ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 1, Issue 4, October 2012

FPGA Implementationof Secret Communication with VHDL

D.Anvesh Kumar[1], Y.V.AdiSatyanarayana[2],P.Prasannamuralikrishna[3]

[1]M.Tecch, ECE,Dr.SGIET, Markapur, A.P, India
[2]Associateprofessor, Department of ECE, Dr.SGIET, Markapur, A.P, India
[3]Assistant professor, Department of ECE, Dr.SGIET, Markapur, A.P, India

***Abstract*—This paperpresents a communication of data secretly using AES Algorithm i.e we first send the data (plain text) and the key which is of 64 bit into the encryption process .The output of this process will be cipher text. This cipher text is then fed into the decryption process and then the data (plain text) is got as output. In this project we add the key and shuffle the data it is very hard for the unknown person to find out the original data. Since for each key there will be a change in the cipher text and so the person has to know the key in order to find out the original data.The AES algorithm involves the process of giving the data and key as input to the encryption block and then implementing several blocks such as key schedule block, control block, round function block, etc.In this project, we aim at designing a high speed and high performance but cost-effective FPGA based processor which encrypts and /or decrypts the data that has to be transferred from one PC to the other, based on the Advanced Encryption Standard (AES) algorithm.**

***Keywords* -Cryptography, AES, FPGA, security, communications.**

I. INTRODUCTION

A. Cryptography

The branch of cryptology dealing with the design of algorithms for encryption and decryption, intended to ensure the secrecy and/or authenticity of messages. An original message is known as the plain text, while the coded message is called the cipher text. The process of converting the plain text to cipher text is known as enciphering or encryption; restoring the plain text from the cipher text is deciphering or decryption. The many schemes used for enciphering constitute the area of study known as cryptography. Such a scheme is known as a cryptographic system or a cipher.Techniques used for deciphering a message without any knowledge of the enciphering details fall into the area of cryptanalysis. The cryptanalysis are what the lay persons calls “breaking the code”. The areas of cryptography and cryptanalysis together are called cryptology.

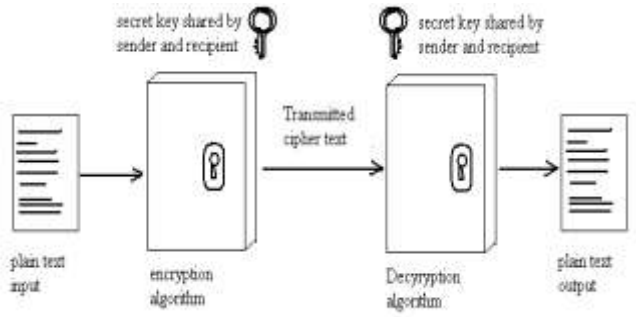


Figure 1 - Simplified model of conventional encryption

A. Encryption

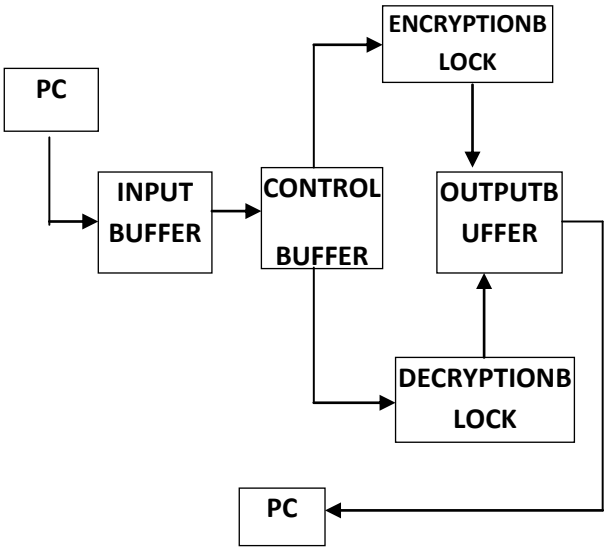
The original intelligible message or data that is fed into the algorithm is the input. Encryption algorithm performs varies substitutions and transformations on the plain text. Secret key is also given as input to encryption algorithm. The key is a value independent of the plain text. The algorithm reproduces the different output depending on the specific key being used at the time. The exact substitutions and transformation performed by the algorithm depend on the key.The output depends on the plain text and the secret key. For a given message, two different keys will produce two different cipher texts. The cipher text is an apparently random stream of data, as it stands, is unintelligible.

B. Decryption

This is essentially the encryption algorithm run in reverse. It takes the cipher text and the secret key and produces the original plain text

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 1, Issue 4, October 2012

II.BLOCK DIAGRAM



III.ALGORITHM REVIEW

A. *Data Encryption Standard (AES)*

The AES block cipher algorithm was developed by researchers at IBM and was fine-tuned by government agencies, the National Security Agency (NSA) and the National Institute of Standards and Technology (NIST). The American National Standards Institute (ANSI) adopted AES as the federal standard for encryption of commercial and sensitive data. This is defined in Federal Information Processing Standards (FIPS 46, 1977) published by NIST [1], [2].The AES algorithm has a regular structure that lends itself to pipelining and simple data manipulations to permit fast operations. AES is a symmetric encryption algorithm where the same key is used for both encryption and decryption. AES takes a 64-bit key and a 64-bit block of data as inputs, and outputs 64-bits of encrypted data. The actual key is only 56-bits and the remaining bits, i.e., the least significant bit (LSB) in every byte can be used as parity. The same basic AESing can be used for both encryption and decryption. The Data Encryption Standard (AES) shall consist of the following Advanced Encryption Algorithm (AES) and Triple Advanced Encryption Algorithm (TDEA, as AEScribed in ANSI X9.52). These devices shall be AESigned in such a way that they may be used in a computer system or network to provide cryptographic protection to binary coded data. The method of implementation will depend on the application and environment. The devices shall be implemented in such a way that they may be tested and validated as accurately performing the transformations specified in the following algorithms.

B. *Advanced Encryption Algorithm*

The algorithm is AESigned to encipher and decipher blocks of data consisting of 64 bits under control of a 64-bit key1. Deciphering must be accomplished by using the same key as for enciphering, but with the schedule of addressing the key bits altered so that the deciphering process is the reverse of the enciphering process. A block to be enciphered is subjected to an initial permutation *IP*, then to a complex key-dependent computation and finally to a permutation which is the inverse of the initial permutation *IP*-1. The key-dependent computation can be simply defined in terms of a function *f*, called the cipher function, and a function *KS*, called the key schedule.A AEScription of the computation is given first, along with details as to how the algorithm is used for encipherment. Next, the use of the algorithm for decipherment is AEScribed. Finally, a definition of the cipherfunction*f*is given in terms of primitive functions that are called the selection functions Si and the permutation function *P*. *Si*, *P*and *KS*of the algorithm are contained in Appendix.The following notation is convenient: Given two blocks *L*and *R*of bits, *LR*denotes the block consisting of the bits of *L*followed by the bits of *R*. Since concatenation is associative, *B1B2...B8*, for example, denotes the block consisting of the bits of *B1*followed by the bits of *B2*...followed by the bits of *B8*.

C. *Enciphering*

A sketch of the enciphering computation is given in Figure(2). The 64 bits of the input block to be enciphered are first subjectedto the following permutation, called the initial permutation *IP*:

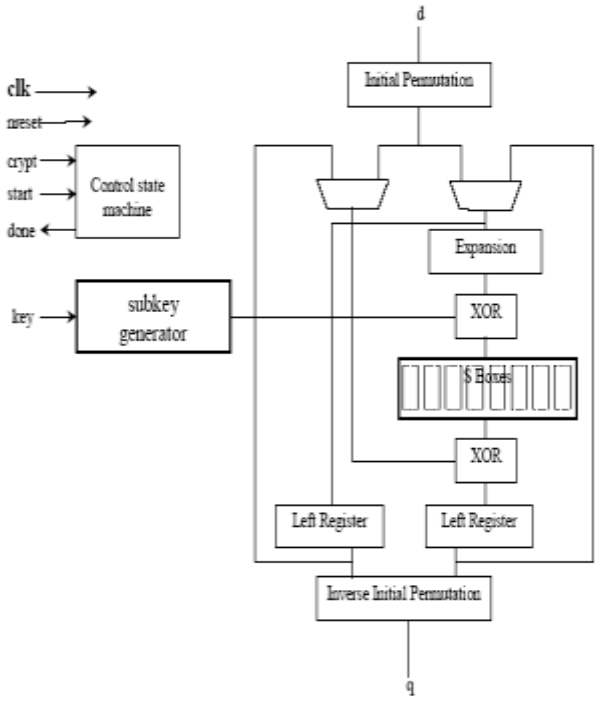


Figure 2 – Enciphering

That is the permuted input has bit 58 of the input as its first bit, bit 50 as its second bit, and so on with bit 7 as its last bit. The permuted input block is then the input to a complex key-dependent computation AEScribed below. then subjected to the following permutation, which is the inverse of the initial permutation:

Table1 – Initial permutation

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Table 2 – Inverse initial permutation

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

That is, the output of the algorithm has bit 40 of the preoutput block as its first bit, bit 8 as its second bit, and so on, until bit 25 of the preoutput block is the last bit ofthe output. The computation which uses the permuted input block as its input to produce the preoutput block consists, but for a final interchange of blocks, of 16 iterations of a calculation that is AEScribed below in terms of the cipher function *f*which operates on two blocks, one of 32 bits and one of 48 bits, and produces a block of 32 bits. Let the 64 bits of the input block to an iteration consist of a 32 bit block *L* followed by a 32 bit block *R*. Using the notation defined in the introduction, the input block is then *LR*. Let *K* be a block of 48 bits chosen from the 64-bit key. Then the output *L'R'*of an iteration with input *LR*is defined by

The output of that computation, called the preoutput, is

L'= R 3.1
R'= L⊕f(R, K)…….. (1)

Where ⊕ denotes bit-by-bit addition modulo 2.As remarked before, the input of the first iteration of the calculation is thepermuted input block. If *L'R'*is the output of the 16th iteration then *R'L'*is the preoutput block. At each iteration a different block *K*of key bits is chosen from the 64-bit key AESignated by *KEY*. With more notations we can AEScribe the iterations of the computation in more detail. Let *KS*be a function, which takes an integer *n* in the range from 1 to 16 and a 64- bit block *KEY* as input and yields as output a 48-bit block *Kn*that is a permuted selection of bits from *KEY*. That is *Kn=KS (n, KEY) 3.2* with *Kn*determined by the bits in 48 distinct bit positions of *KEY*. *KS*is called the key schedule because the block *K*used in the *n*'th iteration of (1) is the block *Kn*determinedby (2).10As before, let the permuted input block be *LR*. Finally, let *L()*and *R()*be respectively*L*and *R*and let *Ln*and *Rn*be respectively *L'* and *R'* of (1) when *L*and *R* are respectively *Ln-1*and *Rn-1*and *K* is *Kn*; that is, when n is in the range from 1 to 16

Ln= Rn-1 3.3
Rn= Ln-1⊕f(Rn-1,Kn)…….(2)

The preoutput block is then *R16L16*. The key schedule *KS* of the algorithm is AEScribed in detail in the Appendix. The key schedule produces the 16 *Kn*, which are required for the algorithm.

D. Deciphering

The permutation *IP-1*applied to the preoutput block is the inverse of the initial permutation *IP* applied to the input. Further, from (1) it follows that

R = L' 3.4
L= R'⊕f(L',K)……(3)

Consequently, to *decipher*it is only necessary to apply the *very same algorithm to an enciphered message block*, taking care that at each iteration of the computation *the same block of key bitsKis used*during decipherment as was used during then cipherment of the block. Using the notation of the previous section, this can be expressed by the equations:

Rn-1= Ln 3.5
Ln-1= Rn⊕f(Ln,Kn)…….. (4)

Where now *R16L16*is the permuted input block for the deciphering calculation and *L0R0*is the preoutput block. That is, for the decipherment calculation with *R16L16*as the permuted input, *K16*is used in the first iteration, *K15*in the second, and so on, with *K1*used in the 16th iteration.

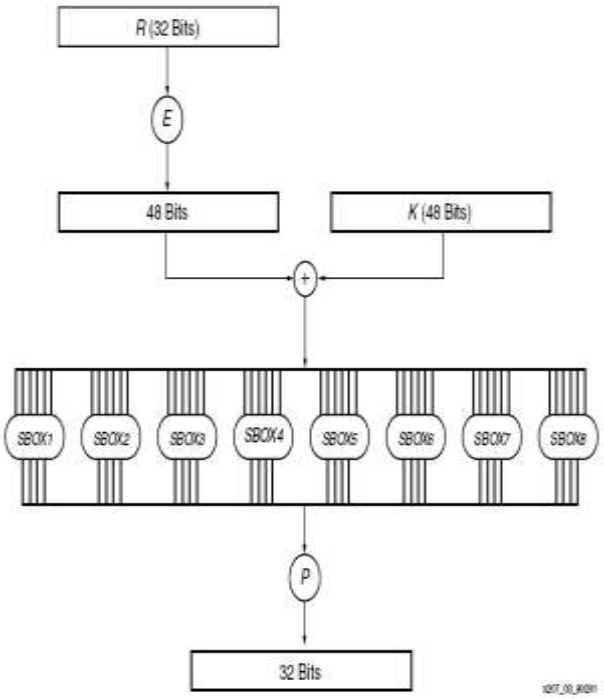


Figure 3 - A sketch of the calculation of *f(R, K)*

Let *E*denote a function that takes a block of 32 bits as input and yields a block of 48 bits as output.

Let*E*be such that the 48 bits of its output, written as 8 blocks of 6 bits each, are obtained by selecting the bits in its inputs in order according to the followingtable 3Thus the first three bits of *E(R)*are the bits in positions 32, 1 and 2 of *R*while the last 2 bits of *E(R)*are the bits in positions 32 and 1.

Each of the unique selection functions *S1***,** *S2*,...,*S8*, takes a 6-bit block as input and yields a 4-bit block as output and is illustrated by using a table 3 containing the recommended *S1*:

Table 3 – E bit selection table

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	33

IV.PRIMITIVE FUNCTIONS FOR THE ADVANCED ENCRYPTION ALGORITHM

The choice of the primitive functions *KS*,*S1***,...,***S8*and *P*is critical to the strength of an encipherment resulting from the algorithm. Specified below is the recommended set of functions, AEScribing*S1***...** *S8*and*P*in the same way they are AEScribed in the algorithm. For the interpretation of the tables AEScribing these functions, see the discussion in the body of the algorithm. The primitive functions *S1*...*S8*are:

S1

14 4 13 1 2 15 11 8 3 10 6 12 5 9 0 7
0 15 74 14 2 13 1 10 6 12 11 9 5 3 8
4 1 14 8 13 6 2 11 15 12 9 7 3 10 50
15 12 8 2 4 9 1 7 5 11 3 14 10 0 6 13

S2

15 1 8 14 6 11 3 4 9 7 2 13 12 0 5 10
3 13 4 7 15 2 8 14 12 0 1 10 6 9 11 5
0 14 7 11 10 4 13 1 5 8 12 6 9 3 2 15
13 8 10 1 3 15 4 2 11 6 7 12 0 5 14 9

S3

10 0 9 14 6 3 15 5 1 13 12 7 11 4 2 8
13 7 0 9 3 4 610 2 8 5 14 12 11 15 1
13 6 4 9 8 15 3 0 11 1 2 12 510 14 7
1 10 13 0 6 9 8 7 4 15 14 3 11 5 2 12

S4

7 13 14 3 0 6 9 10 1 2 8 5 11 12 4 15
13 8 11 5 6 15 0 3 4 7 2 12 1 10 14 9
10 6 9 0 12 11 7 13 15 1 3 14 5 2 8 4
3 15 0 6 10 1 13 8 9 4 5 11 12 7 2 14 18

S5

2 12 4 1 7 10 11 6 8 5 3 15 13 0 14 9
14 11 2 12 4 7 13 1 5 0 15 10 3 9 8 6
4 2 1 11 10 13 7 8 15 9 12 5 6 3 0 14
11 8 12 7 1 14 2 13 6 15 0 9 10 4 5 3

S6

12 1 10 15 9 2 6 8 0 13 3 4 14 7 5 11
10 15 4 2 7 12 9 5 6 1 13 14 0 11 3 8
9 14 15 5 2 8 12 3 7 0 4 10 1 13 11 6
4 3 2 12 9 5 15 10 11 14 1 7 6 0 8 13

S7

4 11 2 14 15 0 8 13 3 12 9 7 5 10 6 1
13 0 11 7 4 9 1 10 14 3 5 12 2 15 8 6
1 4 11 13 12 3 7 14 10 15 6 8 0 5 9 2
6 11 13 8 1 4 10 7 9 5 0 15 14 2 3 12

S8

13 2 8 4 6 15 11 1 10 9 3 14 5 0 12 7
1 15 13 8 10 3 7 4 12 5 6 11 0 14 9 2
7 11 4 1 9 12 14 2 0 6 10 13 15 3 5 8
2 1 14 7 4 10 8 13 15 12 9 0 3 5 6 11

Table 4 - Primitive function

16	**7**	**20**	**21**
29	**12**	**28**	**17**
1	**15**	**23**	**26**
5	**18**	**31**	**10**
2	**8**	**24**	**14**
32	**27**	**3**	**9**
19	**13**	**30**	**6**
22	**11**	**4**	**25**

A. Key schedule calculations

Recall that *Kn*, for 1□*n*□ 16, is the block of 48 bits in (2) of the algorithm. Hence, to AEScribe*KS*, it is sufficient to AEScribe the calculation of *Kn*from *KEY* for *n* = 1, 2,..., 16. That calculation is illustrated in Figure 4**.** To complete the definition of *KS*it is therefore sufficient to AEScribe the two permuted choices, as well as the schedule of left shifts. One bit in each 8-bit byte of the *KEY*may be utilized for error detection in key generation, distribution and storage. Bits 8, 16... 64 are for use in assuring that each byte is of odd parity. The table has been divided into two parts, with the first part determining how the bits of *C( )*are chosen, and the second part determining how the bits of *D( )*are chosen. The bits of *KEY* are numbered 1 through 64. The bits of *C( )*are respectively bits 57, 49, 41,..., 44 and 36 of *KEY*, with the bits of *D( )*being bits 63, 55, 47,..., 12 and 4 of *KEY*.With *C( )*and *D( )* defined, we now define how the blocks *Cn*and *Dn* are obtained from the blocks *Cn- 1*and *Dn-1*, respectively, for *n* = 1, 2,..., 16. That is accomplished by adhering to the following schedule of left shifts of the individual blocks:

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 1, Issue 4, October 2012

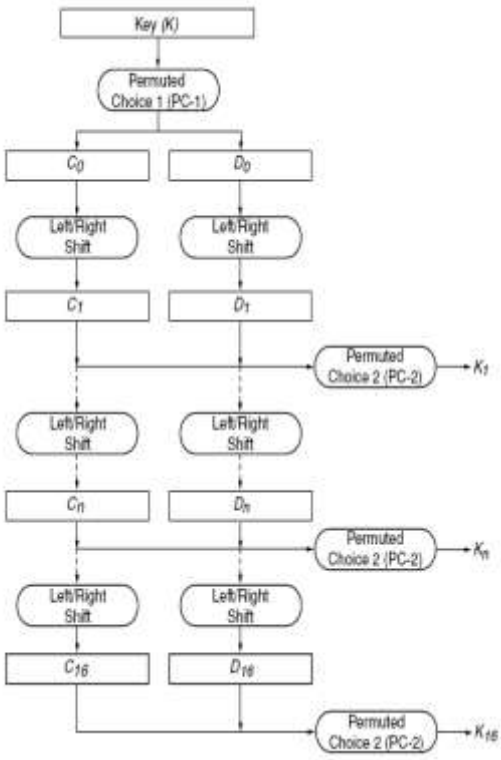


Figure 4 - Key Schedule Calculation

Table5– Permuted choice1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

The number of left shifts according to the iterations can be calculated using a standard formula and found to be the following

Table 6 – Iteration number and left shifts

Iteration number	No of left shifts	Iteration number	No of left shifts
01	1	09	1
02	1	10	2
03	2	11	2
04	2	12	2
05	2	13	2
06	2	14	2
07	2	15	2
08	2	16	1

For example, *C3* and*D3*are obtained from *C2* and *D2*, respectively, by two left shifts, and *C16* and *D16* are obtained from *C15* and *D15*, respectively, by one left shift. In all cases, by a single left shift is meant a rotation of the bits one place to the left, so that after one left shift the bits in the 28 positions are the bits that were previously in positions 2, 3,..., 28, 1.

V. CONCLUSION

Therefore, the first bit of *Kn*is the 14th bit of *CnDn*, the second bit the 17th, and so on with the 47th bit the 29th, and the 48th bit the 32nd.gate counts that we were not able to implement the AES algorithm in synthesis level. So in future if we use a higher version of IC so that we can also implement AES algorithm in synthesis level.

REFERENCES

[1] J. ZAMBRENO , D. NGUYEN AND A. CHOUDHARY , "Exploring area/delay trade-offs in AES FPGA implementation", Proc. Int. Colif, Field Programmable Logic and Its Applications (FPL), Lecture Notes in Computer Science, Vol. 3203 (Springer-Verlag 2004), pp. 575-585.

[2] R. Torrance, D. James, "The State-of-the-Art in IC ReverseEngineering," Cryptographic Hardware and Embedded Systems —CHES 2009, pp. 363–381,Oct 2009.
[3] R. Gennaro, A. Lysyanskaya, T. Malkin, S.Micali and T. Rabin, "Algorithmic Tamper-Proof Security: Tampering," Theory of CryptographyConference (TCC), pp. 258-277, Feb2004.

[4] J.W. Lee, D. Lim, B. Gassend, G. E. Suh, M.vanDijk, And S. Devadas."A technique to build a secret key in integrated circuits with identification and Authentication applications", Proceedings of the IEEE VLSI Circuits Symposium, June 2004.43
[5] J. Guajardo, S. Kumar, G. Schrijen, and P. Tuyls."FPGA IntrinsicPUF and Their Use for IPProtection.Cryptographic Hardware andEmbedded Systems", CHES 2007, pp.63–80, Oct 2007.
[6] P.Tuyls, G.-J. Schrijen, B.Skoric, J. van Geloven, Verhaegh, and R.Wolters.Coatings CryptographicHardware and Embedded Systems CHES2006, pages 369–383, Oct 2006.
[7] R. S. Pappu. "Physical one-way functions."PhD thesis.MassachusettsInstitute of Technology. Mar 2001.

D. Anvesh Kumar Completed B.Tech in Electronics and Communication Engineering From JNTU, Kakinada, A.P. He is pursuing M.Tech in ECE with specialization DECS in Dr.SGIET, JNTU Kakinada, A.P, India.

Prof.Y.V.Adisatyanarayana was born in Yerragondapalem, Prakasam Dt, A.P, India. He Completed B.E in ECE from MADRASUniversity, Chennai. He has completed M.Tech CSE From JNTU,Anatapur.He is doing P.hD in JNTU, Kakinada, A.P. India. Presently he is working as Associate Professor at Dr.SGIET, JNTU, Kakinada. A.P, India.

Prof.Prasannamuralikrishna Completed B.Tech in ECE from Aurangabad University,Maharashtra**.** He has completed M.Tech in ECE from JNTU, A.P. He is doing P.hD in JNTU, Kakinada. Presently he is working as Assistant Professor at Dr.SGIET, JNTU, Kakinada. A.P, India.