Indonesian Journal of Electrical Engineering and Computer Science
Vol. 18, No. 2, May 2020, pp. 774~781
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v18.i2.pp774-781

DES encryption and decryption algorithm implementation based on FPGA

Subhi R. M. Zeebaree
Duhok Polytechnic University, Technical College of Informatics, Information Technology Department, Iraq

Article Info

Article history:

Received Sep 7, 2019
Revised Nov 8, 2019
Accepted Nov 22, 2019

Keywords:

Cryptography
DES algorithm
FPGAs
VHDL
Block cipher

ABSTRACT

Nowadays there is a lot of importance given to data security on the internet. The DES is one of the most preferred block cipher encryption/decryption procedures used at present. This paper presents a high throughput reconfigurable hardware implementation of DES Encryption algorithm. This achieved by using a new proposed implementation of the DES algorithm using pipelined concept. The implementation of the proposed design is presented by using Spartan-3E (XC3S500E) family FPGAs and is one of the fastest hardware implementations with much greater security. At a clock frequency of 167.448MHz for encryption and 167.870MHz for decryption, it can encrypt or decrypt data blocks at a rate of 10688Mbps.

Corresponding Author:

Subhi R. M. Zeebaree
Information Technology
Duhok Polytechnic University.
Email: subhi.rafeeq@dpu.edu.krd

1. INTRODUCTION

Within the last decade, there has been a vast increment in the accumulation and communication of digital computer data in both the private and public sectors. Much of this information has a significant value, either directly or indirectly, and requires protection. It is common to find data transmissions, which constitute monetary transfers of billions of dollars daily. Sensitive information concerning individuals, organizations, and corporate entities are collected by Federal agencies in accordance with statutory requirements and is processed in computer systems. This information requires some type of protection, and cryptographic protection may be specified by the authority responsible for the data [1]. For efficient computation process, High performance Field Programmable Gate Array (FPGA) devices can be depended which produce capability of implementing parallel computing via constructing parallel Processing Elements (PEs) called virtual processors. Due to that fact that FPGAs perform as special purpose devices, hence, any system implementing any system grounded on FPGAs provides more rapidly and precise results than those provided based on PCs, even when parallel processing techniques depended for PCs [2]. FPGAs are ideal for the implementation of the cryptographic algorithms. They represent the reconfigurable platform that gives time and cost-effective solutions as compared to ASICs that are expensive and require the largest development time [3]. It provides far above the ground performance than software implementations and can be reconfigured on the fly to store the updated encryption or decryption standard [3]

Many previous researchers like FOZIA HANIF [3], J. P. Kaps and C. Paar [4], M. McLoone and McCanny [5], Nazar A. Saqib K. M. A [6], Abd El-Latif, Hamed, Hasaneen [7], Seddik Bri [8], Soufiane Oukili [9], Noor Najeeb [10], they used FPGA to implemented DES encryption/decryption methods using different architectures. Most problems involving complex computations can be solved by implementing them using FPGAs devices characterized by high speed, high performance compared personal computers.

An FPGA implementation of efficient image encryption algorithm using a chaotic map has been proposed proposed by [11]. The Failed Path Fixes technique proposed by [12] to reduce the timing violation in the FPGA prototyped design. Ooe [13] proposed a system uses an FPGA home hub as its local analytic engine with an IoT platform to store the sensory data. The architecture of the Hybrid Multilayered Perceptron HMLP neural network for implementation on FPGA is proposed by [14]. An automatic car parking system using FPGA based on emergency conditions was proposed by [15] to detect the driver's condition and perform specific tasks. Several Public Key Cryptography (PKC) algorithms based on the perspective of researchers" effort invented in the last four decades addressed by [16]. Cryptographic hashing method using for secure and similarity detection in distributed cloud data been explained by [17]. A first quantum alternative of the scheme Key-Policy Attribute Based Encryption proposed by [18], where the information, the encryption/decryption key, and the attributes are made of qutrits. The concept of distributed Searchable Asymmetric Encryption (SAE) introduced by [19], which is useful for security and can enable search operations on encrypted data. Dynamic crypto processor used for selected symmetric key cryptographic ciphers depended by [20], and provided an implementation of 16bit cryptographic processor that performs logical and arithmetic operations, and key expansion operation on FPGA. The quantitative analysis and comparison of some symmetric key cryptographic ciphers (DES, 3DES, AES, Blowfish, RC5, and RC6) provided by [21]. An efficient protocol produced by [22] that assures the confidentiality of the RFID system by encrypting the messages communicated between tags and readers and the freshness of the messages by using pseudorandom number generator.

In this paper, DES encryption/decryption algorithms were implemented using FPGA devices. The use of these devices greatly reduced the time required to encrypt or decrypt secure information. The rest of this paper is organized as follows. Section 2 mentions DES encryption /decryption algorithms. Section 3 explains organization of the proposed system in details. Section 4 illustrates Implementation Results. Section 5 presents the conclusion.

2. RESEARCH METHOD

This section deals with the depended methodology to illustrate the semaphores of the algorithms relate to the proposed system. In addition, the architectures of the proposed system will be explained.

a. Data Encryption Standard (DES)

The most widely used encryption scheme is based on the Data Encryption Standard (DES) adopted in 1977 by the National Bureau of Standards, now the National Institute of Standards and Technology (NIST), as Federal Information Processing Standard 46 (FIPS PUB 46). For DES, data are encrypted in 64-bit blocks using a 56-bit key. The algorithm transforms 64-bit input in a series of steps into a 64-bit output shown in Figure 1. The same steps, with the same key, are used to reverse the encryption [23].

Figure1. Encryption and Decryption with DES [23]

b. DES Encryption

The encryption process is made of two permutations (P-boxes), which I call initial and final permutations, and sixteen Feistel rounds [24, 25]. The overall scheme for DES encryption is illustrated in Figure 2. As with any encryption scheme, there are two inputs to the encryption function: the plaintext to be encrypted and the key. In this case, the plaintext must be 64 bits in length and the key is 56 bits in length [23].

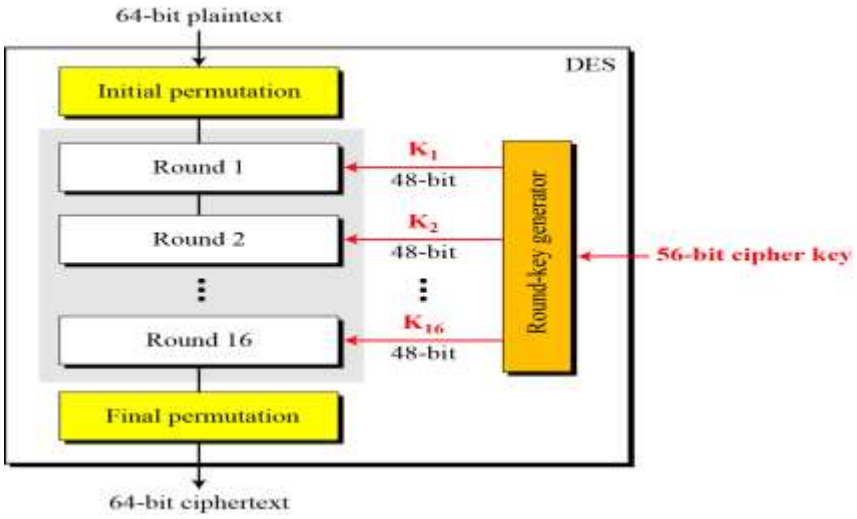


Figure 2. General Structure of DES [24]

c. DES Decryption

As with any Feistel cipher, decryption uses the same algorithm as encryption, except that the application of the subkeys is reversed [23].

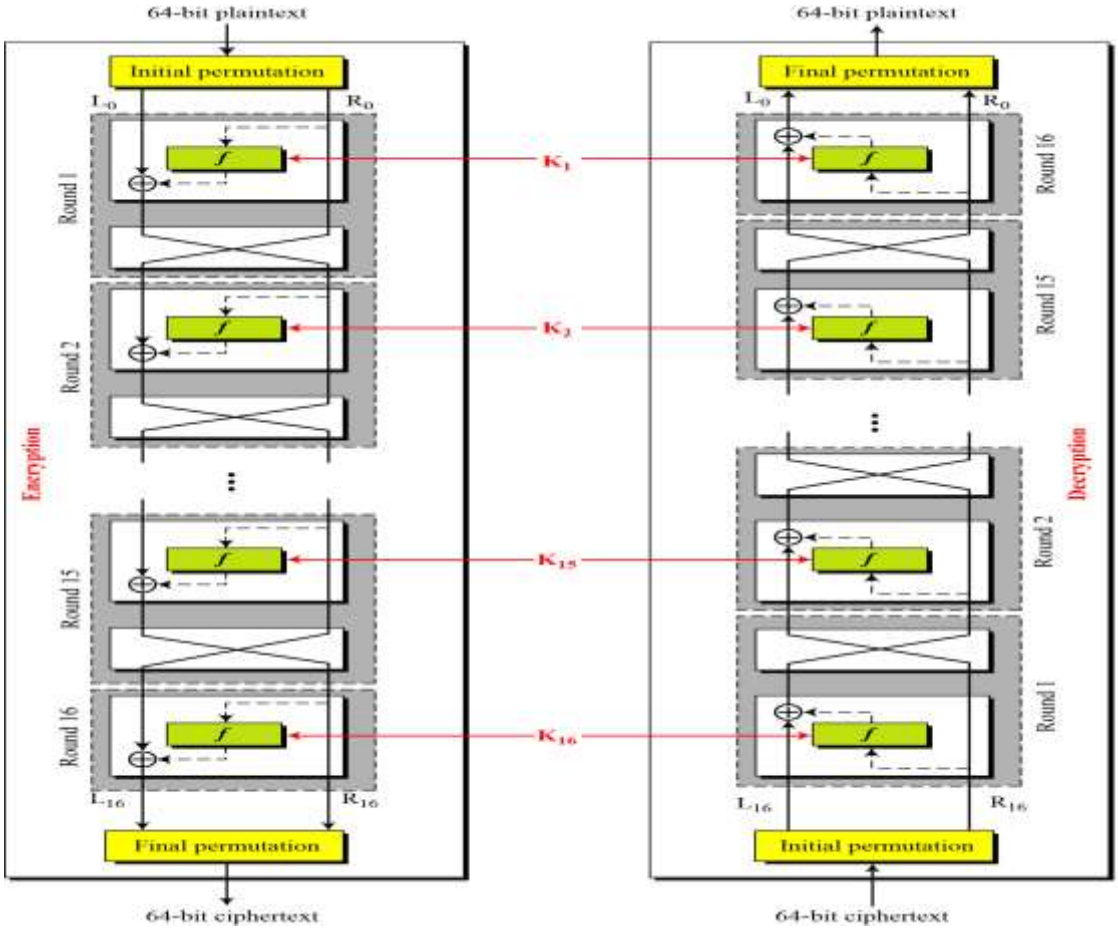


Figure 3. DES Cipher and Reverse Cipher for the first approach [24]

d. Organization of the Proposed Systems

DES encryption and decryption algorithm used as a case study to carry out the code-encrypt/decrypt process using FPGA devices. The goal of using these devices is to greatly reduce the time required to break the code of the above algorithm.

e. Encryption Algorithm Architecture

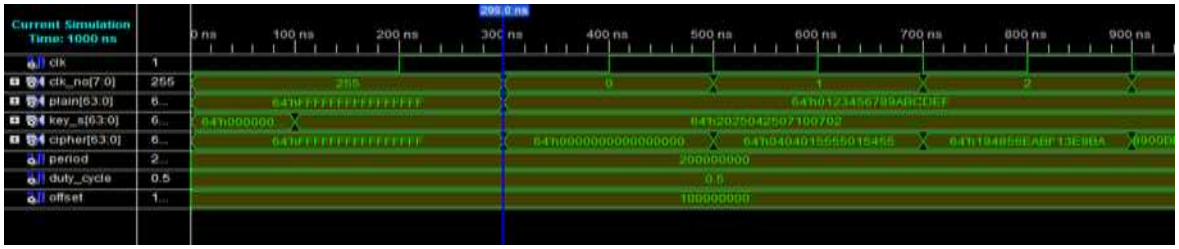
DES encryption algorithm was built within the FPGA device to encrypt any given plain text. The system consists of one FPGA with one encryption algorithm designed as pipeline fashioned consists of sixteen stages, the first stage was used to generate C0, D0 TO C16, D16 and L0, R0.

The maximum frequency can be achieved is (167.44MHz), It means that the maximum time required to encrypt any given plain text using this algorithm is (1/167.44MHz= 5.972 nanosecond).

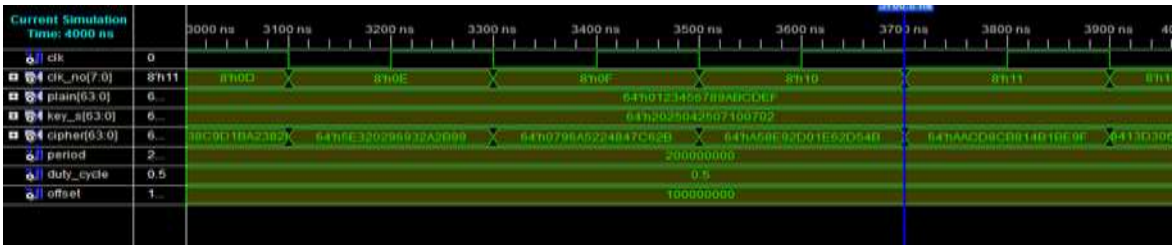
a) Related diagrams forr encryption algorithm

The encryption algorithm block diagram Shown in Figure 4 and the simulation process of encryption algorithm is shown in Figure 5. The timing summery in synthesis report which the maximum frequency shown in Figure 6. Table 1 illustrates the resources for the required design.

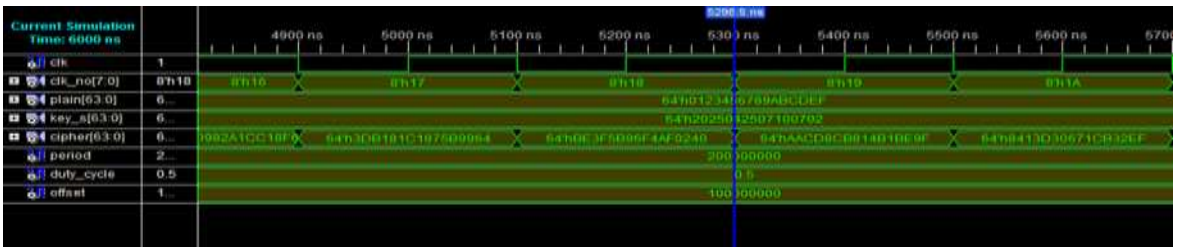
Figure 4. Encryption Algorithm Block Diagram



A: Encryption process at clock 0H



B: Encryption process at clock 11H



C: Encryption process at clock 19H

Figure 5. Simulation Process for Encryption Algorithm

Clk: The clk signal is fed into all 16 stages and plaintext queue and ciphertext queue. It is used to synchronize the events of the system and has a frequency of about (167.44 MHz).
Plain (63 downto 0): These Data lines are fed into the system represented by 64-Bit and are represented by the plaintext encrypted unknown key. The Plain text is saved into 2D array of signals (8 blocksx56 bits) (vectors: (0 to 7) (55 downto 0)).
Cipher (63 downto 0): These are 64-Bit lines is output from last stage (stage 17).
Clk_no (7 downto 0): It is a signal used to display the block number in the simulation process.
key_s (63 downto 0): Through these lines, the key was entered to the system to encrypt plaintext.

When the system was tested the plaintext was 8 blocks, So it needs 17 clock cycles (11H in Hexa) to encrypt the first plaintext and obtain the first ciphertext block, and the second plaintext will be encrypted after clock 18, and the first will be repeated after (17+8) 25 (19H in Hexa) clocks.

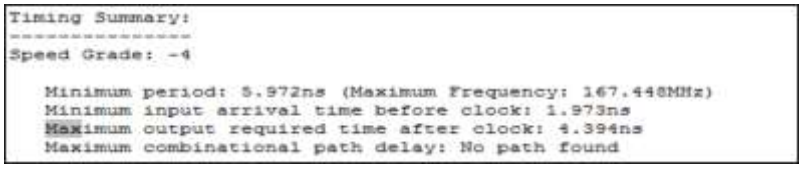


Figure 6. Timing Summary for Encryption Algorithm

Table 1. FPGA Device Utilization Encryption Algorithm

Logic Utilization	Used	Available	Utilization
Number of Slices	2143	14752	14%
Number of Slice Flip Flop	1160	29504	3%
Number of 4 input LUTs	3967	29504	13%
Number of bonded IOBs	193	250	77%
Number of GCLKs	1	24	4%

f. Decryption Algorithm Architecture

The designing of the DES Decryption algorithm was built within the FPGA device to decrypt any given cipher text. The system consists of one FPGA with one decryption algorithm designed as pipeline fashioned consists of sixteen stages; first stage was used to generate C16, D16 TO C0, D0 and L16, R16. This system uses the same algorithm as encryption, except that the application of the subkeys is reversed.

The maximum frequency can be achieved is (167.870MHz), It means that the maximum time required to encrypt any given plain text using this algorithm is (1/167.870MHz= 5.956 nanosecond).

a) Related diagrams forr Decryption algorithm

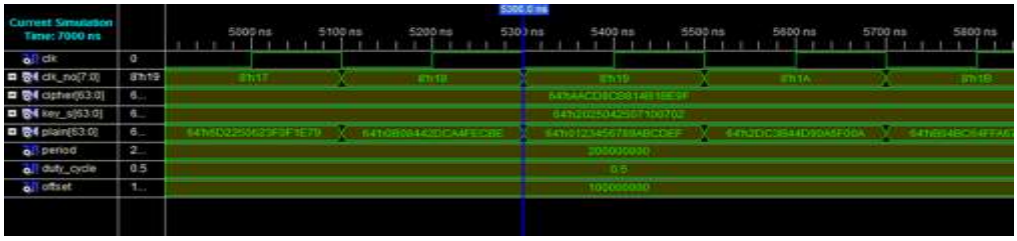
The decryption algorithm block diagram Shown in Figure 7 and the simulation process of encryption algorithm is shown in Figure 8 and the timing summery in synthesis report which the maximum frequency shown in Figure 9, Table 2. illustrates the resources for the required design.

Figure 7. Decryption Algorithm Block Diagram

Clk: The clk signal is fed into all 16 stages and plaintext queue and ciphertext queue. It is used to synchronize the events of the system and has a frequency of about (167.870MHz).
Cipher (63 downto 0): These Data lines are fed into the system represented by 64-Bit and are represented by the ciphertext decrypted by unknown key. The Plain text is saved into 2D array of signals (8 blocksx56 bits) (vectors: (0 to 7) (55 downto 0)).

Plain (63 downto 0): These are 64-Bit lines is output from last stage (stage 17).
Clk_no (7 downto 0): It is a signal used to display the block number in the simulation process.
Key_s (63 downto 0): Through these lines, the key was entered to the system to encrypt plaintext or to decrypt the **ciphertext.**

A: Decryption Process at Clock 0H



B: Decryption Process at Clock 11H

C: Decryption Process at Clock 19H

Figure 8. Simulation Process for Decryption Algorithm

```
Timing Summary:
---------------
Speed Grade: -4

   Minimum period: 5.957ns (Maximum Frequency: 167.870MHz)
   Minimum input arrival time before clock: 1.973ns
   Maximum output required time after clock: 4.965ns
   Maximum combinational path delay: No path found
```

Figure 9. Timing Summary for Decryption Algorithm

Table 2. Fpga Device Utilization Decryption Algorithm

Logic Utilization	Used	Available	Utilization
Number of Slices	2172	14752	14%
Number of Slice Flip Flop	1168	29504	3%
Number of 4 input LUTs	4036	29504	13%
Number of bonded IOBs	193	250	77%
Number of GCLKs	1	24	4%

3. RESULTS AND ANALYSIS

Xilinx System Generator implements the High-Level Language design of DES Encryption and Decryption Algorithm. The design is simulated over System Generator, Xilinx ISE 10.1 and has been implemented over XC3S500E Spartan-3E FPGA.

Table 3. Comparison with other Implementions

Implementation	Device used	CLB slices	frequency (MHz)	Throughput (Mbps)	Design
FOZIA HANIF[3]	XC3S1600E	1344 + 120 BRAMs.	310.174	1240	
J. P. Kaps and C. Paar[4]	XCV4028EX	741	25.18	402.7	16-stage pipeline Designs
M. McLoone and J. McCanny[5]	XCV1000	6446	59.5	3808	16-stage pipeline Designs
Abd El-Latif, Hamed, Hasaneen [7]	XC3S500E	2062	124.734	7983	16-stage pipeline Designs
Seddik Bri[9]	XC3S500E	2046	147.71	9453.47	16-stage pipeline Designs
Soufiane Oukili[8]	XC3S500E	2625	161.03	10305.95	16-stage pipeline Designs
My Implementation	**XC3S500E**	**2143**	**167.870**	**10688**	**16 pipelining**

As its clear from the Table 3, DES Encryption and Decryption algorithms have been implemented on many different platforms and techniques like [3], [4], [5], [7], [9] and [8]. From the results in the table, I found that the proposed 16-stage pipelined design gives 10688 (Mbps) more throughput than the designs of the mentioned references. Finally, from the comparison, I noticed that my implementation is competitive with the reported implementations.

4. Conclusion

However, in terms of frequencies, range in which these devices can work is limited and it is much less than those required for the microprocessors of PCs. In spite of this, they can give results much faster than PCs. Adding to that, FPGA devices are better used in building special purpose systems because these devices have much less cost than PCs with the same number of devices used. Another advantage of FPGAs related to the energy consumed which is much lower than that of PCs.

In this paper, an efficient FPGA implementation of the DES encryption/decryption algorithms based on pipelining concept is presented. The goal of using this concept is to achieve highest possible throughput. The 16-stage pipelining design. At a clock frequency of 167.448 MHz for encryption and 167.870MHz for decryption, the 16 pipelining design can encrypt/decrypt data block at a rate of 10688Mbps. The proposed implementation has been compared with other recent hardware implementations. The comparison has indicated that highest throughput can be achieved by the proposed FPGA implementation.

Finally, designing any system by the use of the (FPGA) devices affects the allowed frequencies. If a higher frequency is chosen than that which is permitted by the (Synthesizer), it will cause a system failure and raise the FPGA device temperature and make it out of the service. So, the system designer must choose the frequency which will be used as a clock generator of a frequency slightly less than that allowed by the (Synthesizer). In the future, it can be breaking other breakable encryption algorithms that more complex than the system designed for more benefits.

ACKNOWLEDGEMENTS

Full thanks expressed to Duhok Polytechnic University (DPU). Great thanks to Dr. Riyadh Zaghlol, Dr. Amira Bibo, and Diyar Qader Zeebaree.

REFERENCES

[1] K. V. R. Kumar, "Analysis & Implementation of DES using FPGA Symmetric Encryption", International Journal of Engineering Technology, Management and Applied Sciences, vol. 4, no. 1, pp. 14–22, 2016.
[2] S. R. M. Zeebaree, A. B. Sallow, B. K. Hussan, S. M. Ali, "Design and Simulation of High-Speed Parallel/Sequential Simplified DES Code Breaking Based on FPGA", 2019 International Conference on Advanced Science and Engineering (ICOASE), IEEE Xplore, pp. 76-81, 2019.
[3] F. H. Khan, R. Shams, A. Hasan, and N. Hasan, "Implementation of Data Encryption Standard (DES) on FPGA", *J. Comput. Sci. Newports Inst. Commun. Econ.*, vol. 5, no. April, pp. 47–59, 2016.
[4] A. J. Elbirt and C. Paar, "An FPGA implementation and performance evaluation of the Serpent block cipher", IEEE Computer Society Digital Library, vol. 1, pp. 33–40, 2000.
[5] M. M. and J. V. McCanny, "New wideband/dualband CMOS LC voltage- controlled oscillator", *Comput. Eng.*, vol. 152, no. 3, pp. 189–209, 2005.
[6] N. A. Saqib, F. R. Henriquez, A. D. P´erez, "A compact and efficient fpga implementation of the DES algorithm" *International Conference on Reconfigurable Computing and FPGAs ReConFig'04, Colima, Mexico,* 2004.

[7] E. A. M. H. K. M. A. Abd El-Latif, H. F. A. Hamed, E. A. M. Hasaneen, "FPGA Implementation of the Pipelined Data Encryption Standard (DES) Based on Variable Time Data Permutation", The Online Journal on Electronics and Electrical Engineering (OJEEE) vol. 2, no. 3, pp. 298–302, 2010.
[8] S. B. S. Oukili, "High throughput FPGA implementation of data encryption standard with time variable sub-keys", *Int. J. Electr. Comput. Eng.*, vol. 6, no. 1, pp. 298–306, 2016.
[9] S. Oukili and S. Bri, "FPGA implementation of Data Encryption Standard using time variable permutations" *Proc. Int. Conf. Microelectron. ICM*, vol. 2016–March, pp. 126–129, 2016.
[10] N. N. Qaqos, "Efficient Hardware Implementation of the Pipelined DES Encryption Algorithm Using FPGA", Al-Rafidain Engineering vol. 2 no. 5, pp. 63–74, Dec. 2014.
[11] H. A. Abdullah and H. N. Abdullah, "FPGA implementation of color image encryption using a new chaotic map", IJEECS, Vol. 13, No. 1, pp. 129-137, January 2019.
[12] S. Savugathali, M. Mustapa, M. S. Razali, F. F. Zakaria, "Timing violation reduction in the fpga prototyped design using failed path fixes and time borrowing techniques", IJEECS, Vol. 14, No. 2, pp. 628-636, May 2019.
[13] L. Y. Ann, P. Ehkan, M.Y. Mashor, S. M. Sharun, "FPGA-based architecture of hybrid multilayered perceptron neural network", IJEECS, Vol. 14, No. 2, pp. 646-652, May 2019.
[14] C. Ooi, W. Tan, S. Cheong, Y. Lee, V. M. Baskaran, Y. Low, "FPGA-based embedded architecture for IoT home automation application", IJEECS, Vol. 14, No. 2, pp. 949-956, May 2019.
[15] K. J. Yong, M. H. Salih, "Design and implementation of embedded auto car parking system using FPGA for emergency conditions", IJEECS, Vol. 13, No. 3, pp. 876-883, March 2019.
[16] J. I. Ahmad, R. Din, M. Ahmad, "Analysis Review on Public Key Cryptography Algorithms", IJEECS, Vol. 12, No. 2, pp. 447-454, November 2018.
[17] A. M. D. Masood, S. K. Muthusundar, "Cryptographic Hashing Method using for Secure and Similarity Detection in Distributed Cloud Data", IJEECS, Vol. 9, No. 1, pp. 107-110, January 2018.
[18] G. Mogos, "Quantum Key-Policy Attribute-Based Encryption", IJEECS, Vol. 7, No. 2, pp. 542-550, August 2017.
[19] S. Yin, L. Teng, J. Liu, "Distributed Searchable Asymmetric Encryption", IJEECS, Vol. 4, No. 3, pp. 684-694, December 2016.
[20] M. Saad, K. Youssef, M. Tarek, H. Abdel-Kader, "Architecture of ASIP Crypto-Processor for Dynamic Runtime Security Applications", IJEECS, Vol. 4, No. 2, pp. 412-423, Noveber 2016.
[21] M. Saad, K. Youssef, M. Tarek, H. Abdel-Kader, "Quantitative Analysis and Comparison of Symmetric Cryptographic Security Algorithms", IJEECS, Vol. 4, No. 1, pp. 116-124, October 2016.
[22] Z. Shi, Y. Xia, C. Yu, "Strong RFID Mutual Authentication Protocol Based on a Lightweight Public-key Cryptosystem", IJEECS, Vol. 12, No. 3, pp. 2320-2326, March 2014.
[23] W. Stallings, "Cryptography and Network Security Principles and Practice", Pearson India; 7th edition, October 16, 2018.
[24] B. Forouzan, "Cryptography and Network Security", McGraw-Hill, 2008.
[25] S. Kumar and A. Swamy, "FPGA Implementation of a Nakagami-m fading channel Simulator using Random Number Generator", *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 1, , pp. 133 - 140October 2016.

BIOGRAPHIES OF AUTHOR

Dr. Subhi Rafeeq Mohammed Zeebaree is an Assistant Professor in Computer Engineering. He is the Director of Culture and Student Activities Center at Duhok Polytechnic University (DPU). He got his BSc, MSc and PhD Degrees from University of Technology-Baghdad-Iraq at 1990, 1995 and 2006 respectively. He is an Assistant Professor since 2012. He started teaching and supervising post-graduate courses and students (PhD and MSc) since 2007 and until now. Twenty-five of his PhD and MSc students completed their studding and got their degrees. Now there are number of PhD and MSc students under his supervising inside and outside of Iraq. He has joint PhD supervisions with UTM (Malaysia), Firat University (Turkey), and EMU (Cyprus). He was the Chairman of Scientific and Research Advices Committee of DPU for five years. He is a member of IEEE Iraq-section. He was the chair of two international conference sponsored by IEEE institution (ICOASE2018 and ICOASE2019). His Official Email: subhi.rafeeq@dpu.edu.krd .