Efficient Hardware Implementation of the Pipelined DES Encryption Algorithm Using FPGA

Noor Najeeb Qaqos
E_mail:Noor_najeeb_2006@yahoo.com
Technical Institute of Shekhan – University of Polytechnic, Duhok

Abstract

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 superpipelined concept. DES are simulated using Xilinx 9.2i software with the use of VHDL as the hardware description language and implemented using Spartan-3E FPGA kit. The DES Encryption algorithm achieved a high throughput of 18.327 Gbps and 3235 number of Configurable Logic Blocks (CLBs), obtaining the fastest hardware implementation with better area utilization. Comparison is made between the proposed implementation and other recent implementations. The comparison results indicate that a high throughput with optimized resource utilizations can be achieved using a superpipelined concept on the proposed design in a single FPGA chip.

Keywords: ***DES Encryption Algorithm, FPGA, Superpipelining Concept, Spartan 3E-Kit, VHDL, Xilinx ISE 9.2i.***

تنفيذ مادي كفوء بأسلوب خطوط الأنابيب لخوارزمية التشفير DES باستخدام FPGA

نور نجيب قاقوس
قسم تقنية المعلومات – المعهد التقني/ شيخان - جامعة دهوك التقنية

الملخص

يقدم هذا البحث تنفيذاً مادياً قابلاً لإعادة التهيئة ذو كفاءة عالية لخوارزمية DES باقتراح تنفيذ جديد لهذه الخوارزمية باستخدام أسلوب خطوط الأنابيب الفائقة السرعة. استخدم برنامج Xilinx 9.2i بالاعتماد على لغة وصف الكيان المادي VHDL القابلة للتنفيذ على FPGA Chip Spartan-3E Kit لمحاكاة الخوارزمية المقترحة. أظهرت الخوارزمية الكفاءة العالية بمقدار 18.327 Gbps وباستغلال (3235 CLBs) فقط من حجم رقاقة FPGA المستخدمة وسرعة تنفيذ مع استغلال جيد للمصادر. بينت مقارنة النتائج بين التنفيذ المقترح مع النتائج الخاصة ببناءات أخرى مقدمة موخرًا كفاءة عالية للنموذج المقترح واستغلال المصادر بشكل مثالي باستخدام أسلوب خطوط الأنابيب الفائق السرعة المقترح والمنفذ على رقاقة FPGA مفردة.

Received: 3 – 11 - 2013 **Accepted: 5 – 1 - 2014**

1- Introduction

Data Encryption Standard (DES) is the most well-known cryptographic mechanism in history [1]. It begins with the work of Feistel at IBM in the early 1970s and culminating in 1977 with the adoption as a U.S. Federal Information Processing Standard for encrypting unclassified information. A fast new DES implementation using software proposed by E. Biham [2]. K. Wong, *et al.* [3] described a single-chip implementation of the data encryption standard (DES) using Xilinx XC4000 series field programmable gate array technology under the XACT step design flow integration system. A fast FPGA implementations of DES are designed, implemented and compared various DES architectures by J. Kaps, *et al.* [4]. ASIC implementation of DES algorithm for network encryption at 10 Gbps and beyond proposed by Wilcox, *et al.* [5]. C. Patterson [6] described a high performance DES encryption in Virtex FPGAs using Jbits. A novel method for implementing the key schedule of DES for high performance on FPGA proposed by M. MaLoone, *et al.* [7]. Efficient and compact FPGA implementation of DES algorithm presented by N. A. Sadiq [8]. V. Patel [9] presented a high performance reconfigurable hardware implementation of the DES algorithm achieved by combining pipelining concept with novel skew core key scheduling method. A DES algorithm implemented on FPGA using pipelined concept based on variable time data permutation described by K. M. A. Abd El-Latif, *et al.* [10]. Another research described by K. M. A. Abd El-Latif, *et al*. [11] for hardware implementation of DES using pipelining concept with time-variable key. An optimized DES Encryption algorithm implemented on FPGA Spartan-3E described by A. Singh, *et al*. [12]. Security enhancement of pipelined DES algorithm implemented by U. R. Kumari [13]. FPGA implementation of DES algorithm based on real time data security applications presented by S. Manikonda [14].The Data Encryption Standard (DES) is a block cipher which means that during the encryption process, the plain-text is broken into fixed length blocks in 64 bits and each block is encrypted at the same time by using 56 bits key. This paper describes the hardware implementation of 16- stage DES. Each stage is divides into many stages and super pipelined resulting in 119 stage DES algorithm. It allows 119 data blocks to be processed simultaneously resulting in an impressive gain in speed. It also supports the use of different keys every clock cycle, thus improving overall security since users are not restricted to using the same key during any one session of data transfer. The design of DES algorithm implemented on Xilinx Spartan-3E FPGA technology [15] due to a reconfigurable hardware offers high speed similar to VLSI and high flexibility similar to software and was coded using VHDL language [16].

This paper is organized as follows: Besides this introductory section, Section 2 describes the DES algorithm briefly, a new proposed implementation of the DES Encryption algorithm using superpipelining concept is described in section 3. FPGA implementation summary and results are presented in section 4. Section 5 compares the achieved results with other previous DES implementations. Finally, conclusions are drawn based on my results.

2- DES Algorithm

The DES algorithm is a block cipher used to encrypt/decrypt 64 bits input plain text data with cipher key of 64 bits as shown in Fig. 1. DES consists of 16 rounds, the input data firstly permutated and then split into two half, right and left half each of them 32 bits in length. The 32 bit right half will be become the left half for the next round while the 32 bits left half is sent to *f*-function block. The output of the function (*f*) is X-ored with left half to produce 32 bits right half for the next round. The process is repeated for 16 round of DES algorithm.

After 16 rounds, the output is sent to final permutation which is an inverse to initial permutation to produce 64 bits cipher text [17].

2-1 Key Generation

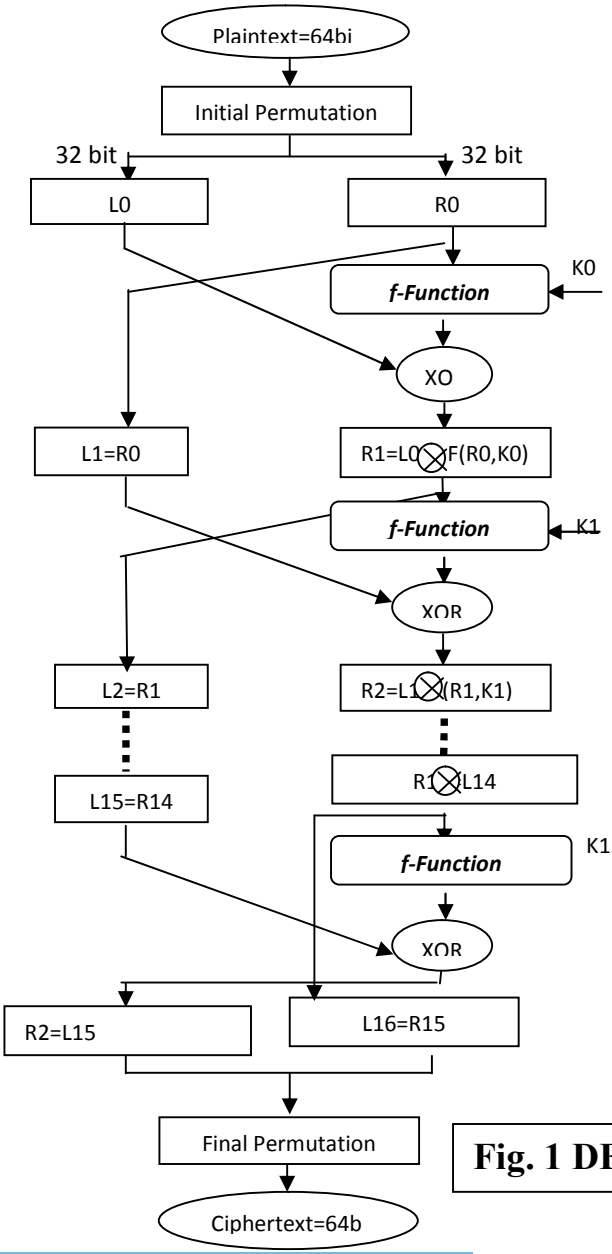
The four operations used to generate 48 bits sub-key for 16 rounds are shown in Fig. 2. Although the input key used is 64 bits, but the actual key used is only 56 bits. Steps to generate the sub-key described as follows [17]:-

1- the initial 64-bit key is inserted, a parity drop occurs in which every 8th bit of the key is used only for parity check and so its final size is reduced to 56-bits.

2- the key splits into two equal halves of 28 bits lengths of data.

3- a left rotate operation, one or two bits depending on the number of round is applied to the left and right half. One rotated bits for rounds 1,2,9,16 and two rotated bits for other rounds.

4- a final compression P-box occurs on the data after rotation to produce 48 bits sub-key.



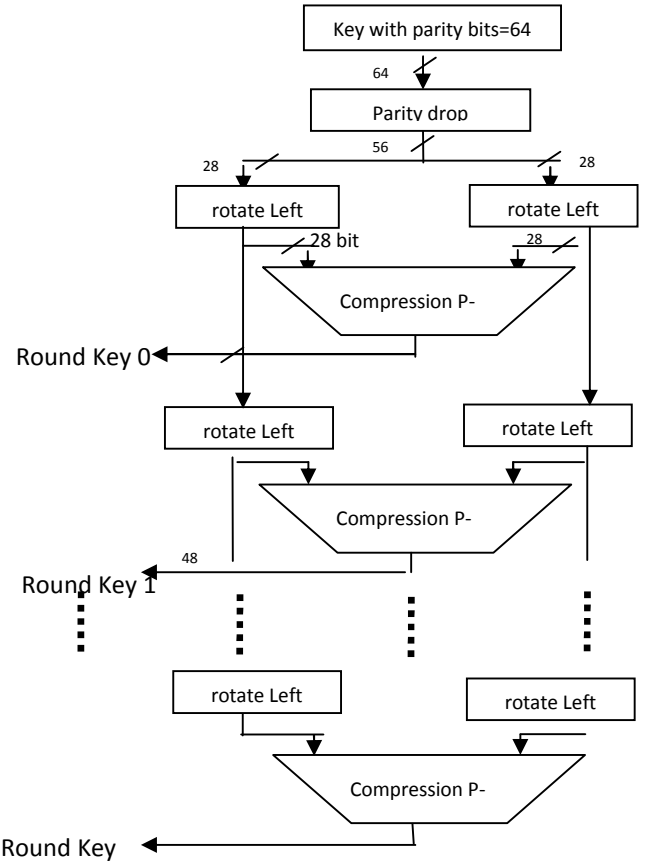


Fig. 2 Key Generation

Fig. 1 DES Encryption Algorithm

2-2 *f*-Function block

The operations of *f*-function block are shown in Fig. 3. The expansion of 32 bits right half to 48 bits to be processed through XOR function with the round key. The output of XOR operation is sent to eight substitutions box to convert 48-bits input to 32- bits output. A 32 bits output from S-boxes is fed to straight permutation as a final operation [17].

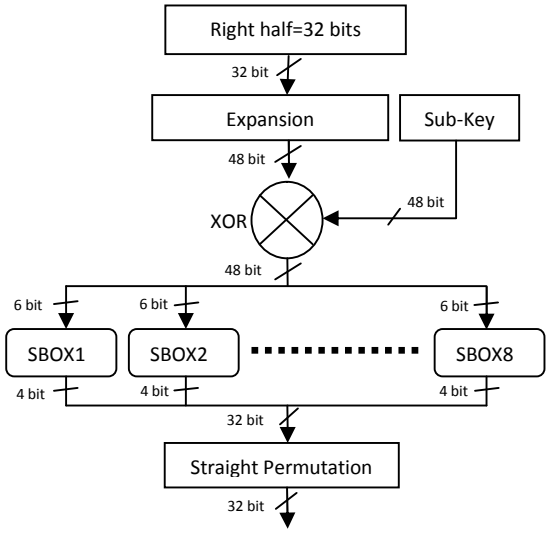


Fig. 3 *f* – Function

3- A New Proposed Hardware Implementation

This paper proposed a new hardware implementation of 16 rounds (stages) DES algorithm based on superpipelining concept. Instead of using 16 pipelined stages to implement the DES algorithm, super pipelining concept is used by splitting each stage into more stages where each stage is simpler (does less work) and thus the clock speed can be increased, meaning that increasing in the throughput of the proposed design, but the *latency*, measured in clock cycles, for any instruction to complete has increased. The proposed hardware implementation of DES algorithm for one round using super pipelining concept is shown in Fig.4. It can be seen that the one round of DES algorithm splits to 7 stages. This means that seven clock cycles are needed to implement one round using super pipelining concept.

The proposed hardware implementation of 16 round DES algorithm is shown in Fig. 5, For the first stage it can be seen that:-

1. the Sub-Key1 and the output of Expansion P-box are ready at stage 6, stage4, respectively. A synchronization process must be done between them. A delay with 2 clock cycle is done to the output of Expansion P-box so that these two output are ready at the same time to be processed by XOR-operation of *f*-Function block .
2. the left half and the output of straight P-box are ready at stage3, stage11, respectively. A synchronization process must be done between them. A delay with 8 clock cycle is done to the output of straight P-box so that these two output are ready at the same time to be processed by XOR-operation .
3. the right half and the output of XOR-operation are ready at stage3, stage12, respectively. A synchronization process must be done between them. A delay with 9 clock cycle is done to the output of right half so that these two output are ready at the same time to be processed as an input to the second round .

it can be seen from the first stage that the number of clock cycle used to obtain the output is 12 clock cycle (1 clock cycle for permutation block and 11 clock cycles for stage 1) .
For the second stage it can be seen that:-

1. the Sub-Key2 and the output of Expansion P-box are ready at stage 7, stage 13, respectively. A synchronization process must be done between them. A delay with 6 clock cycle is done to the output of Sub-Key2 so that these two output are ready at the same time to be processed by XOR-operation of *f*-function.
2. the left half and the output of straight P-box are ready at stage12, stage18, respectively. A synchronization process must be done between them. A delay with 6 clock cycle is done to the output of straight P-box so that these two output are ready at the same time to be processed by XOR-operation .
3. the right half and the output of XOR-operation are ready at stage12, stage19, respectively. A synchronization process must be done between them. A delay with 7 clock cycle is done to the output of right half so that these two output are ready at the same time to be processed as an input to the third round .

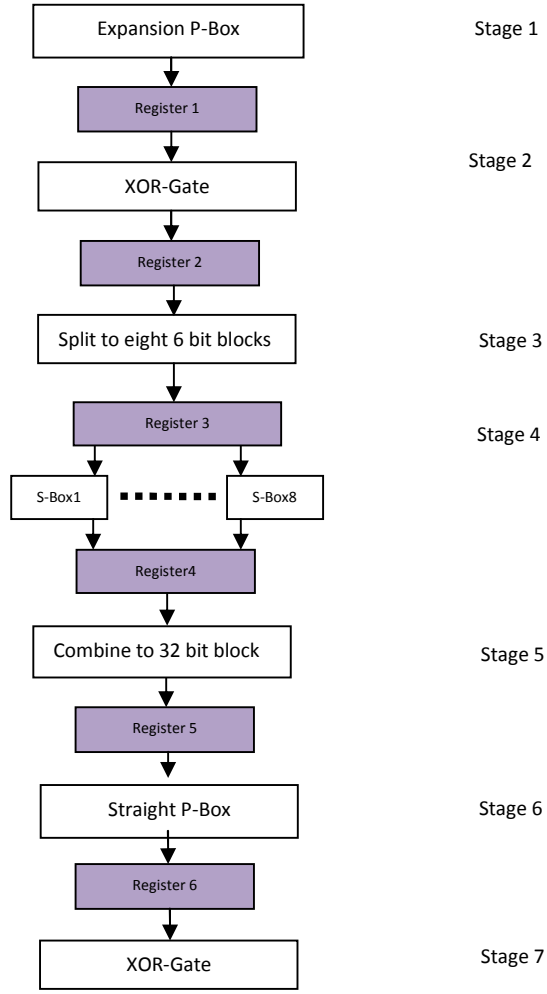
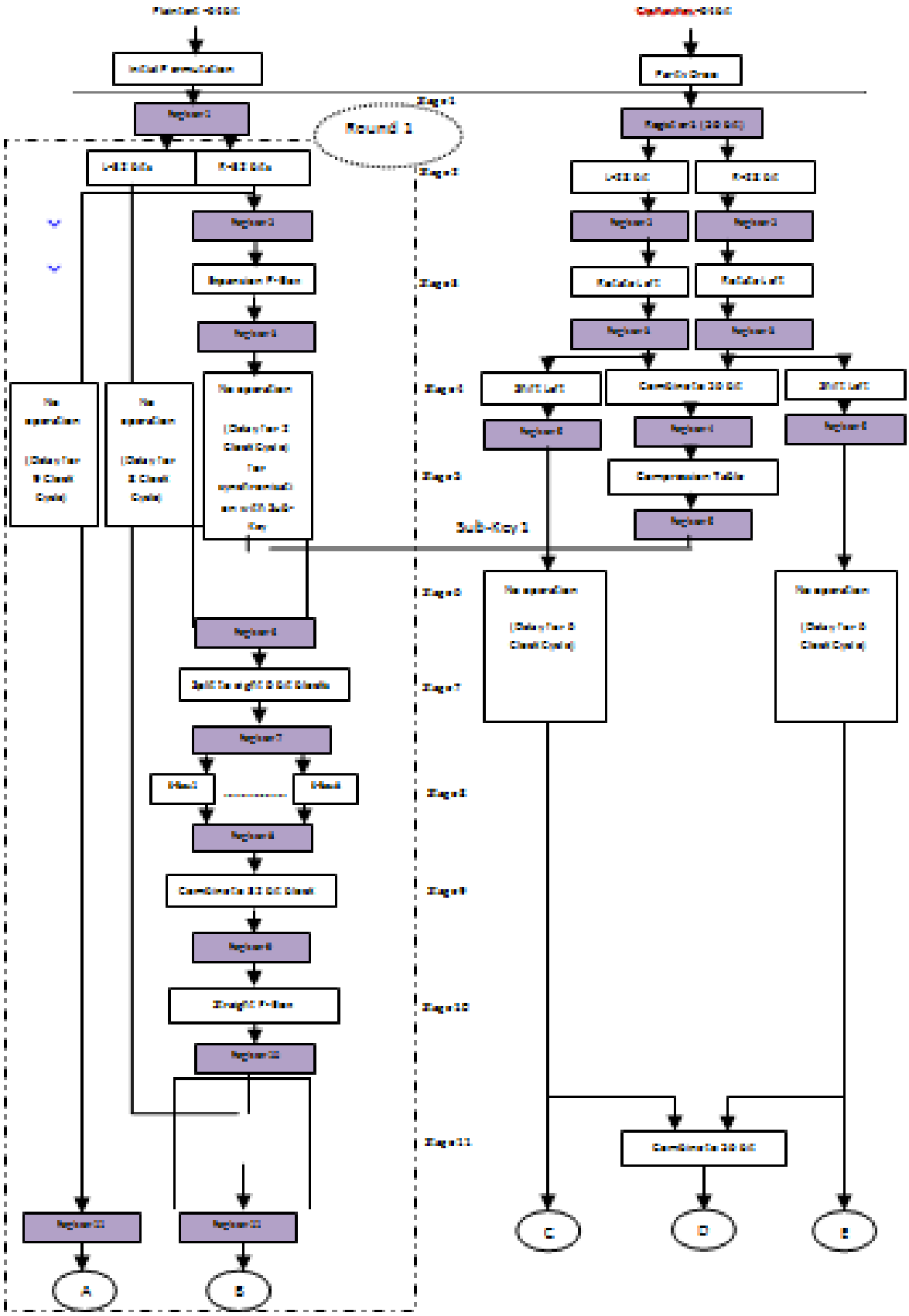


Fig. 4 A proposed implementation of one round DES using super pipelining concept

due to a symmetric of stages (1,2,3….,15,16), the same process of stage 2 is repeated and implemented to stages (3-16), which means that the number of clock cycles used to implement stages (2,3……,15,16) equal to (15×7=105 clock cycles). The final steps are the combination of left and right side together, final permutation and the output (ciphertext)which required 3 clock cycles. The total number of cycles used to implement the complete design equal to 11+105+3=119 clock cycles.



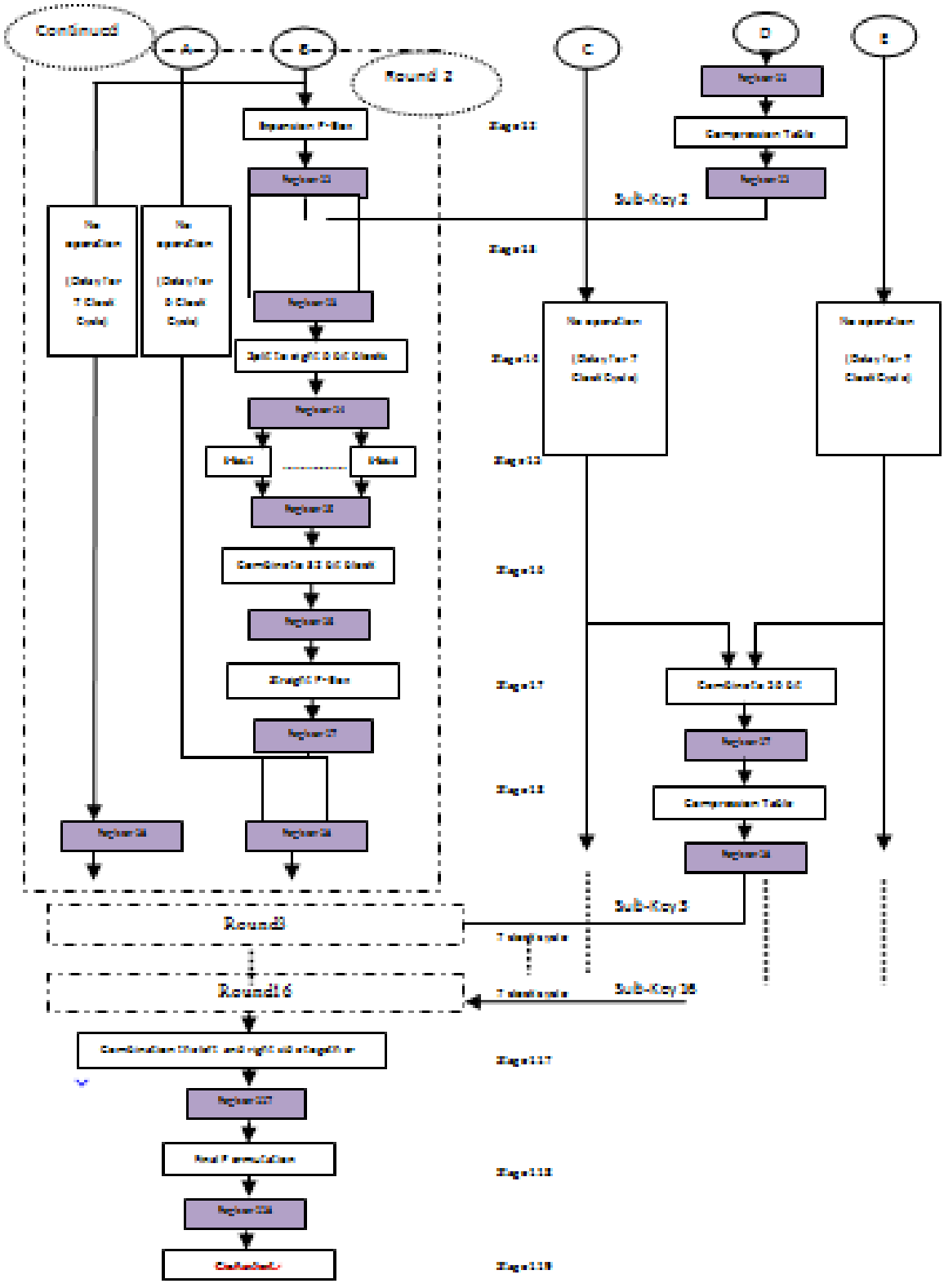


Fig. 5 Hardware implementation of 16 round DES algorithm using super pipelining concept

4- Input-Output Management of the Proposed Implementation

The plaintexts and cipherkeys are stored in an array inside the FPGA as the inputs plaintext 0-3 and cipherkeys 0-3 to the two 4× 1 Multiplixer. By using the existed four switches (Sw0-Sw3) in SPARTAN-3E kit, the selection of desired plaintextand cipherkey using (sw0,sw1)and (sw2,sw3), respectively can be treated as an inputs to the DES algorithm implemented on FPGA chip. The ciphertexts are displayed using Liquid Crystal Display (LCD) screen in SPARTAN-3E kit. The hardware implementation of input-output unit is shown in Fig. 6.

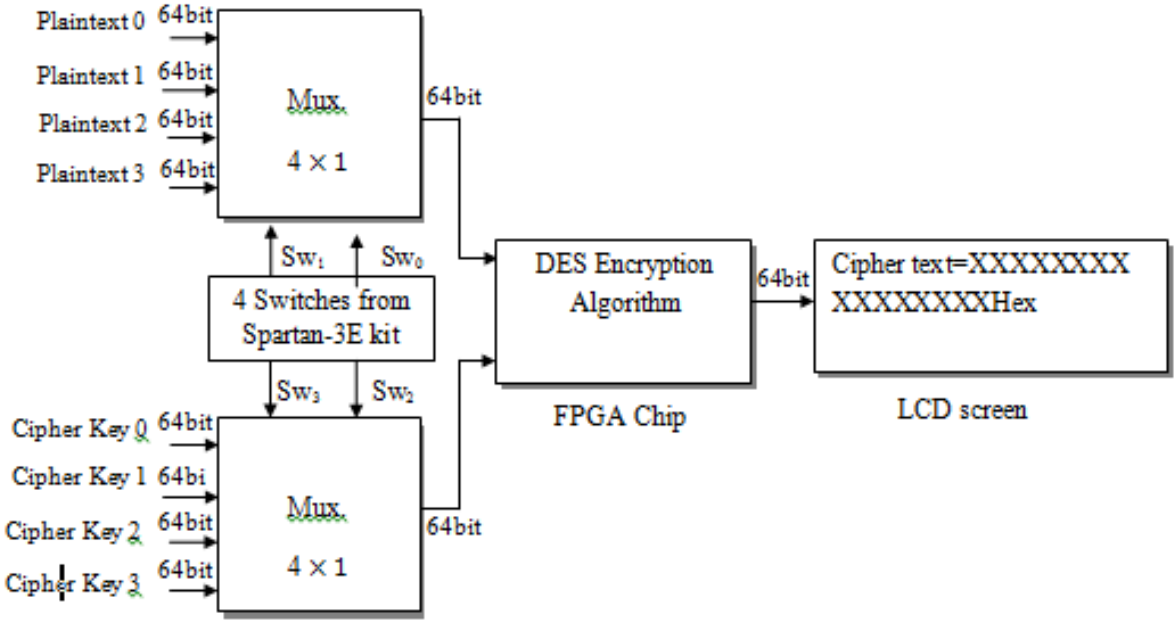


Fig. 6 A hardware implementation for input/output unit for the DES Encryption algorithm.

5- Implementation Summary and Results

Efficient hardware implementation of the DES Encryption algorithm was accomplished on Spartan 3E family FPGA chip XC3s500e-5fg320. The design was simulated by Xilinx 9.2i and coded using VHDL language. The utilization resources and the maximum frequency used for implementing the DES algorithm are shown in Table1. The design operates at a frequency 286.369 MHz and it takes 119 clock cycle latency first time only and then encrypts on data block per clock cycle. Therefore, the achieved throughput is (64×286.369)=18327 Mbps. Fig.7,8 show the timing diagram and full schematic of the implementation of DES Encryption algorithm. Table 2 shows the plaintexts, keys and its corresponding cipher texts. It can be seen from the Fig. 8 that the inputs are Plaintext[0:63] , CipherKey[0:63] , rst and clk and the output is Ciphertext[0:63].The inputs and the corresponding outputs after 119 clock cycle are shown in Table 2.

Table1: Utilization resources and maximum frequency for the proposed implementation

Resources or Frequency	*exploited*	*Total*	*Utilization ratio*
Number of Slices	3236	4656	69%
Number of Slice Flip Flops	3712	9312	39%
Number of 4 input LUTs	5904	9312	63%
Number of bounded IOBs	186	190	97%
Number of GCLKs	1	24	4%
Maximum Frequency	286.369MHz		

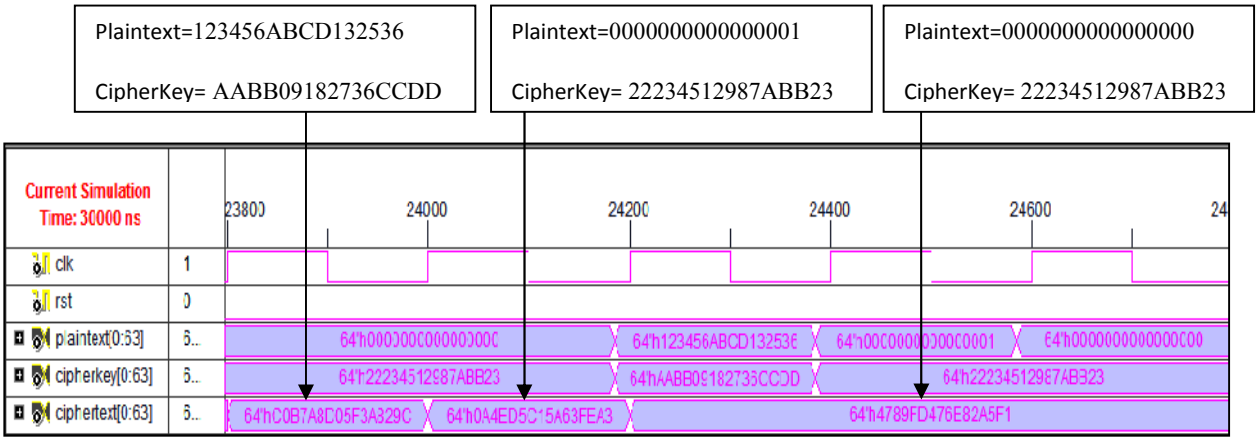


Fig.7: Simulation timing diagram of the proposed DES Encryption algorithm.

Table 2: Some of Plaintext, Cipherkeys and its Ciphertext.

Plaintext	Cipher Key	Cipher text
123456ABCD132536	AABB09182736CCDD	C0B7A8D05F3A829C
0000000000000000	22234512987ABB23	4789FD476E82A5F1
0000000000000001	22234512987ABB23	0A4ED5C15A63FEA3

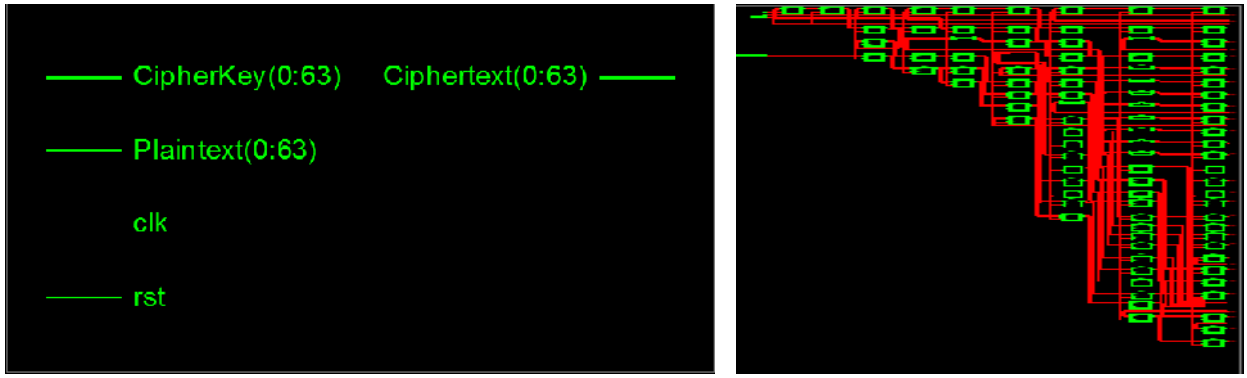


Fig. 8: New DES schematic generated by Xilinx ISE 9.2i tool

6- Performance comparison

A comparison is made between this proposed design with some other DES hardware implementation. Table3 shows the performance figures for some DES hardware implementation. Achieved results of my work are competitive with the existing implementations.

The throughput of various design and implementation strategies for several hardware implementation of DES that have been reported in the comparison ranges from 26.7 to 7983Mbps.The fastest FPGA implementation of DES at [6] using a java-based (Jbits) achieves a data rate of 10752 Mbps, but the key schedule is computed entirely in software.

The implementation of the DES at [9] uses pipelined design with skew core key-scheduling where different keys are loaded every clock cycle. The implementation of a free DES cores at [18] is used a pipeline concept in ECB mode and performs a data rate of 3052 Mbps. A DES implementations at [4] uses pipeline concept for both 2 and 4 stages achieving throughput of 183.8 Mbps and 402.7 Mbps, respectively.

A VLSI implementation of the DES at [5] uses 0.6 micro CMOS technology. A novel method at [7] for implementing the key schedule using FPGA for high performance design. The FPGA implementation of the pipelined DES algorithm at [10] using a time data permutation which means for the same data and key, the ciphered data is varied with time, so the security of the algorithm is increased and performs a data rate of 7983 Mbps. Another research described by [11] for high-performance reconfigurable hardware implementations of the Data Encryption Standard (DES) algorithm. This is achieved by combining pipelining concept with time-variable key technique achieving throughput of 7268.59 Mbps.

This work is the fastest single-chip FPGA design as compared with other previous implementations with throughput of 18327 Mbps assuming data stream length>>119 and latency of 119 clock cycles. In the proposed design the latency is increased and not an important thing because the design is implemented using pipelined concept which means that the first output is obtained after 119 clock cycle, and the other outputs are obtained in each clock cycle.

Table 3: Performance comparison among different hardware implementations.

Author	Device used	CLB Slices	Allowed frequency (MHz)	Throughput (Mbps)	Design strategies
K. Wong, *et al.* [3]	XC4020E	438	10	26.7	Non-pipeline (One round design)
E. Bilam [2] (software)	Alpha8400	-----	300	127	
J. P. Kaps and C. Paar [4]	XCV4028EX	741	25.18	402.7	16-stage pipeline Designs (Latency=16)
Free-DES[18]	XCV400	5263	47.7	3052	
M. McLoone and J. McCanny [7]	XCV1000	6446	59.5	3808	
C. Patterson (Jbits) [6]	XCV150	1584	168	10752	
Sandia Laboratories [5]	ASIC	------	--------	9280	
V. Patel, *et al.*[9]	XC3S500E	2814	111.882	7160	
K. M. A. Abd El-Latif, H. F. A. Hamed, E. A. M. Hasaneen [11]	XC3S500E	2566	113.75	7268.59	
K. M. A. Abd El-Latif, H. F. A. Hamed, E. A. M. Hasaneen [10]	XC3S500E	2062	124.734	7983	
Proposed Design	**XC3S500E**	**3254**	**286.369**	**18327**	**119-stage super pipelined Design(Latency=119)**

6- Conclusions

In this paper, an efficient FPGA implementation of the DES Encryption algorithm based on super pipelining concept is presented. The goal of using this concept is to achieve highest possible throughput. In the 119-stage super pipelining design, data blocks can be loaded every clock cycle and after latency with 119 clock cycle the ciphered data will appear with every clock cycle. At a clock frequency of 286.369 MHz, the 119 superpipelining design can encrypt data block at a rate of 18327Mbps. 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.

References

[1] D. Kahn: The Code breakers: the story of secret writing, MacMillan publishing, 1996.
[2]E. Biham, "A fast new DES implementation in software," Proc. 4th Int. Workshop on Fast software Encryption, FSE '97, Haifa, Israel, Jan. 1997, pp. 260–271.
[3] K. Wong, M. Wark and E. Dawson, " A Single-Chip FPGA Implementation of the Data Encryption Standard (des) algorithm," In: IEEE Globecom Communication Conf., Sydney, Australia, Vol. 2, 1998, pp. 827–832.
[4] J. P. Kaps and C. Paar, "Fast DES implementations for FPGAs and its application to a Universal key-search machine," In: Proc. 5th Annual Workshop on selected areas in cryptography-Sac' 98, Ontario, Canada, Springer-Verlag,1998,pp. 234–247.
[5] Wilcox, Pierson, Robertson, Witzke and Gass, "A DES ASIC Suitable for Network Encryption at 10 Gbps and Beyond," CHES'99 ,LNCS 1717, 1999, pp. 37 - 48.
[6] C. Patterson, "High Performance DES Encryption in Virtex FPGAs Using Jbits, " Field-Programmable Custom Computing Machines, FCCM'00, USA, 2000, pp. 113-121.
[7] M. McLoone and J. McCanny, "High-performance FPGA implementation of DES using a novel method for implementing the key schedule," IEE Proceedings: Circuits, Devices & Systems, Vol. 150**,** 2003, pp. 373–378.
[8]N. A. Sadiq, F. R. Henriquez and A. D. Perez, "A compact and efficient FPGA implementation of the DES algorithm," International conference on reconfigurable computing and FPGAs. Sep. 20-21, 2004.
[9] V. Patel, R. C. Joshi, A. K. Saxena, "FPGA Implementation of DES Using Pipelining Concept With Skew Core Key-Scheduling," Journal of Theoretical and Applied Information Technology, Vol. 5, No3, March, 2009, pp. 295-300.
[10] K. M. A. Abd El-Latif, H. F. A. Hamed and E. A. M. Hasaneen, "FPGA implementation of the pipelined Data Encryption Standard (DES) based on variable time data permutation," the online journal on electrics and electrical engineering (OJEEE), Vol. (2), No. (3), 2010, pp. 298-302.
[11]K. M. A. Abd El-Latif, H. F. A. Hamed and E. A. M. Hasaneen, "Hardware Implementation of DES Using Pipelining Concept with Time-Variable Key," 22nd international conference on microelectronic (ICM), 2010.
[12] A. Singh and M. Bansal, "FPGA implementation of optimized DES encryption alogorithm on Spartan-3E," International journal of scientific and engineering research, Vol. 1, 2010.
[13] U. R. Kumari and T. K. Rasagna, "Implementation of pipelined data encryption standard for security enhancement through verilog," International journal of computer applications and technology, Vol. 1, 2012, pp. 4-8.

[14]S. Maniknda and S.Tandle, "Implementation of data encryption algorithm for FPGA based real-time data security applications," International conference on electronics and communication engineering, April 28th-29th, 2012.
[15] "Spartan-3E FPGA Starter Kit Board User Guide", © 2006-2008 Xilinx, Inc., UG230 (v1.1) June 20, 2008.
[16] D. Perry, "VHDL: Programming by Example", McGraw-Hill,New York, *ISBN* 0-07-140070-2, 2002.
[17] B. Furouzan, "Data communications and networking", McGraw-Hill, New York, NY 10020, ISBN-13 978-0-07-296775-3 to 0-07-296775-7, 2007.
[18] Core(2000), F.D.: (2000) URL: http://www.free-ip.com/DES/.

The work was carried out at the college of Engineering. University of Mosul