FPGA based Secure Biomedical Image Transmission

Jinu Elizabeth John
Department of ECE
Amrita Vishwavidyapeetham
India

Ajay Daniel Peter, PhD.
Temple University
Philadelphia,USA

ABSTRACT
This paper presents a reconfigurable, high performance hardware implementation of highly secure biomedical image transmission system which can be used for sending medical reports in military and high security environments. The algorithm for encryption is based on DES algorithm with a novel skew core key scheduling. The encrypted image is not intelligible to an intruder, but the recovered image has high level of clarity. This type of encryption can be used in applications where we need to discourage eavesdropping from co-channel users or RF scanners. The biomedical image encryption technique is implemented on Virtex 5 XC5VLX110T Field Programming Gate Arrays (FPGA) technology and NET FPGA. Final 16-stage pipelined design is achieved with encryption rate of 35.5 Gbit/s and 2140 number of Configurable logic blocks (CLBs).

General Terms
Image Encryption,Security,Algorithms

Keywords
Steganography,Image, Encryption, Eavesdropping.

1. INTRODUCTION
In this era of rapid developments in internet and multimedia systems the need for effective ,secure and reliable processing ,storage and transmission of images has increased. If a highly confidential medical image report or secret image is to be transmitted among the communicators, then they must use a highly secure method so that the transmitted image is not interpreted by an intruder who is attacking the communication network. So image steganography techniques are the most inevitable module in secure image transmission as they serve as a powerful measure against eavesdropping and are needed to ensure privacy in multimedia transmission in internet. Encryption algorithms have been developed as a mechanism for providing this security and there is a need to perform these algorithms on data in real time.In typical image encryptors the image is digitized and the resulting sequence is encrypted into an unintelligible image to avoid eavesdropping. The use of different keys in every clock cycle, make the scrambled image seem unintelligible and very tough to break making time slot based image steganography suitable for the most sensitive strategic communications .The use of entirely different key sets every clock cycle improves the overall security of the device.This supports the Electronic Code Book (ECB) mode of operation. FPGA implementation of image encryptor or decryptor was accomplished on a Virtex 5-xc5vlx110t-3ff1100 and NET FPGA using Xilinx Foundation Series 12.1 as synthesis. Feistal algorithm implementations on reconfigurable hardware provides major benefits over VLSI (very large scale integrated circuits) and software platforms since they offer high speed similar to VLSI and high flexibility similar to software. On the other hand, reconfigurable devices are attractive since the time and costs of VLSI design and fabrication can be reduced. Moreover, they offer high potential for reconfiguration and experimenting on multiple architectures or several revisions of the same architecture.

In a typical image encryptions, the clear image is digitized and the digital sequence is scrambled into an unintelligible image in order to avoid eavesdropping. The recorded image is subdivided into smaller blocks of 64 bit. A 64 bit novel DES encryption algorithm is used to rearrange blocks within each segment .For transmission the rearranged blocks are brought together. The use of different keys in every clock cycle, make the scrambled image seem unintelligible and very tough to break making time slot based image scrambling suitable for the most sensitive strategic communications. The scrambling is based on a 16 –stage pipelined DES algorithm with a novel skew core key scheduling. It allows simultaneous processing of 16 data blocks, resulting in an impressing gain in speed. This the overall security of the speech scrambling improved since it uses different keys every clock cycle and therefore the users are not restricted to the use of same key at any time of data transfer.This design is implemented on Virtex 5 FPGA and NET FPGA

2. IMAGE SECURITY SYSTEMS
In image scrambling systems, the recorded multimedia image is modified by a known scrambling algorithm so as to make the scrambled image unintelligible does not convey any information of the original secret image. This image scrambling algorithm is governed by a specific code or "key". Different scrambled signals can be obtained if different keys are used. The image that is thus scrambled is transmitted. At the receiver the scrambled signal is again modified by a descrambling algorithm under the control of a specific key. This ultimately results in an image, which resembles the original secret image exactly. In a correctly operating system, the Scrambling Key and Descrambling Key are identical, and the descrambling algorithm is the inverse of the scrambling algorithm. Therefore, whatever the scrambling algorithm does, the descrambling algorithm undoes. If the Scrambling Key and Descrambling Key are different, then the descrambling algorithm will not recover the original signal properly. This paper uses a pipelined DES algorithm with a novel key scheduling to scramble communications.

3. BACKGROUND
In this project the scrambling and descrambling is done by means of DES algorithm. The data encryption standard (DES) is the best known and most widely used private key encryption algorithm developed by IBM in 1977 as a modification of an earlier system known as Lucifer[1].The overall scheme of DES algorithm is illustrated (see Figure.1).DES is a Feistel cipher which operates on two inputs: the 64 bit plain text to be encrypted and 56-bit secret key. Precisely, the input key is specified as 64 bits, 8 bits of which is used for parity checking. With a key length of 56 bits 2^56 combinations are possible, and therefore the cryptanalytic works seem very tough. The encryption

proceeds in 16 stages or rounds. From the input key K, sixteen 48-bit sub keys Ki are generated, one for each round[1]. Within each round, 8 fixed, carefully selected 6-to-4 bit substitution mappings (S-boxes) S, collectively denoted S, are used. The 64-bit plaintext is divided into 32-bit halves L0 and R0. Each round is functionally equivalent, taking 32-bit inputs Li−1 and Ri−1 from the previous round and producing 32-bit outputs Li and Ri for $1 \leq i \leq 16$

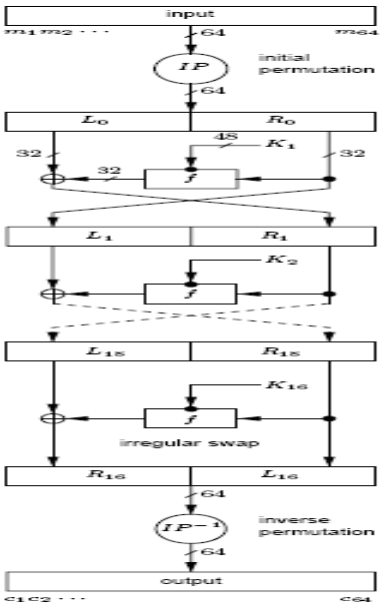

Fig 1 DES Algorithm Description

3.1 F function

The f function of DES algorithm is made up of four functions: Expansion, Xor, Substitution, Permutation. The right half of each round carries out a key-dependent substitution on each of 8 characters, then uses a fixed bit transposition to redistribute the bits of the resulting characters to produce 32 output bits.(see figure 2)

3.2 Key Scheduling

In DES algorithm 16 different sub keys each of 48 bit wide is developed from a single 56 bit key. These operations make use of tables PC1 and PC2 which are permuted choice 1 and permuted choice 2 [6].The 8 bits of 64 bits is discarded by PC1 .The remaining 56 bits are permuted and assigned to two 28-bit variables C and D; and then a cyclic shift operation is carried out on each half. That is, for 16 iterations, both C and D are rotated either 1 or 2 bits, and 48 bits (Ki) are selected from the concatenated result. This process is repeated for each stage of 16 stage pipeline. In rounds, 1, 2, 9 and 16 of DES algorithm the halves are shifted one position to left and for all other rounds it is shifted to left by two places.(see figure 3).

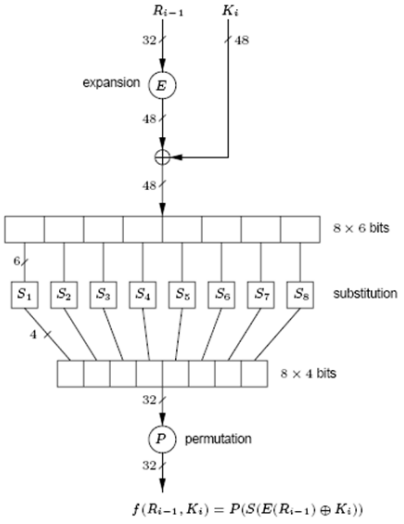

Fig 2 f box

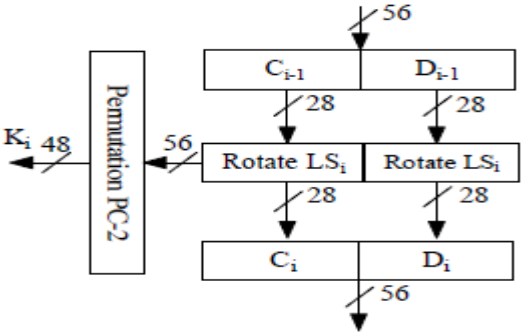

Fig 3 Key Scheduling

3.3 Pipelined Implementation

The ECB mode of DES algorithm is implemented in this paper as it can be easily pipelined[8].This pipelined DES design increases the speed and throughput of DES significantly[6]. If a combinational digital circuit can be divided into stages, we can insert buffers (registers) at proper places and convert the circuit into a pipelined design [2]. Adding pipeline into a combinational design can only increase a system's throughput. Such an approach does not reduce the delay in an individual task. Actually, because of the overhead introduced by the registers and non-ideal stage division, the delay will be worse than that of the non-pipelined design [2].(see figure 3)

3.4 Skew Core Implementation of DES

For the 16-stage pipelined DES design, the sub keys are pre-computed and it is necessary to control the time at which the sub keys are available to each function f block [6]. This is accomplished by addition of an array of D flip flops that delays the individual sub-keys by required amount [6, 9].

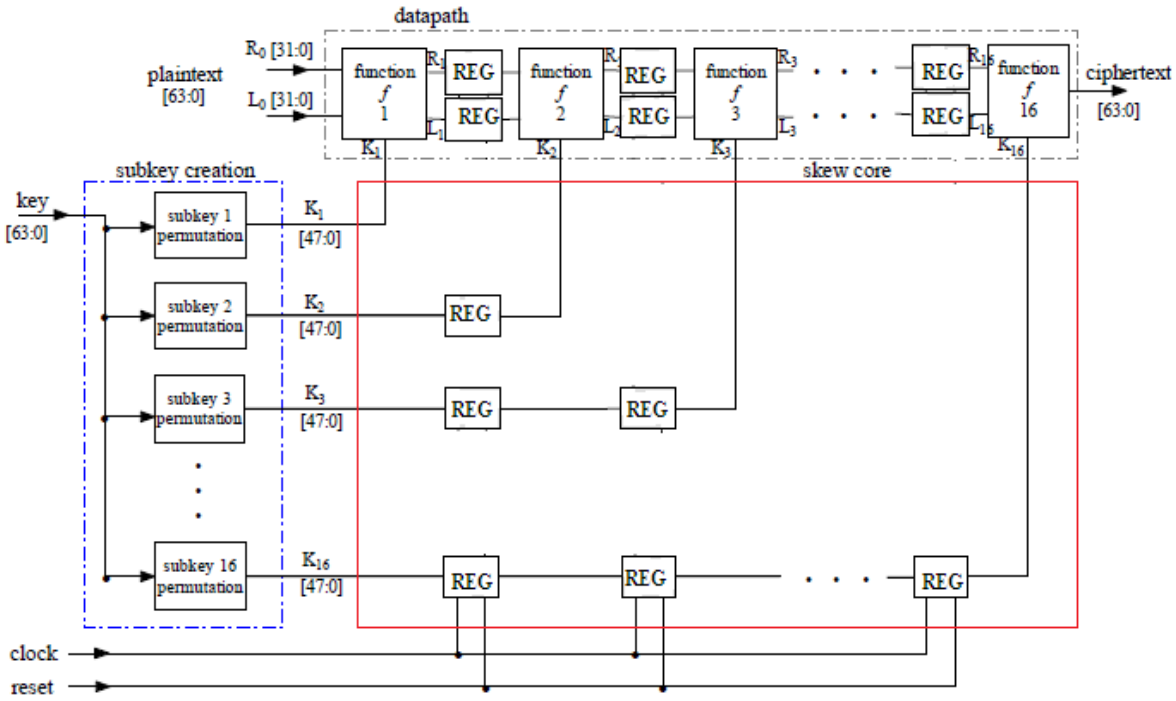

Fig 4: Skew Core Key Scheduling

4. IMAGE PROCESSING USING SKEW CORE DESIGN

Image encryptions described in this paper actually digitizes the conversation at internet and applies a cryptographic technique to the resulting bit stream. The Original image is divided into a number of blocks or segments each of 64 bit wide. The transformed image is then fed to the modified DES encryption algorithm. The main idea is that an image can be viewed as an arrangement of blocks. The transformation process refers to the operation of dividing and replacing an arrangement of the original image. The image is thus encrypted in blocks to produce a series of cipher texts. The cipher texts in decimal format is combined in a matrix form to obtain the required encrypted image in Matlab.The decryption can also be done in the reverse manner to obtain the original image. The scrambling technique presented in this paper offers high speed and throughput alongside improved levels of security. The encrypted and decrypted image is shown in Figure 8. The chaotic sequence is implemented in the DES algorithm to improve the initial keys and the iterating operations, so that the biomedical image encryption is combined with DES algorithm [12].

5. IMPLEMENTATION RESULTS

FPGA implementation of DES algorithm was accomplished on Virtex 5 FPGA,Xilinx as synthesis tool and Modelsim 6.2c as simulation tool. The design was coded using Verilog HDL language.It occupied 2140 (45%) CLB slices, 1808 (19%) slice Flip Flops and 187 (80%) I/Os. It takes 16 clock cycles latency first time only then encrypts one data block (64-bits) per clock cycle. Therefore, the achieved throughput is 35.5 Gbits/s. Full design schematic and simulation window are shown.(see figure5).

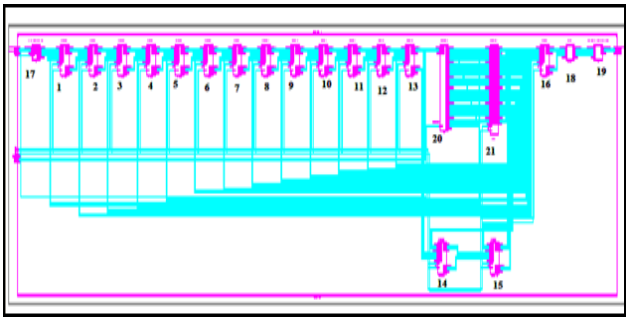

Fig 5: Full DES design schematic generated by Xilinx ISE tool

BLOCKS:1 to 16 (Round Function),17(Initial Permutation),18(Swap),19(Inverse initial Permutation),20(Key Top),21(Skew Core)

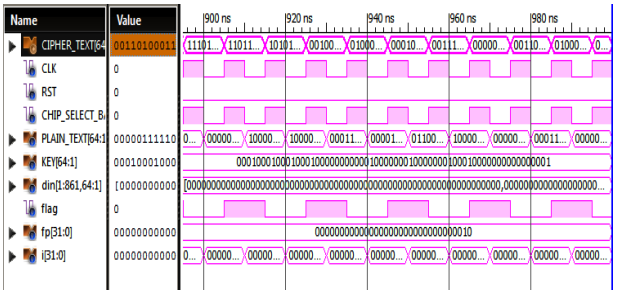

Fig 6: Simulation Window of DES design

6. PERFORMANCE COMPARISON

The fastest DES software implementation achieves a throughput of 127 Mbit/s on a 300MHz Alpha 8400 processor[4]. A VLSI implementation of DES on static 0.6 micron CMOS technology at [7] is the fastest implementation of DES reported in the literature. The image encryption scheme was implemented in design with skew and also without skew core key scheduling and the device utilization details are shown in figure. And it is found that pipelined DES has high speed, high data throughput and less CLB utilization. The performance analysis in terms of area ,timing and power was obtained using Synopsys tool.

COMPARISON OF SPECIFICATIONS FOR DES HARDWARE AND SOFTWARE IMPLEMENTATIONS:

DEVICE USED	SYSTEM CLOCK (MHz)	DATA RATE (Giga bits/second)
XC4020E	10	0.0267
Alpha 8400	300	0.127
XC4028EX	25.18	0.4027
XCV400	47.7	3.052
XCV1000(McLoone,McCanny)	59.5	3.808
ASIC	--	9.280
XCV150	168	10.752
*XE3S500E (SPARTAN 3E)	50	3.2
*XC5VLX110T (VIRTEX 5)	550	35.5
*NET FPGA	400	25.58

* AS PER OUR IMPLEMENTATION RESULTS

Fig 7: Various Implementation Results

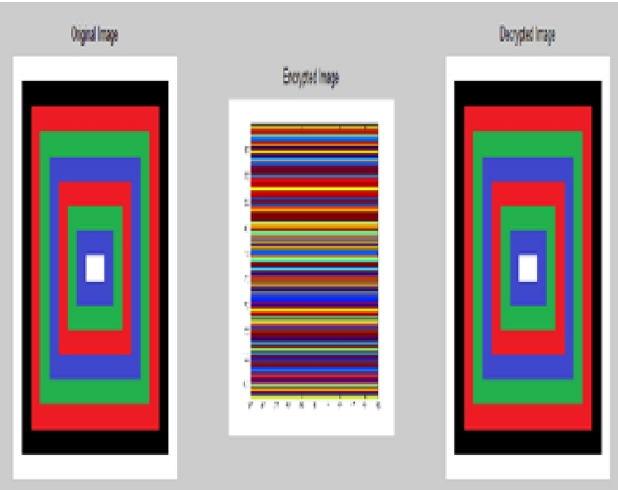

Fig 8: Orginal ,Encrypted and Decrypted Image

8. CONCLUSION

This paper describes a high speed, high throughput image scrambling system. A 16-stage pipelined novel implementation of DES algorithm design is presented here for scrambling. The input image is split into blocks of 64 bits and it allows the processing of 16 data blocks simultaneously. Image data blocks can be loaded every clock cycle and after an initial delay of 16 clock cycles the corresponding encrypted/decrypted voice data blocks will appear on consecutive clock cycles. Different keys can be loaded every clock cycle allowing the possibility of using multiple keys in any one session of data transfer. In general, hardware implementations of encryption algorithms and their associated keys are physically secure, as they cannot easily be modified by an outside attacker. The 16-stage pipelined design can encrypt or decrypt data blocks at a rate of 35.5Gbit/sec.

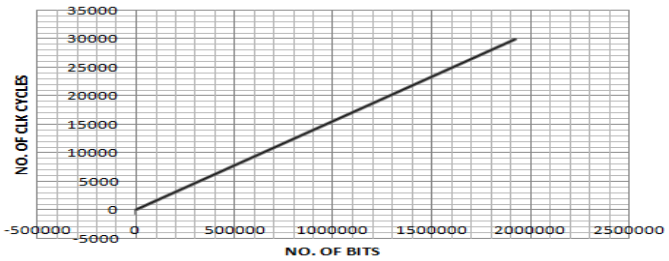

Fig 9: No. of Bits vs No. of Clock Cycles

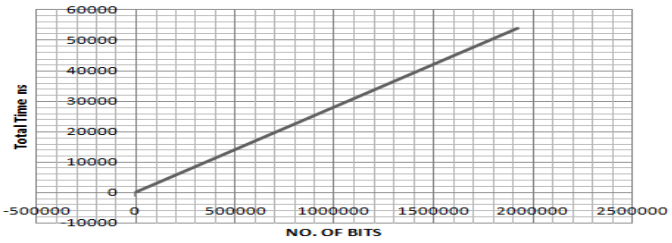

Fig 10: No. of Bits vs Time

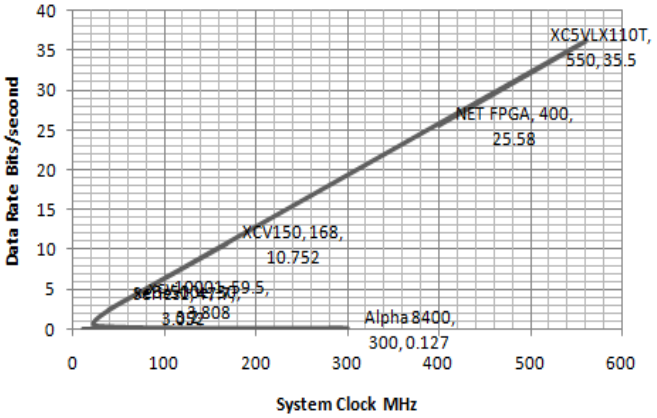

Fig 10: System Clock vs Data Rate

9. REFERENCES

[1] Alani M.M "DES96-improved DES security" 7th International Multi-Conference on Systems Signals and Devices(SSD),2010

[2] First Int. Workshop on Cryptographic hardware and embedded systems, CHES '99, Worcester, MA, USA, August 1999 (Springer-Verlag, 1999), pp. 37–48

[3] Patterson, C. (Xilinx Inc.): 'High performance DES encryption in virtex FPGAs using Jbits'. Proc. IEEE

Symp. on Fieldprogrammable custom computing machines, FCCM '00, Napa Valley, CA, USA, April 2000 (IEEE Comput. Soc., CA, USA, 2000), pp. 113–121

[4] Kaps, J., Paar, C.: 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 (1998) 234–247.

[5] Core(2000), F.D.: (2000) URL: http://www.free-ip.com/DES/.

[6] McLoone, M., McCanny, J.: High-performance FPGA implementation of DES using a novel method for implementing the key schedule. IEE Proc.: Circuits, Devices & Systems 150 (2003) 373–378.

[7] J Wilcox, D., Pierson, L., Robertson, P., Witzke, E.L., Gass, K.:A DES asic suitable for network encryption at 10 Gbs and beyond. In:CHESS 99, LNCS 1717 (1999) 37–48.

[8] Biham, E.: 'A fast new DES implementation in software'. Proc. 4th Int. Workshop on Fast software Encryption, FSE '97, Haifa, Israel, Jan. 1997 (Springer-Verlag, 1997), pp. 260– Van Der Lubbe, J.C.A.: 'Basic methods of cryptography' (Cambridge University Press, 1998)Proceedings of TENCO' 97,IEEE, December 1997

[9] Vishwanath Patel. C Joshi, A.K Saxena: FPGA implementation of DES using pipelining concept with skew core key scheduling, Journal of Theoritical and Applied Information Technology(2005-2009)

[10] Nascimento, J.C. ; Figueiredo, M.A.T. ; Marques, J.S. Image Processing,2013 IEEE Transactions on Activity Recognition Using a Mixture of Vector Fields

[11] Zhang Yun-Peng ; Liu Wei ; Cao Shui-ping ; Zhai Zheng-jun ; Nie Xuan ; Dai Wei-di Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on Digital image encryption algorithm based on chaos and improved DES

[12] Bin, Ling ; Lichen, Liu ; Jan, Zhang Advanced Computer Control (ICACC), 2010 2nd International Conference on IEEE Conference Publications Image encryption algorithm based on chaotic map and S-DES