International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Issue 1, Volume 2 (January 2015) www.ijirae.com

All Algorithms Were Subject To Assessment Process Performed By Various Groups of Cryptographic Researchers All Over the World

Sharif Yahou[1*] Mohammad Reza Behforooz[2]
[1]Department of physics, Mahabad Branch, Islamic Azad University, Mahabad, Iran
[2] Department of physics, Mahabad Branch, Islamic Azad University, Mahabad, Iran

Abstract— In this paper Advanced Encryption Standard (AES) algorithm is implemented, that can process with the data blockof 128 bit and cipher key length of 128 bit. The usage of 128 bit cipher key to achieve the high security, because cipher key is difficult to broken. As result of this we obtain a secure transmission of data in both encryption and decryption. While computing the existing AES, it takes more area, so we are going to implement the new algorithm for mix Column in AES flow. When we use a new mix column instead of existing one, we can obtain a less area compare to existing AES algorithm.

Keywords— bit, block, algorithm, data

I.INTRODUCTION

The need of privacy has become a high priority for both government and civilians desiring protection from signal and data interception. Widespread use of personal communication devices has only increases demand for a level security on previously insecure communication by using DES algorithm. For a long time, the Data Encryption Standard (DES) was considered as a standard for the symmetric key encryption. Data block and key length of DES algorithm has a only of 56 bits. The 56 bits of data block and key is to be small and can easily broken. For this reason in September 1997, the National Institute of Standards and Technology (NIST) promoted worldwide research into a replacement for DES, or the widely accepted Data Encryption Standard. AES algorithm candidates of 15 members were announced in August 1998. Next all algorithms were subject to assessment process performed by various group of cryptographic researchers all over the world. NIST selected the 5 algorithm in August 2000 they are, RC6, Mars, Rijndeal, Serpent, and Twofish as the final competitors. On October 2, 2000, NIST announced that the Rijndeal algorithm was the winner. As per the Reijndeal data block and key size of multiple of 32 bits, with a minimum of 128 bits and maximum of 256 bits. As a result of breaking of data block and key can be difficult [1]. Advanced Encryption Standard (AES) algorithm also known as Rijndeal [2]. AES has fixed data block size of 128 bits and key size of 128, 192 or 256 bits. AES minimize cost, focusing on efficiency reduced overall hardware complexity. By incorporating most of the algorithm complexity into the controller, components are reused and efficiency increased. A Verilog hardware implementation is also presented, utilizing a field programmable gate array (FPGA) as a prototyping platform. Thus, the design can be easily migrated to an ASIC implementation in an SoC [3]. This paper is organised as follows. Section deals with the introductionabouttheCryptographyalgorithm generation. Section II deals with the AES algorithm .Section III describes abouttheImplementationofAESalgorithm. Section IV shows the simulation results and the last Section V concludes the paper and followed by References.

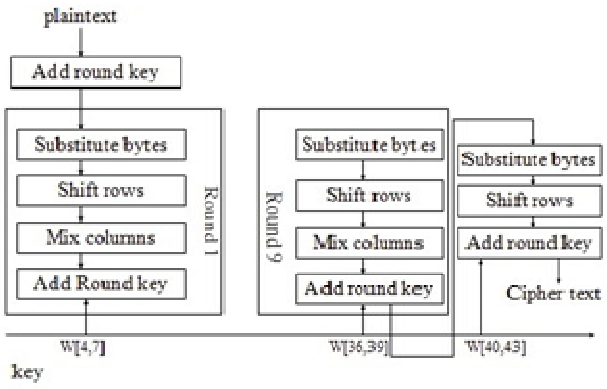

Fig 1. AES encryption structure

		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

TABLE I S-BOX TABLE

II.AES ALGORITHM

AES algorithm is a symmetric block cipher that can encrypt and decrypt the information. Encryption can convert the original data in to unintelligible data is called cipher text. Decryption can convert the cipher text form in to original data, which is called plain text.

A. AES Encryption

The AES algorithm (AES-128) operates on a 128 bit block of data size and 128 bit key size and executed N-1 loop times. A loop is called a round and the number of iterations of a loop N can be 10. The first and last rounds differ from other rounds in that there is an a additional AddRoundkey transformation at the beginning of the first round and no Mixcolumns transformation is performed in the last round. AES encryption is given in Fig.1.

1) SubBytes Transformation: The SubBytes transformation is a nonlinear byte substitution, operates on each of the state bytes independently. The SubBytes transformation is done by using S-box. S-box is a precalculated substitution table contains 256 numbers (from 0 t0 255) and their corresponding resulting value. More details of the calculating the S-box table refers to [4]. S-box table as shown in Table I.

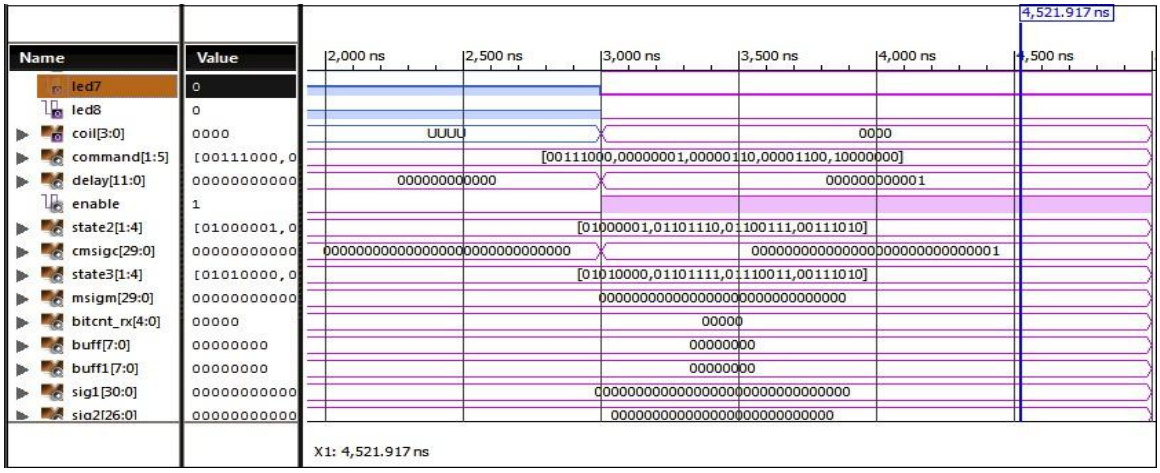

Fig 3. AES encryption structure

2) ShiftRows Transformation: In ShiftRows transformation, the rows of the state are left shifted in cyclic manner. 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.

3) MixColumns Transformation: In MixColumns transformation, the columns of the state are considered as polynomials over GF (28) and multiplied by modulo x4 + 1 with a fixed polynomial c(x), given by: c(x)={03}x3 + {01}x2 + {01}x + {02}.

4) AddRoundKey Transformation: In the AddRoundKey transformation, a Round Key is added to the State - resulted from the operation of the MixColumns transformation - by a simple bitwise XOR operation. The RoundKey of each round is derived from the main key using the KeyExpansion algorithm [1]. The encryption/decryption algorithm needs eleven 128-bit RoundKey, which are denoted RoundKey [0].

B. AES decryption

Decryption is a reverse of encryption which inverse round transformations to computes out the original plaintext of an encrypted cipher-text in reverse order. The round transformation of decryption uses the functions AddRoundKey, InvMixColumns, InvShiftRows, and InvSubBytes successively.

*1) AddRoundKey***:** AddRoundKey is its own inverse function because the XOR function is its own inverse. The round keys have to be selected in reverse order. The description of the other transformations will be given as follows.

2) InvShiftRows Transformation: InvShiftRows exactly functions the same as ShiftRows, only in the opposite direction. The first row is not shifted, while the second, third and fourth rows are shifted right by one, two and three bytes respectively.

*3) InvSubBytes transformation***:** The InvSubBytes transformation is done using a once-precalculated substitution table called InvS-box. That InvS-box table contains 256 numbers (from 0 to 255) and their corresponding values.

4) InvMixColumns Transformation: In the InvMixColumns transformation, the polynomials of degree less than 4 over GF(28), which coefficients are the elements in the columns of the state, are multiplied modulo ($x4 + 1$) by a fixed polynomial $d(x) = \{0B\}x3 + \{0D\}x2 + \{09\}x + \{0E\}$, where {0*B*}, {0*D*}; {09}, {0*E*} denote hexadecimal values.

In the next section, a description of the proposed design based on FPGA implementation of AES encryption/decryption function is detailed.

III.FPGA IMPLEMENTATION OF AES

Fig.2 shows the detailed design of AES core based on FPGA implementation, where the control signals are described in Table II.

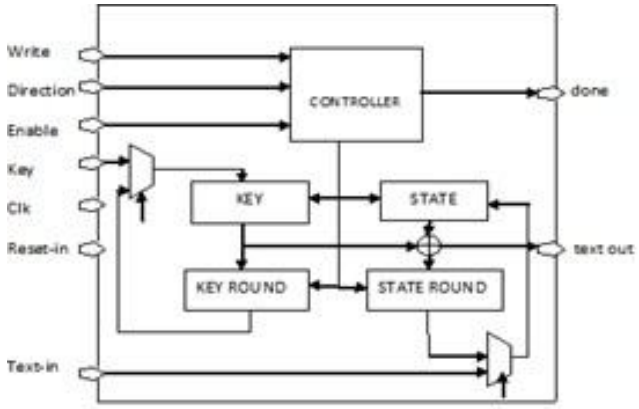

Fig 2. Architecture of AES core

The total design has 390 pins. It requires the text in, text out and key which have a 128 bits length. And the controls signals using to control the proper operations of the core are clock (clk), reset in, write, direction, done and enable pins. The Key block loads keys and combines with Key Round block to perform Key Expansion transformation, and generates proper Roundkeys under the control signals from the Controller block. Controller block takes write signal, direction signal, and enable signal from outside and generates all the control signals for the whole system. The plain text (text_in) and key is loaded only when the write signal makes a low-high-low transition (basically a pulse). The process is going to complete when the done signal is pulsed after some clock cycles from the write signal goes low. The "done" signal actives only in one clock cycle. Each round key as well as round is completed in one clock cycle.

However, the round key is finished before the round is calculated by one clock cycle. Hence, combining with one clock cycle for registering the input, a total clock cycle need for processing 128-bit data is 13 clocks in encryption mode. In decryption, eleven round keys must be completed before the first round is calculated. Because the last round key is used in the first round process, it takes 25 clock cycles to complete. With using the above iterative looping approach, a minimal number of clock cycles required performing Encryption/decryption for each data block of 128-bit.

IV. SIMULATION RESULTS

The design has been coded by Verilog HDL. All the results are synthesized and simulated basing on the Quatus 9.0, the Model Sim - Altera 6.4a and EP20K400CB652C7 device. The results of simulating the encryption/decryption Algorithms from the ModelSim simulator.

V. CONCLUSION

This paper mainly focused in implementation of AES encryption and decryption standard AES-128. All the transformations of both Encryption and Decryption are simulated using an iterative design approach in order to minimize the hardware consumption.. This method can make it a very low-complex architecture, especially in saving the hardware resource in implementing the AES InverseSub Bytes module and Inverse Mix columns module. As the S - box is implemented by look-up-table in this design, the chip area and power can still be optimized. The new Mix Column transformation improves the performance of the inverse cipher and also reduces the complexity of the system that supports the inverse cipher. As a result this transformation has relatively low relevant diffusion power .This allows for scaling of the architecture towards vulnerable portable and cost-sensitive communications devices in consumer and military applications.

REFERENCES

[1] Riccardo Marino, "Patrizio Tomei. Nonlinear Control Design Geometric, Adaptive and Robust". Pearson Education Limited, 2000.

[2] Isidori Albort. "Nonlinear Control System". Berlin Springer Verlag, 1989.

[3] Isidori Albort, Keener J, Gori-Giorgi C, et al. "Nonlinear decoupling via feedback: a differential geometric approach". IEEE Trans Automatic Control, ,26(3) 331-341981.

[4] Lu Qiang, Mei Sheng-Wei, Sun Yuan-Zhang. "Nonlinear system control for electric power system (2nd edition)". Beijin: Tinghua university publishers (in Chinese), 2008.

