IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016

Synthesis and Simulation of FPGA Based RC4 Encryption Method

Renu M. Sharma[1], Pallavi Choudhary[2]

Shri Balaji Institute of Technology & Management, ECE Department, Betul (M.P.), India[1]

Assistant Professor, Shri Balaji Institute of Technology & Management, ECE Department, Betul (M.P.), India[2]

Abstract: RC4 has been the most famous stream cipher in the history of symmetric key cryptography. RC4 has been designed by Ron Rivets in 1987; it is the most widely deployed commercial stream cipher. The applications of RC4 are in network protocols such as WPA, WEP, SSL and in Secure SQL, Microsoft Windows, Apple OCE, etc. In this project, it has been focus on the synthesis and simulation of hardware of RC4 algorithm which can be implemented further.. The cipher was a trade secret but in 1994 it was leaked out. RC4 is extremely fast and its design is simple. This project deals with RC4 key stream generator, within the scope of the model of an exchange shuffle, in order to achieve better security. The main factors in RC4's success over such an extensive range of applications are its speed and simplicity. For Efficient implementations can be easily modification/editing can be performed either in software or / and hardware as per necessity. The RC4 code is written in Verilog language and based on synthesis result it can be further downloaded on FPGA for its hardware realization.

I. INTRODUCTION

Cryptography is a Greek word. It is associated with Coded Text. Cryptography is Process for converting encrypted information of plaintext into an intermediate form which is known by Cipher Text. It is used to secure information in storage or transit. The main cryptography deals with resolution of problems, which are associated with integrity, secrecy and authentication. Cryptography is also related with Protocol. A protocol is set of standard procedures or sequences of actions. Protocols are designed to fulfill the desired action and concern with many dimensions. Thus, a cryptographic protocol is a protocol that uses cryptography. This cryptographic protocol uses an algorithm to prevent attempts of thefts and invasions. Now a day's cryptography has achieved milestone and strongly Implemented using terms of Theory of Statistics and Theory of No's. In order to handle all the cryptographic problems, various cryptographic algorithms have been invented. Depending on the complexity of these problems Algorithms for cryptographic are categorized in many ways. A much known is the RC4 stream cipher. Normally, security occurs as a result of having a huge No. of different transformations. Then, if a rival acquires some cipher text, a vast No. other plaintext messages could have produced presumably that are exact as cipher text, one for each of the possible keys. Cryptography is one of the sections of Cryptology, which is further divided into secret codes versus ciphers. Cryptography is opposite to Steganography. In Steganography existence of a message seeks to hide. The cryptography seeks to provide a message inscrutable even when the message is completely exposed. Cryptography includes at least: Key generation, confidentially (Secrecy) Message authentication (integrity). RC4 was designed by Ron Rivest of RSA Security in 1987; Even though the RC4 cipher is officially named "Rivest Cipher 4", the RC acronym is alternatively understood to stand for "Ron's Code". RC4 was initially a commercial secret, but in September 1994 a description of it was anonymously posted to the Cipher Punks mailing list so it leaked out in 1994. It was soon spread on the script new group, and on the Internet. Because the algorithm is known, it is no longer a commercial secret. It has become part of some frequently used encryption protocols and standards, including WEP (Wired Equivalent Privacy) and WPA (Wi-Fi Protected Access) for wireless cards. The main factors which helped its supply over such a wide range of applications consisted in its awesome speed and simplicity. The RC4 stream cipher works in two phases. The key setup phase and pseudorandom key stream generation phase. Both phases must be performed for every new key [01]. This paper deals with a new reduced hardware implementation of the RC4 stream cipher. This hardware supports variable key lengths from 1 byte to 256 bytes. It uses only one 256 bytes S-array and one 256 bytes K-array, but previous design uses one 256 bytes K-array and three 256 bytes S-arrays[01]. This implementation needs three clock cycles per byte generation, in the key setup phase and three clock cycles per byte generation in the pseudorandom key stream generation phase. Clearly, there have two ways for implementing any algorithm, either hardware or software. The choice of platform mainly depends on algorithm flexibility its performance and its cost. For securing the high speed networks the hardware always appears to be the final choice. Since Cryptographic algorithms implements hardware they are more secure rather than Software. Also they execute faster operation than software. Uses of FPGAs in hardware implementations for cryptographic applications make the system more competent in all aspect. Their ability to reform and re-programmed make it more computationally in-depth (comprehensive) operations of a range of ciphers depending on security and application requirements. They are more cost effective to ASIC or VLSI design which has a much longer design cycle. This paper mainly focuses the attention on suitability of hardware implementation for RC4 stream cipher. In this paper hardware implementation

DOI 10.17148/IJARCCE.2016.5461

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016

of RC4 stream cipher is presented. Since the dawn of humanity, one of the strongest intrinsic characteristics of the human mind has been its affinity towards secrecy. Be it in terms of secret possessions or secret intentions, the human race has interacted and evolved within the veils of secrecy often regarded as the best form of security. In the modern information age we live in, the notion of security often translates to the idea of confidentiality of information; especially of digital information utilized and transacted over various communication networks. It is important to note that the requirement for confidentiality and security is in fact entirely a social construct. If there was nobody in the universe interested in the piece of information you hold so dear, you may never feel the need for secure possession and transmission of information in the first place. In fact, we still judge security qualitatively in terms of the eagerness and competence of an adversary who may be interested in the information that is kept secure.

Cryptology – Inherently a social science refers to the art of bridging this bizarre gap between an entirely social awareness called Security and the logical base of mathematics, computer science and allied domains. After some research on the web to find an interesting cryptographic primitive to implement, we decided to implement RC4. We chose this stream cipher for each and every reason. First of all, this cipher is one of the most widely used stream cipher. Moreover it is used by really important and famous protocols and standards such as SSL, TSL, WEP, etc. Another reason for this choice is that it is well known for its simplicity and efficiency and we wanted to see if we could really optimize the performance of this cipher after implementing a first basic version. So after presenting the RC4 stream cipher and explaining the way it works, we will present our work on the subject.

II. LITERATURE REVIEW

This stream cipher was invented in 1987 by one of the inventors of the RSA public key cryptography algorithm and co-founders of RSA security i.e. Ron Rivest. Even though the RC4 cipher is officially named "Rivest Cipher 4" or "Ron's Code 4". Other encryption algorithm i.e. RC5, RC6 and RC2 also exist. The trade secret behind RC4 was revealed in September 1994 when the description of the cipher was sent to the Cypherpunks mailing list (group of people interested in privacy and cryptography who used this mailing list to communicate). After that, the description was posted on many website and the genuineness of the information was confirmed as the resulting outputs of the described cipher were matching the outputs of licensed RC4. RC4 had a really large success thanks to its simplicity and efficiency. It was used in many popular standards and protocols such as WEP, WPA, SSL or TLS. RC4, a fast output-feedback cipher, is one of the popular cryptosystems on the Internet, mostly used as the default cipher for SSL/TLS applications. Ron Rivest in 1987 initiated this application for RSA Data Security Inc. and kept as a trade secret until it leaked out in 1994 and is now available for public analysis. RC4 is currently being standardized by the IETF under the name "Arc four". RSA DSI denied that the published algorithm is in the RC4 form, but experimental RESULT showed that it produces the similar outputs as the RC4 software. The RC4 key stream generation (creation) algorithm updates the RC4 internal state and generates one byte of key stream. The key stream is XORed to the plaintext to generate the cipher text. RC4 is comprised of two algorithms: the KSA (Key Scheduling Algorithm) which turns a random key (whose typical size is 40-256 bits) into an initial (inceptive) permutation S of {0,..., N-1}, which uses the secret key to create a pseudo-random inceptive state, and the PRGA (Pseudo Random Generation Algorithm), which generates the pseudo-random stream to Produce a pseudo-random output sequence**.**

III. PRINCIPLE

The RC4 algorithm generates a pseudo-random key stream that is then used to generate the cipher text by XORing it with the plaintext. It is called pseudorandom because it generates a sequence of numbers that only approximates the properties of random numbers. The sequence of bytes produced is not random since the output is always the same for a given input but it has to approximate random properties to make it harder to crack. The key stream is generated from a variable length key using an internal state composed of the following elements:

- A 256 bytes array (denoted S) containing a permutation of these 256 bytes
- Two indexes i and j, used to point elements in the S array (only 8 bits are necessary for each index since the array only have 256 elements).

RC4 uses a variable length key from 1 to 256 bytes to initialize a 256-byte array. The array is used for subsequent generation of pseudo-random bytes and then generates a pseudorandom stream, which is XORed with the plaintext/cipher text to give the cipher text/plaintext. There are two 256-byte arrays, S-Box and K-Box. The S-array is filled linearly, such as S0=0, Sl=l, S2=2, ...S255=255. The K-array consists of the key, repeating as necessary times, in order to fill the array. The RC4 stream cipher works in two phases.

The key setup phase & the pseudorandom key stream generator phase. Both phases must be performed for every new key. RC4 uses two counters, i and j, which are initialized to zero. In the key setup phase the S-box is being modified according to pseudo-code:

Key setup phase:
For i= 0 to 255
j = (j + Si + K1) mod 256
swap Si and Sj
Once the key setup phase is completed the second phase encrypts or decrypts a message. The pseudorandom number generator (PRGN) phase is described by the following pseudo code:
Key stream generation phase:
i=(i+ 1)mod256
j = (j + S) mod 256
swap S, and Sj

IJARCCE

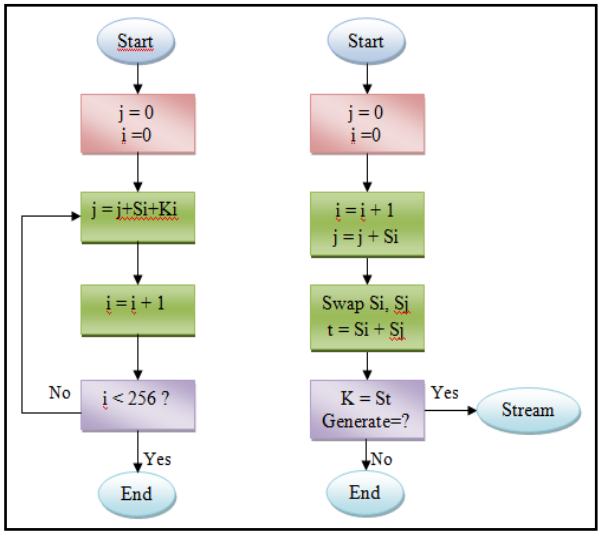
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016

t = (Si + Sj) mod 256
K = Si
The key stream K is XORed with the plaintext/cipher text to produce cipher text/plaintext.

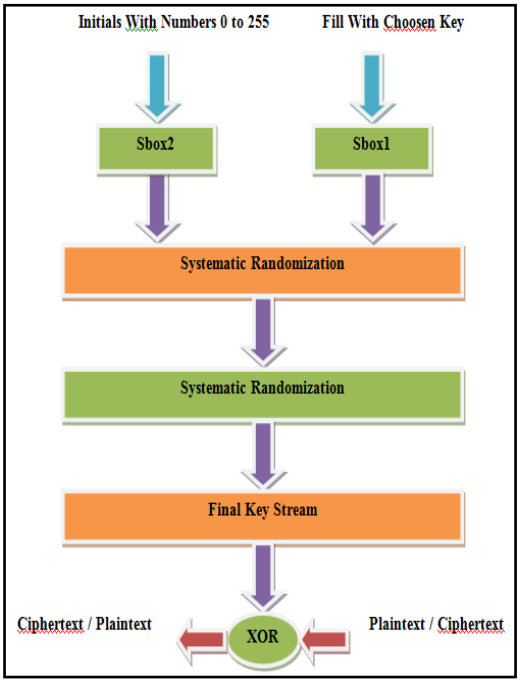
The main principles of this project are as follows:-

- To generate 8-bit security key stream to encrypt and decrypt the data.
- To Implement RC4 algorithm in Verilog language.
- To generate simulation and synthesis result for RC4 Stream Cipher Algorithm.



Flowchart of RC4 Phases

The basic theme of this project is to generate simulation and synthesis report for RC4 encryption algorithm.



Flow Diagram of Plaintext to Ciphertext

The Verilog language is used for coding of RC4 algorithm. The simulation and synthesis report is generated for Virtex-5 FPGA. As 40 to 256 bits can be used for key size in Key Scheduling Algorithm (KSA) therefore in this project 256 bits key size is chosen. The simulation result is verified using draft-josefsson-rc4-test-vectors-02 (This document contains test vectors for the stream cipher RC4)

IV. CONCLUSION

The RC4 stream cipher algorithm used variable key length from 1 byte to 256 bytes. It provides high flexibility for many applications with any key length from 1 byte to 16 bytes. RC4 is simple and cost effective algorithm and possible to implement in both hardware and software. This project will provides the detailed simulation and synthesis results of RC4 algorithm which can be further implemented in the hardware.

REFERENCES

[1]. Rourab Paul, Amlan Chakrabarti and Ranjan Ghosh, "Hardware implementation of four byte per clock RC4 algorithm," in Journal of latex class files Vol. 6 No. 1, Jan. 2007.
[2]. Jaya Dofe and Manish Patil,"Hardware implementation of modified RC4 stream cipher using FPGA," IOSRJEN, vol. 02, Issue 06, pp. 1447–1450, Jun. 2012.
[3]. Poonam Jindal and Bramhajit Singh,"A survey on RC4 stream cipher," IJCNIS, vol. 7, pp. 37–45, Jun. 2015.
[4]. Rajendar Racherla and S. Nagakishor Bhavanam,"Design and simulation of enhancing RC4 stream cipher for Wi-Fi securityusing Verilog HDL," IJERA, vol. 1, Issue 3, pp. 653–659.
[5]. Sultan Weatherspoon,"Overview of IEEE 802.11b security," Network Communication Group, Intel Technology Journal Q2, 2000.
[6]. IEEE STD 802.11. IEEE Standard: Hardware implementation of the RC4 stream cipher-P.kitsos, G. Kostopoulos, N. Sklavos and O.Koufopavlou.VLSI design laboratory.
[7]. Claude E. Shannon. Communication theory of secrecy systems. Bell Systems Technical Journal, 28(4): 656–715, 1949.
[8]. BluetoothTM. Bluetooth Specification, v4.0, June2010. E0 encryption algorithm described in volume 2, pages 1072 Available onlineat http://www.bluetooth.org. 3621of Lecture Notes in Computer Science, pages 97–117. Springer,2005.
[9]. Yi Lu and Serge Vaudenay.Cryptanalysis of Bluetooth keystream generator two-level E0. In Pil Joong Lee, editor, ASIACRYPT, volume 3329 of Lecture Notes in Computer Science, pages 483–499. Springer, 2004.
[10]. Yi Lu and Serge Vaudenay.Cryptanalysis of an E0-like combiner with memory. J. Cryptology, 21(3):430–457, 2008.
[11]. Yi Lu, Willi Meier, and Serge Vaudenay. The conditional correlation attack: A practical attack on Bluetooth encryption. In Victor Shoup, editor, CRYPTO, volume.