FPGA Implementation of Smart Cryptography Algorithm

B. Prathiba, E. Lakshmi Prasad, N. B. Hulle, Sarika R. Khope

Abstract: ***Mceliece algorithm is one of the simplest algorithm which is widely used in cryptography application for encrypting and decrypting the data. In the proposed work, mceliece algorithm able to allow an infinite number of characters, but for every nine characters, it considered as one iteration due to the fixed length of matrix size (3*3). Findings: This mceliece application running over micro blaze processor, and this processor implemented on Spartan-3E FPGA. The entire experimental setup is implemented using Xilinx platform studio 14.3 and targeted on Spartan 3E-1600E FPGA.***

Index Terms: Mcecliece cryptosystem, micro blaze processor, Encryption, Decryption.

I. INTRODUCTION

Cryptography is a technique that is widely used for protection of data [1]. Especially public- key cryptography provides additional benefits for security purposes. There are various Cryptographic techniques such as AES Algorithm, DES algorithm, Triple DES and so on [2], out of which simplest approach is mceliece algorithm.

Mceliece is a public key cryptosystem algorithm developed by Mceliece [3]. Mceliece cryptosystem is mainly based on linear error-correcting codes. It is the first public key cryptography used in coding theory for encrypting and decrypting the messages, it is faster than other algorithms. But the mceliece algorithm is rarely used due to its large key size and low data rate.

The main objective of this paper is to propose mceliece algorithm for encryption and decryption using micro blaze processor implemented on Spartan-3E FPGA.

S. R. Shrestha [4] proposed the examples of attacks that will occur in mcecliece cryptosystem. Also, they have explained the process of increasing the security without increasing the key size. They also gave a brief explanation about Goppa codes on which mcecliece system is based. T.P. Berger [5] proposed Implementation of mcecliece cryptosystem on reconfigurable hardware where the mceliece scheme was first implemented on Xilinx Spartan-3 FPGA.

B.Biswas [6] introduced mceliece cryptosystem based on moderate density parity check (MDPC) codes in which soft decoding algorithms are used where the key size is reduced to 25% compared with existing methods.

M.P Priyanka [8] proposed a novel architecture for purpose of key generation, encryption, decryption which is realized on Virtex-5 FPGA and tested with Xilinx software.

Manuscript published on January 30, 2020.
* Correspondence Author

Prathiba B*, Department of E&TC, G. H. Raisoni Institute of Engineering & Technology, Pune, India.
Email: balireddyprathibha@gmail.com
E. Lakshmi Prasad, Sr. DFT Engineer, Xilinx India Pvt. Ltd., India. Email: lakshmi_prasad2@yahoo.com
N. B. Hulle, Department of E&TC, G. H. Raisoni Institute of Engineering & Technology, Pune, India.. Email: nagnath.hulle@raisoni.net
Sarika R Khope, E&TC, G.H.Raisoni Institute of Engineering & Technology, Pune, India. Email: sarika.gabhane@raisoni.net

II. METHODOLOGY

1. Mceliece encryption process:

In encryption process, messages are considered as inputs and which will be arranged in the form of the matrix [i][j] and key length matrix[i][j] is fixed [5]. To get an encrypted message, multiply the message matrix[i][j] and key length matrix[i][j]. Note the matrix size of the message and key length is (3*3). Therefore, for every nine characters considered as one iteration, likewise it allows infinite characters but it determines the number of iterations of an output message. Encryption process as shown in figure-1.

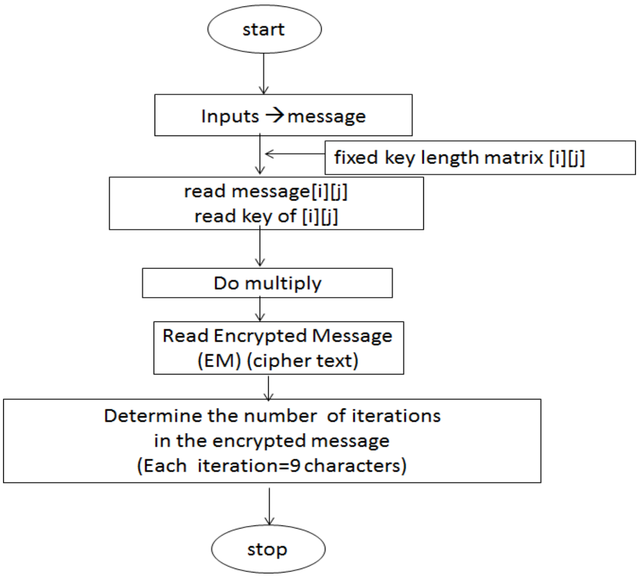

Figure 1: Encryption process

2. Mceliece Decryption process:

In decryption process the inputs are encrypted message [i][j] and inverse key matrix [i][j] of fixed length.

Here, the same encryption process is followed for decryption, but reduction technique is used. Decryption process as shown in figure-2.

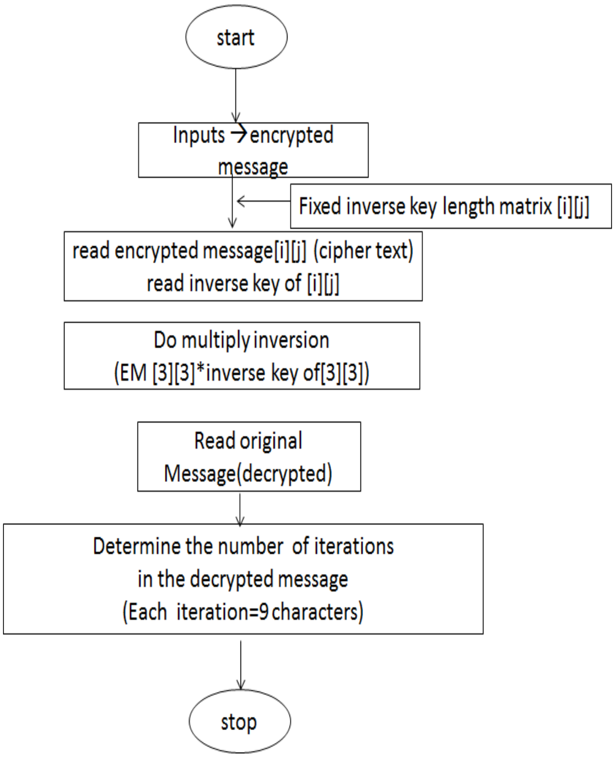
start
Inputs → encrypted message
Fixed inverse key length matrix [i][j]
read encrypted message[i][j] (cipher text)
read inverse key of [i][j]
Do multiply inversion
(EM [3][3]*inverse key of[3][3])
Read original Message(decrypted)
Determine the number of iterations in the decrypted message
(Each iteration=9 characters)
stop

Figure 2: Decryption process

III. IMPLEMENTATION OF MICRO BLAZE PROCESSOR

The initial step is to create a new Base System Builder (BSB) project using Xilinx Platform Studio (XPS). Next, the step is to select the interconnect type i .e., Processor local bus (PLB) and board selection. In this experimental work, Spartan- 3E FPGA boards is selected and specifies the board configurations according to requirements. Next step in the system configuration, we have to select the Single micro blaze processor instead of the dual micro blaze processor system [6]. Next step is to select the peripheral configuration, where we select the processor and its clock frequency. By clicking the finish button, it allows to open the new BSB project. Implementation of BSB project as shown in figure-3.

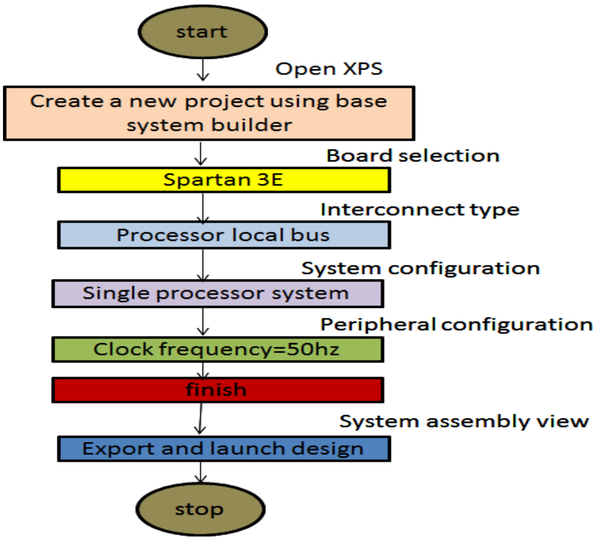

Figure 3: Flowchart for implementing micro blaze processor

In the Xilinx XPS window, a synthesized netlist of micro blaze processor can view in Graphical design tab. Micro blaze setup is ready which consists of several peripherals such as PLB, UART, timer, and bus interfaces, and input and output ports.

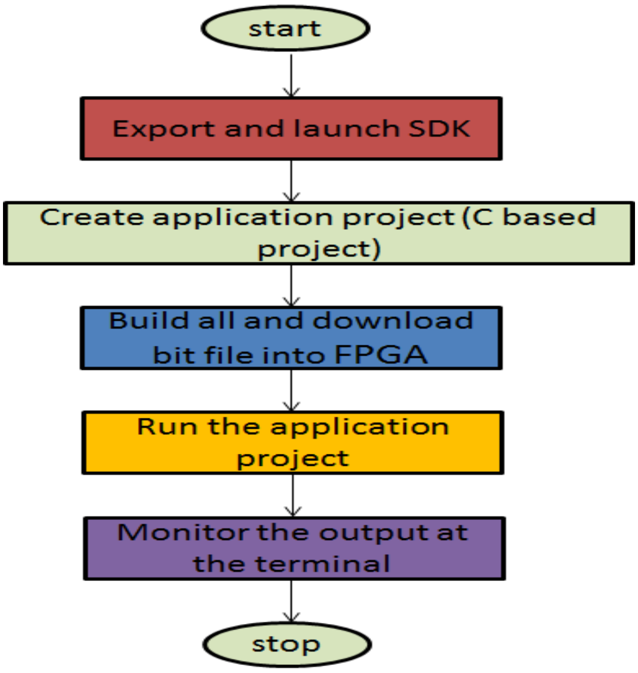
start
Export and launch SDK
Create application project (C based project)
Build all and download bit file into FPGA
Run the application project
Monitor the output at the terminal
stop

Figure 4: Flowchart for SDK

Then export this design from EDK to SDK by clicking on export design tab. After exporting and launching the SDK environment, we must create a 'c' application. After creating an application, we must compile and select the build all option and import the bit file into FPGA which contains micro blaze processor.

Then run the application of mceliece algorithm over on Spartan- 3E FPGA. The resultant output appears in the terminal window. Implementation of the SDK project as shown in figure-4.

Retrieval Number: E6126018520/2020©BEIESP
DOI:10.35940/ijrte.E6126.018520
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

IV. EXPERIMENTAL RESULTS

In the proposed work, mceliece algorithm application running on micro blaze processor, this processor is implemented on the Spartan 3E FPGA board [6]. The communication PC and FPGA is interfaced by using UART. The inputs are given to the FPGA and after computation, the output response of FPGA can observe on the terminal window [7] [8].

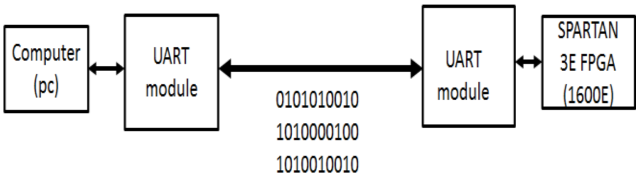

Figure5: Communication between PC and FPGA

The implementation of the communication process between PC and FPGA as shown in figure-5 and photocopy is shown in figure 6.

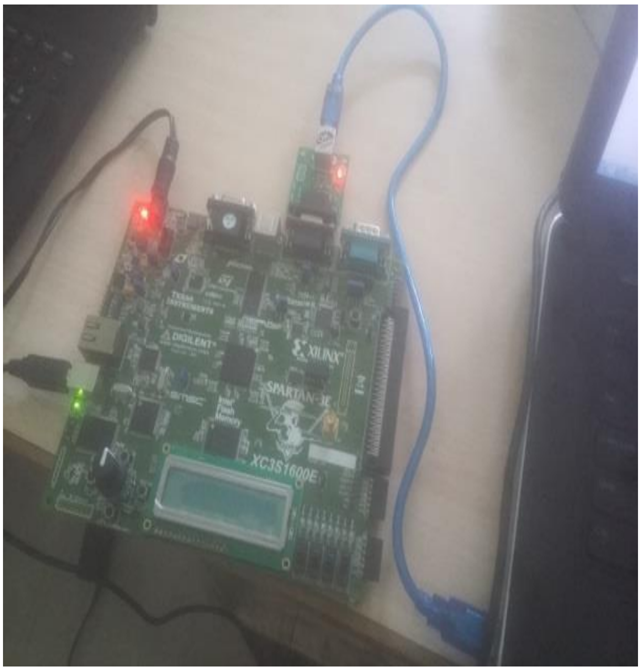

Figure 6: photo copy of communication process

Synthesis report:

Micro blaze device utilization summary is shown in table-1, the total number of LUTs is 5,599. The total number of Dual RAMs are 342. The number of IOB flip flops are 102. The total number of slice flip flops are 4,551 and the total number of occupied slices are 5,129.

Table 1: synthesis report

Logic utilization	**Used**	**Available**	**Utilization**
Number of slice flip flops	4,551	29,504	15%
Total number of 4-input LUTs	5,599	29,504	18%
Number of bonded IOBs	128	250	51%
Number of DCMs	2	8	25%

Simulation results:

The minimum period is required to transmit a bit is 18.705ns. The maximum path delay taken by the processor from one node to other is 2.912 ns. The actual clock for direct and derivative periods is 6.00ns and 19.040ns. The maximum frequency taken by the processor is 535 MHZ. The total number of slices utilization is 51% and the LUTs utilization is 18%.

Constraint report:

The TS_sys_clk_pin constraints require the period of 20.000ns. The TS_clock_generator_0_clock_generator_0_SIG_DCM1_CLKX requires the period of 10.000ns. Thus, the period requirement of various constraints is given in the following table-2.

Table 2: Constraint report

Constraint	Period Requirement	Actual Period		Paths Analyzed	
		Direct	Derivative	Direct	Derivative
TS_sys_clk_pin	20.000ns	6.000ns	19.040ns	3	248328
TS_clock_generator_0_clock_generator_0_SIG_DCM1_CLK0	20.000ns	16.780ns	N/A	242537	0
TS_clock_generator_0_clock_generator_0_SIG_DCM1_CLK2X	10.000ns	4.800ns	9.520ns	3	5968
TS_clock_generator_0_clock_generator_0_SIG_DCM0_CLK0	10.000ns	9.520ns	N/A	5435	0
TS_clock_generator_0_clock_generator_0_SIG_DCM0_CLK90	10.000ns	7.900ns	N/A	533	0

Retrieval Number: E6126018520/2020©BEIESP
DOI:10.35940/ijrte.E6126.018520
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

V.CONCLUSION

In this paper, mceliece algorithm is designed for cryptographic applications using micro blaze processor. Here, mceliece encryption and decryption algorithm is tested with a plain text using mceliece algorithm. In this algorithm, it allows infinite number of characters but due to the limitation of matrix size, the number of iterations is taken place for encrypting and decrypting the data. For every nine characters one iteration is considered. The whole process is implemented on Spartan -3E FPGA. The minimum amount of time is required for bit transmission is 18.605ns.

REFERENCES

1. Forouzan and Mukhopadhyay, "Cryptography and Network Security", 2nd Edition, McGraw Hill.
2. Jignesh R.Patel, Rajesh S.Basode, Vikas Kaul "Hybrid security algorithm for data transmission using AES- DES",.IJAIS-2012.
3. J.L. Beuchat, N.Sendrier, A.Tisserand , and G.Villard " FPGA Implementation of a recently Published Signature Scheme". Technical report, INRIA 2004.
4. S. R. Shrestha and Y.S. Kim, "New design of McEliece type cryptosystem," in Proc. 2013 the 23th Joint Conf. Commune 2013.
5. T.P. Berger, P.L.Cayrel, P.Gaborit, and A.Otmani , "Reducing the key length of McEliece Cryptosystem," in *Proc. AfricaCrypt'09*, 2009.
6. T. Hard castle "Spartan-6 FPGA Dual-Lockstep Micro blaze Processor with Isolation Design Flow", Xilinx Application Note Author, XAPP584 (vol.0) 2012.
7. B.Biswas and N.Sendrier, "McEliece cryptosystem implementation: Theory and practice," in *Proc. Int. Workshop Post-Quantum Cryptography*, 2008.
8. M.P Priyanka, E.Lakshmi Prasad, A.R.Reddy, "FPGA implementation of image encryption and decryption using AES-128bit core" International Conference on Communication and Electronics Systems (ICCES), Pages: 1 - 5, DOI: 10.1109/CESYS.2016.7889929, 2016.

AUTHORS PROFILE

B. Prathiba, currently working as an Asst.Prof. in the department of E&TC, G.H.Raisoni Institute of Engineering and Technology, Pune. She has published the articles in various international publications in the area of Wireless Sensor Networks. Her interested areas are Wireless Sensor Networks, Microcontrollers and VLSI.

E. Lakshmi Prasad, currently working as Sr. DFT Engineer, Xilinx India Pvt Ltd. He completed his Ph.D in the department of ECE from JNTUA. He has published several articles in both national and international journals in the area of Network on chips. He also served as Front-End VLSI design engineer for two years in ISHITV technologies. His interested areas are scalable architectures, system on chip, Network on chip, low power, and low latency design applications.

N. B. Hulle, currently working as an Associate.Prof. in the department of E&TC, G.H.Raisoni Institute of Engineering and Technology, Pune. He has published the articles in various international publications in the area of VLSI, Network Security and Cryptography. His interested areas are VLSI and Control Systems.

Sarika Khope, currently working as an Asst.Prof. in the department of E&TC, G.H.Raisoni Institute of Engineering and Technology, Pune. She has published the articles in various international publications in the area of VLSI,. Digital Systems. Her interested areas are Machine Learning and VLSI.