ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Simulation and Synthesis for TACIT Network Security in Hardware Description Language Environment

Yogesh Kumar Sharma*, Dr. Manoj Kumar
* Research Scholar, Mewar University, Chhittorghara, India
Dept. of Mathematics, R.K (PG) College, Shamli, U.P, India
yks_mzn@rediffmail.com

Abstracts

In cryptography, it is necessary to encrypt and decrypt the data for code storage and security. Sometimes it is very difficult to secure physically all access networks. The research paper has introduced a new block cipher cryptographic symmetric key algorithm named "TACIT Encryption and Decryption Technique" and its implementation in hardware description language environment. There are already many algorithms supporting to encryption and decryption process over networks, but limited to their block size and key size. In the new TACIT network security algorithm, the key size and text size may be of 'n' bits and it provides better results if key size is larger than block size. In the research, it is emphasized to develop chip for network security and it verified by experimentation on Sparten-3 FPGA synthesis. It is possible to enhance network performance and security by exploiting modern features in Field Programmable Gate Arrays (FPGA), which allow the modeling of encryption and decryption algorithm on System-on-Programmable-Chip (SOPC). The work is carried out on modeling and simulation tools, Xilinx ISE 14.2 and Model SimEE 10.1b student's edition of Mentor Graphics Company.

Keywords: Application Specific Integrated circuits (ASIC), Field Programmable Gate Array (FPGA), Network on chip (NOC), System on chip (SOC), Very High Speed Integrated Circuit hardware Description language (VHDL).

Introduction

Intellectual property (IP) based networks [1] like the internet is known for their scalability and resilience [2] to partial failures. Such networks are ideal for sending short control messages with less or no call holding times because their reliance are based on datagram [2, 8] forwarding on hop-by-hop [2] technique. A unique address is assigned to each terminal in the network so messages or connections can be routed to the correct recipients. Asynchronous transfer mode (ATM) technology [1, 16] can permit a much higher degree of predictability to be engineered relatively simply, because of their capability of inherent resource partitioning and connection oriented broadband networks [14]. They are thus best suited for transport of streams with high quality of service (QOS) requirements and processing for long call holding times. When communication is taken over untrusted medium [11] or any internet based network, there is the problem of safe and secured data transmission. Cryptography [11] helps in secured data transfer and has been utilized to handle such issues since year.

In cryptography plaintext [10, 11] information is transferred. When information is transferred, encryption is done at transmitting end and same decryption on receiving end as shown in the figure 1.

Encryption [11] is the process of transforming information with the help of an algorithm called a cipher [20] to make it secure unreadable to anyone except those possessing special knowledge, usually called key. The data is encrypted with key, the resultant text is referred as cipher text. The reverse process, on the receiving end i.e., to make the encrypted information readable again, is referred to as decryption.

In Cryptography, key management [15] is the policy which involves encryption algorithms along with an efficient approach of because there are different algorithms that offer different degree of security [7].

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

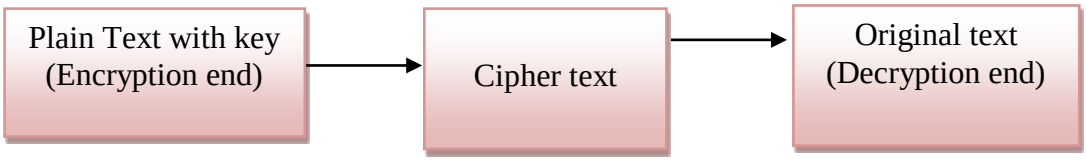

Fig. 1 Encryption and decryption process

Encryption algorithms and techniques are classified according keys size, block size and type of transformation end. In the broad way they can be classified as the symmetric and asymmetric key algorithms. In symmetric algorithm, [11] the sender and receiver both share same key and without key it is not possible to encrypt and decrypt the messages. In asymmetric algorithms require two key values, one for encryption and another for decryption of data. In such a way, it is difficult to decrypt message without the decryption key. Tomas and Rene have reported both algorithms for their application in Third Generation Cellular Networks. Based upon the algorithm The DES (Data Encryption Standard) and triple DES was implemented in compliance with the NIST [15] Data Encryption Standard [11].

Tools Utilized*:* Design and FPGA implementation includes the following software development tools: Project Navigator Application ISE 14.2 of Xilinx Company is a tool to design the IC and to view their RTL (Register Transfer Logic) schematic. Model SimEE 10.1b students edition of Mentor Graphics Company is used for simulation and debugging the functionality. The hardware chip implementation is done using VHDL programming language.

The research work is an advanced work over the research work done by *Prosanta Gope, Ajit Singh, Nikhil Pawha, Ashwani Sharma*, [15] "An Efficient Cryptographic Approach for Secure Policy Based Routing (TACIT Encryption Technique)," 978-1-4244-8679-3/11/©2011 IEEE Page (1-4), The future scope of mentioned paper was hardware implementation of TACIT network security Algorithms, which is the work of our paper**.** The paper is organized as follows, Section I presents the introduction and the tools utilized. Section II discusses the NOC Model for the data transfer. Section III presents the data encryption for TACIT logic and section IV presents data decryption for TACIT logic. Section V presents the synthesis and experimental environment. Section VI describes result and performance evaluation and conclusion is presented in Section VII.

Data encryption algorithm for tacit logic

To implement the TACIT Logic for data communication between two nodes of NOC, the following algorithm has been used. The corresponding VHDL coding has been developed and results

Step 1: Reading of the text file is done and key is applied on the text [10].

Step 2: The ASCII value of each character is read, after reading the character from the text file corresponding to the text [10].

Step 3: The specific value of 'N' bits key is XORed with text. [10].

Step 4: Apply the TACIT logic which is (n^k xor k^k). Here *n* is the value computed from step.

Step 5: Converting the value into binary value [10].

Step 6: reverse operation is performed on the binary string value. [10].

Step 7: The decimal value of corresponding text is found [10].

Step 8: Cipher text is formed due to the unicode character corresponds to decimal value is formed. Formed value is nothing but the cipher text generated [10].

Step 9: The same procedure is continued with each character. Steps from 1 to 7 are carried out, for the next characters of the file until, End of File (EOF) is reached [10].

Data decryption algorithm for tacit logic

The decryption algorithm at the receiving end follows the following steps.

Step 1: First character of the cipher text is read and corresponding decimal value is gotten [10].

Step 2: Evaluating the corresponding binary value then reversing the binary value [10].

Step 3: Inverse of the tacit logic (k^k XOR n^k) is applied to decrypt the encrypted data [10].

Step 4: Performing XOR operation with n bit key value [10].

Step 5: The corresponding characters of the value from step 4 are determined [10].

Step 6: With the help of key value, reshuffling is done [10].

Step 7: For each character decryption, repeating the steps (1 to 6) till the end of the cipher text [10].

4.1 Functional Description

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Input value
a. Inputs: Keyboard Enter data into system.
Clock: This is the clock used for the core operations.
b. Reset: Asynchronous reset, that sets the device to a known state.
Modes of Operation:
a. ASCII Input Mode: The input from the keyboard is considered to be encoded in ASCII.
b. Hexadecimal Mode: The input from the keyboard is considered to be encoded in Hexadecimal, thus the only valid characters are 0-9, A-F.
c. Encryption Mode: Device is configured to receive data and output cipher text.
d. Decryption Mode: Device is configured to receive cipher text and output data.

Experimentation & synthesis environment

The synthesis work is done using FPGA kit Spartan-3E [6]. It is a Micro Blaze Development Kit [6] board having the unique features of the Spartan-3E FPGA family and provides a convenient development board for embedded processing applications. The board highlights these features, Spartan-3E specific features, Parallel NOR Flash configuration [6], Multiboot FPGA configuration from Parallel NOR Flash PROM [6] , Embedded development [18], SPI serial Flash configuration [6] , Micro Blaze 32-bits embedded RISC processor, Pico Blaze 8 bits embedded controller, DDR memory interfaces, 10-100 Ethernet and UART. The experiment set up block diagram is shown in figure 3. Experiment is carried out to validate the data transfer among inlets/outlets using Spartan -3E FPGA. Analog/audio signal of 1 KHz is generated with the help of function generator. The analog signal is converted to digital signal using ADC. The output of ADC module is connected to FPGA kit. The synthesized program is loaded into FPGA and checked for data transfer among two nodes with their addresses in VHDL program. The data transferred can be seen over FPGA kit using LEDs or LCD. The output of FPGA is given to Digital to Analog converter (DAC) and converted signal is displayed by Digital Storage Oscilloscope (DSO). The displayed signal is same of 1 KHz which is attributed to the correct data transfer over the FPGA and validates the simulation results. The simple experiment suggests that the data transfer using TACIT network security algorithm is feasible.

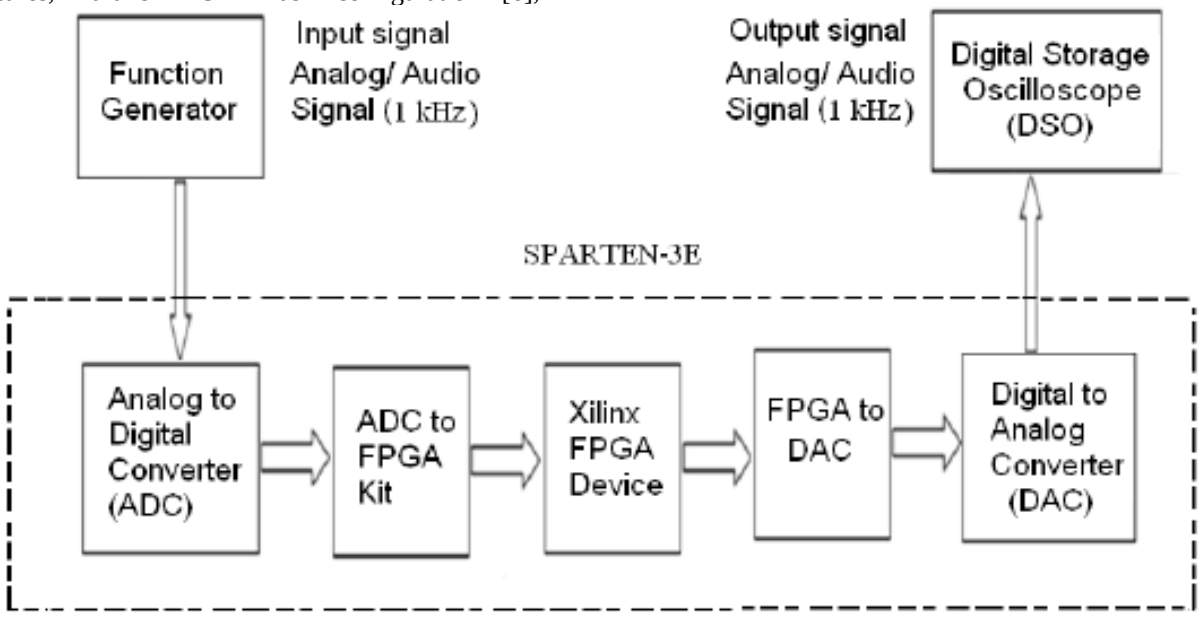

Fig. 3 Experimental set up of testing environment

Results & performance evaluation

Data Encryption

In the screenshots of Figure 4, *text* is the input *text [0-N-1]* which is encrypted in ASCII format for each character. *Key* size is considered as *'N'* bits and each character is encoded with *key* size individually. *Key* is in integer form and *key_value* is *'N'* bits binary value of *Key*. *Xor_value* is the result after XOR of text and *key_value*. *tacit_logic_1* is the value of TACIT operation n^k *tacit_logic_2* is the value of TACIT operation k^k. *tacit_logic* is the actual value of TACIT logic n^k XOR k^k, *reverse value* is the value of reverse operation on *tacit_value*. After it the corresponding decimal value is kept by intermediate signal *decimal_value*. *ciper_text* is the intermediate value of

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

cipher text in encryption logic and *ctext* represents the actual cipher text.

Encryption of plaintext *text [ ]= [N-1 : 0]* by the external *key[ ] = [N-1 : 0]* will produce a cipher text *ctext [] = [N-1 : 0]*. For encryption, *RESET* is set to high at first but low for the rest of time RESET is low. Positive clock pulse *clk* is applied for the synchronization.

Step input 1: reset = 1, positive clock pulse *clk* ia applied and tenn run in Modelsim simulator.
Step input 2: reset = 0, positive clock pulse *clk* , *Key* = input in decimal, *text = [N-1 : 0]* block size and run in modelsim simulator.
Fig. 4 shows the simulation results of data Encryption, for a word length of 'N' bits. The key size is also taken '*N*' bits.

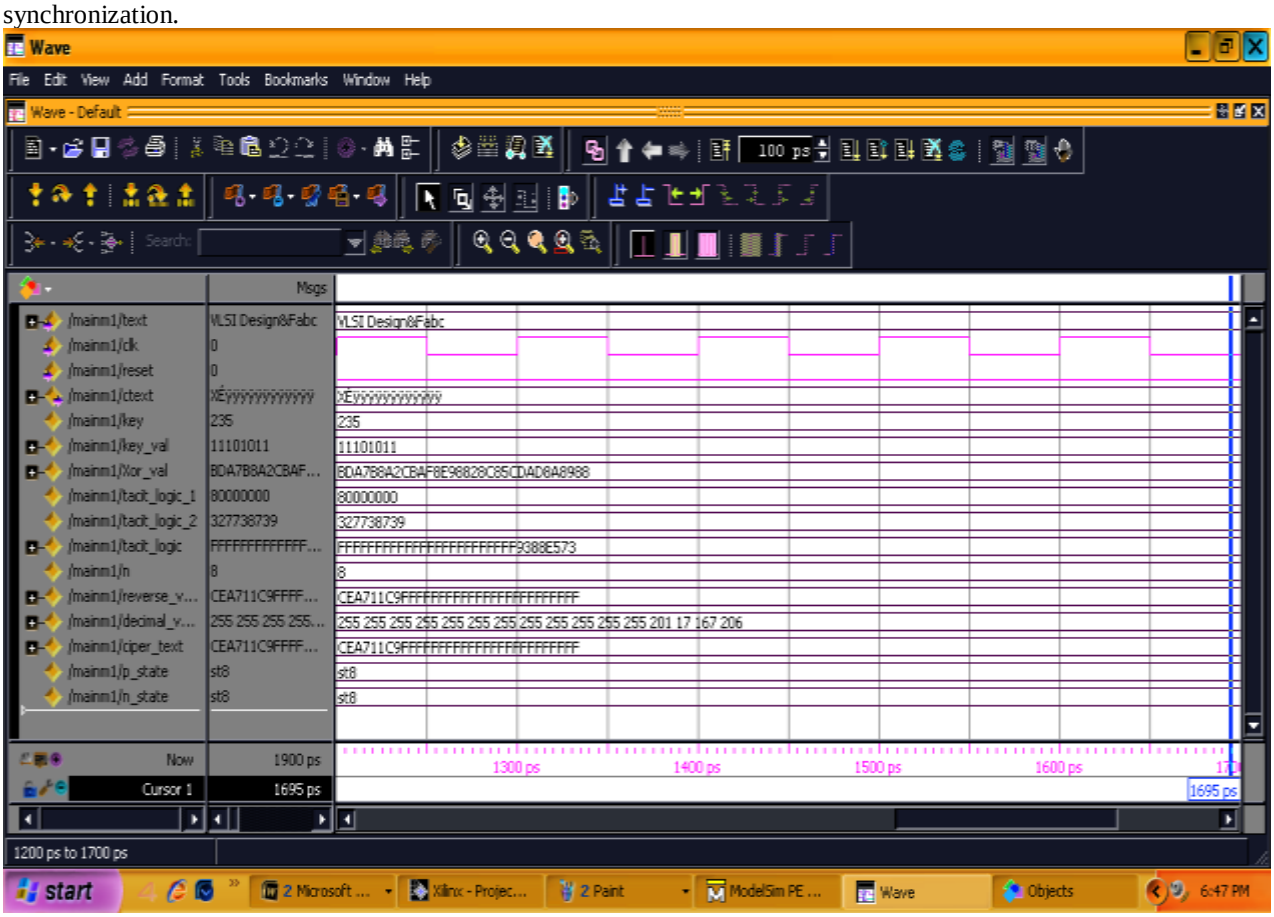

Fig. 4 Modelsim Simulation (Data Encryption of 'N' bits block size

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

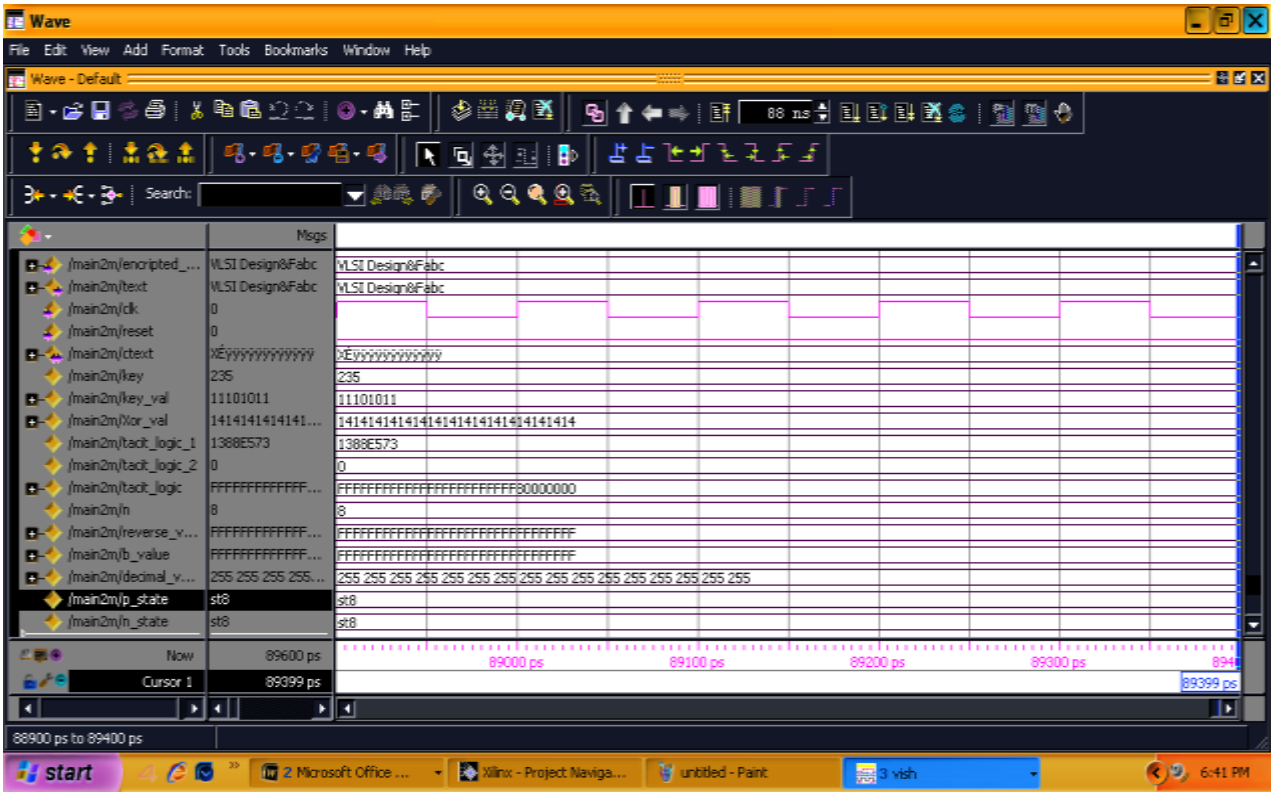

Fig. 5: Modelsim Simulation (Data Decryption for 'N' bits block size)

Data Decryption

In the screen shot of decryption logic shown in Fig.5, *ctext* is the input cipher text to decryption logic *ctext* [0-127] which is decrypted in ASCII format for each character. *Key* is in integer form and *key_value* is *'N'* bit binary value of *Key*. *Xor_value* is the result after XOR of text and *key_value*. *tacit_logic_1* is the value of TACIT operation n^k). *tacit_logic_2* is the value of TACIT operation k^k. tacit_logic is the actual value of TACIT logic n^k XOR k^k, *reverse value* is the value of reverse operation on *tacit_value.* After it the corresponding decimal value is kept by intermediate signal *decimal_value*. *ciper_text* is the intermediate value of cipher text in encryption logic and *ctext* represents the actual cipher text.

The decryption of cipher text *ctext[N-1: 0][F823AB657CED ..1894]* using the same key *k [N-1:0]* produces the original plaintext *text [N-1:0] = [F413579ABCDE2468..]* For decryption, *RESET* is set to high at first but low for the rest of time *RESET* is low. Positive clock pulse *clk* is applied for the synchronization.

Step input 1: reset = 1, positive clock pulse *clk* ia applied and run in Modelsim simulator.
Step input 2: reset = 0, positive clock pulse *clk*, *Key* = input in decimal, *ctext = [N-1: 0]* block size and run in modelsim simulator.

Fig. 5 shows the simulation results of data decryption, for a word length of 128 bits. Data decryption depends on the decoding the exact data and key sharing.

RTL View of chip

The Register Transfer logic (RTL) view for *'N'* bit TACIT encryption and decryption logic is shown in figure 6 and figure 7 respectively. Table 1 lists the pins detail of both algorithms chips .The RTL is extracted from Xilinx ISE 14.2.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Table 1 Pin description for TACIT encryption and decryption logic

Pin	Size	Functional Description
reset	1 bit of std_logic	used for synchronization of the components using clk
clk	1 bit of std_logic	Signal produce to clock signal with any duty cycle
text [N-1:0]	'N' bits of std_logic_vector	Block size of text input for encryption and output at decryption logic.
cipher_ text (N-1 :0)	'N' bits of std_logic_vector	Cipher block text value after decryption and input to decryption logic.
key	N' bits of std_logic_vector,	control signal to protect text at transmitting and receiving end

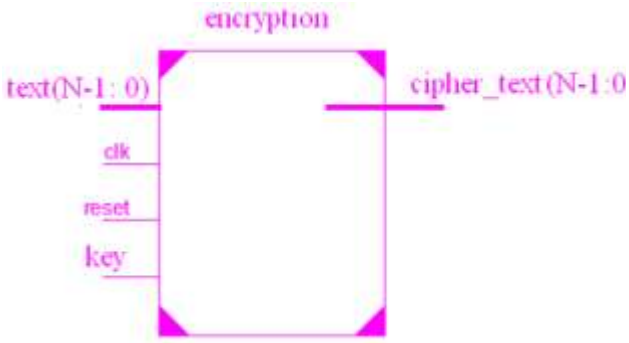

Fig. 6 RTL view of encryption logic

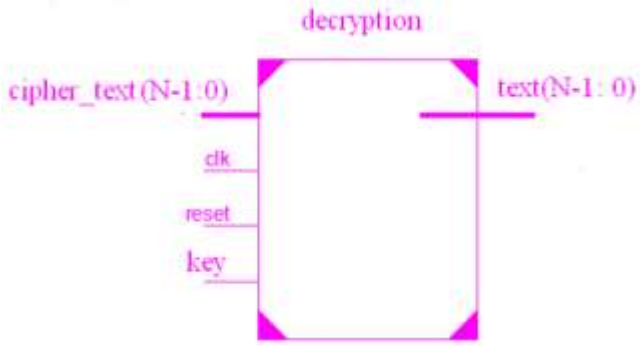

Fig. 7 RTL view of decryption logic

Device Utilization and timing Summary for encryption

Device utilization is the hardware and logic circuitry required to implement the design. It consist of no. of slices, flip flops, gates, combinational logic circuitry and input/output blocks. Selected Device: 2vp2fg256-6, this is the device for targeting FPGA with Speed Grade: - 6. Table 2 shows the summary of device utilization in encryption logic for 8 bits, 64 bits, 128 bits and 'N' bits block size.

Table 2 Device utilization in encryption logic

Device part	Utilization			
	8 bits block size	64 bits block size	128 bits block size	'N' bits block size
Number of Slices	16 out of 1408, 1%	32 out of 1408, 2%	72 out of 1408, 5%	99 out of 1408, 7 %
Number of Slice Flip Flops	19 out of 2816, 0%	55 out of 2816, 1%	124 out of 2816, 4%	169 out of 2816 ,6%
Number of 4 input LUTs	23 out of 2816, 0%	27 out of 2816, 0%	32 out of 2816, 1%	56 out of 2816, 2%
Number of bonded IOBs	9 out of 140, 6%	65 out of 140, 46%	129 out of 140, 92%	129 out of 140, 94%
Number of GCLKs	1 out of 16, 6%	1 out of 16, 6%	1 out of 16, 6%	1 out of 16, 6%

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Device Utilization and timing Summary for decryption

Selected Device: 2vp2fg256-6, this is the device for targeting FPGA with Speed Grade: -6. Table 3 shows the summary of device utilization in encryption logic for 8 bits, 64 bits; 128 bits and *'N'* bits block size.

Table 3 Device utilization in decryption logic

Device part	Utilization			
	8 bits block size	**64 bits block size**	**128 bits block size**	**'N' bits block size**
Number of Slices	10 out of 1408, 0%	43 out of 1408, 3%	79 out of 1408, 5%	98 out of 1408, 7 %
Number of Slice Flip Flops	17 out of 2816, 0%	74 out of 2816, 2%	138 out of 2816, 4%	171 out of 2816, 6%
Number of 4 input LUTs	13 out of 2816, 0%	70 out of 2816, 2%	134 out of 2816, 4%	197 out of 2816, 7%
Number of bonded IOBs	17 out of 140, 12%	129 out of 140, 92%	137 out of 140, 97%	137 out of 140, 98%
Number of GCLKs	1 out of 16, 6%	1 out of 16, 6%	1 out of 16, 6%	1 out of 16, 6%

Timing Summary

Timing details provides the information of delay, minimum period, minimum input arrival time before clock and maximum output required time after clock. Timing summary includes the parameters minimum period, maximum frequency, Minimum input arrival time before clock, Maximum output required time after clock and total memory usage.

Table 4 Timing details for encryption logic

Device part	Utilization			
	8 bits block size	**64 bits block size**	**128 bits block size**	**'N' bits block size**
Minimum period	1.495ns	1.507ns	1.519ns	2.05 ns
Maximum Frequency	638.978 MHz	643.350 MHz	658.328 MHz	665.358 MHz
Minimum input arrival time before clock	2.725ns	2.729 ns	2.748 ns	2.815 ns
Maximum output required time after clock	4.483ns	4.491ns	4. 499 ns	4.503 ns
Total Memory usage	78040 kB	82136 kB	87832 kB	92137 kB

Table 5 Timing details for decryption logic

Device part	Utilization			
	8 bits block size	**64 bits block size**	**128 bits block size**	**'N' bits block size**
Minimum period	1.488ns	1.495 ns	1.510 ns	2.03 ns
Maximum Frequency	629.723 MHz	633.628 MHz	653.628 MHz	665.628 MHz
Minimum input arrival time before clock	2.725ns	2.746ns	2.748ns	2.845 ns

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Maximum output required time after clock	4.413ns	4.433ns	4.449 ns	4.502 ns
Total Memory usage	75992 kB	77016 kB	79064 kB	88187 kB

Conclusion

The hardware chip implementation of TACIT logic for encryption and decryption is implemented in Xilinx 14.2 and functional simulated in Modelsim 10.1 b. In the introduction, it is cleared that the research work is an extended research work over research paper mentioned in reference 15. The Hardware implementation is done sequentially for one character (8 bits), 64 bits, 128 bits and 'N' bits with key size of 'N' bits, and it means sender and receiver both can enjoy with same key. The main advantage of the TACIT logic is that we can choose the block size and key size both. In future, compression techniques could be implemented along with encryption and decryption process.

References

1. Aye Sandar Win "Design and Construction of Microcontroller Based Telephone Exchange System" World Academy of Science, Engineering and Technology, Vol. 46, pp (60-67), 2008.
2. Andreas Hansson, Kees Goossens and Andrei Radulescu "A Unified Approach to Mapping and Routing on a Network-on-Chip for Both Best-Effort and Guaranteed Service Traffic" Hindawi Publishing Corporation VLSI Design Volume 2007, pp (1-16).
3. David Atienzaa, Federico Angiolini, Srinivasan Murali,A ntonioPullini dLucaBeninic, Giovanni De Michelia, "Network-on-Chip design and synthesis outlook" Integration The VLSI Journal Elsevier, Vol. 41 , pp(340-359), 2008.
4. Ganghee Lee, Kiyoung Choi, and Nikil D. Dutt, "Mapping Multi-Domain Applications onto Coarse-Grained Reconfigurable Architectures" IEEE Transaction on Computer Aided Design of Integrated Circuits and Systems, Vol. 30, No. 5, pp (637-650) , May 2011.
5. Hiroaki Morino Thai Thach Bao Nguyen Hoaison Hitoshi Aida Tadao Saito "A Scalable Multistage Packet Switch for Terabit IP Router Based on Deflection Routing and Shortest Path Routing" © 2002 IEEE, pp (2179-2185)
6. Hyung Gye Lee and Naehyuck Chang, Umit Y. Ogras and Radu Marculescu "On-Chip Communication Architecture Exploration: A Quantitative Evaluation of Point-to-Point, Bus, and Network-on-Chip Approaches" ACM Transactions on Design Automation of Electronic Systems, Vol. 12, No. 3, pp (1-20), August 2007.
7. Hao Tian, Ajay K. Katangur, Jiling Zhong Yi Pan "A Novel Multistage Network Architecture with Multicast and Broadcast Capability" The Journal of Supercomputing, Springer, Vol.35, 2006, pp (277–300)
8. Paolo Meloni, Igor Loi, Federico Angiolini, Salvatore Carta,"Area and Power Modeling for Networks-on-Chip with Layout Awareness" Hindawi Publishing Corporation VLSI Design, Volume 2007, pp (1-12)
9. Prosanta Gope, Ashwani Sharma Ajit Singh Nikhil Pahwa "An Efficient Cryptographic Approach for Secure Policy Based Routing (TACIT Encryption Technique)", Conference Proceedings, IEEE Xplorer, (2011), pp (359-363)
10. Thomas Eisen barth, "Cryptography and Cryptanalysis for embedded" Dissertation Ph.D Faculty of Electrical Engineering and Information Technology Ruhr University Bochum, Germany July 2009.
11. Xinmiao Zhang, and Keshab K. Parhi "High-Speed VLSI Architectures for the AES Algorithm", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 12, No. 9, pp (957-968), Sep. 2004.
12. Zhao D, Wang Y "SD-MAC: design and synthesis of a hardware-efficient collision free QoS-aware MAC protocol for wireless Network-on-Chip" IEEE Transactions Computing TC Vol. 8, pp (1046–1057), 2008
13. J. J. Shen, C. W. Lin, and M. S. Hwang," "Security enhancement for the optical strong password authentication protocol ",ACM SIGOPS Operating Systems review ,vol .37 no.2pp 7- 12 2003.
14. E. J. Yoon, E. K. Ryu, and K. Y. Yoo, " A secure user authentication scheme using hash functions" ,ACM SIGOPS Operating Systems review ,vol .38 no. 2 pp 62-68 2004.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

15. E. J. Yoon, E. K. Ryu, and K. Y. Yoo, "Efficient remote user authentication scheme based on generalized elgamal signature scheme", IEEE Trans. Consumer electronic, vol.50 no.2 pp612-614, May 2004.
16. E. J. Yoon, E. K. Ryu, and K. Y. Yoo, "Further improvements of an efficient password based remote user authentication scheme using smart cards",IEEE Trans. Consumer electronic, vol.50,no.2 pp.612-614,May2004.
17. H. M. Sun, "An efficient remote use authentication scheme using smart cards," IEEE Transactions on Consumer Electronics, vol. 46, no. 4, pp. 958–961, 2000.
18. H. M. Sun and H. T. Yeh, "Further cryptanalysis of a password authentication scheme with smart cards," IEICE Transactions and Communications, vol. E86, no. 4, pp. 1412–1415, 2003.
19. H.M. Sun and H.T. Yeh and B.T. Hseih, Security of a remote user authentication scheme using smart cards", IEICE Transactions on communications, vol. E87, No.1, pp192-194, Jan. 2004.
20. H.T. Liaw, S.W. Fan and W. C. Wu, "A Simple Password Authentication using a polynomial", ACM SIGOPS Operating system Review, Vol -38 no 4 pp. 74-79, 2004.
21. H. Y. Chien, J. K. Jan, and Y. M. Tseng, "An efficient and practical solution to remote authentication: smart card", Computers & Security, vol. 21, pp. 372–375, 2002.
22. J. J. Shen, C. W. Lin, and M. S. Hwang, "A modified remote user authentication scheme using smartcards", IEEE Transactions on Consumer Electronics, vol. 49, no. 2, pp. 414–416, 2003.
23. J. K. Jan and Y. Y. Chen, " 'Paramita wisdom' password authentication scheme without verification tables", The Journal of Systems and Software, vol. 42, pp. 45–57, 1998.
24. J. K. Lee, S.R. Ryu and K. Y. Yoo, "Fingerprint –based remote user authentication scheme using smart cards", IEEE Electronic Letters, vol 38 no 12 2002.
25. K. C. Leung, L. M. Cheng, A. S. Fong, and C. K. Chan, "Cryptanalysis of a modified remote user authentication scheme using smart cards," IEEE Transactions on Consumer Electronics, vol. 49, no. 4, pp. 1243–1245 2003.