International Journal of Engineering & Technology, 7 (1.1) (2018) 612-617

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

FPGA based asymmetric crypto system design

V. Narasimha Nayak[1*], M. Ravi Kumar[2], K. Anusha[3], Ch. Kranthi Kiran[4]

[1]*Assistant Professor, Department of ECE, K L E F, AP, India.*
[2]*Assistant Professor, Department of ECE, K L E F, AP, India.*
[3]*Assistant Professor, Department of ECE, K L E F, AP, India.*
[4]*Assistant Professor, Department of ECE, K L E F, AP, India.*

Abstract

In the network security system cryptography plays a vital role for the secure transmission of information. Cryptography is a process of integrating and transferring the data to the genuine users against any attacks. There are two types of Cryptographic algorithm: Symmetric and Asymmetric algorithms. In the symmetric type cryptography, single key is used for both encryption and decryption. Symmetric algorithms are fast and simple. Asymmetric cryptographic algorithm uses different keys such as public key to encrypt the message at sender and private key which is known only to receiver for decrypting the encrypted message. Asymmetric algorithms are more secure and difficult, to decrypt the message unless hacker acquires the knowledge of private key. A new Asymmetric algorithm with Error Detection and Correction mechanism is proposed that can reduce hardware, and improves decryption time and security. Proposed Asymmetric algorithm uses the few properties of: RSA, Diffie-Hellman and ElGamal Algorithms. Performance of asymmetric algorithms is compared with proposed algorithm, which is designed using Verilog HDL. Algorithms are synthesized, simulated, implemented using Vivado and targeted for Artix-7 XC7A100T-1CSG324Carchitecture.Chipscope Pro logic analyzer-Virtual Input Output core is binded to design for hardware debugging, to monitor and capture the output signals at selected specified state by applying random input stimuli at runtime in Nexys4 DDR FPGA Board.

1. Introduction

Presently many advanced technologies are evolving in the network security system [1]. In the network security systems, if the data is insecure attackers can hack the information [2]. So there must be security for the information by which users can avoid information theft. There are two methods by which the security increases are cryptography and stenography. Cryptography is the art of converting the original message into unreadable code, which allows the information to be secret. Steganography is an art of hiding the actual data using duplicate data. The term cryptography is a Greek word which means "secret writing". It is an art and science of converting the information so as to make that information secure and immune from attacker.

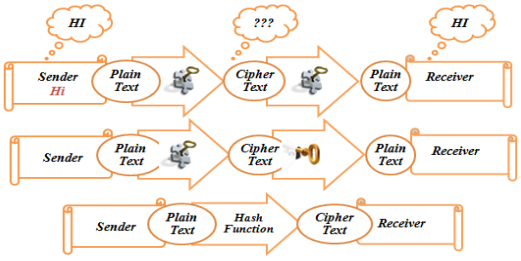


Fig. 1: Types of cryptographic algorithm

Cryptography involves the process of encryption and decryption. Encryption is defined as the converting the plain text or information in to cipher text and decryption involves converting the cipher text in to plain text. Privacy, Authentication, Integrity, Non- repudiation and key exchange are the five primary functions involved in the cryptography. There are three types of cryptographic algorithms as shown in Fig 1, they are: 1. Symmetric key cryptography, 2. Asymmetric key cryptography and 3. Hash functions [18]. Symmetric key cryptography: Symmetric key cryptography is also called as secret-key or conventional cryptography or a single key cryptography. Fig 2 shows the encryption and decryption of Symmetric key cryptography, same key is used for both encryption and decryption. Alice uses public key for converting plain text in to encrypted text i.e. cipher text. Bob receives cipher text uses private key to obtain the plain text. Only authorized people would know it. Symmetric key cryptography is primarily used for confidentiality. DES, Triple DES, AESis the examples of Symmetric key cryptography [3] [4].

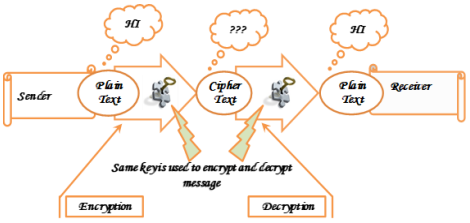


Fig. 2: Symmetric key type cryptosystem

Asymmetric key cryptography: Asymmetric key cryptography is also called as public key cryptography. Different keys are used for encryption and decryption. Fig 3 shows the Asymmetric key cryptography with encryption and decryption process. The transmitter uses public key for converting plain text in to encrypted text i.e. cipher text.

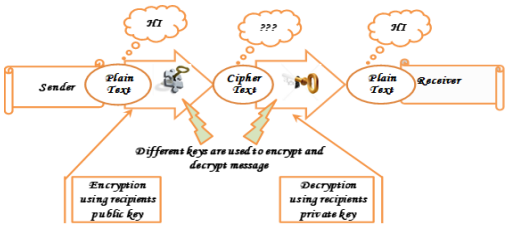


Fig. 3: Asymmetric key cryptography

The receiver uses private key to obtain the plain text from cipher text. Diffie–Hellman, ElGamal, and Elliptic curve cryptography are the examples of Asymmetric key cryptography. Hash function is also called as one way encryption; it doesn't use any key rather, but uses a fixed length hash value. It is estimated in view of plaintext that makes it difficult to recover the length of plain content. Generally, for the digital fingerprints, file's content is protected with hash algorithm; it is used to confirm that the file has not been attacked by any virus.

2. Literature survey

Present day life is injected with advanced innovation i.e. with the digital technology which is developed within few decades. Usage of internet has increased much rapidly in almost all the activities like online shopping, online games etc. There are many other applications like ATM security, trusted computing, military applications. Though using internet has been made very easy, there is another important factor which should be considered that is the security and privacy [5]. By using the cryptography technique these constraints can be removed. There are many algorithms developed by the scientists for the secure communication. Basically there are three types of cryptosystems: Symmetric key cryptography, Asymmetric key cryptography and hash functions.

In 2017, VANGA ODELU proposed a new RSA based CP-ABE scheme [6]. For each decryption and encryption, this scheme provides constant size secret keys and cipher texts (CSKC) with an expressive AND gate access structure without using bilinear maps. It is also shown that it is secure against attacks.

In 2015 Juliet N. Gaithuru presented a paper which shows a comprehensive review of the asymmetric key algorithms from its beginning [7]. Comparison of different algorithms in terms of their encryption, decryption, security levels, implementation area, advantages and disadvantages are shown. It also mentions that research in cryptography is moving towards quantum computing.

To improve the performance of DES, 3DES and AES instruction set extensions for a reduced instruction set computer processor are presented by Sean O'Melia in 2010[8]. He presented a discussion on the existing methods for the performance improvement and then a detailed discussion on targeted processors and targeted cryptographic algorithms. Along with the functional units description of the custom instructions were also given which implements the logical an arithmetic operations. This proposed method showed positive results in terms of size of the code and improved the encryption and decryption throughput for all the targeted algorithms.

In 2015 Yi et al proposed a distributed ElGamal Cryptosystem based on the Pailier cryptosystems [9]. It deals with simpler the distributed key generation and the distributed decryption of the information from a larger domain. Through this method the analysis on the security of their proposed variant of ElGamal encryption and they demonstrated that their construction is not secure as claimed. A feasible attack on the variant of the Elgamal encryption method is seen.

Prashant Sharma proposed a novel algorithm Modified Elgamal Cryptosystem Algorithm (MECA) [10]. In this paper since security using Elgamal encryption/decryption algorithm can be breached as is not secured against adaptive attacks, a new Modified Elgamal Cryptosystem Algorithm has been proposed to improve the security. This algorithm improves the security for encrypting long messages and secure against any attacks on Elgamal. The discrete logarithm problem and Integer factorization problem that is difficult to solve is the prime factor of the security. Even MGCA can be attacked by trying all the possible keys. Yet compared to regular Elgamal getting factors of large numbers into prime factors and solving discrete logarithms problems is still difficult for any brute force attacks. The disadvantage is since it is a one way function it cannot be used as authentication and execution process is slowed down.

3. Asymmetric cryptography techniques

RSA Algorithm

RSA is one of the most popular Asymmetric key cryptography for the transmission of secure data which is developed by the Ron Rivest, Adi Shamir, and Leonard Adleman in 1978. It is a block cipher system [16]. Fig 4 shows the encryption and decryption process. A public key is created and published which is based on two prime numbers along with an auxiliary value and that prime numbers must be kept secret. To decrypt the data, there must have some knowledge on prime numbers and key generation pair. Typically for the key exchange or digital signature or for the block data encryption RSA is used. In the RSA algorithm there are major steps, they are: key generation, encryption and decryption [11].

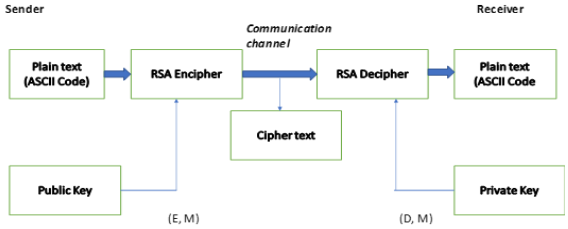


Fig. 4: RSA algorithm process

Encryption process begins with plain text and it is divided into blocks and translated to a process unknown format called cipher text. Cipher text is transmitted to the receiver. Two prime numbers 'P' and 'Q'. These two numbers are random prime numbers and P should not be equal to Q. Using prime numbers calculate RSA modulus 'n' by multiplying the two prime numbers P and Q i.e. n=P*Q. The equation phi-nv=(P-1)*(Q-1) is obtained from P & Q. calculate Eulerstoitent function 'phi-nv'. Generate a public key, theinteger value 'e' such that 'e' should be less than and it should be coprime to phi-nv. Generate a private key, then calculate the value of 'd' by using the equation de=1 +k phi-nv/n, where d is a decryption key. To encrypt the plain text 'M' consider the equation, $C=M^e \bmod n$, where C is the cipher text. After the transmission of cipher text to the receiver, to obtain the plain text 'M' use the equation $M=C^d \bmod n$.

Steps involved:

1. Select two random prime numbers P and Q by consider the condition that P should not be equal Q.
   P=3 and Q= 5.
2. Calculate RSA modulus 'n' by multiplying the two prime numbers P and Q nv=P*Q
   Nv=15
3. Calculate Euler's toient function 'phi-nv' by using the equation phi-nv= (P-1)*(Q-1).
   phi-nv=8
4. Select an integer 'E' for the encryption by consider two conditions 1<E< phi-nv and gcd(phi-nv,E). Selected integer 'E' should satisfy both the conditions.
   E=7
5. Calculate decryption key'd' using the equation de=1+ mod 40. 'd' is a private key.
   d=7
6. Let us consider the message as M=2
   For the encryption process, $C=M^E \bmod n$.

Where C=Cipher text, M=plain text, n=RSA modulus.
$C=2^7 \bmod 15$
$C=128 \bmod 15=8$
C=8 is a cipher text.

7. For the decryption of the cipher text which is transmitted to the receiver, calculate $M=C^E \bmod n$.
$M=8^7 \bmod 15$
$M=2097152 \bmod 15=2$
M= 2 is the original message.

Fig 5 shows the RSA algorithm encryption process. Here d is a private key, which is at receiver. The message which is to be transmitted is taken as '2'. After encryption process the cipher text obtained is cipher text 8, is transmitted to the receiver. Fig 6 shows the decryption process. Decryption process is opposite to encryption process; from the above steps, message is obtained. The output obtained is 2 which is the original message.

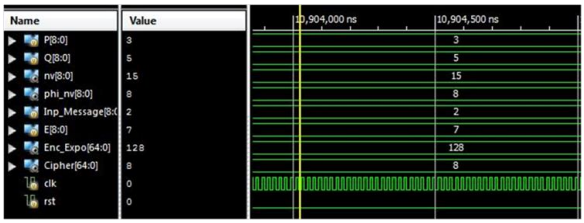


Fig. 5: RSA algorithm encryption results

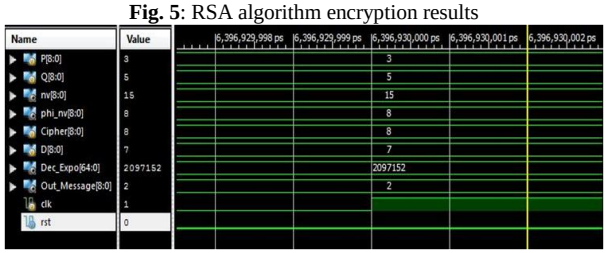


Fig. 6: RSA algorithm decryption results.

Diffie-Hellman algorithm

Diffie–Hellman (D-H) algorithm was published after the RSA algorithm [12]. D-H algorithm securely distributes keys over open communication channel [19]. Alice and Bobgenerate of a share secret key and then communicate among them uninterruptedly. Information should be exchanged using an unsecure communication channel. Intruder which is eve or spy cannot know the shared secrete key, as shown in Fig 9 . Let us consider an example with step by step process

Diffie –Hellman key exchange steps:

1. Consider two parties namely 'Alice' and 'Bob'; select a prime number 'P' and a primitive element 'G'.
P=23, G=5
2. 'Alice' selects an integer 'a' which is a secrete key and then transmits 'A' to the 'Bob'. Here 'A' is obtained from the equation $A=G^a \bmod P$
$a=4$, $A=5^4 \bmod 23=4$
3. Similarly Bob selects an integer 'b' which is also a secrete key and then transmits "B' to 'Alice'. Here $B=G^b \bmod P$.
$b=7$, $B=5^7 \bmod 23=17$
4. Alice calculates $s=B^a \bmod P$.
$s=17^7 \bmod 23=8$.
5. Bob calculates $s=A^b \bmod P$
$s=4^7 \bmod 23=8$.
6. Thus the two parties 'Alice' and 'bob' shares a same secrete key s=8.

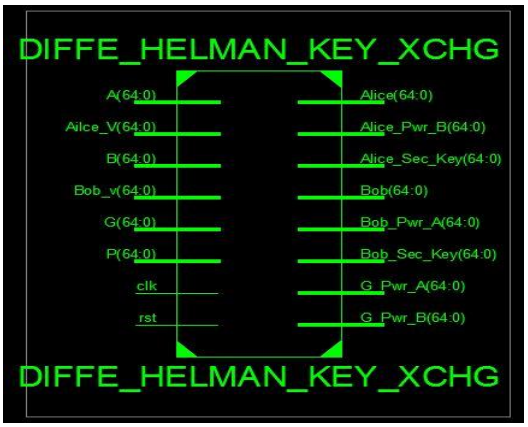


Fig. 7: Diffie –Hellman key exchange

Now both 'Alice' and 'Bob' have the same value 's'. In D-H algorithm both 'a' and 'b' values are secrete and remaining values P, G, A, B are public. After both 'Alice' and 'Bob' calculates the shared key then it can be used for the encryption key which is known only to them for transmitting data over the open communication channel.

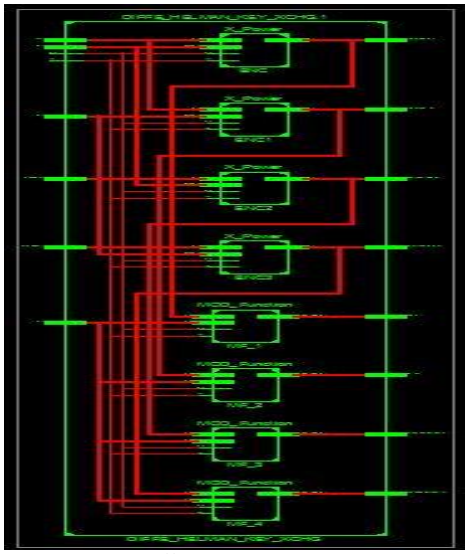


Fig. 8: Diffie-Hellman RTL diagram

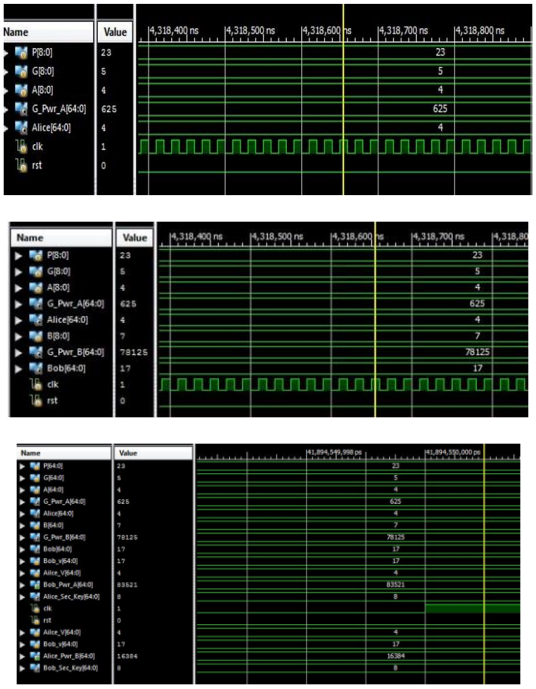


Fig. 9: Diffie-Hellman algorithm results

Fig 7 and Fig 8 shows the Diffie –Hellman key exchange and RTL schematic. Fig 9 shows the results obtained for the Diffie – Hellman algorithm. There are some advantages and disadvantages of the D-H algorithm. The main advantage of using D-H algorithm is that, it is easy perform exponents compared to solving the discrete logarithms [13] and there is no transmission of secrete key across the channel [14].

ElGamal algorithm

In 1985 TaharElGamal introduced the ElGamal algorithm for the public key cryptography. In D-H algorithm both parties should have an interaction to compute a common private key. This may lead to a problem if both parties cannot interact in time with each other due to some delays in communication process or if the receiver is unavailable. This problem can be solved by using the ElGamal algorithm by using a random exponent [15]. Private exponent in the D-H algorithm is replaced with a random exponent in the ElGamal algorithm. Hence there is no need of receiving party necessicity while encryption process is running. In ElGamal algorithm there three major components, they are: key generation, encryption and decryption.Consider two parties 'X' and 'Y'. Now message has to be transmitted from X to Y. Firstly Y has to consider some parameters like p, α and β. Here p is a prime number, α is a primitive element. Before the encryption starts Y has to send public keys p and α to the X. Y has to calculate β from the equation β= α^a mod p. Here 'a' is a private key which should not be transmitted and kept secret. Now for the transmission of the message 'M' from X to Y encryption of the message should be done. The encryption process is divided in to two parts y1 and y2. C (C1, C2) is the cipher text which is divided two parts. X has to compute C1 using the equation C1= α^k mod p. Here k is a random integer. Then compute the part in the cipher text C2 using the equation C2=M β^k mod p. Now the cipher text C (C1, C2) is transmitted to Y. For the decryption of the message 'M', Y should use his private key 'a' by using the equation M=C2^(p-1-a) mod p. Here D is a decrypted message; by this the original message can be obtained. ElGamal algorithm steps:

1. Alice has to choose a prime number 'P', primitive element G and private key 'X'.
   P=23, G=5 and X=6
2. compute H= G^X mod P
   H=5^6 mod 23=8
3. Alice transmits public keys P, G values to the transmitter 'Bob'.

Encryption process:

4. Bob, for the encryption of the message 'M' cipher text is divided in to two parts C1 and C2.
   Let M=11.
5. Calculate C1=G^R mod p
   R=3;C1=10,
6. Calculate C2= M H^R mod p. Where R is random integer.
   C2=66
7. Now the cipher text C (C1, C2) i.e.C (10, 66) is transmitted from Bob to Alice.

Decryption process

8. Then for the decryption of the cipher text. Calculate s=$C1^X$ mod P and M=C2 $s^{(P-2)}$ mod P where M is a decrypted message which is the original plain text.
   s=6 and M=11.
   M=11 is the original message.

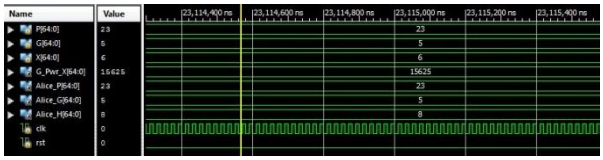


Fig. 10: ElGamal Algorithm Key generation pair

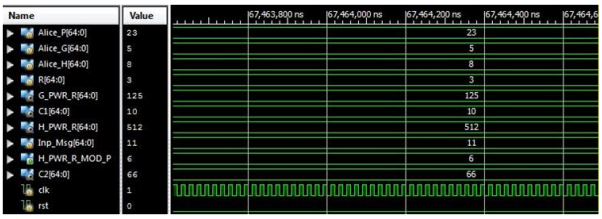


Fig. 11: ElGamal algorithm encryption results

Fig 10 shows ElGamal Algorithm Alice key pairand Fig 11 shows the encryption results of ElGamal algorithm. ElGamal algorithm increases the security level by the complex cipher text generation. Fig 12 shows the ElGamal algorithm decryption results. The cipher text size is twice the plain text so it is difficult for the attackers to decrypt the message.

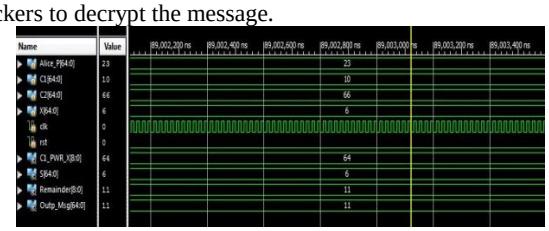


Fig. 12: ElGamal Algorithm Decryption results

4. Proposed asymmetric algorithm

Asymmetric algorithm with Error Detection and correction mechanism is proposed, with few properties of RSA and Diffie Hellman and Elgamal algorithm. Security level is enhanced by Hamming Error Detection and Correction, for data authentication. Selection of prime numbers, generation of public and private key pairs is obtained using RSA algorithm. Diffie-Hellman Algorithm is used for the key generation and Elgamal increases the security level by generating complex cipher text. To improve the security level circular shift operation is performed on precipher text. The encryption process starts with prime number selection, obtaining public and private key pairs. Input message is converted to precipher text, then Hamming Error Correction and Detection mechanism is applied, with an additional overall parity bit is considered for both ciphers, then C1 and C2 is obtained. The cipher text C (C1, C2) is transmitted to the receiver. Security level is raised by using Diffie- Hellman algorithm a secret shared key 's' is generated between Alice and Bob. This shared key is xored with private key'd' at transmitter to obtain 'k' and 'k' is transmitted to the receiver. Both sender and receiver will have shared key's'. Original decryption key'd' is obtained by performing xor operation between 'k' and pre shared key's'. Original message is obtained at receiver by performing reverse operation to encryption. Decrypted message is authorized, by gratifying two conditions, and then obtained message is an original plain text. The security level of the asymmetric algorithm depends on the length of message proportional to P&Q values which indirectly proportional to N value.

Proposed Asymmetric Algorithm Encryption and Decryption steps:

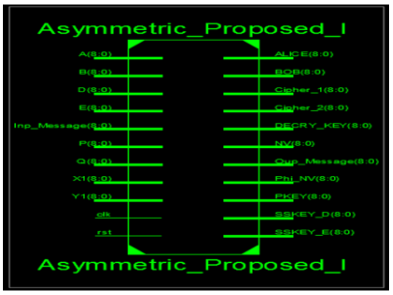
1. Prime numbers p=11 and q=3
2. N=pq=33
3. Phi =(p-1)(q-1)=20
4. Public key e =3
5. Private key d=7
6. Input message =5
7. Encryption process
   a) Divide the message in two 2 parts T1 and T2
   b) T1=(m+e)mod n=(5+3)mod 33=8 mod 33=8
   c) T2=(m+d)mod n=(5+7)mod 33=12 mod 33=12
   d) Divide T1 and apply right circular shift to right part and left circular shift to left part T1=000000 1000, Hamming Error Correction with an additional overall parity bit is included to obtain cipher text as c1=00000 0100=4

e) Divide T2 and apply right circular shifting to right part and left circular shifting to left part T2=00000 1100,Hamming Error Correction with an additional overall parity bit is included to obtain cipher text as c2= 00000 0110=6
f) The cipher textc (c1,c2)= (4,6) is transmitted to the receiver

8. Key generation using Diffie –Hellman method:
a) Both agree on two random numbers x=23 and y=2
b) Sender chooses a=2 and sends A=y^a mod x=2^2 mod 23=4.
c) Receiver chooses b=4 and sends B=y^b mod x=2^4 mod 23=16.
d) Sender computes s=B^a mod x=16^2 mod 23=3.
e) Receiver computes s=A^b mod x=4^4 mod 23=3.
f) Both have same key s=3.
9. To send private key k=d xor s=7 xor 3= 4.
10. Send encrypted key k to the receiver.
11. At the receiver for the private key d=k xor s =4 xor 3=7.
12. Decryption process:
a) Divide c1 and apply left circular shift to right part and right circular shift to left partof cipher C1=00000 0100, remove the parity bits and detect any errors and correct them,to obtain T1=00000 1000=8.
b) Divide c2 and apply left circular shift to right part and right circular shift to left part of cipherto C2=00000 0110, remove the parity bits and detect any errors and correct them, to obtainto obtainT2=000000 1100=12.
c) m=T2-d=12-7=5.
d) To verify whether obtained m is true or not: Two equations should be satisfym+e mod n=T1 and m+d mod n=T2. 5+3 mod 33=8 and 5+7 mod 33=12
i.To find e using the equation: e=(T1+d)-(m+d). e=(8+7)-(5+7)=3
ii.If both equations satisfy and equal to T1 & T2 then, the obtained message m is accurate.

5. Simulation results

Proposed Asymmetric algorithm is designed, simulated using Verilog HDL. Fig 13 shows the RTL schematic of proposed Asymmetric algorithm. Symmetric algorithm is faster but security level is less due to key distribution. This can be overcome using Asymmetric algorithm; high security can be achieved but consumes much time for the processing due to exponential and mod functions. The exponential operation is replaced with the addition in the proposed asymmetric algorithm; occupies less hardware resources, which improves decryption speed and time. Fig 14 shows the Encryption results at the sender and Fig 15 shows the Decryption results at the receiver. Fig 16 shows comparison of different parameters like Number of slice registers, Number of slice LUTs, Number of fully used LUT-FF pairs, Number of BUFG/BUFGCTRLs and Number of DSP48E1s. By the observation it is clear that the proposed Asymmetric algorithm showed better performance in terms of Number of slice registers, Number of slice LUTs, Number of fully used LUT-FF pairs and Number of DSP48E1s.



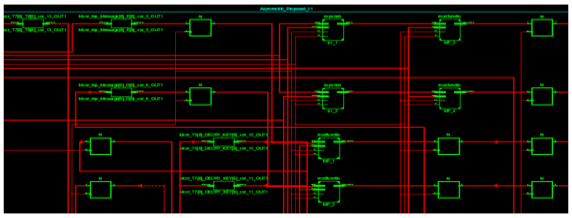


Fig. 13: RTL schematic of proposed asymmetric algorithm

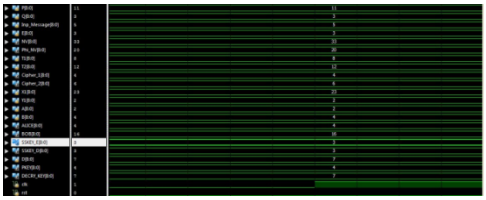


Fig. 14: Proposed asymmetric algorithm encryption results

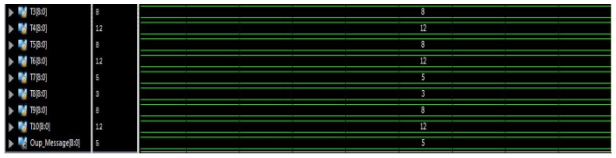


Fig. 15: Proposed asymmetric algorithm decryption results

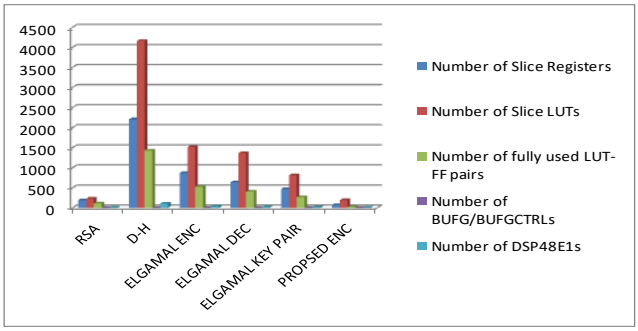


Fig. 16: Comparison of different parameters

Fig 17 shows implementation of Method-I using Virtual Input Output (VIO) core logic analyzer binded to design. Due to less no. of GPIO resources available on FPGA, to monitor Ciphers C1, C2 which are huge bit length, VIO Core generates the required no. of inputs and capture the outputs.

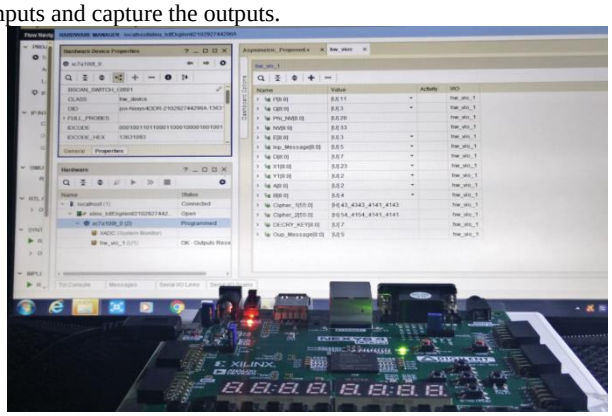


Fig. 17: Implementation of proposed method using VIO Core Logic analyzer

6. Conclusion

To increase the performance of the asymmetric crypto algorithms by reducing hardware in terms of slicing the exponential operation which improves the factors like less hardware resources, better speed and low latency with more surety by translating cipher text to DNA. Proposed asymmetric algorithm is based on three algorithms like RSA, ElGamal and Diffie- Hellman Key exchange protocol. Limitations of asymmetric algorithms are its speed and the limitation of Symmetric algorithms is key distribution and

security levels. Resource occupancy report on fig 16depict various parameters like Number of slice registers, Number of slice LUTs, Number of fully used LUT-FF pairs, Number of BUFG/ BUFGCTRLs and Number of DSP48E1s. The observation noticeably points that the proposed Asymmetric algorithm shows better performance in terms of hardware resources with low latency. Chipscope Pro logic analyzer-Virtual Input Output core is binded to design for hardware debugging which is targeted for Artix-7 XC7A100T-1-CSG324C architecture. To monitor and capture the output signals at selected specified state by applying random input stimuli at runtime in Nexys4 DDR FPGA Board.

References

[1] Krishna BM, Habibulla Khan GL, Lohitha B, Bhavitha E, Sri PT & Kumar BA, "FPGA Implementation of DNA Based AES Algorithm For Cryptography Applications", *International Journal of Pure and Applied Mathematics*, Vol.115, No.7 , (2017), pp.525-530.

[2] Krishna BM, Khan H & Madhumati GL, "Reconfigurable pseudo biotic key encryption mechanism for cryptography applications", *International Journal of Engineering & Technology*, Vol.7, No.1.5, (2017), pp.62-70.

[3] Krishna BM, Khan H, Madhumati G, Kumar KP, Tejaswini G, Srikanth M & Ravali P, "FPGA Implementation OF DES Algorithm Using DNA Cryptography", *Journal of Theoretical and Applied Information Technology*, Vol.95, No.10, (2017), pp.2147-2158.

[4] Krishna BM, Madhumati, GL, Ganesh MSR, Bhargav Y, Krishna VMV & Kumar OM, "Biometric Based Industrial Machine Access Control System Using FPGA", J*ournal of Theoretical and Applied Information Technology*, Vol.79, No.1,(2015).

[5] Krishna BM, Madhumati GL & Khan H, "Design of Dynamically Reconfigurable Input/Output Peripheral Based Wireless System", *Indian Journal of Science and Technology*, Vol.9, No.30, (2016), pp.1-9.

[6] Krishna BM, Madhumati G & Khan H, "Dynamically Evolvable Hardware-Software Co-Design Based Crypto System Through Partial Reconfiguration", *Journal of Theoretical & Applied Information Technology*, Vol.95, No.10, (2017), pp. 2159-2169.

[7] Krishna BM, Madhumati GL & Khan H, "FPGA Implementation Of Partially Reconfigurable DNA Cryptography Methods Through Wireless Using Zigbee", *ARPN Journal of Engineering and Applied Sciences*, Vol.11, No.21, (2006), pp.12514-12522.

[8] Beebe NH, "A Bibliography of Publications in Computer Networks and ISDN Systems", Journal of Advanced Research in Dynamical and Control Systems, Vol.9, (2014), pp.1566-1586.

[9] Krishna BM, Nayak VN, Alekhya PDS, Sree KS, Dhruvitha M & Yashwanth V, "FPGA implementation of DNA based S-DES cryptography technique", *International Journal of Pure and Applied Mathematics*, Vol.115, (1994), pp.233-239.

[10] Krishna BM, Chowdary GR, Vardhan GC, Ram KS, Kishore P, Madhumati GL & Khan H, "FPGA based wireless electronic security system with sensor interface through GSM", *Journal of Theoretical & Applied Information Technology*, Vol.89, No.2, (2016), pp.489-494.

[11] Rao IRSN, Krishna BM, Shameem S, Khan H & Madhumati GL, "Wireless Secured Data Transmission using Cryptographic Techniques through FPGA", *International Journal of Engineering and Technology (IJET)*, (2016), pp.0975-4024.

[12] Chowdary MU, Krishna BM, Madhumati KMG & Khan H, "ZigBee Based Wireless Data Transmission with LDPC codes using FPGA", *International Journal of Engineering and Technology*, Vol.8, No.2, (2016), pp.653-659.

[13] Reddy AG, Das AK, Yoon EJ & Yoo KY, "A secure anonymous authentication protocol for mobile services on elliptic curve cryptography", *IEEE Access*, Vol.4, (2016), pp.4394-4407.

[14] Wu Z, Su D & Ding G, "ElGamal algorithm for encryption of data transmission", *International Conference on Mechatronics and Control (ICMC)*, (2014), pp.1464-1467.

[15] Chen C, Wang T, Kou Y, Chen X & Li X, "Improvement of trace-driven I-Cache timing attack on the RSA algorithm", *Journal of Systems and Software*, Vol.86, No.1, (2013), pp.100-107.

[16] Wang CH, Chuang CL & Wu CW, "An efficient multimode multiplier supporting AES and fundamental operations of public-key cryptosystems", *IEEE transactions on very large scale integration (VLSI) systems*, Vol.18, No.4, (2010), pp.553-563.

[17] Wang D, Jiang Y, Song H, He F, Gu M & Sun J, "Verification of Implementations of Cryptographic Hash Functions", *IEEE Access*, Vol.5, (2017), pp.7816-7825.

[18] Imamoto K & Sakurai K, "Design and Analysis of Diffie-Hellman-Based Key Exchange Using One-time ID by SVO Logic", *Electronic Notes in Theoretical Computer Science*, Vol.135, No.1, (2005), pp.79-94.