Analysis of 64- bit RC5 Encryption Algorithm for Pipelined Architecture

Ashmi Singh
NRI Institute of Information Science & Technology bpl

Puran Gour
NRI Institute of Information Science & Technology bpl

Braj Bihari Soni
NRI Institute of Information Science & Technology bpl

ABSTRACT
In modern days data transmission through a channel requires more security. Security based more important transmission is comparatively better & believable than simple transmission. The aim of this work to use RC5 algorithm for encryption and decryption of data for secure data transmission from one place to another place for proper communication purposes. Today this is utmost importance to send information confidentially through network without any risk for hackers or unauthorized possibility to access from the network. This urgently require security implementation devices in network for well secured transmission of data. Symmetric encryption cores provide data protection through the use of secret key only known as encryption, whereas decryption deals with the yield at the end of communication path. Today world require secure transmission through cryptographic algorithm. Keeping view in mind the proposed well defined RC5 architecture have been taken, based on the fact for suitability of each operation for encryption, high speed processing and possibility of area reduction. The work results of the study clearly indicate that logic implementation by this hardware is maximum clock frequency of 179 MHz and areas reduced to 50% as compare with the results of design of previous worker. The propose design is described in verilog, synthesized by Xilinx synthesis technology.

Keywords
RC5, Encryption, pipeline**,** Verilog HDL

1. INTRODUCTION
Modern age requires secured and believable data transmission through network. The fast growth of wireless communication technology extend possibilities for search of new type of data services which must be free from different threats, cheating and attack. So, the utmost importance is to secure transmission of datas. The primary method used for protecting valuable data through encryption. Many encryption schemes constitute the area of cryptography. Cryptography device is growing rapidly through adoption of computer technology [5]. The invention of sophisticated and high level computer broaden the new dimension and direction for cryptography devices. The design for objective of cryptographic ciphers is still under investigation and is not well understood. There is lot of scope to maximize objective results through change of design or other means. Normally, in cryptography process, plain test is converted into cipher text at the transmitter side and again it is converted into plain text at the receiver side. The proposed pipeline designed architecture based on RC5 encryption algorithm which is used as symmetric block cipher which operates on W-bit wide data for r-rounds using b-bytes of key to encrypt the data. It is simple algorithm which uses round wise data dependent encryption techniques. A typical cryptograph had been presented by [7] in which proposed new hardware dedicated to RC5. In the proposed RC5 dedicated hardware, by introducing an architecture suitable for each operation used for the encryption, high speed processing, and area reduction can be realized. The high performance pipeline hardware proposed in [2] in which implementation of RC5 that improves the response time of the system with less resources and performance of this architecture provide high throughput. In [3] represents RC5 algorithm and its implementation for low power and in complex architecture. Our work aim to design objective based pipeline architecture specifically based for low delay with high frequency. The introduction & reconfigurable architecture of pipeline design has been already presented in the text. For synthesis proposed design the parameters were taken i.e. W=32, r=12, and b=16. The proposed design described in verilog, synthesized by Xilinx Synthesis Technology (XST). The specific manifold changes in design are already proposed in 1, 2, 5, 6, 7, 8, 9, 11, 12, 13, 14 & 15.

2. RELATED WORKS
Many existing hardware architecture have been published by many earlier workers which are designed based on RC5. Bevil et.al.(2012) designed pipeline based FPGA architecture for RC5 encryption by applying increasing throughput and area reduction can be realized.

Some authors have published work on alternative 12 stage pipelined architecture with reduced waiting and response time through sliding window architecture. Whereas, Elkeelany and Olabisi (2008) proposed a pipelined based architecture with special synchronization for the process of encryption which yielded the throughput ranging between 300 to 450 mb/sec. In, 2011 Yoshikawa and Sakaun introduced a dedicated hardware for the RC5. Our aim is to introduce 12 stage pipeline based architecture for RC5 by further increase the throughput.

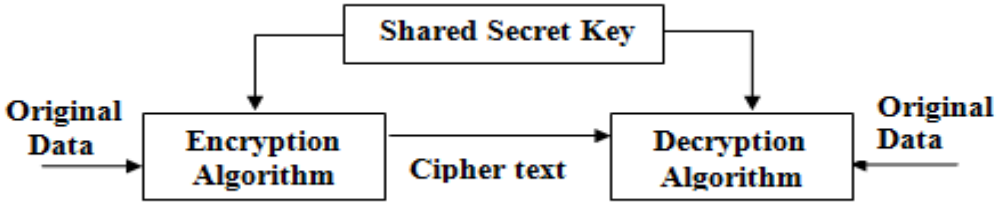

Fig 1: Block Diagram of Symmetric Encryption Process

3. DESCRIPTION RC5ALGORITHM

RC5 encryption algorithm was designed and proposed by Rivest of (MIT) and this is first appeared in 1994. RC5 analysis (RSA Laboratories) is still have great scope and is in progress and this can periodically updated to reflect any additional findings. RC5 is a symmetric block cipher which is used as the same secret cryptographic key for both encryption and decryption, a variable number of rounds, with a variable block size and a variable-length key it is a parameterized algorithm. This whole concept provides the opportunity for great flexibility in, performance characteristics and the level of security. [1,2,3,16 & 17].

RC5 algorithm got designated as RC5-w/r/b. The number of bits in a word w, is a parameter of RC5. Different choices of this parameter result in different RC5 algorithms. With a variable number of rounds, RC5 is iterative in structure. with second parameter of algorithm is r, is called number of rounds. A variable-length of secret key is used by RC5. The third parameter of RC5 is the key length b(in bytes). The all parameters are summarized as follows :

w: The word size, in bits. The standard value is 32bits and allowable values are 16, 32 and 64. To generate the ciphertext and plaintext blocks size are each 2w bits , RC5 encrypts two-word blocks.

r: The number of rounds. Allowable values of r are 0 to 255. Also, the expanded key array S contains T = 2 (r + 1) words.

b: The number of bytes in the user's key K. expected values of b are 0 to 255.

There are three components in RC5: key expansion algorithm, encryption algorithm, and decryption algorithm. These routines consist of three primitive operations (and their inverse): words addition , bitwise XOR, and data-dependent left rotation of cx by dx denoted by cx <<< dx. Note that only the log2(w) low order bits of y affect this rotation. So secret key K is expanded to fill a key array whose size depends on the number of rounds in Key expansion routine. The key array S is then used in encryption. The description of the encryption algorithm is given.

3.1 Key Expansion

In key-expansion algorithm expands the secrete key K to fill the key array S, so that S combine an array of t = 2(r + 1) , determined by K. It uses two word-size magic constants Pw and Qw defined for arbitrary w as shown below:

Pw = Odd ((e − 2)2^w)

Qw = Odd ((Ø− 1)2^w)

where

e = 2.71828 . . . (base of natural logarithms)

O= (1 +_5)/2 = 1.61803 . . . (golden ratio)

Odd(x) is equal to x.

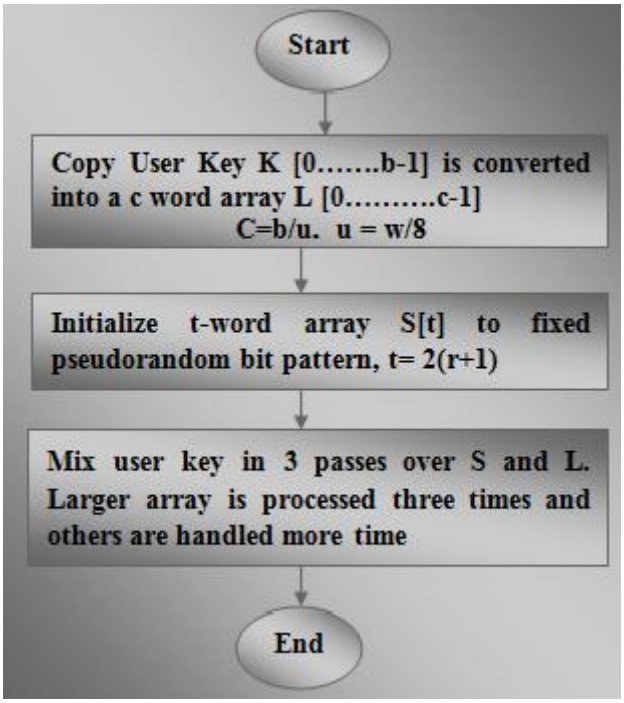

Fig 2: RC5 Key Expansion Process

3.1.1 First algorithmic step of key expansion:

This step is to copy user key K [0, 1 . . . b − 1] converted into c-word array L[0, 1, . . . , c − 1] of c = b/u words, here u = w/8 it is the bytes/word. This first step will be achieved by the following pseudo code operation:

for j = b – 1 down to 0 do L[j/u] = (L[j/u] <<< 8) + K[j]; where all bytes are unsigned and the array L is initially zeroes.

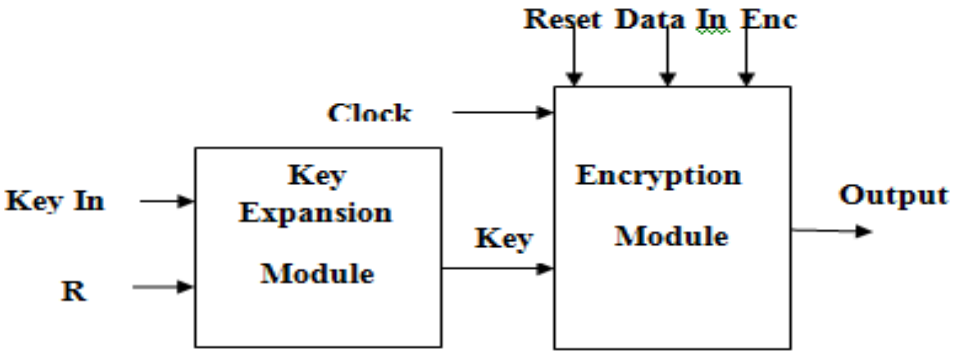

Fig 3: RC5 Encryption Block Diagram

3.1.2 Second algorithmic step of key expansion:

This step is to initialize array S to a particular fixed pseudo-random bit pattern T=2(r+1), using an modulo 2w determined by two magic constants Pw and Qw.

S[0] = Pw:

For j = 1 to t − 1 do

S[j] = S[j − 1] + Qw

3.1.3 Third algorithmic step of key expansion:

This step is to mix in the secret key in three passes over the arrays element L and S. Due to the potentially different sizes of L and S, the larger array is processed three times, and the other array will be handled more after.

u = v = 0;

G = H = 0;

do 3_ max (t, c) times:

G = S[u] = (S[u] + G + H) <<< 3

H = L[v] = (L[v] + G + H) <<< (G + H);

u = (u + 1) (mod t);

v = (v + 1) (mod c).

Note that with the key-expansion function it is not so easy to found K from S, due to the one-wayness.

3.2 Encryption:

The input block to RC5 consists of two w-bit words given in two registers, W0 and X0. In the registers L1 and R1 the o/p is placed there also. As explained in RC5 uses expended key array S[0, 1, . . . , T − 1], consisting of T = 2(r + 1) words. The From the user's secret key K the key-expansion algorithm initializes S. However, the S table in RC5 encryption is not like an S- box used by DES. The formulas is given in code of encryption algorithm as shown below:

W0 = G + S[0];

X0 = H + S[1];

for i = 1 to r do

Wi = ((Wi-1 XorXi-1) <<< Xi-1) + S[2i];

Xi = ((Xi-1 Xor Wi) <<< Wi) + S[2i + 1];

The output is in the registers W1 and X1. The RC5 encryption algorithm is illustrated as shown in Figures 3, respectively.

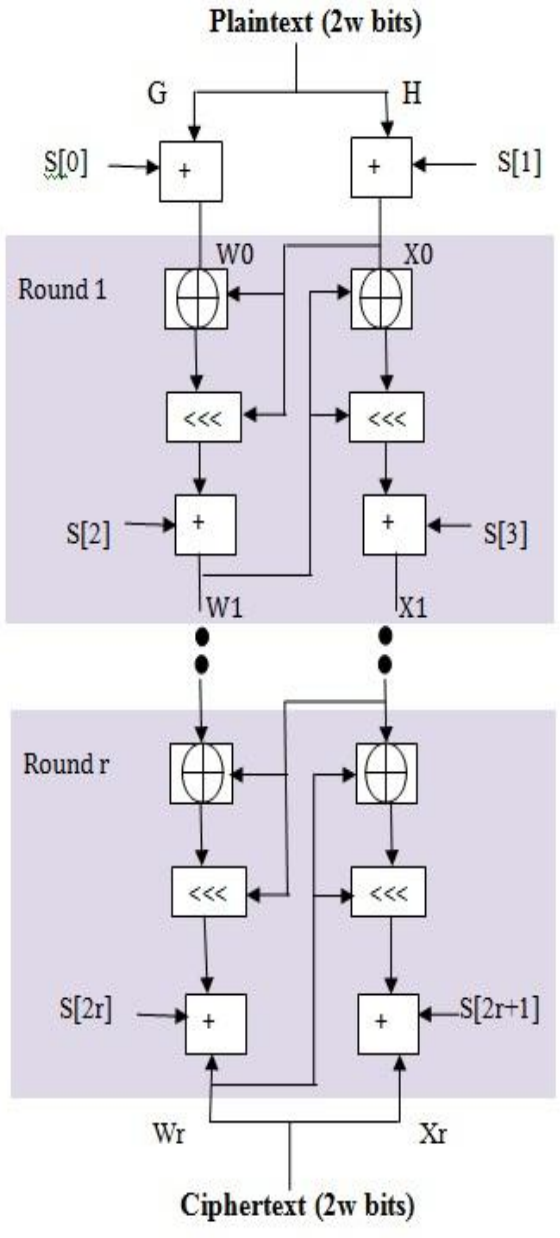

Fig 4: RC5 Encryption Process

4. PROPOSED ARCHITECTUR

In order to realize high-speed processing and area reduction, this study introduces arithmetic processes suitable for hardware during encryption. First, a shift process used for encryption is replaced by a bit selection process. Because of this substitution, the shift process can be realized in wiring. The mixture process usually requires t x 3 = 78 calculations. The proposed architecture divides the processing by inserting a register between calculations and introducing a loop process, which reduce the required calculations to 26. Dividing the processing of each round enables high speed processing. As Fig. 4 shows, RC5's round processing enciphers a 64-bit plaintext. However, the actual processing is performed every 32 bits. Therefore, the round is divided into the left and right parts to perform left and right processing with two clock signals, which raises the operating frequency and improves latency. The proposed model uses to RC5-32/12/16 Parameters. Which means two 32-bit word plaintext and cipher text , 12 no of rounds and 16- byte (128-bit) user secret key. By providing these parameters as original data to the circuit this model is made flexible. Hence it can be modified to suit whatever one's goal. Choice of *r*, for instance, affects the encryption speed and security. The word size also affects both security and speed. For example, choosing the value of parameter *w* larger than the size of the register CPU can degrade encryption speed. To have a fixed set of parameters it is also unusual and risky. Finally extra ff's has been reduced so our design pattern has been changed. In each table, Slice" expresses a block composed of FF and LUTs, and Mux.

In this proposed a new pipelined pattern which got a best performance and Result compared to in device Vertex-2. In designed RC5 pipelined technique extra FF's has been reduced so pipelined pattern has been changed. Finally RC5 encryption pipelined work in artix-7 gives better performance than Vertix-2. In designed architecture Implementation results shown in table 2 i.e. reduced time response and area with increase frequency compare to that table1.

5. IMPLEMENTATION RESULTS

The proposed 12-stage RC5 pipeline model is synthesis in Xilinx 13.4.The input specifications to the algorithm are 64-bit data, 12 rounds, with 16-byte key. This is the most widely used RC5 configuration for obtaining optimal results both in terms of speed and security. The round wise analysis is performed with different number of pipeline stages and rounds which has shown that the 12-stage pipeline is more effective both in area and delay. The proposed design is described with Verilog, synthesized by Xilinx Synthesis Technology (XST).

The encryption is done in XILINX Artix-7 target device XC7A8-3CSG324. The parameters taken are w = 64, r=12and b=128. The plain text is given as input to encryption core then Cipher is obtained. For comparison, the RC5 algorithm used in section 3 was described in verilog HDL. This paper refers to the algorithm used for comparison as target device vertex-2 results some references in table 1 and table 2 show the proposed pipelined technique in artix-7 results.

In each table, Slice" expresses a block composed of FF and LUTs, and Mux. The RTL schematic of various block are shown.

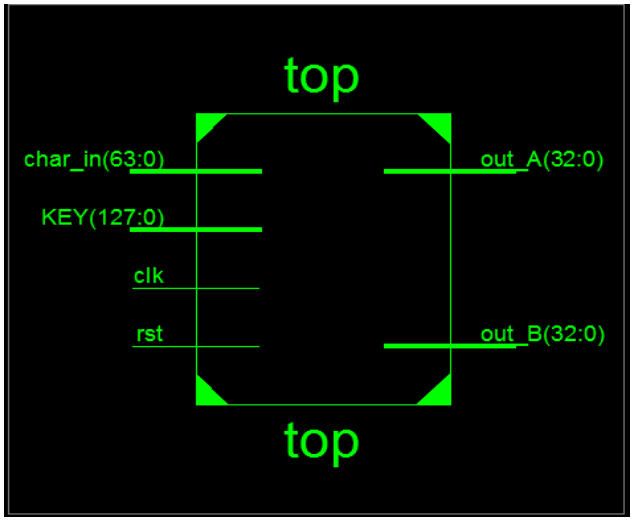

Fig 5: RTL1 schematic of encryption of RC5

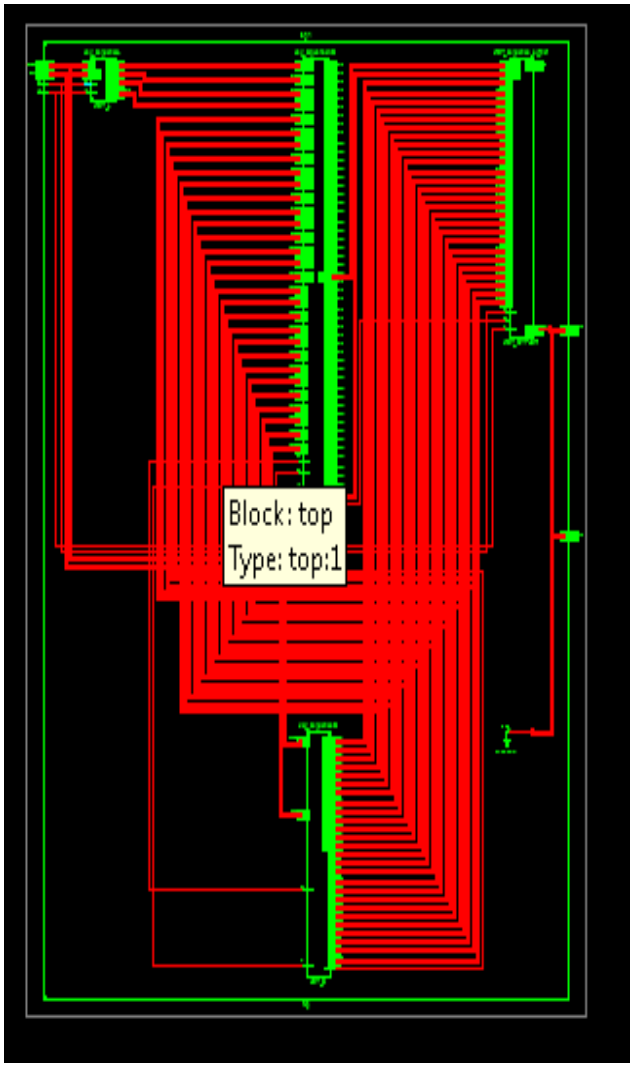

Fig 6 : RTL2 schematic of encryption of RC5

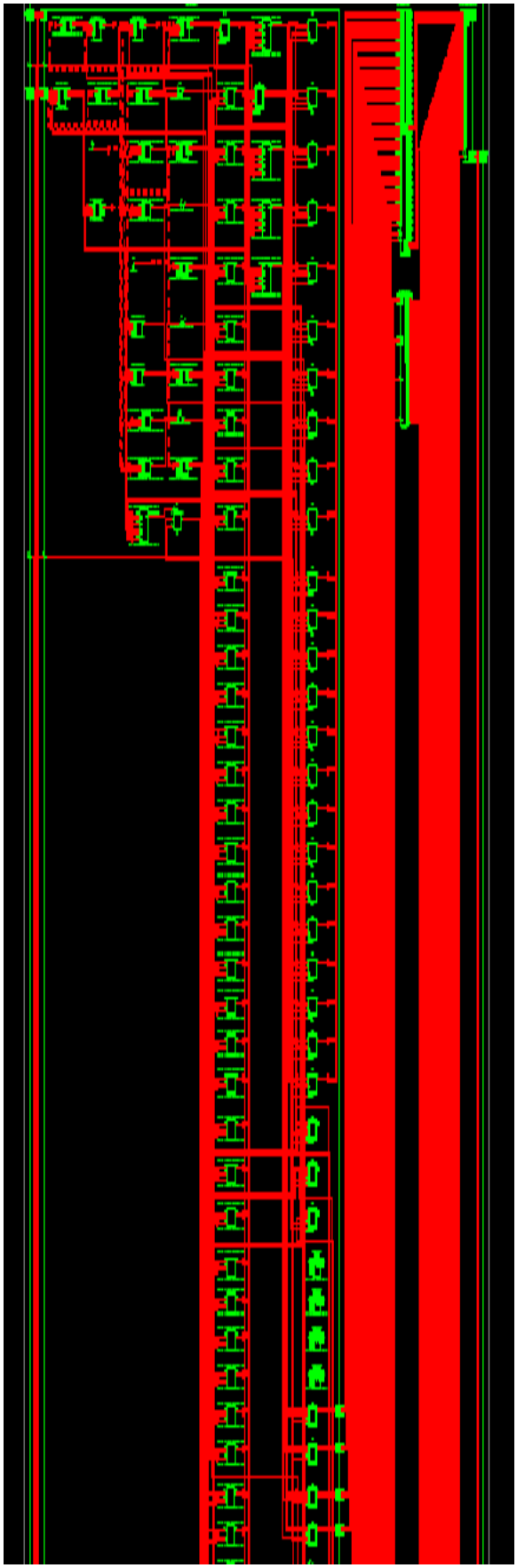

Fig 7 : Schematic for Proposed Pipelined model

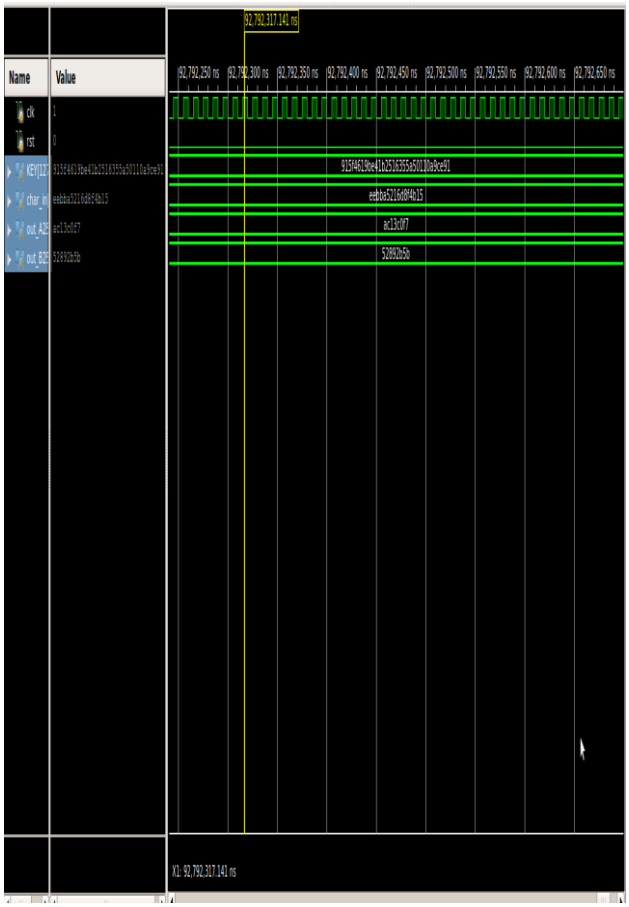

Fig 8: RTL simulation on Encryption process

Finally designed work on device artix-7 is better than work done on Vertix-2 because yet today no any work was done on artix-7 only done in vertex -2. So proposed work on new pipelined pattern in Artix-7 and proofed that artix-7 is better than vertex-2.

The proposed design has resulted in less number of slices with reduced response time. The area-delay product has shown a reduction of nearly 50% when compared with the reference [2] of RC5 design. The simulation of the pipelined design is shown in the figure 8, which clearly illustrates the pipelining mechanism of the proposed model for 12 rounds and encrypted data is shown at output after every clock pulse after initial delay

The implementation results are presented in table 1 & 2.

Table 1. Result of work on device Vertex-2

Architecture	Slices	Delay (ns)	Area* delay	Frequency (MHz)
A.Ruhan Bevi	1698	8.62	14363	116

Table 2. Result of work on device Artix-7

Architecture	Slices	Delay (ns)	Area* delay	Frequency (MHz)
Proposed Architecture	1607	5.587	13543	179

The proposed architecture result clearly shows drastic change which clearly indicate the importance of results with respect to various parameters taken to compare different devices**.**

6. CONCLUSION

In this paper represent RC5 encryption algorithm Reduced the delay and increased the frequency and it's Top module simulation results in model sim 13.4. The proposed design is described in verilog, synthesized by Xilinx synthesis Technology(XST). Result shown in term of frequency and area i.e. 179 MHz and area is reduced 50% in comparision with work done by different devices.[2] The work results are in confirmity with the trend comparision of results published by other workers in various devices.

7. REFERENCES

[1] R.Sanju Abraham & A.Arun, 2013, " Design of RC5 Algorithm using Pipelined Architecture" International Journal of Advanced Research in Computer Engineering & Technology(IJARCET), Vol 2 pp.647-651.

[2] A. Ruhan Bevi1, S.S.V. Sheshu & S. Malarvizhi , 2012, "FPGA based pipelined Architecture for RC5 encyption", IEEE conference on (DICTAP) , pp 214-219.

[3] A. Ruhan Bevi1, S.S.V. Sheshu & S. Malarvizhi ,2012, " FPGA based Sliding Window Architecture for RC5 Encryption", International Conference on Advances in Computing, Communications and Informatics (ICACCI) pp 614-618.

[4] Bahram Rashidi , 2012, "FPGA implementation of optimized the 64- bit RC5 encryption algorithm", Elixir Elec. Engg. 51 pp 10700-10703.

[5] Dhanashri H. Gawali and Vijay M. Wadhai, 2012, "RC5 algorithm: potential cipher solution for security in wireless body sensor networks" International Journal Of Advanced Smart Sensor Network Systems (IJASSN), Vol 2, No.3, pp 1-6.

[6] Harsh kumar verma, and ravindra kumar singh, 2012 " performance Analysis of RC5, Blowfish and DES Block Cipher Algorithms " International Journal Of Computer Application (IJCA) Vol 42 No 16 pp 5775-6004, .

[7]. Masaya Y., and K. Sakaun , 2011 "Dedicated hardware for RC5 cryptography and its Implementation".

[8] Mohamed, A.B. ; Zaibi, G. ; Kachouri, A. 2011, "Implementation of RC5 and RC6 block ciphers on digital images", pp 1-6.

[9] Mohamed, A.B. Zaibi ,& G. Kachouri, 2011, "Implementation of RC5 and RC6 block ciphers on digital images" ISBN: 978-1-4577-0413-0, IEEE .

[10] Ronald L. Rivest, 1995 "The RC5 Encryption Algorithm" Springer-Verlag, pp 87 – 96.

[11] Samir Palnitkar "Verilog HDL: A Guide to Digital Design & Synthesis", ISBN: 978-81-775-8918-4 .

[12] William Stallings, 2010, "Cryptography and Network Security: Principles and Practice", ISBN-13: 978-0136097044.

[13] Bernard Menezes, 2010, "Network Security & Cryptography",ISBN 9788131513491. [14] Zhigang wu and Wei wang, 2011, "Pipelined Architecture for FPGA Implementation of Lifting-Based DWT".

[15] Juha Kukkurainen, Mikael Soini, and Lauri Sydanheimo, 2010, "RC5-Based Security in Wireless Sensor Networks: Utilization and Performance".

[16] Omar Elkeelany and Adegoke Olabisi, 2008, " Performance Comparisons, Design, and Implementation of RC5 Symmetric Encryption Core using Reconfigurable Hardware", pp 49-55.

[17] V Chaitanya Tummalapalli, and MD Khwaja Muinnuddin Chisti, 2012, " Implementation of Low power Algorithm in Xinnx FPGA".