from the Key expansion. These Nb words are added into the columns of the state.

2.2 Decryption Process

For decryption, the same process occurs simply in reverse order- taking the 128-bit block of cipher text and converting it to plaintext by the application of the inverse of the four operations. AddRoundKey is the same for both encryption and decryption this process is inverse of encryption process. The last round values of both data and key are first round inputs for the decryption process and follows in decreasing order.

3. LITERATURE REVIEW

Literature review has been carried out related to the work to find out the current research. Abstracts of some of most relevant research works are reported as follows –

- Bin Liu, Student Member, IEEE, and Bevan M. Baas, Senior Member, IEEE " Parallel AES Encryption Engines for Many-Core Processor Arrays" IEEE TRANSACTIONS ON COMPUTERS, VOL. 62, NO. 3, MARCH 2013.

In this paper, exploring different granularities of data-level and task-level parallelism, map 16 implementations of an Advanced Encryption Standard (AES) cipher with both online and offline key expansion on a fine-grained many-core system. The design shows 2.0 times higher throughput than the TI DSP C6201, and 3.3 times higher throughput per unit of chip area and 2.9 times higher energy efficiency than the GeForce 8800 GTX.

- Kamalika Datta ,Vishal Shrivastav ,Indranil Sengupta, Hafizur Rahaman "Reversible Logic Implementation of AES Algorithm"8th international conference on design & technology Integrated systems in nanoscale era,978-1-4673-6040-1/13/$31.00,2013 IEEE.

In this paper,reversible logic implementation of a cipher,namely128-bits AES.various AES functional blocks have been synthesized using reversible gates,using which overall reversible architecture has been proposed.hardware complexity has been evaluated using the number of reversible gates and quantum cost.

Vol 06, Article 01531; February 2015
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

- N.Anitha Christy and P.Karthigaikumar "FPGA implementation of AES algorithm using Composite Field Arithmetic"International conference on devices circuits and systems,978-1-4577-1546-4/12/$26.00,2012 IEEE.

In this paper,low area advanced encryption standard -128 bit algorithm is proposed.This technique is implemented using composite field arithmetic in byte substitution block,inverse byte and key expansion block of AES algorithm to increase the throughput .

- Hong Trang;Nguyen Van Loi;"An efficient FPGA implementation of the advanced Encryption Standard algorithm"978-1-4673-0309-5/12,2012,IEEE

In this paper,FPGA implementation of the advanced encryption standard algorithm is presented.This implementation is compared with other work to show efficiency,low latency,high throughput.

- Nabihah Ahmad, N.; Hasan, R.; Jubadi, W.M; "Design of AES S-Box using combinational logic optimization", IEEE Symposium on Industrial Electronics & Applications (ISIEA), pp. 696-699, 31.00,2010 IEEE.

In this paper, combinational logic design of S-box implemented in FPGA. Architecture employs a Boolean simplification of truth table of logic function for reducing delay.

- Marcelo Barcelos, Ricardo Reis "An IP of an Advanced Encryption Standard For Altera Devices",IEEE Symposium on Integrated circuits and systems Design (SBCCI'02)0-7695-1807-9/02,2002, IEEE. Federal Information Processing Standards Publication 197 November 26, 2001 Announcing the ADVANCED ENCRYPTION STANDARD (AES).

In this paper,128 key size is used is used.IP uses a VHDL description optimized to Altera devices .Use for better performance with less use of area.

- Girish Kumar P, Mahesh Kumar "Implementation of AES algorithm using verilog" International Journal of VLSI and Embedded Systems,vol-4,Article 05090;june 2013.

In this paper, encryption and decryption carried out with key length of 256 bits.

CONCLUSION

From review of various papers concluded that Rijndael algorithm is best suited for security of data/images, using different key length to find better result .Protection of data is much more important in application like Military communication, Forensics, Robotics, Intelligent systems. If key length increase security increase but speed of operation decrease. We will try compare various research in field of speed/area/power.

REFERENCES

[1] Bin Liu, Student Member, IEEE, and Bevan M. Baas, Senior Member, IEEE " Parallel AES Encryption Engines for Many-Core Processor Arrays" IEEE TRANSACTIONS ON COMPUTERS,VOL. 62, NO. 3, MARCH 2013

[2] Kamalika Datta ,Vishal Shrivastav ,Indranil Sengupta, Hafizur Rahaman "Reversible Logic Implementation of AES Algorithm"8th international conference on design & technology Integrated systems in nanoscale era,978-1-4673-6040-1/13/$31.00,2013 IEEE.

[3] N.Anitha Christy and P.Karthigaikumar "FPGA implementation of AES algorithm using Composite Field Arithmetic"International conference on devices circuits and systems,978-1-4577-1546-4/12/$26.00,2012 IEEE

[4] Hong Trang;Nguyen Van Loi;"An efficient FPGA implementation of the advanced Encryption Standard algorithm"978-1-4673-0309-5/12,2012,IEEE.

[5] Nabihah Ahmad, N.; Hasan, R.; Jubadi, W.M; "Design of AES S-Box using combinational logic optimization", IEEE Symposium on Industrial Electronics & Applications (ISIEA), pp. 696-699, 31.00,2010 IEEE.

[6] Marcelo Barcelos,Ricardo Reis "An IP of an Advanced Encryption Standard For Altera Devices",IEEE Symposium on Integrated circuits and systems Design (SBCCI'02)0-7695-1807-9/02,2002, IEEE.

[7] Girish Kumar P,Mahesh Kumar "Implementation of AES algorithm using verilog"International Journal of VLSI and Embedded Systems,vol-4,Article 05090;june 2013.

[8] Hrushikesh S.Deshpand, Kailash J.Karande, Altaaf O.Mulani "Efficient Implementation Of AES Algorithm on FPGA" Progress In Science in Engineering Reaserch Journal ,PISER 11,vol.02,ISSN 2347-6680 (E)@2014.

[9] J. Nechvatal et.al., Report on the development of Advanced Encryption Standard, NIST Publication, Oct,2000.

[10] J. Daemen and V Rijmen, "AES Proposal:Rijndael", AES Algorithm Submission, September 3, 1999.

[11] National Institute of Standards and Technology (NIST), Data Encryption Standard (DES), National Technical Information Service, Spring field, VA 22161, Oct. 1999.

[12] N Radhika, Obili Ramesh, Priyadarshini, "Design and Verification of Area-Optimized AES Based on FPGA using verilog HDL"International Journal of Engineering Trends and Technology-volume 4 Issue9-Sep 2013.