HDL Implementation of RC4 Stream Cipher for Cryptographic Applications

[1]Leslie Joseph, [2]Binu Manohar
[1] M.Tech Student,
[2]Asst. Professor
[1,2]Mangalam College of Engg,. Ettumanoor.

Abstract: The implementation of cryptographic algorithms plays an important role because of growing requirements of high speed and high level of secure communications. Implementation of cryptographic algorithms on hardware runs faster than on software and at the same time offering more intrinsic security. The paper involves the design of high performance architecture of a well known cryptographic algorithm- RC4 stream cipher and its performance analysis. The cipher architecture is designed using VerilogHDL and finally programmed into Spartan3 FPGA.

General Terms: Cryptography, RC4, security, delay.
Keywords: KSA, PRGA, Verilog HDL, FPGA.

1. INTRODUCTION

The security of sensitive information against 'prying eyes' has been of prime concern throughout the centuries. Therefore, a mechanism is required to guarantee the security and privacy of information. Under the existing circumstances cryptography is the only convenient method for protecting information transmitted through communication networks. The implementation of cryptographic algorithms plays an important role because of growing requirements of high speed and high level secure communications. The field of cryptography deals with the techniques for conveying information securely. The goal is to allow the intended recipients of a message to receive the message properly while preventing eavesdroppers from understanding the message[2].

The message in its original form is called plaintext. The transmitter in a secure system will encrypt the plaintext in order to hide its meaning. This reversible mathematical process produces an encrypted output called ciphertext. Cryptography is the only practical means to provide security services in many applications. Research into cryptography has exploded in the last 18 years and a variety of cryptographic algorithms and techniques have emerged.

In cryptography, RC4 (also known as ARC4 or ARCFOUR meaning Alleged RC4) is the most widely used software stream cipher and is used in popular protocols such as Secure Sockets Layer (SSL) (to protect Internet traffic) and WEP (to secure wireless networks). It is remarkable for its simplicity and speed in software. The paper involves the design of the conventional and fastest known architecture of the RC4 stream cipher and their hardware implementation in FPGA. The performance of the design is analyzed based on delay and area.

2. CONVENTIONAL RC4 STREAM

CIPHER

The RC4 stream cipher was designed by Ron Rivest for RSA Data Security in 1987. While it is officially termed "Rivest Cipher 4", the RC acronym is alternatively understood to stand for "Ron's Code". It uses S-box S, an array of length N, where each location of S stores 1 byte (typically= 256). A secret key k of size l bytes is used to scramble this permutation. Array K of length N holds the main key, with secret key k repeated as K[y] = k[y mod l].RC4 has two components, namely the Key Scheduling Algorithm (KSA) and the Pseudo-Random Generation Algorithm (PRGA). The KSA uses the key K to generate a pseudo random permutation and PRGA uses this pseudorandom permutation to generate arbitrary number of pseudorandom key stream bytes. [1].

Key Scheduling Algorithm
1. Procedure KSA (Secret Key K).
2. Initialize S {0, 1 …N-1} and j 0.
3. For i=0 … N-1 do
   Increment j = j + S[i] +K[i].
   S[i] = S[j].
4. End for
5. Return Sbox.
6. End procedure.

Pseudo Random Generation Algorithm
1. Procedure PRGA.
2. Initialize i and j to 0.

3. While TRUE do
Increment i=i+1 and j= j+S[i].
S[i] = S[j].
Z= S[S[i] +S[j]].
4. End while.
5. End procedure.

The output key stream Z is XOR-ed with the plaintext (byte per byte) to generate the cipher text at the sender end and is XORed back with the cipher text to get back the plaintext at the receiver end. The major disadvantage of this conventional RC4 it can only take one plaintext at a time i.e., the rate is very less. As a result the cost of its hardware will be more .But due its simplicity and elegance the RC4 is most commonly used. To avoid the rate performance disadvantage, there is a great need of high performance architecture for the cipher.

3. HIGH PERFORMANCE ARCHITECTURE FOR RC4

As suggested in [1], for the high performance architecture, we consider the generation of two consecutive values of Z together, for the two consecutive plaintext bytes to be encrypted. Also, the architecture combines the ideas of pipeline. [1].

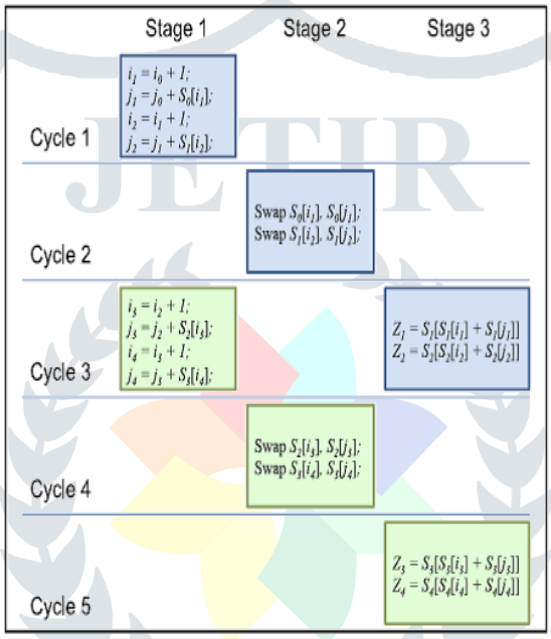

Figure 1: Fastest Architecture concept [1]

Using this concept a fastest architecture for RC4 can be implemented. So, RC4 algorithms can be used in a wild range of applications; including the SSL (Secure Sockets Layer) developed by Netscape Communications Co., SSH (Secure Shell), IPSec (IP Security Protocol), Kerberos and PGP (Pretty Good Privacy). Besides networking applications, RC4 algorithms are also applied to data storage. UNIX system provides system tools to encrypt files, word processors such as Microsoft Word and Adobe Acrobat integrate the encryption/decryption functions internally. Also, many data compressing tools such as Zip and RAR provide encryption. On static storages such as DVDs, RC4 algorithms are used to protect copyright by preventing illegal copying.

4. SIMULATION RESULTS

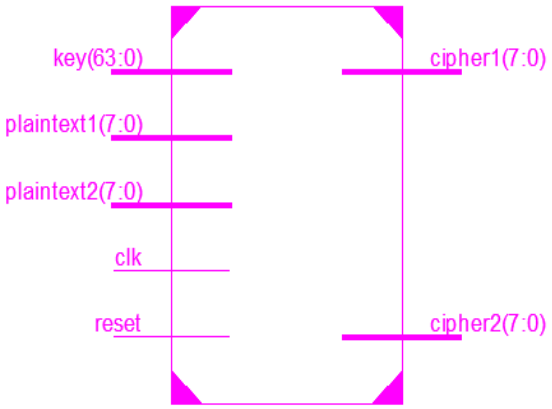

Figure 2: RTL of Encryption

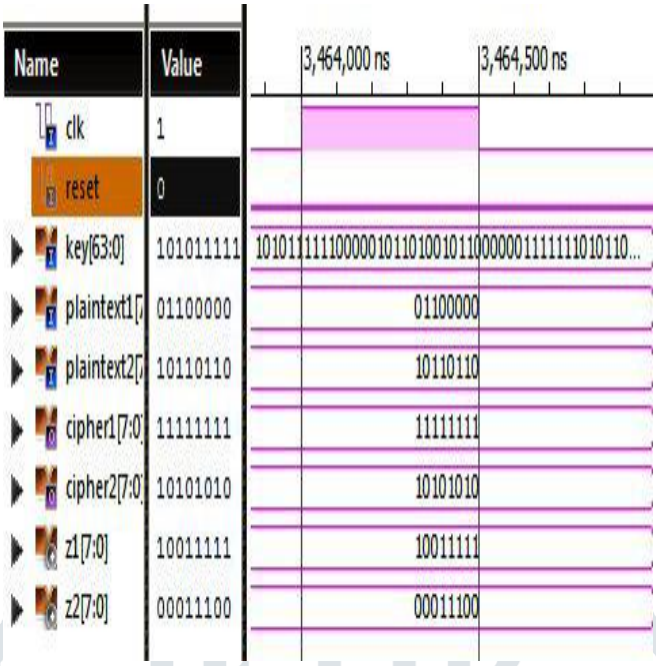

Figure 3: Simulation Result of RTL encryption

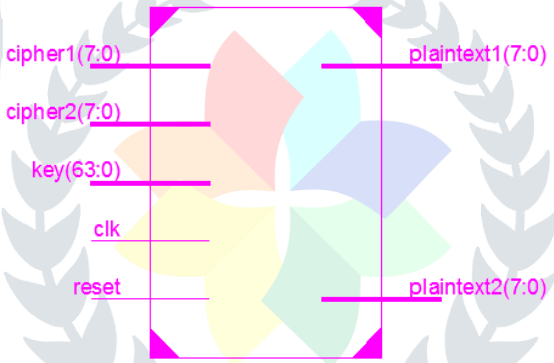

Figure 4: RTL of RC4 Decryption

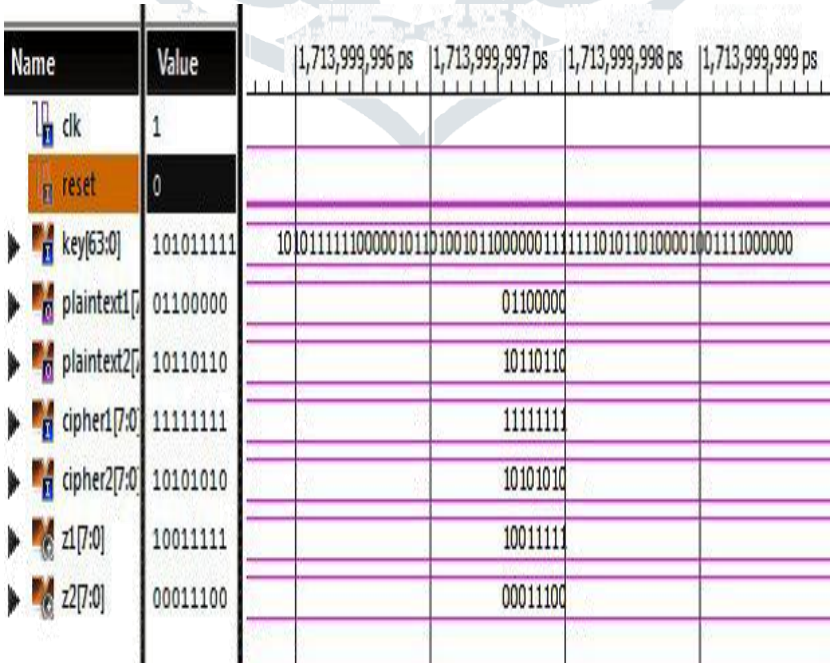

Figure 5: Simulation Result of RC4 Decryption

5. PERFORMANCE ANALYSIS

5.1 Timing Report

```
Timing Summary:
---------------
Speed Grade: -4

   Minimum period: 108.130ns (Maximum Frequency: 9.248MHz)
   Minimum input arrival time before clock: 113.998ns
   Maximum output required time after clock: 8.856ns
   Maximum combinational path delay: 8.813ns
```

5.2 DEVICE UTILLIZATION REPORT

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	15422	20480	75%
Number of Slice Flip Flops	7398	40960	18%
Number of 4 input LUTs	29581	40960	72%
Number of bonded IOBs	98	333	29%
Number of GCLKs	3	8	37%

6. CONCLUSION

Due to the great use of RC4 in cryptographic applications a fast architecture is needed, as a result a new RC4 architecture is implemented and analyzed.

7. REFERENCES

[1] Sourav Sen Gupta, Anupam Chattopadhyay and Koushik Sinha, "High Performance Hardware Implementation of RC4 Stream Cipher", IEEE Trans. on Software Eng., Vol.63 (5):730-743, April 2013.
[2] "Enhancing Jian Xe, Xiaozhong Pan "An Improved RC4 Stream Cipher"International Conference on Computer Application and System Modelling (ICCASM 2010).
[3] RC4 algorithm for WLAN WEP protocol" IEEE Transactions on control and decision Conference, 2010.
[4] P.kitsos, G. Kostopoulos, N. Sklavos and O.Koufopavlou, VLSI design laboratory. "IEEE Std 802.11. IEEE Standard: Hardware implementation of the RC4 stream cipher"