International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 4 2389 - 2392

Design and Implementation of Rijindael's Encryption and Decryption Algorithm using NIOS-II Processor

Monika U. Jaiswal
Student
Electronics Department
JDCOEM
Nagpur, India
E-mail: monika.jaiswal89@gmail.com

Prof. Nilesh A. Mohota
Professor
Electronics Department
JDCOEM
Nagpur, India
E-mail:nileshmohota@gmail.com

Abstract - One of the foremost vital problems in communication customary is that the secure transport protocols. This paper can offer a doable resolution for Rijindael's encryption and decoding algorithmic program using NIOS II processor, provided by ALTERA to be enforced in FPGA. We are going to see the performance of Rijindael's AES using NIOS II/e (economic), NIOS II/s (standard) and NIOS II/f (fast). The FPGA has the potential of data processing and hardware modification. The NIOS II is a versatile embedded processor family that represents high performance, lower overall cost, power consumption, complexity combining several functions into one chip. The look of the Rijindael algorithmic program supported "NIOS II + FPGA" are able to do a better processing speed whereas it occupies comparatively low resources. The AES algorithmic program is written in VHDL and is interfaced with the system using general purpose input and output (GPIO) and also the management part is enforced in software in NIOS II integrated development environment (IDE). The implementation is completed on Cyclone II FPGA kit.

Keywords- *Rijindael's algorithm, AES, DES, FPGA, SoPC, NIOS II*

*****

I. INTRODUCTION

The successor of CPLD (Complex Programmable Logic Device), the FPGA (Field Programmable Gate Array) is a digital hardware programmable element used for digital signal processing and growing into the power electronics area. FPGA offers one or additional silicon chip implementations just like the NIOS II processor created by ALTERA just for FPGA. NIOS II is a 32 bit fixed point high performance processor and has separate buses to information and program memory that is usually referred to as Harvard design, additionally has Reduced Instruction Set Computer (RISC) design. It has thirty-two general purpose registers, and 6 control registers used to manage the processor status. For an extended time, the data encryption standard (DES) was thought-about as a standard for the symmetric key encoding. DES includes a key length of 56 bits. However, this key length is currently thought of little and may simply be broken. For this reason, the National Institute of Standards and Technology (NIST) opened a proper necessitate algorithms in Sep 1997. A group of fifteen AES candidate algorithms were declared in August 1998. In August 2000, NIST designated 5 algorithms: Mars, RC6, Rijndael, Serpent and Twofish as the final competitors. These algorithms were subject to additional analysis before the choice of the simplest algorithm for the AES. Finally, on October 2, 2000, office proclaimed that the Rijndael algorithmic rule was the winner. The Rijndael algorithmic rule is that the new Advance Encryption Standard (AES) suggested by the US National Institute of Standards and Technology (NIST) [1]. Rijndael's algorithmic rule is chosen as AES as a result of factors like efficiency, security, performance and adaptability each in hardware and software system platforms and additionally includes a ease of writing the code for algorithmic rule in numerous programming languages. Additional significantly it needs less memory and therefore best fitted to restricted space environments. It's a block cipher algorithmic rule that encrypts blocks of 128, 192 or 256 bits. Therefore, the matter of breaking the key becomes harder.

The Advanced encryption standard (AES) specifies cryptographical rule which will be used to defend electronic data. With the speedy development and wide application of computer and communication networks, the information security has aroused high attention. info security is applied to the political, military, diplomatic fields and common fields of people's day to day life. The presently known attacks can be avoided by AES.

II. AES ENCRYPTION

The AES algorithm is a symmetric block cipher that can encrypt and decrypt the information. Encryption converts data to an unreadable form which is called as cipher-text. Decryption of the cipher-text converts the data back into its original form, which is called as the plain-text [2].

.

A variable block length of 128,192 and 256 bits is supported by Rijandael's AES. The following are four different round transformations: ByteSub, ShiftRow, MixColumn and AddRoundkey. The first and last rounds differ from other rounds as there is an additional AddRoundKey transformation

at the beginning of the first round and no Mix Coulmns transformation is present in the last round.

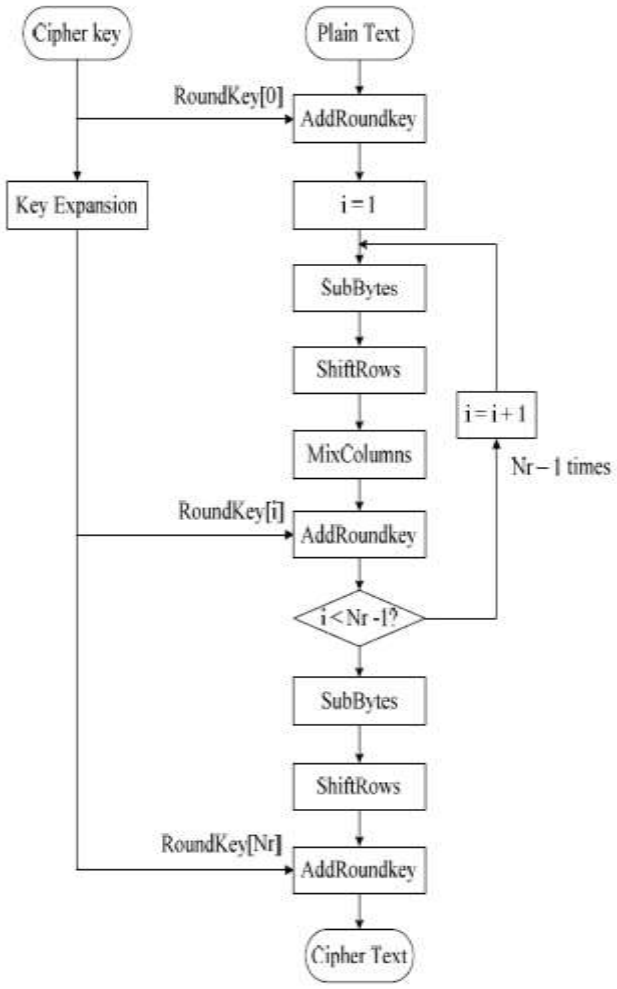

Figure 1. AES encryption structure

A. *SUB BYTES TRANSFORMATION:*

The Sub Bytes transformation is a non-linear byte substitution method and operates on each of the state bytes independently. The Sub Bytes transformation is done using a pre calculated substitution table called as S-box. That S-box table is having 256 numbers (from 0 to 255) and their corresponding resulting values. In this design, we are using a look-up table as shown in Table I. Using S- box, each byte xy (in hexadecimal) in the matrix is substituted with another byte by looking for the entry in the x-row and the y-column of the table. This is a more efficient method than directly implements the multiplicative inverse operation followed by affine transformation of the polynomial. This approach is used to avoid complexity of hardware implementation. This process has the advantage of performing the S-box computation in a single clock cycle and thus latency is reduced.

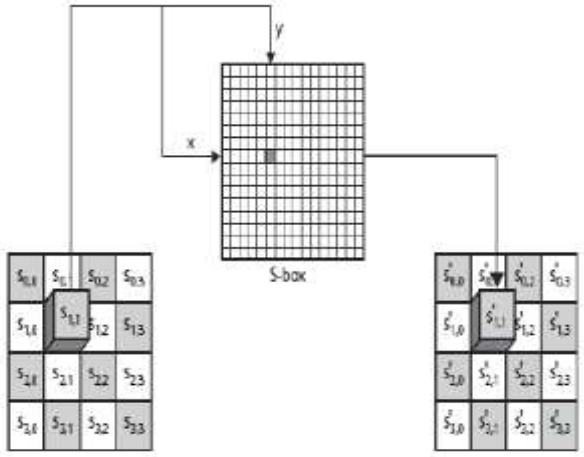

Figure 2. Sub bytes transformation process

TABLE I. S-BOX TABLE

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	1	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	4	c7	23	c3	18	96	5	9a	7	12	80	e2	eb	27	b2	75
4	9	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	0	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	2	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	6	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	8
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	3	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

B. *SHIFTROWS TRANSFORMATION:*

In Shift Rows transformation method, the rows of the state matrix are cyclically left shifted over different offsets. Row 0 is not shifted by any value; row 1 is shifted by one byte to the left; row 2 is shifted by two bytes to the left and row 3 is shifted by three bytes to the left as shown in following figure.

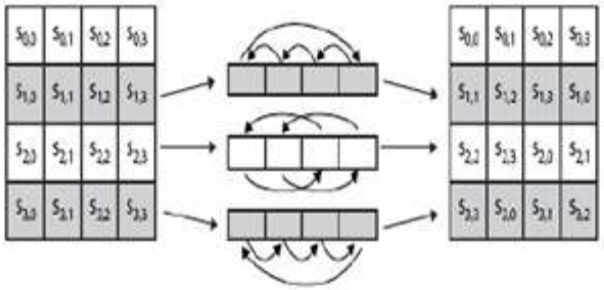

Figure 3. Shift row transformation process

ISSN: 2321-8169
XXX – YYY

C. MIXCOLUMNS TRANSFORMATION:

In MixColumns transformation process, the columns of the state are considered as polynomials over GF (28) and multiplied by modulo x4 + 1 with a fixed polynomial given by c(x),

$c(x) = \{03\}x3 + \{01\}x2 + \{01\}x + \{02\}.$

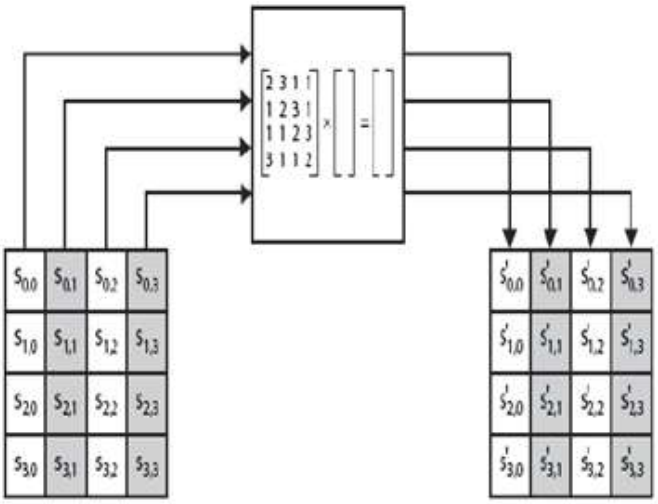

Figure 4. Mix columns transformation

D. ADDROUND KEY OPERATION:

The XOR operation is performed between the state and the round key that it is generated from the main key by the Key Generation method. The matrix of keys is represented by w columns. Add Round Key is used both in the encryption and decryption algorithms. The XOR operation is conducted on byte basis, where the new output byte S'x,y is given by Sx,y ⊕ Kx,y.

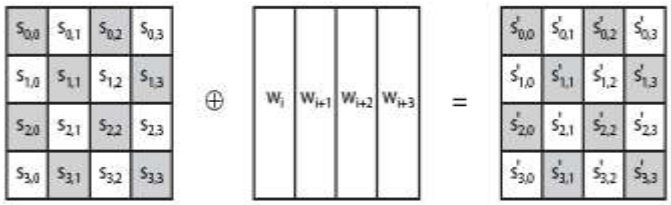

Figure 5.Add round key transformation process

III. AES DECRYPTION

A. Decryption method is a reverse of encryption method that is inverse round transformations for determining the initial plaintext of an encrypted cipher-text in reverse order. The round transformation of decryption uses the subsequent four transformations:

Add round Key, Inv mix Columns, Inv Shift Rows, and Inv Sub Bytes.

B. ADD ROUND KEY OPERATION:

Add Round Key is its own inverse function as the XOR function is having its own inverse value. The round keys are selected in reverse order [5]. The description of the other transformations is given below.

C. INV MIXCOLUMN TRANSFORMATION:

In Inv MixColumn transformation process, the columns of the state are considered as polynomials over GF (28) and multiplied by modulo x4 + 1 with a fixed polynomial given by $c(x)^{-1}$,

$c(x)^{-1} = \{0b\}x3 + \{0d\}x2 + \{09\}x + \{0e\}.$

D. INV SHIFT ROWS TRANSFORMATION:

Inv Shift Rows exactly operates as Shift Rows, only in the opposite direction. The first row is not shifted by any value, while the second, third and fourth rows are shifted right by one, two and three bytes respectively.

E. INV SUB BYTES TRANSFORMATION:

The Inv Sub Bytes transformation is done using a once-pre calculated substitution table called as Inv S-box able. Inv S-box table is having 256 numbers (from 0 to 255) and their corresponding values. Inv S-box is presented in following Table II.

TABLE II INVS-BOX TABLE

I \ Y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	52	9	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	8	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	5	b8	b3	45	6
7	d0	2c	1e	8f	ca	3f	0f	2	c1	af	bd	3	1	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	1a
b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
c	1f	dd	a8	33	88	7	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	4	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

IV. FPGA AND DEVELOPMENT TOOLS

In this project the FPGA kit ALTERA DE2 are going to be used. This board has an EP2C35F672C6 FPGA, from Cyclone II family. It presents several peripherals, like memories, growth pins, CODEC (Encoder/Decoder) and oscillators. The most project package used is that the ALTERA Quartus II web Edition. At this program, all the hardware is represented victimisation block diagrams and hardware description languages (HDL), like VHDL and Verilog. To style systems that have several blocks interconnected by a bus, Quartus II has as complement a package known as SOPC Builder. This package features a library of parameterized elements, as external memory controllers, timers, parallel ports et al., which may be enclosed in a system interconnected by an Avalon Bus, an intelligent bus generated at the system compilation in SOPC Builder. Additionally, SOPC Builder has in its library

the NIOS II Processor, a 32-bit processor to be enclosed within the style. Once the system is already completed, SOPC Builder generates the optimized bus, and exports the complete system to the Quartus II. Once the hardware project is complete, the program compiles all diagrams and outlines archives and interprets into connections within the FPGA, then the user will simulate the practicality of his/her project by using an input vector and analysing the output results [3].
The AES, written in hardware is interfaced with the NIOS II processor system through 'GPIOs'. The inputs and control of an AES is written in 'C-language,' in eclipse IDE. The results are analyzed on the personnel computer (PC) in IDE console window. The performance of the proposed system design is compared with hardware and software implementation of the same application [4].

V. CHARACTERISTICS OF THE NIOS II PROCESSOR

Most of the systems built using SOPC Builder have a processor. The NIOS II Processor is a 32-bit fixed point high-performance processor created to be used only in FPGA. This processor has separated buses to data and program memories, which is often called Harvard architecture, and also has Reduced Instruction Set Computer (RISC) architecture.
To increase the performance, we use NIOSII/f (fast), which results in a larger processor that uses more logic elements, but is faster and has more features, such as multiplication and others; the NIOSII/s (standard), which creates a processor with balanced relationship between speed and area, and some special features; the NIOSII/e (economic), which generates a very economic processor in terms of area, but very simple in terms of data processing capability.

VI. DESIGN OF SYSTEM USING NIOS II PROCESSOR

NIOS II is a soft-core processor which is reconfigurable unlike microcontroller. Soft core means the processor core is not fixed in silicon and can be targeted to any ALTERA FPGA family. Reconfigurable means that one can add or remove features to meet performance or price goals. The system design consists of a NIOS II processor core, a set of on chip peripherals like JTAG UART to communicate with host PC through the USB cable, SRAM that is used by the CPU as a working memory, PIO parallel input output ports, LED and SWITCHES as one of the input output checking device [4] [8].

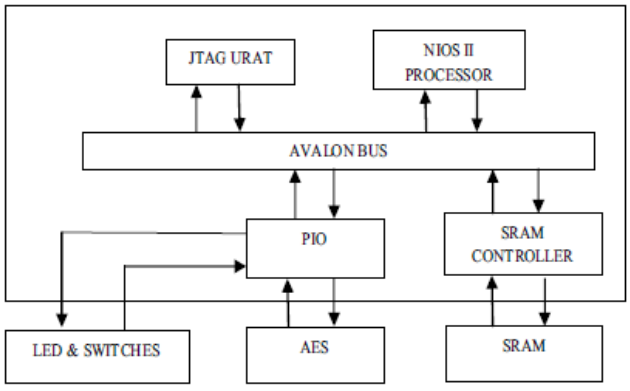

Figure 6. System Design Using NIOS II Processor

VII. CONCLUSION:

Thus with the help of NIOS II and FPGA Rijindael's encryption and decryption algorithm will be implemented. The performance of the system will be calculated by using performance counter for the NIOS II/e, NIOS II/s and NIOS II/f. We can also increase the performance of the system by introducing the custom hardware. The combined design using hardware and software is known as Co-design.

REFERENCES

[1] Dr. Tariq Jamil, "The Rijindael Algorithm" Department of Electrical and Computer Engineering, Sultan Qaboos University (Oman). 0278-6648/04 2004 IEEE
[2] Shunwen Xiao, Yajun Chen, PengLuo, "The Optimized Design of Rijindael Algorithm Based on SOPC", College of Physics and Electronic information China West Normal University, Nanchong, China, 978-0-7695-3922-5/09 2009 IEEE
[3] Andre Luis PescoAlcalde, MarcioSilveiraOrtmann, Samir Ahmad Mussa, "NIOS II Processor Implemented in FPGA: An Application on Control of a PFC Converter" Federal University of Santa Catarina (UFSC), Department of Electrical Engineering (EEL), Power Electronics Institute (INEP), 978-1-4244-1668-4/08 2008, IEEE
[4] Meghana A. Hasamnis, Shri Ramdeobaba college of Engg and Management, S. S. Limaye, Jhulelal Institite og Technology,
"Custome Hardware Interface using NIOS II Processor through GPIO", department of Electronics Engg., Nagpur, India, 978-1-4577-2119-9/12/ 2011 IEEE
[5] Madhav M. Deshpande, Meghana A. Hasamnis, "Design of Encryption System using NIOS II Processor", Electronics Department, R.C.O.E.M, Nagpur University, International Journal of Computer Applications, Volume 68- No. 21, April 2013
[6] N. Sklavos and O. Koufopavlou, Member IEEE, "Architectures and VLSI Implementations of the AES-Proposal Rijindael", Electrical and Computer Engineering Department, University of Patras, Greece, 0018-9340/02 2002 IEEE
[7] R. Sever, N. Ismailoglu, M. Askar, Y.C. Tekmen, "A High Speed ASIC Implementation of the Rijindael's Algorithm," 2004 IEEE International Symposium on Circuits and Systems, May 2004, Vancouver, Canada
[8] Refik Sever, A. NeslinIsmailoglu, Yusuf C. Tekmen, Murat Askar, BurakOkcan, "A High Speed FPGA Implementation of the Rijindael Algorithm", Ankara, Turkey, Proceedings of the EUROMICRO Systems on Digital System Design (DSD'04) 0-7695-2203-3/04 IEEE