FPGA IMPLEMENTATION OF DES USING PIPELINING CONCEPT WITH SKEW CORE KEY-SCHEDULING

[1]VISHWANATH PATEL, [2]R. C. JOSHI, [3]A. K. SAXENA

[1]M-Tech Student, Department of Electronics & Computer Engineering, IIT Roorkee, India-247667
[2]Prof., Department of Electronics & Computer Engineering, IIT Roorkee, India-247667
[3]Prof., Department of Electronics & Computer Engineering, IIT Roorkee, India-247667

E-mail: vishwanath03@gmail.com, rcjosfec@iitr.ernet.in, kumarfec@iitr.ernet.in

ABSTRACT

This paper presents a high-performance reconfigurable hardware implementation of the Data Encryption Standard (DES) algorithm. This is achieved by combining pipelining concept with novel skew core key scheduling method and compared with previous illustrated encryption algorithms. The DES design is implemented on Xilinx Spartan-3e Field Programming Gate Arrays (FPGA) technology. Final 16-stage pipelined design is achieved with encryption rate of 7.160 Gbit/s and 2814 number of Configurable logic blocks (CLBs). This result is among the fastest hardware implementations with better area utilization.

Key Words: *Data Encryption Standard (DES) algorithm, Field Programming Gate Arrays (FPGA), pipelining, key scheduling, skew core concept.*

1. INTRODUCTION

The DES algorithm is a private-key encryption algorithm, which was developed by IBM and has been a federal standard since 1977 [1]. Presently replaced by the Advanced Encryption Standard (AES) algorithm, but still used widely in the public domain and provides a basis for comparison for new algorithms.

A 16-stage pipelined DES Algorithm hardware implementation is outlined in this paper. It allows 16 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 is implemented on Xilinx Spartan FPGA technology. Implementing cryptographic algorithms 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. VLSI implementations are fast but must be designed all the way from behavioral description to the physical layout. They have to follow an expensive and time consuming fabrication process. Software implementations offer high flexibility but they are not fast enough for the applications where time factor is vital.

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 reprogramming and experimenting on multiple architectures or several revisions of the same architecture.

The rest of this paper is organized as follows: Section 2 describes the DES algorithm. Our proposed DES architecture and its implementation on a reconfigurable hardware device are presented in Section 3 and Section 4. Section 5 gives implementation summary. Section 6 compares the achieved results with the previous DES implementations. Conclusions and references are given in Section 7 and 8 respectively.

2. DES ALGORITHM DESCRIPTION

An outline of DES is shown in Fig. 1. It is a block cipher operating on 64- bits blocks of plaintext utilizing a 64-bits key. Every eight bit of

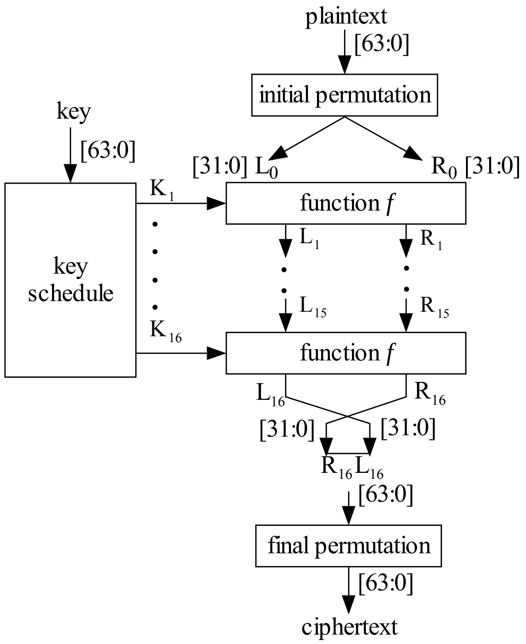


Fig. 1. DES algorithm block diagram

the 64-bits key is used for parity checking and otherwise ignored. After an initial permutation, the 64-bits input is split into a right (R_0) and left half (L_0), each 32 bits in length. DES has 16 iterations or rounds. In each round a function f is performed in which the data is combined with a 48-bits permutation of the key. After the 16th iteration, the right (R_{16}) and left (L_{16}) halves are concatenated and a final permutation, which is the inverse of the initial permutation, completes the algorithm.

2.1 *f* – Function

The function f of the DES algorithm is made up of four operations. Firstly, the 32-bits right half of the plaintext R_0 is expanded to 48-bits and then XORed with a 48-bits sub-key K1. The result is fed into eight substitution boxes (s-boxes), which transform the 48-bits input to a 32-bits output. Finally, a straight permutation (P-permutation) is performed, the output of which is XORed with the initial left half, L_0 to obtain the new right half R_1. The original right half R_0becomes the new left half L_1. This is shown in Fig. 2.

2.2 Key-Scheduling

The initial step in the in this procedure is to remove the parity check bits in the 64-bit key. Every eighth bit is used for parity checking, leaving 56-bits. A different 48-bits sub key is now generated for each of the 16 rounds of DES. The sub-keys are determined by first splitting the 56-bits into two 28-bits lengths of data. Then both halves are shifted left by either one or two bits depending on the round number.

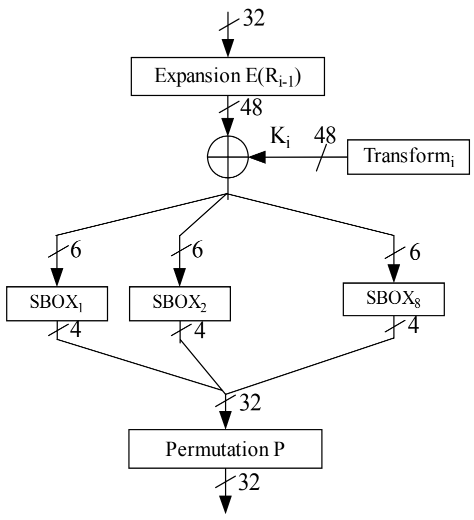


Fig. 2. f – function

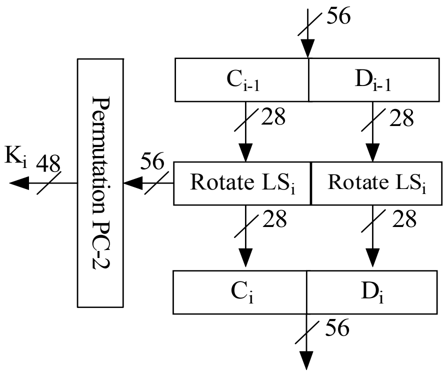


Fig. 3. One round of Keyschduling unit.

3. PIPELINING THE DES ALGORITHM

Pipelining is wildly use method in large design for speed enhancement. The iterative nature of the DES algorithm makes it ideally suited to pipelining and that can be 4, 6, 8 or 16 stages. The DES algorithm implementation presented in this paper is based on the ECB mode with 16 stages pipelining. Although the ECB mode is less secure than other modes of operation, it is commonly used and its operation can be pipelined [9].

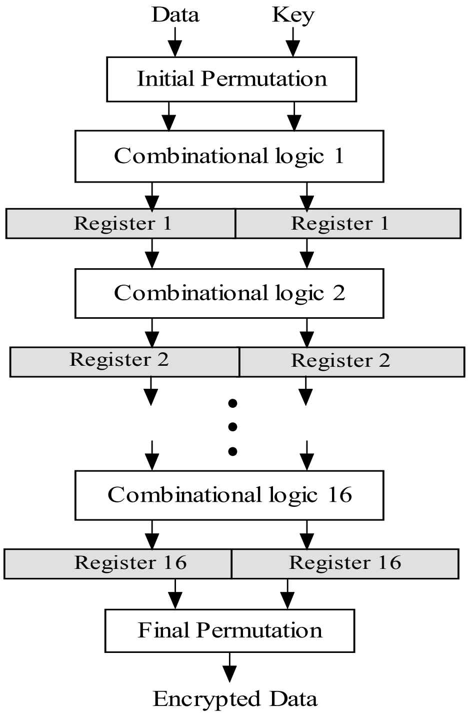


Fig. 4. DES Pipeline Architecture.

4. SKEW CORE KEY-SCHEDULING

In the implementation of the DES algorithm key schedule employed here same as above stated. the sub-keys are pre-computed that can also be done by direct mapping [6] of given key in required sub-keys but both ways using only wiring resources, So this part will be executed very fast and no optimizations would have any stage pipelined DES design and key-scheduling, it is necessary to control the time at which the sub-keys are effect. For maintaining proper synchronization in 16-available to each function *f* block. This is accomplished by the addition of a skew [6] that delays the individual sub-keys by the required amount. The skew consists of 48 bits array of register shown in Fig. 5. An outline of this key scheduling method is provided in Fig. 6, since the DES algorithm consists of 16 rounds, the skew

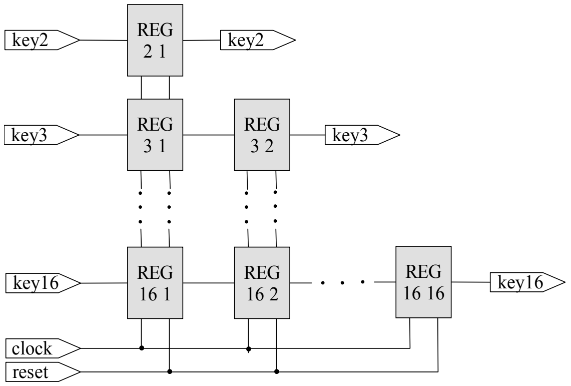


Fig. 5. Register arrays.

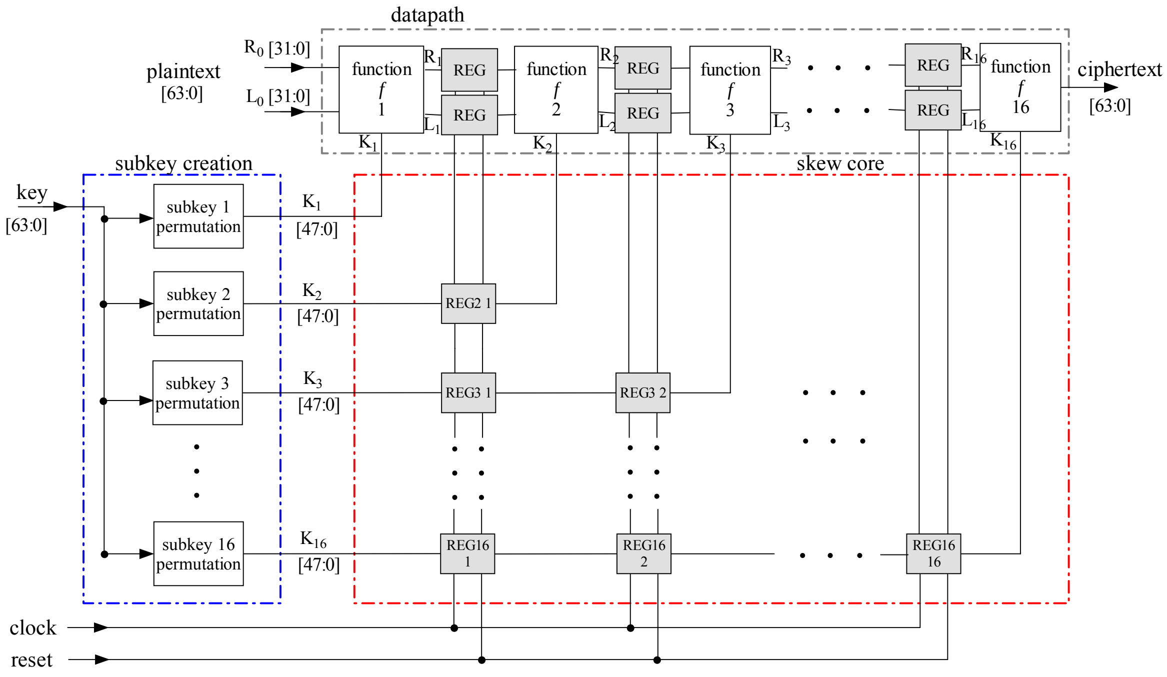
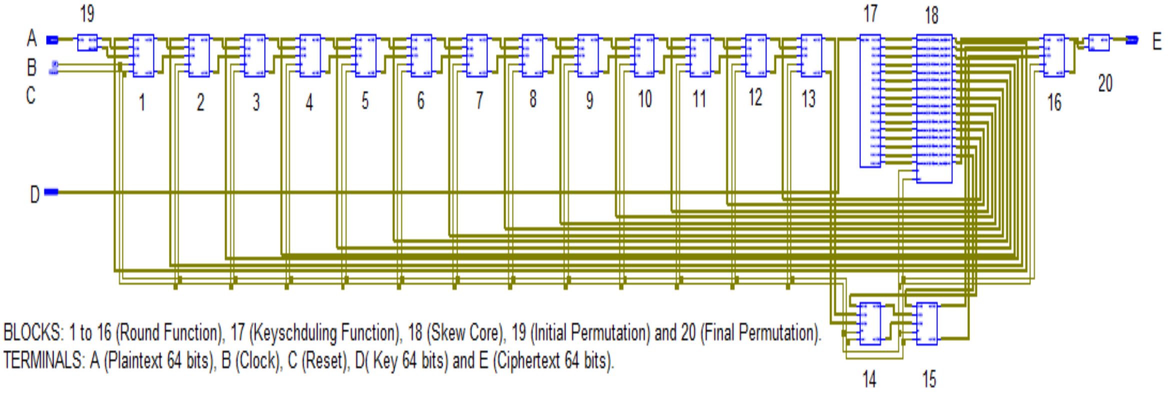


Fig. 6. Skew core Key-scheduling in DES design.

core is set to loop 15 times since a register is not required to delay the first sub-key. The number of resister in each sub-key path same as plaintext passes the no. of register before reaching respective round as shown in Fig. 6. It is noticeable that design of these registers same as used in Round blocks.

5. IMPLEMENTATION SUMMARY

FPGA implementation of DES algorithm was accomplished on a Spartan-3e device XC3s500e-4bg320 using Xilinx Foundation Series F 9.2i as synthesis and Modelsim 6.3f as simulation tool. The design was coded using VHDL language. It occupied 2814 (60%) CLB slices, 1704 (18%) slice Flip Flops and 186 (80%) I/Os. The design achieves a frequency of 111.882 MHz. It takes 16 clock cycles latency first time only then encrypts one data block (64-bits) per clock cycle. Therefore, the achieved throughput is (111.882 ×64) = 7.160 Gbits/s. full design schematic and simulation window shown in fig. (15).



BLOCKS: 1 to 16 (Round Function), 17 (Keyschduling Function), 18 (Skew Core), 19 (Initial Permutation) and 20 (Final Permutation).
TERMINALS: A (Plaintext 64 bits), B (Clock), C (Reset), D(Key 64 bits) and E (Ciphertext 64 bits).

Fig. 7. Full DES design schimatic genrated by Xilinx ISE tool.

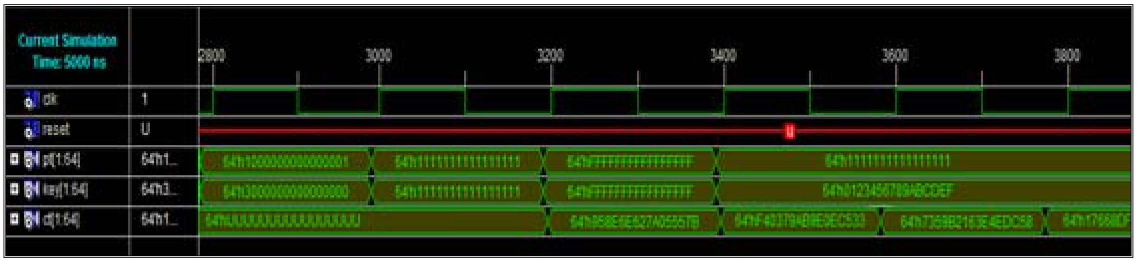


Fig. 8. Simulation Window of DES design.

6. PERFORMANCE COMPARISON

Table 1 shows the performance figures for some representative DES hardware implementations. Notice that the achieved results are competitive with the existing implementations.

A VLSI implementation of DES on static 0.6 micron CMOS technology at [7] is the fastest implementation of DES reported in the literature. Using a pipeline approach, the encryption can be performed at the rate of ≥ 6.7Gbs. Several FPGA implementations of DES have been reported in the literature achieving throughput ranges from 26 to 10752 Mbits/s using different design strategies. A DES implementation at [5] is a free DES cores which uses pipeline approach in ECB mode and achieves a data rate of 3052 Mbits/s. A java-based (Jbits) DES implementation at [8] achieves the fastest encryption rate of 10752 Mbits/s. DES implementation at [6] implements both 2-stage and 4-stage pipeline approaches obtaining throughput of 183.8 Mbits/s and 402.7 Mbits/s respectively. Almost all FPGA architectures for DES implement use partially or fully pipeline approaches.

Now compare our Design with various claims of DES implementation based on pipeline approach in

Table 1 Performance comparison.

Manufacture	Device Used	CLB slices	System clock(MHz)	Data rate(Mbit/s)	
Wong et al. (3)	XC4020E	438	10	26.7	Non-pipeline, One round Design
Bilam (9) (software)	Alpha 8400	----	300	127	16 stage Pipeline Designs
Kaps and Paar (4)	XC4028EX	741	25.18	402.7	
Free-DES(5)	XCV400	5263	47.7	3052	
McLoone, McCanny(6)	XCV1000	6446	59.5	3808	
Sandia Laboratories(7)	ASIC	----	----	9280	
Patterson(Jbits)(8)	XCV150	1584	168	10752	
Proposed Design	**XC3S500E**	**2814**	**111.882**	**7160**	

ECB mode shown in Table 1 with name of "Proposed Design", we find that only one claim, A java-based (Jbits) DES implementation [8] is above our design with encryption rate of 10752 Mbits/s, which is fastest FPGA design up to now. However, in this design the key schedule is computed in software and can only support one key per data transfer session. Therefore, performance of the presented DES design is ranked second in over all designs and one of the fastest single-chip FPGA designs.

7. CONCLUSION

This paper describes the design of a high performance silicon intellectual property core for the DES encryption algorithm. A 16-stage pipelined DES algorithm design is presented. This involved the instantiation of the function *f* block 16 times. Data blocks can be loaded every clock cycle and after an initial delay of 16 clock cycles the corresponding encrypted/decrypted 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. At a clock frequency of 111.882 MHz, the 16-stage pipelined design can encrypt or decrypt data blocks at a rate of 7.16 Gbit/sec and should prove very useful in applications where speed is vital as with real-time communications such as satellite communications and electronic financial transactions etc.

8. REFERENCES

[1] Data encryption standard (DES)," National Bureau of Standards (U.S.), Federal Information Processing Standards Publication 46**,** National Technical Information Service, Springfield, VA, Apr. 1977.

[2] http://www.tutorial-reports.com /computerscience/fpga/ overview.php.

[3] Wong, K., Wark, M., Dawson, E.: A Single-Chip FPGA Implementation of the Data Encryption Standard (des) Algorithm. In: IEEE Globecom Communication Conf., Sydney, Australia (1998) 827–832.

[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] Patterson, C.: High Performance DES Encryption in Virtex FPGAs using Jbits. In: Field-programmable custom computing machines, FCCM' 00, Napa Valley, CA, USA,

IEEE Computer. Soc., CA, USA, 2000 (2000) 113–12.

[9] 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. 260–271.

Vishwanath patel received the B.E. degree in electronics & communication engineering from G.E.C. Sagar (M.P.), in 2006. Currently pursuing M-Tech degree in “Solid state Devices and VLSI Technology” at Department of Electronics and Computer Engineering of Indian Institute of Technology Roorkee, India. His research interests are in Low power and High speed digital VLSI circuit.

Dr. Ramesh C. Joshi received the B.E. degree in electrical engineering from Allahabad University in 1967, M.E. and Ph.D. degrees in Electronics and Computer Engineering from University of Roorkee (Now, IIT Roorkee) in 1970 and 1980, respectively. He is currently working as professor in Department of Electronics and Computer Engineering at Indian Institute of Technology Roorkee, India. He has received a Gold Medal by Institute of Engineers in 1978 for best research paper. He has published about 150 research papers in National/International Journal/Conferences and delivered about 20 special lectures in various US and Indian Universities and Organizations. His main research interests are in Parallel & Distributed Processing, Databases and VLSI Design.

Dr. Ashok K. Saxena received M.Sc. from Agra University in 1969, M.Sc.(Tech.) from Department of Electronics and Electrical Engg. from B.I.T.S. in 1971, M.Engg. and Ph.D. from Department of Electronics and Electrical Engineering, UMIST/Sheffield University (UK) in 1975 and 1978, Served CEERI, a sister laboratory of CSIR during 1972-74 working on semiconductor device technology. He is currently working as professor in Department of Electronics and Computer Engineering at Indian Institute of Technology Roorkee, India. He is a fellow of IEL (UK) and Inst. of Phy. (London). He is a senior member of IEEE (USA) and Overseas Advisory Board of IEICE Transactions of Electronics of Japan. He is also a Fellow, Honorary Editor and Member Editorial Board of IETE. The discovery of a level in GaAlAs is christened as ’Saxena’s Deep Donor’ by Philips Research Laboratory, Eindhoven (Netherlands). He is also a winner of INSA Young Scientist, Roorkee University Khosla Award Gold Medal, Kothari Scientific Research Institute Award and S. K. Mitra Memorial Awards (twice) of IETE. He is also a member of Research Board of Advisors of ABI (USA) and has also been honored with the title of ’Man of the Year’ by ABI (USA) and IBC (UK). He has published about 175 research papers in international journals and conference proceedings with very high citation index. He has also written AICTE sponsored nine volumes on the related subjects for working professionals. His main research interests are in III-V Semiconductor Materials & Devices for High frequency/ Optoelectronic Applications and VLSI Technology.