International Journal of Cryptology Research 1(2): 225-233 (2009)

Hardware Implementation of RC4A Stream Cipher

[1]Abdullah Al Noman, [1]Roslina Mohd. Sidek and [2]Abdul Rahman Ramli

[1]Department of Electrical and Electronic Engineering, Faculty of Engineering, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.

[2]Department of Computer and Communication System Engineering, Faculty of Engineering, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.

E-mail: Simbul74@gmail.com, roslina@eng.upm.edu.my, arr@eng.upm.edu.my

ABSTRACT

Cryptography is the only practical method for protecting information transmitted through communication networks. The hardware 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. This paper presents efficient hardware implementation of new stream cipher, RC4A. The proposed hardware implementation achieves a data throughput up to 22.28 MB/sec at frequency of 33.33 MHz and the performance in terms of throughput to area ratio equal to 0.37. The implementation is also parameterized in order to support variable key lengths, 8-bit to 512-bit. The cipher was designed using Verilog hardware description language and implemented into a single Altera APEX™ 20K200E Field Programmable Gate Array (FPGA).

Keywords: Cryptography, FPGA, RC4A, Security, Stream cipher, Throughput, Verilog HDL

INTRODUCTION

Today, with the explosion in the Internet based Electronic Commerce; there is an increasing need to secure the data and commercial transactions. The desire to transmit messages securely is not new. For centuries, people want to keep their communications secret. Therefore, a mechanism is required to guarantee the security and privacy of information that is transmitted over the electronic communications media.

Under the existing circumstances cryptography is growing as a usual solution to such problem offering security services of data privacy, data

integrity, authenticity and non repudiation. 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. 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*. (Eskicioglu et al 2001)

Cryptography is the only practical means to provide security services in many applications. Research into cryptography has exploded in the last 18 years (Schneier 1996) and a variety of cryptographic algorithms and techniques have emerged .RC4A is one of them.RC4A, an RC4 family algorithm designed by Ron Rivest for RSA Data Security, Inc. in 1987(Schneier 1996), developed by S. Paul and B. Preneel (Paul et al 2004) which attempts to increase security without decreasing efficiency. Their approach essentially takes two RC4 instances and crosses information between them. RC4A stream cipher works in two phases, KSA (Key Scheduling Algorithm) phase and PRGA (Pseudo Random number Generation Algorithm) phase. During PRGA two successive output byte are generated. The goal behind RC4A was to increase security primarily by increasing the internal complexity of the algorithm. (Mckague 2005)

Clearly, there have two ways to implement any algorithm, i.e. either hardware or software. The choice of platform depends on algorithm performance, cost and flexibility.

For secure high speed networks the hardware always appears to be the ultimate choice because hardware implementation of cryptographic algorithms is intrinsically more physically secure and run faster than software. In hardware implementations, the flexibility and high speed capability of FPGAs make them a suitable platform for cryptographic applications. Their reconfigurability means that they can be re-programmed to perform the more computationally intensive operations of a range of ciphers depending on security and application requirements (Marnane 2004). They also offer a more cost effective solution than any ASIC or VLSI design which has a much longer design cycle.

In this paper, we mainly focus our attention on suitability of hardware implementation for RC4A stream cipher. In this paper hardware implementation of RC4A stream cipher is presented. We mainly focus our attention on suitability of hardware implementation for RC4A stream cipher.

Because for real time high speed secure communication application hardware is needed to replace software. As far as security concern, RC4A has an improved security over the RC4 against most of the known plaintext attacks (Zoltak 2004).RC4A pseudorandom bit generator passed all the statistical tests listed in (Preneel et al 2003 ;Paul et al 2004).

Hardware implementation of the RC4A stream cipher is presented in this paper. Proposed implementation supports variable key lengths. The Key lengths could be 8 bit to 512 bit. For initialization proposed implementation require 7.67354 µs or 256 clock cycle, for KSA it require 61.257877 µs or 2042 clock cycle and for PRGA it require 22.98 µs or 766 clock cycle. The proposed hardware implementation achieves a data throughput up to 22.28 MB/sec or 171 Mbits/s at frequency of 33.33 MHz.

The cipher was designed using Verilog hardware description language and implemented into a single Altera APEX^{TM} 20K200E Field Programmable Gate Array (FPGA).

The paper is organized as follow. First RC4A algorithm is presented. Design methodology SOC is then discussed. This is followed by architecture of SOC, discussion on the FPGA implementation and performance of the SOC. Finally the conclusion is presented.

RC4A STREAM CIPHER

At FSE 2004, RC4A (Paul et al 2004) was proposed by Souradyuti Paul and Bart Preneel. This cipher is another modification of RC4 which attempts to increase security without decreasing security. RC4A is made through improvement on the RC4, i.e., providing 2 *S* arrays (S1 and S2) that are independent from each other, so that RC4 should not have bias in consecutive output byte. KSA for RC4 is used as that for RC4A, following the manner of paper (Paul et al 2004). To be more specific, in KSA of RC4, the array *S*1 is initialized, using the secret key *K*. *WK*, are generated from the array *S*1 in PRGA (Pseudo Random number Generation Algorithm) of RC4. Then, the array *S*2 is initialized in KSA of RC4, using *WK*. Unlike RC4, in PRGA of RC4A two successive output byte are generated. All the arithmetic operations are computed modulo N (N=256) (Zoltak 2004).

The algorithm and its block diagram are given below.

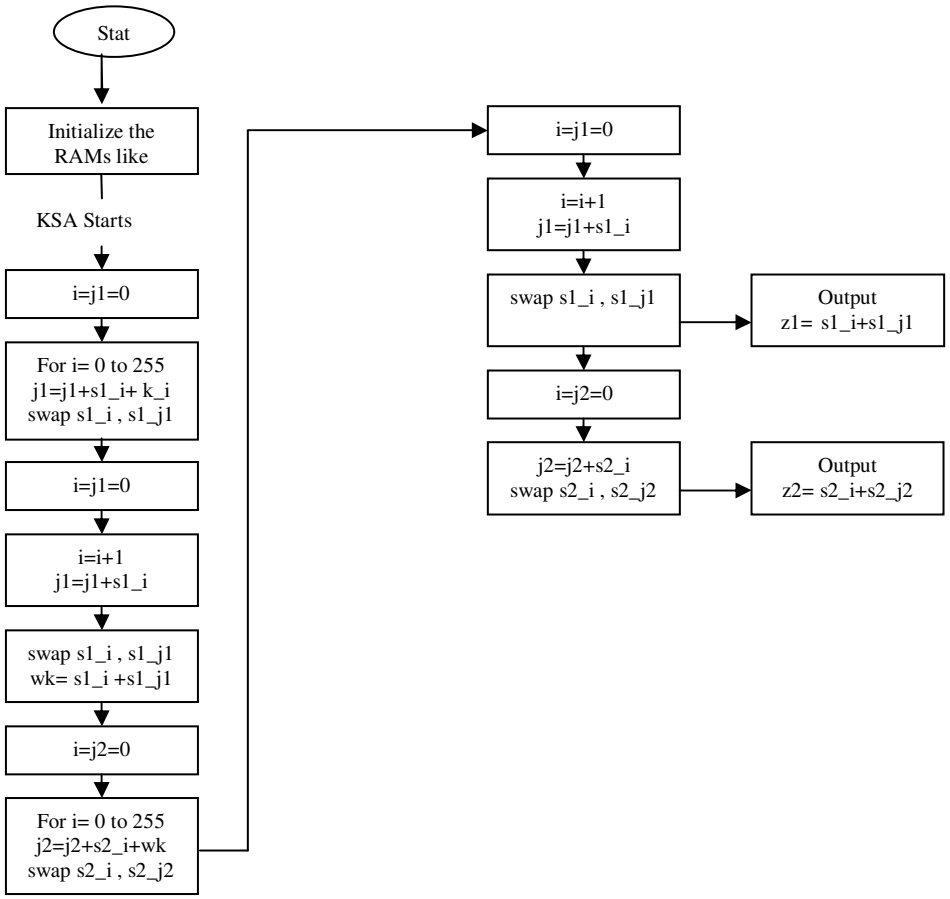

Figure 1: Block Diagram Showing the RC4A Algorithm

Algorithm
KSA (K)

RC4_KSA(K, S1)
For i = 0 …l – 1
WK[i] = RC4_PRGA(S1)
RC4_KSA(WK, S2)

PRGA (S1, S2)
Initialization:
i = 0
j1 = j2 = 0
Generation loop:
i = i + 1
j1 = j1 + S1[i]
Swap(S1[i], S1[j1])
Output z1 = S2[S1[i] + S1[j1]]
j2 = j2 + S2[i]
Swap(S2[i], S2[j2])
Output z2 = S1[S2[i] + S2[j2]

DESIGN METHODOLOGY

Design Cycle

Design cycle for our work consists of the following steps:

- Verilog implementation of the algorithm of RC4A stream cipher.
- Verifying the algorithm on Register-Transfer-level (RTL)
- Synthesis and logic optimization.
- Place and Route for specific device.

Software tools

The entire design was described using Verilog HDL language. Our choice has the advantages to be portable on all circuit design platforms.

For implementing our function on FPGA device, we have used the following development software:

Quartus II 5.0 which is fully integrated package for creating for logic design for Altera FPGA. Altera Quartus II EDA tool provides a complete, multiplatform design environment for system-on-a-programmable-chip (SOPC) design. It supports system level design, FPGA, and CPLD (Complex programmable logic device) design, synthesis, place and rout, verification and device programming. Each stage of design flow can be invoked from the GUI of the Quartus II.

Hardware tools

Altera Nios development kit is a board mounted with EPXAI device from the APEX family (Altera 2006). The part number of the FPGA chip mounted on the Nios development kit used in this thesis is EP20K200EFC484-2X. It provides 8,696 registers; 106496 memory bits; and 200,000 typical gates. It contains an embedded array to implement memory functions and logic array to implement general logic functions. It has 8,320 logic elements grouped into 52 Mega Logic Array Block(LAB) structures, each of which consists of 16 logic Labs and one Embedded System Block(ESB).The ESB provides 2,048 programmable bits that can be configured as product term logic, look up table based logic, or various types of memory. The kit has other peripherals such as external SDRAM (static Dynamic Ram), SDRAM controller, watchdog timer and UART (Universal Asynchronous Receiver and Transmitter). The kit is an ideal platform for system prototyping, emulation, hardware and software development or other special requirements. It provides a flexible, powerful debug and development environment to support the development of systems using APEX devices.

PROPOSED ARCHITECTURE

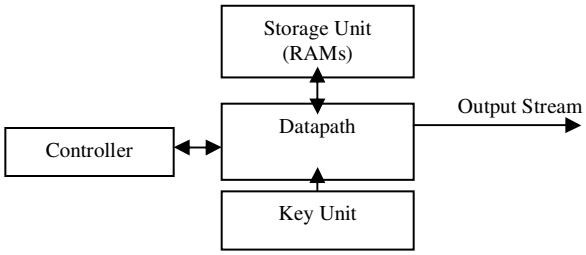

Figure 2: Functional block diagram of the SOC

Figure 2 illustrates the functional block diagram of the SOC proposed in this paper. The SOC consists of Controller, Datapath, Storage Unit and Key Unit. Controller provides essential signals to operate Datapath, Storage Unit and Key Unit. Datapath unit is responsible for the key set up and key stream generation. As per received signal from controller it regenerate required signal for operation of storage and key unit. With the data from storage and key unit it executes arithmetical and swapping operation. And then generate the output.

The storage unit was used to contain memory elements for S1, S2 known as S1 box, S2 box. Storage unit dealt with S1 box and S2 box. Storage Unit consists of six Rams. The Key Unit was used to contain memory elements for K array known as K box to store key of variable length from 8 bit to 512 bit. For this function one RAM was used in the SOC.

HARDWARE IMPLEMENTATION RESULTS

The whole design was analyzed & synthesized by using Altera FPGA device (part number-EP20K200EFC484-2X). Synthesis results for the proposed SOC are shown in table 1. Timing of various stages of the SOC is shown in table 2.

TABLE 1: FPGA Implementation Results

FPGA Device: APEX 20KE (part number-EP20K200EFC484-2X)		
Area allocation	Used/Available	Utilization
Total logic elements	480/8,320	5%
Total pins	146/376	38%
Total memory bits	10,240/106,496	9%

TABLE 2: Timing Results of various stages

Stage	**required time**	**Clock cycle**
Initialization	7.67354 µs	256
KSA	61.257877 µs	2042
PRGA	22.98 µs	766

Compare with other stream ciphers in (Galanis et al 2005) it is observed that proposed SOC required more time and clock cycle for KSA.

Throughput of the SOC can be defined as, *Throughput* = N * clock frequency. Where N is the number of bits produced in every clock cycle (Batina 2004) So, Throughput of proposed SOC is171.11 Mbits/s where for the SOC clock frequency is 32 MHz and N=(512x8)/766 = 5.34. Throughput to area ratio is a measure of Hardware performance of the cipher (Galanis et al 2005). For the presented SOC Throughput to area ratio is 0.36 where Throughput 171.11 Mbits/s and area consumed 480 slices. The performance

in term of throughput proposed SOC is high. But it consumes more area thus it has low throughput to area ratio.

It is reasonable to consume more area and high time for KSA because to increase security of the RC4A primarily internal complexity of the algorithm and number of variables involved in each output byte was increased. By increasing the number of variables involved in each output the size of predictive state was increased, reducing biases. The cost is a large increase in memory requirements as well as set up time as it involves more arithmetic operation (Paul et al 2004; Mckague 2005).

To the best of our knowledge there are no published hardware implementations results for RC4A cipher, which can be compared with our implementation.

CONCLUSION

Hardware implementation of RC4A stream cipher is presented in this paper. Proposed implementation achieved 171Mbits/s data throughput at frequency 32 MHz, using 8 bit word and variable key length, from 8 bit to 512 bit. It offers flexibility as it can be employed in several applications with any key length from 8 bit to 512 bit.

Unlike RC4, RC4A considered as a secure cipher. Therefore, it is a more suitable alternative for long term privacy. Nevertheless, the cost of this improved security is longer encryption time and high consumed area.

In the light of the presented hardware implementation results, it can be said that proposed SOC is flexible solution for any cryptographic system.

REFERENCES

Alan Daly, William Marnane, Tim Kerins and Emanuel Popovici. 2004. An FPGA implementation of a GF(p) ALU for encryption processors**,** *Microprocessors and Microsystems*, **28**(5-6): 253-260.

Batina. 2004, www.cosic.esat.kuleuven.be/publications/article-497.pdf

B. Schneier. 1996. *Applied Cryptography*, 2nd Edition, Wiley, New York.

B. Zoltak. 2004. VMPC One-Way Function and Stream Cipher, Fast Software Encryption, FSE 2004, LNCS 3017, pp.210-225, Springer-Verlag.

B. Preneel *et al.* 2003. NESSIE Security Report, Version 2.0, IST-1999-12324, February 19, 2003.

Eskicioglu, A., Litwin, L., 2001. Cryptography Potentials, *IEEE*, **20**(1): 36 – 38.

Mathew E. Mckague. 2005. Design and analysis of RC4 like stream cipher. *Masters Thesis*, University of Waterloo, Canada, 2005.

Michalis Galanis, Paris Kitsos, Giorgos Kostopoulos, Nicolas Sklavos and Costas Goutis. 2005. Comparison of the Hardware Implementation of Stream Cipher, *The international Arab Journal of Information Technology*, **2**(4).

S. Paul and B. Preneel. 2004. A New Weakness in the RC4 Keystream Generator, Fast Software Encryption, FSE 2004, LNCS 3017, pp.245-259, Springer-Verlag.