Indonesian Journal of Electrical Engineering and Computer Science
Vol. 4, No. 2, November 2016, pp. 412 ~ 423
DOI: 10.11591/ijeecs.v4.i2.pp412-423

Architecture of ASIP Crypto-Processor for Dynamic Runtime Security Applications

Mahaba Saad[1]*, Khalid Youssef [2], Mohamed Tarek[3], Hala Abdel-Kader[4]
[1,3,4]Dept. of Electrical Engineering, Shoubra Faculty of Engineering, Benha University, Cairo, Egypt
[2]Dept. of Communications, Institute of Aviation Engineering &Technology, Aviation, Academy, Giza, Egypt
*Corresponding author, email: eng_mahaba@yahoo.com

Abstract

Nowadays, demands of data security are increasing, especially after introduction of wireless communications to the masses. Cryptographic algorithms are mainly used to obtain confidentiality and integrity of data in communication. Cryptography is usually referred to as "the study of secret". Encryption is the process of converting normal text to unreadable form. There are a variety of encryption algorithms have been developed. This paper provides quantitative analysis and comparison of some symmetric key cryptographic ciphers (DES, 3DES, AES, Blowfish, RC5, and RC6). The quantitative analysis approach is a step towards optimizing the security operations for an efficient next generation family of network processors with enhanced speed and power performance. A framework will be proposed as a reference model for quantitative analysis of security algorithm mathematical and logical operations. This paper also provides a dynamic crypto processor used for selected symmetric key cryptographic ciphers and provides an implementation of 16bit cryptographic processor that performs logical operations and arithmetic operations like rotate shift left, modular addition 2^16, S_box operation, and key expansion operation on spartan6 lower power, xc6slx150L-1lfgg676 FPGA. Simulation results show that developed processor working with high Speed, low power, and low delay time.

Keywords*: cryptographic, quantitative analysis, reference model*

1. Introduction

Cryptography is one of the most critical and necessary element of every network infrastructure and communication. It is an important research topic due to the explosive growth in data communications and Internet services. There are five goals of cryptography; confidentiality, authentication, data integrity, non-repudiation, and service reliability and availability. It can be divided into two families: Asymmetric key cryptography; the data is encrypted with the public key and decrypted with private key. Symmetric key cryptography; encrypt and decrypt data by using a single key. These are based on a mathematical function to encrypt a plain-text message and to produce cipher message. In this Paper a quantitative analysis will be conducted on the Symmetric key algorithms (DES [1, 2], 3DES [2], AES [2, 3], Blowfish [5, 7], RC5 [8, 9], and RC6 [10]) to abstract the algorithm operations in terms of basic operations that could be directly implemented on arithmetic and logic operator according to current network processor architectures. The Complete description of these selected algorithms isn't the scope of this paper. The quantitative analysis approach is a step towards optimizing the security operations for an efficient next generation family of network processors with enhanced speed and power performance. In section 3, a framework will be proposed as a reference model for quantitative analysis of security algorithm mathematical and logical operations. Cryptographic processors are becoming the standard way to enforce data-usage policies. In order to implement crpto processor used for all algorithms we started from quantitative analysis to know comman security operations and used composite operations for designing the processor on spartan6 lower power, xc6slx150L-1lfgg676 FPGA. XOR ($\oplus$) operator and rotate shift left are the most importantly used for all cryptographic algorithms, which there are basic operation. Addition modulo 2^{32} (mod 2^{32}) and multiplication Modulo 2^{32} operations are also used for some cryptographic algorithms.

Received September 30, 2016; Revised October 20, 2016; Accepted October 30, 2016

2. Review of Literature

Once security cryptographic algorithms are designed, it is important to implement these algorithms in software to run on a general purpose processor (GPP). Such an implementation, however, cannot provide the desired performance at the desired power and computing resources. ASIC is used to solve this problem, but to eliminate the inflexibility of ASIC, one could implement these algorithms in FPGA that can be reprogrammed. FPGA implementation is expensive and the same level of difficulty to program as an ASIC. Another implementation is to design a special programmable processor ASIP that is optimized to execute cryptographic algorithms. Due to their flexibility and high throughput, reconfigurable cryptographic processors are alternatives for the implementation of cryptographic algorithms. Cryptographic processors are special type of network processors. So, developing a hardware architecture that provides efficient, high speed at low area and high throughput implementations for crypto-systems has become increasingly important. First, many researches tries to achieve high throughput on cryptographic applications by new instructions introduced to support one or more cryptographic algorithms like Intel, IBM, and Oracle. Second, other researches are a design of 32bit cryptographic processor. In [12] the authors implement 32 bit pipelined processor on FPGA using Verilog, to perform logical and arithmetic operations like rotate word, modular addition modular multiplication, matrix multiplication, fixed coefficient multiplier ,mix column transform using binary extension field operations (2^m) for arbitrary irreducible Polynomial. Simulation results showed that increase overall performance of the speed with low area and low propagation delay. It was concluded in [13] that 32-bit RICS processor introduces modular addition, Polynomial matrix multiplication, Reduction Modulo Multiplication, and Rotate word to increase overall performance of the speed with low area and low power consumption. In [14] describes the design of high performance MIPS Cryptography processor based on 3DES. There are 3 new 32-bit instructions LKLW, LKUW and CRYPT in order to increase the processor functionality and performance. In [15] the authors design to perform addition, multiplication, shift operations, matrix multiplications, fixed coefficient multiplier, mix column transformation, multiplier X (2X+1), and circular shift operations for arbitrary irreducible polynomial. The main objective of these works is to reduce the power consumption, increase the speed of operation and reduce the programming length.

3. Security Operation Reference Model (SORM) As a Framework for Quantitative Analysis of Algorithms

Any security operation can be described by the reference model, it may classify as a composite or primitive (basic /atomic). The primitive is a simple operation which can be implemented by logic gates, and there are two kinds of atomic operations; combinational logic and sequential logic. Composite operation is computations performed by an algorithm. It can be mapped into one or more basic operation. This operation can be memory or memory less, reversible or irreversible, data oriented tabular or not, event based or step based and decomposable or not decomposable as shown in Figure 1. Projection of reference model on cryptographic selected security algorithm operations is listed below.

Reference Model of DES Operations, XOR operation can be classified as primitive, memory less, reversible, and combinational. Rotation Shift Left operation can be classified as atomic, memory (registers of 32bit), clock based, reversible, and sequential. Rotation shift left by two bite is composite operation, memory (registers of 32bit), and clock based, reversible, and decomposable. It can be decomposed into more atomic operation like two register for input and output, rotate shift left for shifting one bit and after another clock cycle make second shift. Permutation operation is composite, data oriented table base, memory less, and decomposable. It can be decomposed into more primitive operation like one table, two register for input and output, and one combination. S-Box operation is composite, data oriented table base, memory less, and decomposable. It can be mapped into more atomic operation like 8 s-box table (s1 to s8 table), three register 48bit for input, 4bit for output from s-box table, and 32bit for final output, and 4 combinations, one for grouping of data 6* 8bit, two for selecting row and column, and one for generate final output from this operation. The all reference model of DES algorithm decomposed operations can be summarized in Table 1. The number of each operation used in this algorithm will be found in the following table.

Reference Model of 3DES Operations, there are 3 different modes in 3DES; in DES-EEE3 mode (encyrpte plainedtext with k1 followed by another encyrpte with K2 followed by another encyrpte with k3), these are exactly the same as regular DES, but it is repeated three times. The all reference model of 3DES algorithm decomposed operations is the same as DES but number of operations in 3DES is multiply by 3 of DES operations.

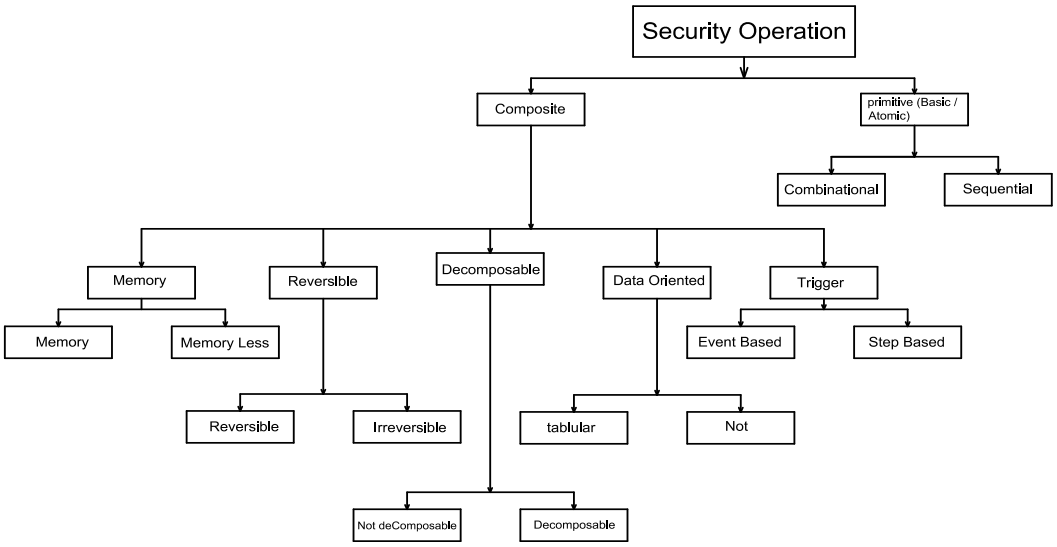

Figure 1. Reference Model of Proposed Security Operation

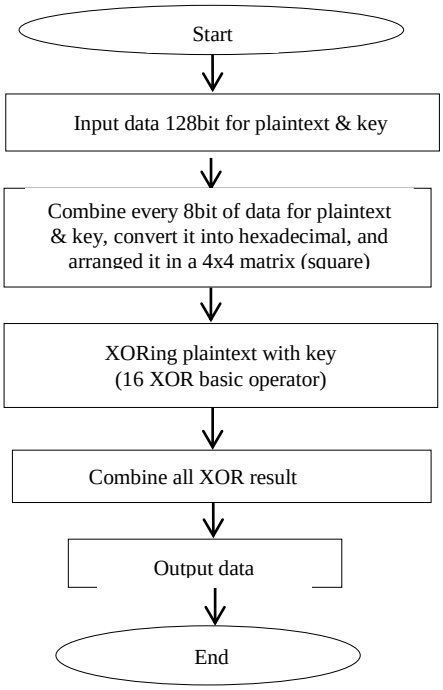

Figure 2. Add Round Key Operation Flowchart

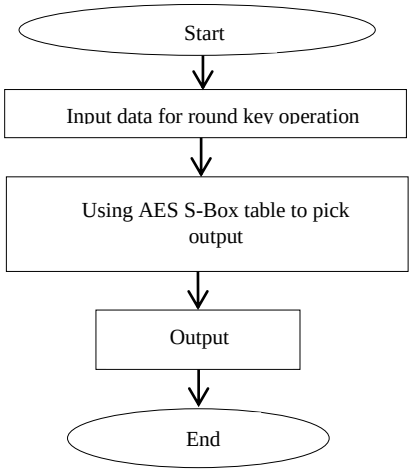

Figure 3. Sub Byte Operation Flowchart

Table 1. Reference Model of All DES Algorithm Operations

Operation	No. of operation in DES	Primitive/Basic/ Atomic	Composite
XOR (⊕)	64	atomic, memory less, reversible, and combinational	
Rotate left shift	4 left shift by one bit	atomic, memory(registers of 32bit) , clock based, reversible, and sequential	
Rotate left shift	12 left shift by two bit		Composite, memory (registers of 32bit), clock based, reversible, and decomposable.
Permutation	16for 32bit input & output , 1 for 64bit input x56bit output, 1 for 64bit input & output, 16 for 56bit input x48bit output, 16 for 32bit input x48bit output, 16 for 64bit input & output		Composite, data oriented table base, memory less, and decomposable.
DES S-Box	16		Composite, data oriented table base, memory less, and decomposable.

Table 2. Reference Model of All AES Algorithm Operations

Operation	No. of operation in AES (10 round)	Basic	Composite
XOR (⊕)	16 per one round for Add key operation, 48 per one round for Mix Column operation, and 20 per one round for Key Expansion operation	Primitive, memory less, reversible, and combinational	
Rotate left shift (Byte shifter)	1 left shift by 1byte, 1 left shift by 2byte, 1 left shift by 3byte		Composite, not data oriented, memory, and decomposable.
Add Round Key Operation	11		Composite, not data oriented table base, memory less, and decomposable
Sub Byte Operation	10		Composite, data oriented table base, memory less, and decomposable
Shift Row Operation	10		Composite, not data oriented, memory, and decomposable.
Mix Column Operation	9		Composite, data oriented table base, memory less, and decomposable.
Key Expansion Operation	10		Composite, data oriented table base, memory, and decomposable.

Table 3. Reference Model of All Blowfish Algorithm Operations

Operation	No. of operation in Blowfish	Basic	Composite
XOR (⊕)	50 for data encryption operation, 68 for key expansion operation	atomic, memory less, reversible, and combinational	
data encryption	1		Composite, memory (registers of 32bit), data oriented table base, and decomposable.
key expansion	1		Composite, memory (registers of 32bit), data oriented table base, and decomposable.

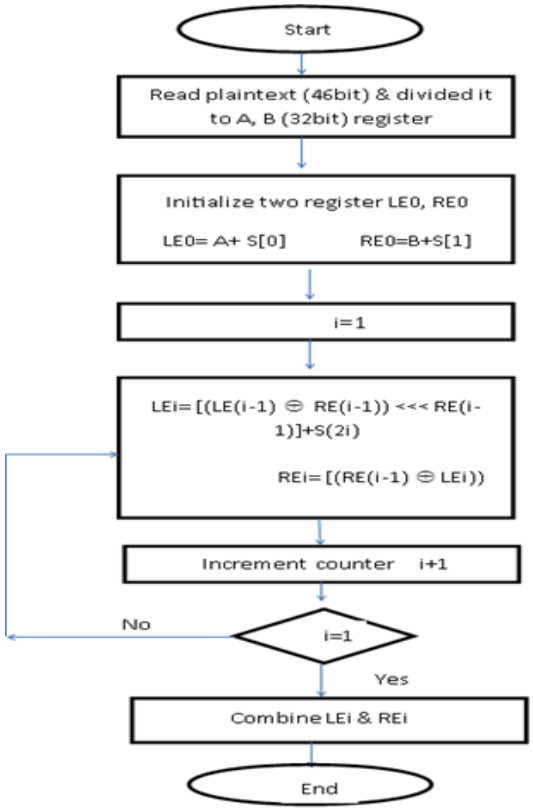

Figure 4. Data Encryption Operation Flowchart with RC5-32/12/16

Reference Model of AES Operations, add Round Key operation as shown in Figure 2 can be decomposed into more atomic operation like two register for input and output, XOR operator, and 2 combinations, one for grouping of data 16* 8bit to be converted to hexadecimal, and one for generate final output from this operation. Sub Byte operation is composite, it can be decomposed into more atomic operation like AES S-Box table, two register for input and output as shown in Figure 3 . Shift Row operation is composite, it can be decomposed into more atomic operation like 4 register for input and one for output, 5 combination for data byte, and 3 rotate left shift operation (byte shifter not bit shifter) are be listed as one 1byte shifter, 2byte shifter, and 3byte shifter. Mix Column operation is so complex operation. It is composite. It can be decomposed into more atomic operation like two tables L and E, registers for input and output, 5 combination, for multiplication operation (16 total multiplications) there are 48 XOR, 64 mathematical addition, and subtraction for hexadecimal value. Key Expansion operation is composite, It can be decomposed into more atomic operation like circular shift, AES S-Box, registers for input and output, 2 combination, 20 XOR operator. The all reference model of AES algorithm decomposed operations can be summarized in Table 2.

Reference Model of Blowfish Operations, data encryption operation is composite, data oriented table base, memory, and decomposable. It can be mapped into more simple operations like 4 S-array (4 byte), registers 32bit for input and output, 1 combination for output cipher text, addition mod 2^{32}, subtraction and 50 XOR operations for generate final output from this operation. Key expansion operation is composite, data oriented table base, memory, and decomposable. It looks like data encryption operation, it can be mapped into more simple operations like registers 32 bit for input and output, 4 S-array (4 byte), addition mod 2^{32},subtraction and 68 XOR operation (18 XORing key with P-array + 50 XOR for data encryption operation) for generate final output from this operation. The reference model of Blowfish algorithm decomposed operations can be summarized in Table 3.

Reference Model of RC5 Operations, Data encryption operation is composite, not data oriented table base, memory, and decomposable. It can be mapped into more simple operations like 32bit registers for input and output, 1 combination for output cipher text, addition and

subtraction mod 2^{32}, 24 XOR operations, and 24 rotate shift left by different value of LE and RE as shown in Figure 4. Key expansion operation is composite, not data oriented table base, memory, and decomposable. It can be mapped into more simple operations like addition and subtraction mod 2^{32},registers for input and output, 78(3t=78) rotate shift left by 3 and 78 rotate shift left by unknown variable of X and Y as shown in Figure 5. The reference model of RC5 algorithm decomposed operations can be summarized in Table 4.

Table 4. Reference Model of All RC5 Algorithm Operations

Operation	No. of operation in RC5	Basic	Composite
XOR (⊕)	24for data encryption operation	atomic, memory less, reversible, and combinational	
Rotate left shift by unknown variable	24 left shift by unknown variable of RE,LE for data encryption operation, 78 shift by 3 and 78 shift by unknown variable for key expansion operation		Composite, memory, clock based, reversible, and decomposable.
data encryption	1		Composite, memory (registers of 32bit), not data oriented table base, and decomposable.
key expansion	1		Composite, memory, not data oriented table base, and decomposable.

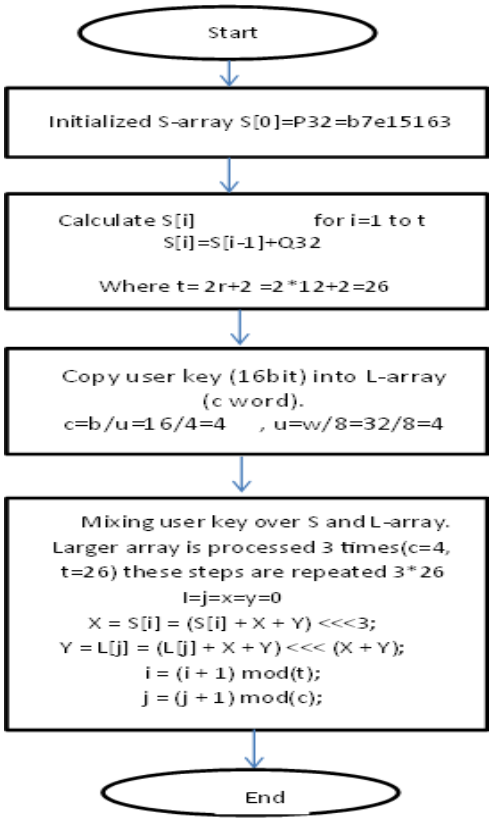

Figure 5. Key Expansion Operation Flowchart of RC5 Algorithm

Reference Model of RC6 Operations, Data encryption operation is composite. It can be mapped into more simple operations like 32bit registers for input and output, 1combination for output cipher text, addition, subtraction, and multiplication mod 2^{32}, 40 XOR operation, and 40

rotate shift left by 5 and 40 shifter by different calculated value of u and t. Key expansion operation is composite. It can be mapped into more simple operations like addition, subtraction, and multiplication mod 2^{32}, registers for input and output, 132 rotate shift left by 5 and 132 rotate shift left by unknown variable of A+B. The reference model of RC6 algorithm decomposed operations can be summarized in Table 5.

Table 5. Reference Model of All RC6 Algorithm Operations

Operation	No. of operation in RC6	Basic	Composite
XOR (⊕)	40for data encryption operation	atomic, memory less, reversible, and combinational	
Rotate left shift by unknown variable	40 left shift by unknown variable z, and 40 left shift by 5 for data encryption operation, 132shift by 3 and 132shift by unknown variable p= A+B for key expansion operation		Composite, memory, clock based, reversible, and decomposable.
data encryption	1		Composite, memory (registers of 32bit), not data oriented table base, and decomposable.
key expansion	1		Composite, memory, not data oriented table base, and decomposable.

4. Comparison between Common Operations used in Security Algorithms

A new comparative study between selected algorithms according to reference model of common operations used in these algorithms is listed in Table 6. The structure of DES, 3DES uses fewer types of atomic and composite operations rotate shift left, XOR, and eight 6bit-to-4bit S-boxes. Compared to AES uses byte shifting, XOR, one 8bit-to-8bit S-box, and field multiplication. Also, for RC5 and RC6 uses byte shifting by different value, XOR, addition mod 2^{32}, subtraction mod 2^{32}, multiplication mod 2^{32}.

5. Crypto-Processor Proposed Architecture

The architecture of processor is shown in Figure 6. 16-bit processor has a complete instruction set, general purpose registers, control unit, decoder, S_box controller, and Arithmetical Logic Unit (ALU). ALU design performs the cryptographic operations like rotate shift left, modular addition 2^16, S_box operation for simplicity, and key expansion operation. For cryptographic processor, it was necessary to create dedicate instruction set.

Table 6. Comparison between DES, 3DES, AES, Blowfish, RC5, RC6

	DES					AES		Blowfish	RC5		RC6		
No. of round	16		3DES			10		16	12		20		
XOR	64		192			84		118	24		40		
S-Box	16		48			10		-	-		-		
Rotate shift left	1bit	2bit	1bit	2bit	1byte	2byte	3byte	-	3	unknown variable	5	3	unknown variable
	4	12	12	36	1	1	1		78	102	40	132	132
Addition mod 2^{32}	-		-			-		64	26+6+3*78= 286		84+43+3*132= 523		
Subtract from 2^{32}	-		-			-		≤ 16	≤ 12		≤ 20		
Multiply mod 2^{32}	-		-			-		-	-		40		

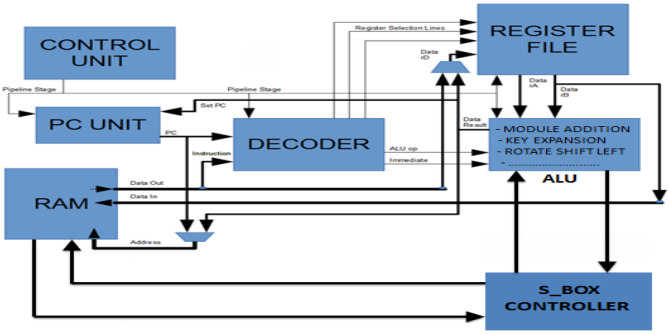

Figure 6. Cryptographic Processor Architecture

5.1. General Purpose Registers (GPR's)

GPR store and save operands and result during program execution so a set of eight 16-bit registers are used. Two registers are the operands to ALU which performs the operation. So, ALU and memories must be able to write/read those registers. Figure 7 represents block diagram of GPR.

5.2. Control Unit

This unit is responsible for telling other units what to do, and when. The control unit is technically a state machine, the main operations performed by it are reset, synchronize everything up and drive the enable bits, instruction fetch, load, address setup, operand fetch and store the result. It will be in ideal state when no operations are performed.

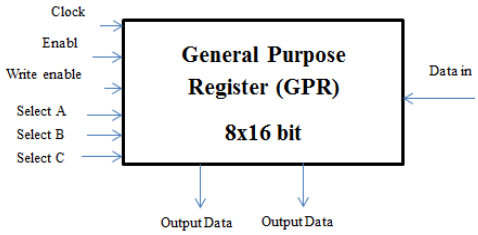

Figure 7. Block Diagram of GPR

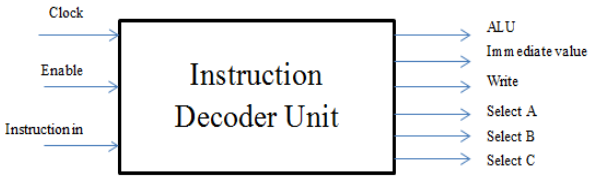

Figure 8. Instruction Decoder Unit Block Diagram

5.3. Decoder Unit

This unit used to statically pull bits out from the instruction stream and forward them on to the other units we connect to: The Register file, ALU, and Control unit. As shown in Figure 8, decoder unit requires a clock, an enable bit, and the instruction it's to decode. The output is all of the selection lines for the register file (A, B and C), the ALU operation (ALU op) to be performed, any immediate value from the instruction and whether the instruction performs a register write.

5.4. Arthimatic Logic unit (ALU)

ALU should take the opcode given by the decoder, along with input data read from the register file, and output a result that can then be written to the register file. It has 16 operations; each one of them was created and converted into a symbol. New operations are developed to perform composite operations like s_box operation, module addition, rotate left shift and key expansion. Figure 9 showed RTL schematic and Figure 10 showed simulation result.

5.5. Program Counter Unit (PC)

PC is just a register (16 bit) containing the location of the currently executing instruction. PC unit will create to manage this. PC unit will obviously hold the current PC, and on command increment it. It will have an input for setting the next PC value, the ability to stop stay at the same location which needed due to pipeline being several cycles long and also set PC to reset vector, which is 0x0000. So, a 2-bit select input can be used to select one of these operations.

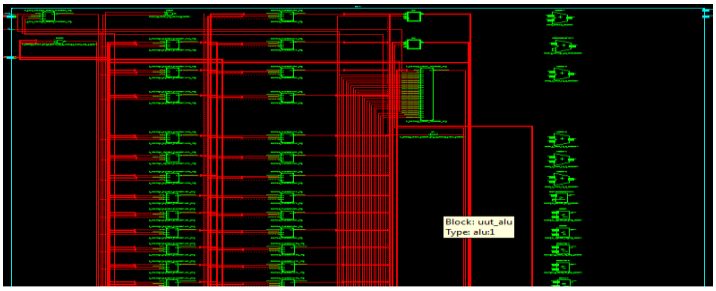

Figure 9. RTL Schematic of ALU

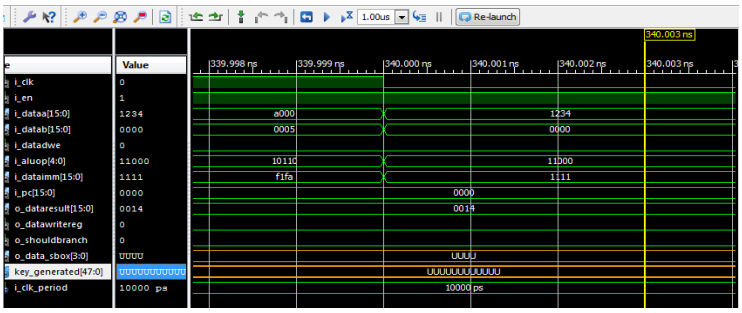

Figure 10. Simulation Result of ALU

Table 7. Implementation Results on xc6slx150L-1lfgg676 FPGA

Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	284	184,304	1%
Number used as Flip Flops	284		
Number of Slice LUTs	595	92,152	1%
Number used as logic	563	92,152	1%
Number used as Memory	28	21,680	1%
Number of occupied Slices	217	23,038	1%
Nummber of MUXCYs used	216	46,076	1%
Number of fully used LUT-FF pairs	188	657	28%
Number of slice register sites lost to control set restrictions	84	184,304	1%
Number of bonded IOBs	58	498	11%

6. Simulation Result of New Instructions Set Of Proposed Architecture

16 bit processor was described using ISE 14.1tool .This tool including the simulator, synthesizer and programming; after synthesized the hardware resource consumption for the complete processor implemented in a Xilinx spartan6 lower power, xc6slx150L-1lfgg676 FPGA is shown in Table 7. Simulation result of four new security composite operations is listed below

6.1. Module Addition 2^{16}Operation

Module addition 2^32 is used in blowfish, RC5, and RC5 algorithms. The novelty is in breaking the original 32-bit input plain text into 16-bit data set and performs module addition 2^16. Figure 11 represents simulation result. A simple description of mod 2^16 addition in terms of:

$$\text{integer addition: } x + y \bmod 2^{16} = \begin{cases} x + y & \text{if there was no a carry out} \\ x + y - 2^{16} & \text{if there was a carry out} \end{cases} \quad (1)$$

6.2 S_box Composite Operation

S-box operation provides an extra layer of security. It is usually the most important task while designing DES algorithm. S-box is a lookup table, using six bits as input and four bits as output. The 6 input bits are split into two groups: the middle four bits indicate the column of the S-box and the two bits on both sides indicate the row of it. Figure 12 represents simulation result.

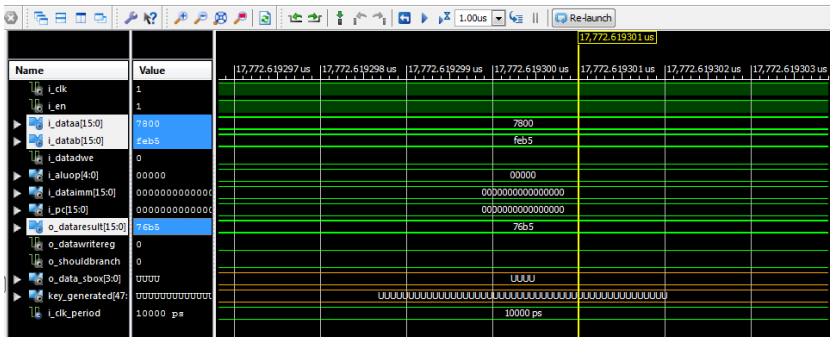

Figure 11. Simulation Result of mod addition 2^16

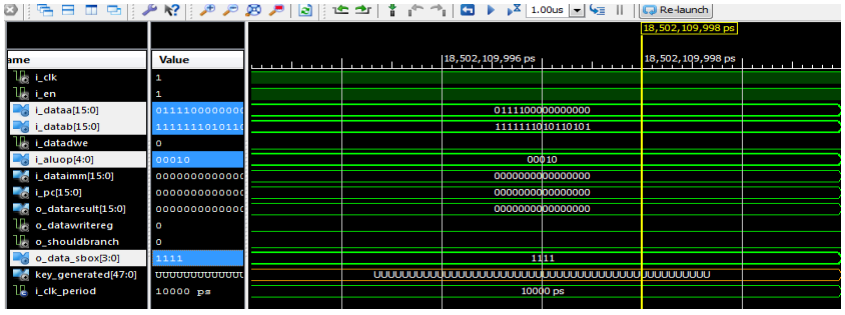

Figure 12. Simulation Result of S_box Composite Operation

6.3. Rotate Shift Left Operation

It is used in many of the sub-key generators for cryptographic algorithms used to left rotate 16bit of data. Figure 13 represents simulation result.

6.4. Key Expansion Composite Operation

Input key is expanded to generate sub keys before data encryption in all selected algorithms. This part is used as separated unit in the algorithm steps, so the novelty is used this as an opcode executed by ALU. Figure 14 represents simulation result.

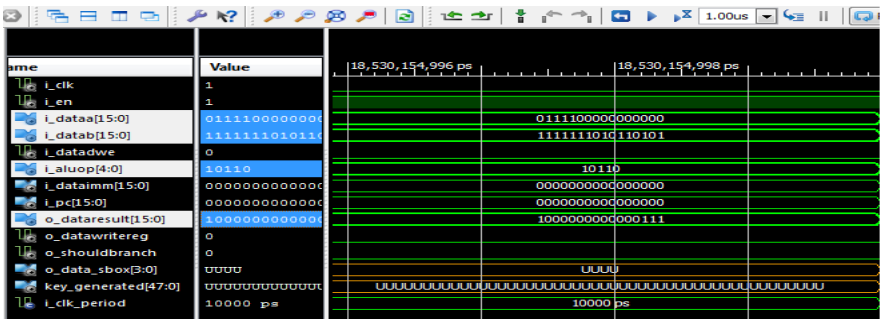

Figure 13. Simulation Result of Rotate Shift Left Composite Operation

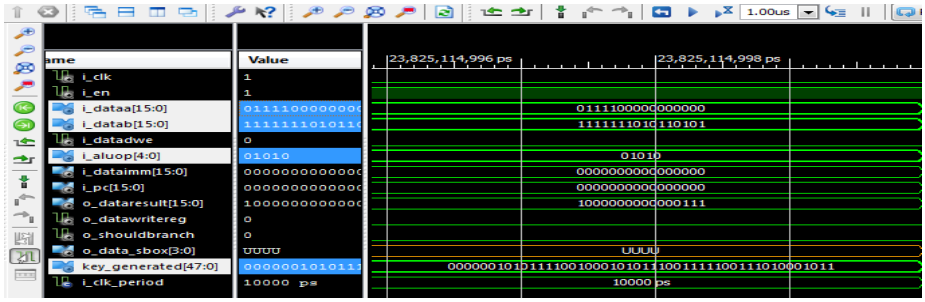

Figure 14. Simulation Result of Key Expansion Operation

7. Timing Analysis

After running the Implement Design process and Synthesize process, Timing Analyzer can be used to perform a detailed time analysis for crypto processor. Table 8 represents these values for crypto processor design.

Table 8. Time Analysis

Maximal frequancy	125.019MHZ
Minimum Input Arrival Time Before Clock	7.512 ns
Maximum Output Required Time After Clock	6.106ns

8. Conclusion

This paper proposes new comparative study between selected algorithms based on a completely defined reference model and framework for security algorithms operations that explained and defined mathematically a framework for common operations used in security algorithms that is better to be interpreted in a new instruction set architecture for next generation network processors. The study held included the following algorithms: DES, 3DES, AES, Blowfish, RC5 and RC6 according to reference model of quantitative analysis for composite and primitive operations. The second part of this paper proposes architecture for 16bit ASIP network processor based on new instruction set architecture for next generation network processors as a dynamic processor than can be perform security operations used in selected symmetric key algorithms, has been designed using VHDL. This 16-bit processor introduces the cryptographic instructions like Modular addition, S_box operation, Key expansion operation, and Rotate left shift to increase overall performance and speed with low area and low power consumption by reducing the number of instruction cycles of executing the algorithm.

References

[1] JO Grabbe. The DES Algorithm Illustrated. http://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm

[2] William Stallings. Cryptography and Network Security: Principles and Practice. Sixth Edition. United States of America: Pearson. 2014.

[3] Douglas Selent. Advanced Encryption Standard. *Rivier Academic Journal.* 2010; 6(2): 1-14.

[4] PG, SM Ankita Verma. Comparative Study of Different Cryptographic Algorithms. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*. 2016; 5(2): 58-63.

[5] KV Saikumar Manku. Blowfish Encryption Algorithm for Information Security. *ARPN Journal of Engineering and Applied Sciences. 2006-2015 Asian Research Publishing Network (ARPN)*. 2015; 10(10):4717-4719.

[6] TS Atia. Development of a New Algorithm for Key and S-Box Generation in Blowfish Algorithm. *Journal of Engineering Science and Technology School of Engineering, Taylor's University*. 2014; 9 (4): 432-442.
[7] PA Tanjyot Aurora. Blowfish Algorithm. *International Journal of Computer Science and Communication Engineering IJCSCE Special issue on "Recent Advances in Engineering & Technology" NCRAET.* 2013; 3(4): 238-243.
[8] SP Vishnu N. Implementation of RC5 Symmetric Key Encryption Algorithm for Secure Communication. *IJRIT International Journal of Research in Information Technology*. 2013; 1(3): 14-17.
[9] HAS Mowafak Hasan. Modified Cryptanalysis of RC5. *The International Arab Journal of Information Technology*. 2006; 3(4): 299-302.
[10] SS Vikas Tyagi. Enhancement of Rc6 (Rc6_En) Block Cipher Algorithm and Comparison with Rc5 & Rc6. *Journal of Global Research in Computer Science*. 2012; 3(4).
[11] M Sumathi, D Nirmala, R Immanuel Rajkumar. Study of Data Security Algorithms using Verilog HDL. *International Journal of Electrical and Computer Engineering (IJECE).* 2015; 5(5): 1092-1101.
[12] M Lalitha Sowmya, B Divya, S Jagadeesh. Design of Custom Instructions in Cryptographic Processor. *International Journal of Engineering Research and Applications (IJERA)*. 2012; 2(5): 1718-1724.
[13] Fathima Shireen Syed Musthak Ahmed G. Krishnamurthy. RISC Based Architecture for Customized Cryptographic Instructions. *International Journal of Engineering Science Invention*. 2013; 2(2).
[14] Kirat Pal Singh Shivani Parmar. Design of High Performance MIPS Cryptography Processor Based on T-DES Algorithm. *International Journal of Engineering Research & Technology (IJERT)*. 2012; (3).
[15] M Praveen Kumar, R Ashwitha, N Jyosna, M Pavani. Design of Cryptographic Processor for Security Algorithm Operations. *International Journal of Engineering Research & Technology (IJERT).* 2013; 2(4).