FPGA Implementation of Digital Data using RSA Algorithm

V.SeethaRama Rao[1], Indrajith Natarajan[2], Manjunath Dachepally[3]

[1]*Assistant Professor at Sreenidhi Institute of Science and Technology, Hyderabad*

[2]*UG Scholarat Sreenidhi Institute of Science andTechnology,Hyderabad*

[3]*UG Scholar at Sreenidhi Institute of Science and Technology, Hyderabad*

Abstract - Secured Communication System plays a major role over unsecured Communication channel. So, Security can be achieved Cryptosystem techniques, which is used for network security for data. It mainly describes architecture and modeling developed for RSA Algorithm which is a public key algorithm. The key generation stage aims to generate a pair of public key and private key, and then the private key will be distributed to receiver according to certain key distribution schemes The extension of this project includes the application of RSA algorithm on FFT data. Fast Fourier Transform is the fastest Fourier transform technique adopted in Signal Processing. This Butterfly model can be implemented on any bits of data effectively. FFT can be implemented using Decimation in time and Decimation in frequency. Converting a raw data into discrete in frequency using FFT and then performingEncryption gives more accurate results. This technique is designed by using Verilog HDL with Xilinx 2017.4 version

Keywords: Cryptosystem, FFT,RSA

I. INTRODUCTION

RSA Cryptography named for its inventors, Rivest, Shamir and adleman is an efficient work for public and private key Cryptography. In public Cryptography encoding key is shared to everyone which is not the case in private Cryptography. RSA algorithm is useful for both Encryption and Decryption of messages so that they remain secure for transmission. Encryption is performed on plain text or message and Decryption is performed on the cipher text. The entire process is based on modular multiplication. There are many methods to implement modular multiplication but the efficient method reduces time and improves the performance of the algorithm. This paper proposes the hardware implementation of RSA algorithm. RSA algorithm includes different modules which together implements algorithm. A random number generator is used to generate numbers randomly which are further given to primary tester. The function of primary tester is to check for the condition of prime. Two such numbers are selected and given to the further modules. Other modules include Euclid algorithm (performs GCD and inverse GCD),Modular multiplication(performs exponentiation and modulus for both message and cipher text).Encryption for public key is represented as (e,n) whereas the Encryption of private key is represented as (d,n).RSA Encryption technique can be performed on any digital data available. This technique can be applied on Fourier transformed data . Fast Fourier Transform is an efficient method for Fourier transform. Fourier Transform converts time domain data into frequency domain data. RSA method can then be applied on Fourier transformed data to make the Encryption more efficient. Reverse process occurs at Decryption method.

II. RSAALGORITHM

RSA algorithm includes different steps:

Step:-1 Consider two random number p, q

Step:-2 Check for the functionality of prime for both the numbers. If both numbers are prime then continue the process, else repeat the procedure until two prime numbers are obtained.

Step:-3 calculate $n=p*q$ and $\Phi(n)=(p-1)*(q-1)$

Step:-4 Choose the value of e such that $(\Phi(n),e)$ are relatively co-prime(GCD =1) and $1<e<\Phi(n)$

Step:-5 Choose the value of d such that d=(e-1)modulus $\Phi(n)$. d and e have inverse relationship.(if d is used at Encryption method then e is used at Decryption method and viceversa). e is considered as public key whereas d is considered as private key.If e is the Encryption key, d is the Decryption key , M is the message or plain text, C is the cipher text then Encryption and Decryption process is as follows C=M^e(mod nM=C^d(mod n)

Fast Fourier Transform is the efficient Fourier transform technique which converts time domain signals into frequency domain signals. It is also called as Butterfly model. Fast Fourier Transform can be done in two ways, Decimation in time and Decimation in frequency. Frequency domain signal can be reverted back using Inverse Fast Fourier Transform technique which is an efficient method for Inverse Fourier transform. Complexity of Fourier Transform will be reduced from O(nlogn) to O(n^2) (n is the data size) if Fast Fourier Transform technique is used. Fast Fourier Transform can be used for Chebyshev algorithms, Hartley algorithms. They can also be used for solving difference algorithms. A Fourier Transformed data can be applied to RSA algorithm which makes the encoded data efficient. The figure below is an example for

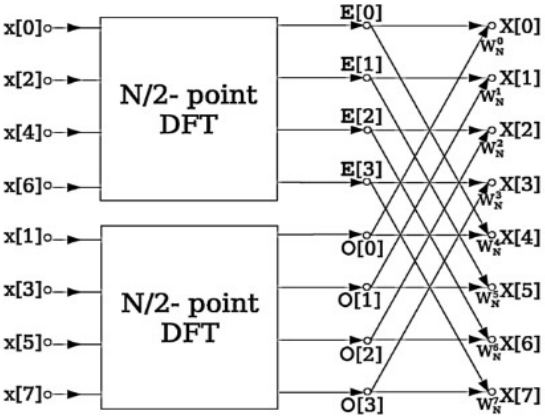


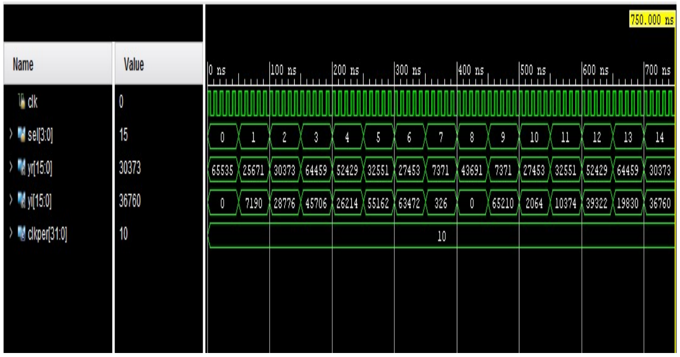
Fig FFT butterfly model.

Above figure is an example for decimation for decimation in time, which rearranges the input values in reverse bitwise order and then generates the output, whereas decimation in frequency gets the reverse bitwise order outputs which are to be rearranged later

III. SIMULATION RESULTS

Below Figure shows the Fourier Transform technique using Fast Fourier Transform.16bit FFT is implemented. These values are considered as 2, 4, 8, 16, 32,64……65536 .Totally we are applying for 16 samples .Fourier Transformed data output can be applied to RSA algorithm to develop public and private

Design Action	Tool Name
Design Entry	Verilog HDL
Synthesis Xilinx	Synthesis Tool(XST)
Simulation	Xilinx 2017.4
Implementation	FPGA Editor, Plan Ahead
Board FPGA	XC3s 500e fg320

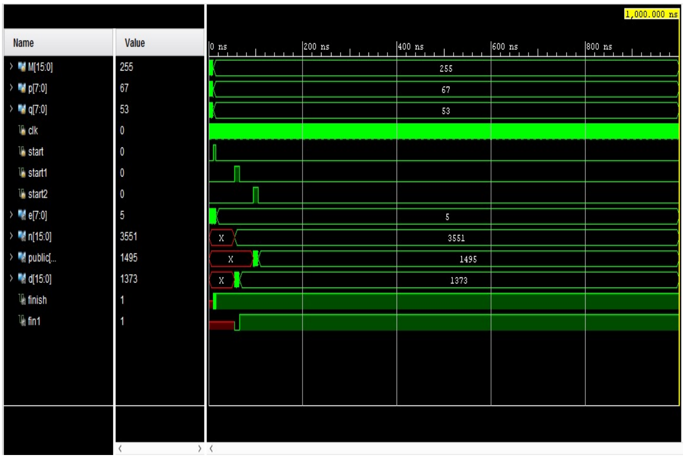


Simulation results of FFT

Below figure shows the Encryption of RSA algorithm using public key. Consider the following example.

- Let the value of p=67, q=53
- According to the formula n=p*q; n=3551
- Value of Φ(n)=3423
- According to Extended Euclid algorithm value of e=5
- Value of d can be obtained as 1373
- Encryption using public key can be defined as C=M^e (mod) n, here e is the public key

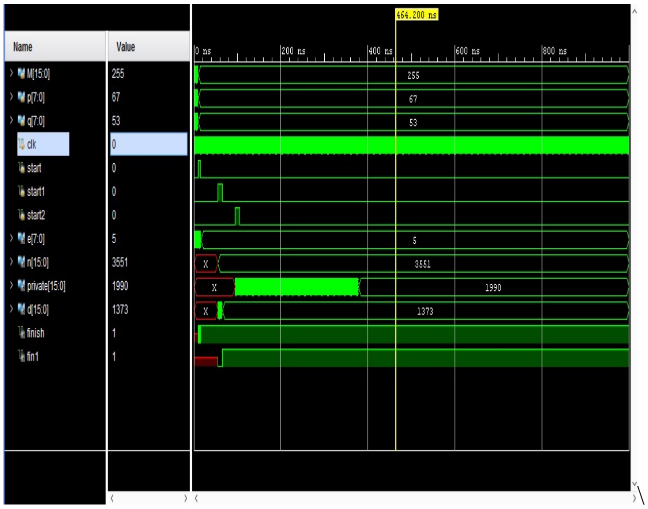
Consider M=255,C=(255^e)mod 3551; C=1495



Name	Value
M[15:0]	255
p[7:0]	67
q[7:0]	53
clk	0
start	0
start1	0
start2	0
e[7:0]	5
n[15:0]	3551
public[...	1495
d[15:0]	1373
finish	1
fin1	1

Simulation results of RSA public key

Below Figure show the Encryption of RSA algorithm using Private key. Considering all the above values, private key Encryption can be obtained as follows. Here d is considered as private key .C=M^d mod n; C= 255^d mod 3551; C=1990

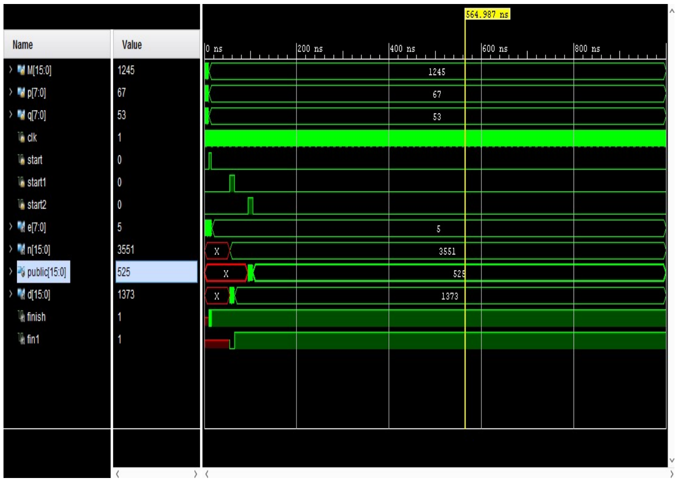


Name	Value
M[15:0]	255
p[7:0]	67
q[7:0]	53
clk	0
start	0
start1	0
start2	0
e[7:0]	5
n[15:0]	3551
private[15:0]	1990
d[15:0]	1373
finish	1
fin1	1

\

simulation results of RSA private key

RSA is an Asymmetric key technique, where private and public key can be used for Encryption and Decryption techniques. If public key is used atEncryption then private key can be used at Decryption and vice versa. Below Figures depicts the Encryption of FFT data using public key i.e and Decryption using private key .Encryption process with public key is as follows: Here values of p and q are similar to the above example and the value of M is considered as 1245.

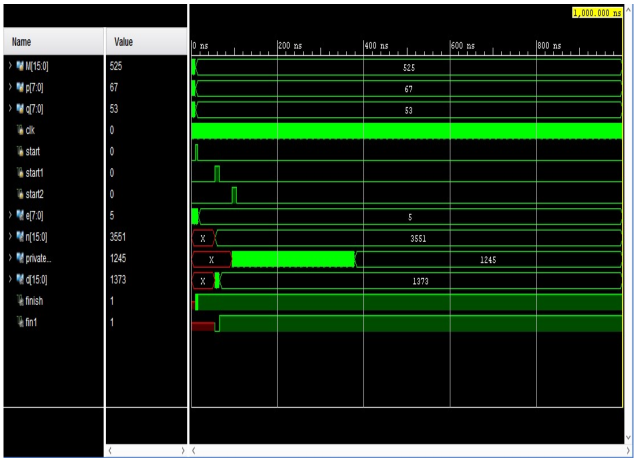


Name	Value
M[15:0]	1245
p[7:0]	67
q[7:0]	53
clk	1
start	0
start1	0
start2	0
e[7:0]	5
n[15:0]	3551
public[15:0]	525
d[15:0]	1373
finish	1
fin1	1

Simulation results of Encryption with public key

Decryption process with private key is as follows:

For this process, output obtained atEncryption process is given as input and output obtained from this process is 1245.

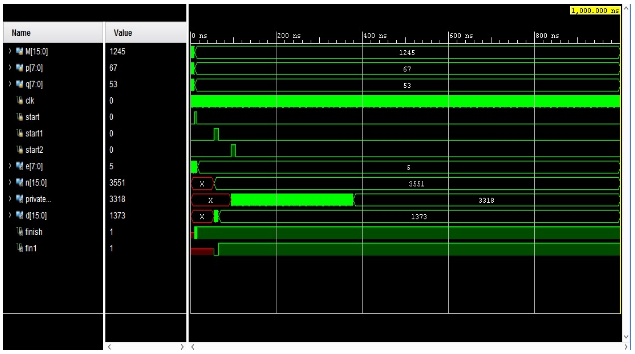


Name	Value
M[15:0]	525
p[7:0]	67
q[7:0]	53
clk	0
start	0
start1	0
start2	0
e[7:0]	5
n[15:0]	3551
private...	1245
d[15:0]	1373
finish	1
fin1	1

Simulation results of decryption with private key

After Decryption process original message is recovered back . Message given is same as message obtained. In this way original signal can be recovered using RSA Decryption technique. Decryption can be verified in two ways using RSA algorithm. The other method is as follows.Below Figures depicts the Encryption of FFT data using private key i.e d and Decryption using public key i.e Encryption process using private key is also follows:

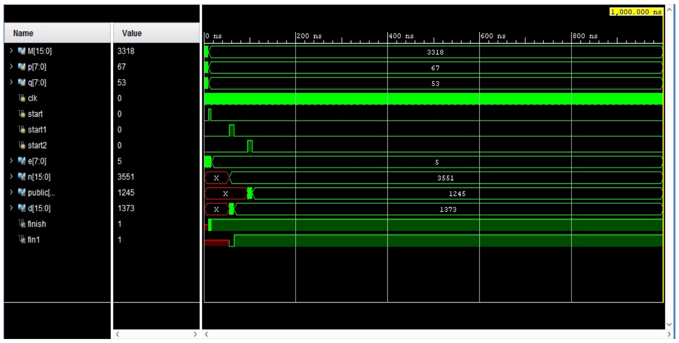
The value of M is considered as 1245.



Name	Value
M[15:0]	1245
p[7:0]	67
q[7:0]	53
clk	0
start	0
start1	0
start2	0
e[7:0]	5
n[15:0]	3551
private...	3318
d[15:0]	1373
finish	1
fin1	1

Simulation results of Encryption with private key

Decryption process with public key is as follows:



Name	Value
M[15:0]	3318
p[7:0]	67
q[7:0]	53
clk	0
start	0
start1	0
start2	0
e[7:0]	5
n[15:0]	3551
public[...	1245
d[15:0]	1373
finish	1
fin1	1

Simulation results of decryption with public key

IV. CONCLUSIONS AND FUTURE SCOPE

RSA algorithm is most commonly known for its asymmetric nature. This nature helps to protect a part of the process from unauthorised users. If Encyrption is done with public key then Decryption can be protected using private key and in the same way ifEncryption is performed using private key, thenEncryption can be protected from unauthorised users and then Decryption can be shared among them. This Asymmetic nature can be performed on any kind of data. In this project we dealth with Fourier Tranformed data. Fast Fourier Transform technqiue is used to make Fourier Transform more efficient and less time consuming. This data is given as a message to RSA algorithm and then Decryption is performed to recover back the message.

This project is done on Xilinx vivado 2017.4 and simulation results are observed. This project can also continued by performing Inverse Fast Fourier Transform technique such that original signal which is applied to FFT can be obtained.

REFERENCES

[1] R. L. Rivest, A. Shamir, and L. Adleman. "A Method for obtainingdigital signatures and public-Key cryptosystems". Comm. ACM, 21:120126,1978.

[2] Schneier Bruce, Cryptographie appliquée - Algorithmes, protocoles etcodes source en C - 2ème édition, International Thomson PublishingFrance, 1997-Applied Cryptography - Protocols, Algorithms, and SourceCode in C - 2nd Edition.

[3] Bouallegue Ridha, Hamdi Omessaad « Sécurité des CryptoSystèmes».*ENIT; SUPCOM, Tunis, Tunisie.2003.*

[4] Alan Daly and William Marnane "Efficient Architectures forimplementing Montgomery Modular Multiplication and RSA ModularExponentiation on Reconfigurable Logic". -University College CorkIreland 2001.

[5] Young Sae Kim, Woo Seok Kang, Jun Rim Choi "Implementation of1024-bit modular processor for RSA cryptosystem" School of Electronicand Electrical Engineering, Kyungpook National University,Korea.2001.

[6] John Fry - Martin Langhammer. "RSA & Public Key Cryptography inFPGA"2000.

[7] A.Mazzero, L.Romano "FPGA-based Implementation of a serialRSA" processor, G.P.Saggese-Universita'degli Studi Napoli "FedericoII" 2002.

[8] Tom Kean "Cryptography Rights Management of FPGA IntellectualProperty Cores" Edinburgh EH8 8YB United Kingdom.1999.

[9] S.H. Tang, K.S. Tsui and P.H.W. Leong “Modular Exponentiationusing Parallel Multipliers” The Chinese University of Hong KongShatin, NT, Hong Kong 2001.