ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Block Ciphers and Data Encryption Standard by using Feistel Decryption Algorithm

A.Swathi
M.Tech Student,
Department of ECE,
Anu Bose Institute of Technology (ABIT),
Paloncha, khammam, India.

T.Venu Gopal
Associate Professor,
Department of ECE,
Anu Bose Institute of Technology (ABIT),
Paloncha, khammam, India.

ABSTRACT:

This paper describes a high-performance reconfigurable hardware implementation of the new Data Encryption Standard (DES) based on variable time data permutation. The permutation choice is variable with time. For the same data and key, the ciphered data is varied with time, so the security of the algorithm is increased. We have used the pipelining concept in our design. Our DES is implemented on Xilinx Spartan-3e (XC3s500e). Final 16-stage pipelined design is achieved with data rate of 7.98 Gbps and 2062 CLB slices. The proposed design is more secure and among the fastest hardware implementations with better area utilization.

Index Terms:

Data Encryption Standard (DES) Algorithm, Field Programmable Gate Arrays (FPGA), Pipelining, Variable Time Data Permutation.

INTRODUCTION:

The Data Encryption Standard (DES) [1],[2] was published by the United States national bureau standards (NBS) in January 1977. It has been used by United States federal agencies since 1977. The Data Encryption Standard (DES) is a block cipher which encrypts data in 64 bits by using 56 bits key. Because of small key size, applications of DES are limited. So we have to increase the security of the algorithm. In order to increase the security of DES, The Data Encryption Standard (DES) Based on Variable Time Data Permutation [3] has been proposed. The proposed scheme uses a permutation box that contains several data permutations to be selected. Every time, a different permutation is selected. This is leads to that the same plain text is ciphered to different ciphered texts by time. As we can see, it is very difficult to hack into the new DES because of it's dynamic behavior.

This paper describes the hardware implementation of 16- stage pipelined DES Based on Variable Time Data Permutation. It allows 16 data blocks to be processed simultaneously resulting in an impressive gain in speed. We implemented the design on Xilinx Spartan FPGA technology. FPGAs are a rapidly class of electronic components. They can be reprogrammed an unlimited number of times and also offer a low cost design. They offer high speed similar to VLSI(very large scale integrated circuit) and high flexibility similar to software, so they can be used in innovative designs. Section II describes the DES algorithm. DES Based on Variable Time Data Permutation [3] is presented in section III. Pipelining the new DES is described in section IV. Section V gives implementation summary. Section VI compares the achieved results with the previous DES implementations. The conclusion is given in Section VII.

Existing System:

DES is a block cipher, which takes 64-bit input and 64-bit key. A 64-bit output is produced as shown in figure (1). The effective key length is 56 bits because 8 bits are used as parity checking bits. The DES algorithm consists of 16 rounds as shown in figure (2). Data is 64-bit firstly permutated and then divided to 32 bits right (R) and left (L). It is processed through DES function as shown in Figure (3) which 32 right bits are expanded to 48 bits to be processed through XOR function with the round key. The XOR output is converted from 48 bits to 32 bits through substitution boxes (S boxes). The S boxes output is XORed with the 32 bits left and the output is the right to the next round. The right bits of the previous round is the left of the next round as illustrated by the following formula:

$$L_i = R_{i-1} \quad , R_i = L_{i-1} \oplus F(R_{i-1}, K)$$

There are 16 round keys that are generated from the main key. Each round has it's sub generated key from the main key. Figure (4) shows the sequence of generating the 16 sub keys.

The main key is firstly permutated and then will be shifted according to the round number. The round number indicates the shift bits. After that, the output is permutated by the second permutation

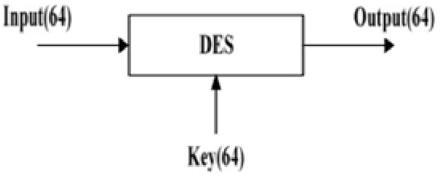


Figure (1): Block diagram of DES

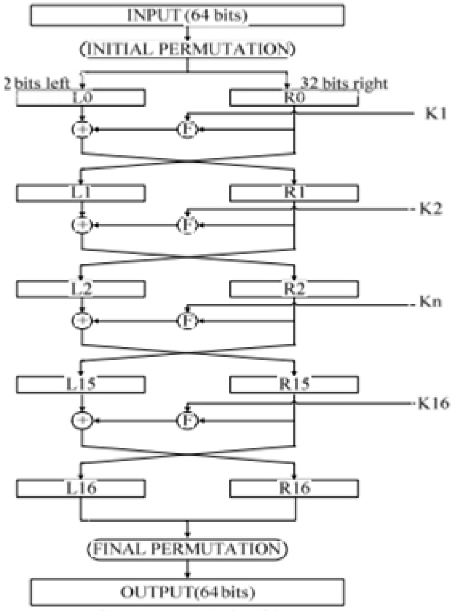


Figure (2): DES algorithm

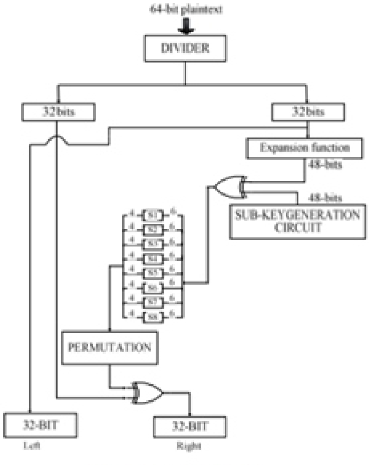


Figure (3): DES function

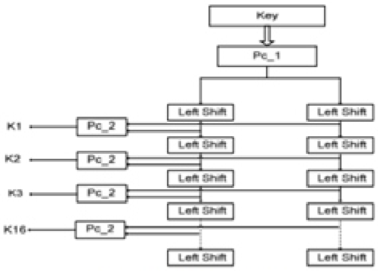


Figure (4): Sub key generation

III.DES BASED ON VARIABLE TIME DATA PERMUTATION:

According to Figure (2), it is clear that this algorithm can be attacked by any method of cryptanalysis linear [4], differential [5], and brute force attack [6]. In order to make the DES algorithm more secure, the new DES [3] shown in Figure (5) has been developed.

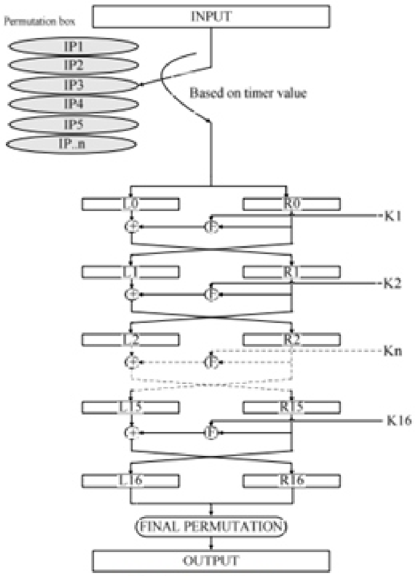


Figure (5): New DES

Proposed System:

The proposed scheme has a permutation box that contains several permutations in order to be used periodically with time. In this scheme, the synchronization between sender and receiver will be based on timer. While sender starts transmission, the timer will be on in both sender and receiver. The proposed scheme changes the permutation of data periodically using a timer control as shown in Figure (6). At the beginning of transmission, sender and receiver timers will be on and for every timer value, the permutation is selected

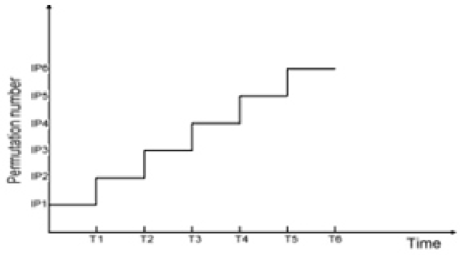


Figure (6): Permutation variation with time

In order to avoid the disadvantages of the synchronization between the sender and receiver, we can transmit an additional data with the ciphered data to indicate the timer value to guide receiver to choose the correct permutation from its permutation box. This behavior increases the size of transmitted bits, but it enhances the algorithm. Sender has a permutation box that periodically selects a permutation from it. Also, the receiver has the same permutation box. The sender and receiver are connected to have the same permutation at any specific time. For any hackers behave, linear [4], differential [5], and brute force [6], detecting the algorithm will be very difficult, because the same plaintext is ciphered to different forms as a function of time. In our design, we use six different permutations. Flow chart shown in figure(7) introduces the method of changing the data permutation with time. Every time, the program checks the timer value. According to the timer value, the permutation is selected from the permutation box.

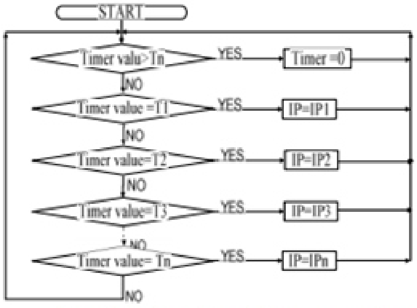


Figure (7): Flow chart of data permutation

PIPELINING DES BASED ON VARIABLE TIME DATA PERMUTATION

Pipelining is used in large designs for maximum throughput. The beauty of a pipelined design is that new data can begin processing before the prior data has finished. The nature of new DES [3] makes it ideally suited to pipelining that can be 4, 6, 8, 16 stages. Our implementation is 16-stage pipelined design as shown in figure (8).

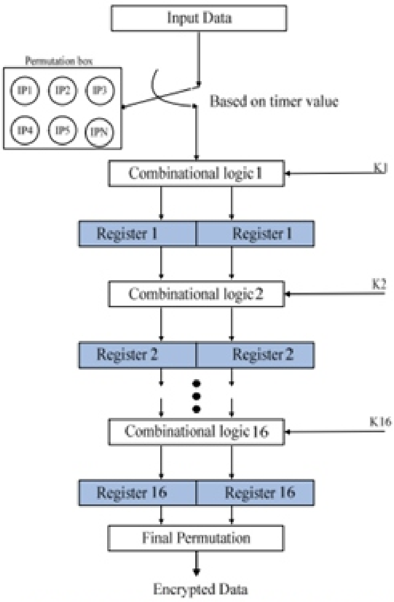


Figure (8): pipelined DES based on variable time data permutation (16-stage)

FPGA implementation of the new DES [3] was accomplished on a Spartan-3e XC3s500e-4fg320 using Xilinx Foundation Series F9.li as synthesis tool. The design was coded using VHDL language. It occupied 2062(44%) CLB slices, 1004 (10%) slice flip flops and 197(84%) I/Os. The design achieves a frequency of 124.73 MHz. It takes 16 clock cycle latency first time only and then encrypts one data block (64-bit) per clock cycle. This is leads to that the throughput = (124.73*64) =7.98 Gbps. Full design schematic and simulation window are shown in figure (9) and figure (10)

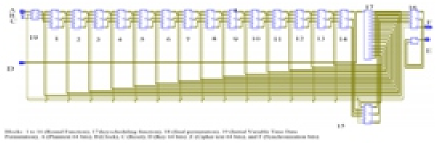


Figure (9): New DES schematic generated by Xilinx ISE tool.

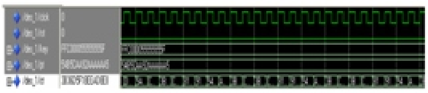


Figure (10): Simulation window of the new DES.

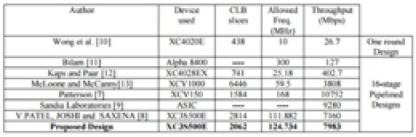
VI.PERFORMANCE COMPARISON

Table (1) shows the performance figures for some DES hardware implementations. Our achieved results are competitive with the existing implementations. Several FPGA implementations of DES have been reported in the comparison achieving throughput ranges from 26 to 10752 Mbps using different design strategies.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

A Java-based (Jbits) DES implementation [7] is the fastest implementation. The key schedule is computed entirely in soft ware. As a result, all cryptographic key input and sub-key generation logic are removed from the pipelined design. A DES implementation at [8] uses a pipelined design with skew core key-scheduling to load different keys every clock cycle.A VLSI DES implementation [9] uses 0.6 μm CMOS technology. Our proposed design is presented in table (1). According to the comparison of our design and the previous designs, we find that there is only one claim. The throughput of javabased (Jbits) DES implementation is above our design with encryption rate 10752 Mbps. However, in this design [7] the key schedule is computed in software. Our implementation is more secure than any presented DES implementation because of time-variant behavior and one of the fastest single-chip FPGA designs.

Table (1): Performance comparison



Author	Device used	CLB slices	Allowed Freq (MHz)	Throughput (Mbps)	
Wong et al. [10]	XC4020E	438	10	26.7	One round Design
Biham [11]	Alpha 8400	----	300	127	16-stage Pipelined Designs
Kaps and Paar [12]	XC4028EX	741	25.18	402.7	
McLoone and McCanny[13]	XCV1000	6446	59.5	3808	
Patterson [7]	XCV150	1584	168	10752	
Sandia Laboratories [9]	ASIC	----	----	9280	
V PATEL, JOSHI and SAXENA [8]	XC3S500E	2814	111.882	7160	
Proposed Design	**XC3S500E**	**2062**	**124.734**	**7983**	

VII. CONCLUSION:

In this work, an efficient and compact FPGA implementation for the pipelined DES based on variable time data permutation is presented. For the same data and key, the ciphered text is variable with time. As a result of this, the security of the algorithm is increased. In the 16-stage pipelined design , data blocks can be loaded every clock cycle and after an initial delay of 16 clock cycle the ciphered data will appear on consecutive clock cycles. At a clock frequency of 124.73 MHz the 16-stage pipelined design can encrypt or decrypt data blocks at a rate of 7.98 Gbps.

REFERENCES:

[1] National Bureau of Standard (U.S.), "Data Encryption Standard (DES)," Federal Information Processing Standards Publication 46, National Technical Information Service, Springfield, VA, April 1977.

[2] National Bureau of Standard (U.S.), "DES modes of operation," Federal Information Processing Standard Publication 81, National Technical Information Service, Springfield, VA, December 1980.

[3] K. M. A. Abd El-Latif, E. A. M. Hasaneen and H. F. A. Hamed, "Improved DES Algorithm Based On Variable Time Data Permutation, "International Conference for Advanced Computer Theory and Engineering (ICACTE), vol. 2, pp. 1381-1388, Cairo, September, 2009.

[4] Mitsuru Matsui, "linear Cryptanalysis Method for DES Cipher," Advances in Cryptology - EUROCRYPT '93, vol. 765, pp. 387-397, May 1993.

[5] E.Biham and A.shamir, "Differential cryptanalysis of DES-like cryptosystem," Journal of cryptology, vol. 4, pp. 3-21, 1991.

[6]http://en.wikipedia.org/wiki/Brute_force_attack.

[7] Patterson, "High Performance DES Encryption in Virtex FPGAs Using Jbits, " Field-Programmable Custom Computing Machines, FCCM'00, pp. 113- 121, USA, 2000.

[8] Vishwanath 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, pp. 295-300, March, 2009.

[9]Wilcox, Pierson, Robertson, Witzke, Gass, "A DES ASIC Suitable for Network Encryption at 10 Gbps and Beyond," CHES'99 ,LNCS 1717, pp. 37 - 48 , 1999.

[10] K. Wong, M.Wark, E. Dawson, " A single-chip FPGA implementation of the data encryption standard(DES) algorithm,"IEEE Globecom Communication, pp. 827-832 vol.2, Sydney, Australia, 1998.

[11] E.Biham, " A Fast New DES Implementation in Software," 4 th International Workshop on Fast Software Encryption , FSE '97, pp.260-271, Israel, 1997.

[12] Jens-Peter Kaps, Christof Paar, "Fast DES Implementations for FPGAs and Its Application to a Universal Key-Search Machine," 5th Annual Workshop on selected areas in cryptography, pp.234- 247 ,Canada 1998.

[13] McLoone, McCanny," High-performance FPGA implementation of DES using a novel method for implementing the key schedule," IEE proc: Circuits, Devices and Systems, Vol 150, pp. 373-378, 2003.