HARDWARE IMPLEMENTATION OF AES-CCM FOR ROBUST SECURE WIRELESS NETWORK

Arshad Aziz and Nassar Ikram

National University of Sciences & Technology (NUST) Pakistan

.

arshad@khi.paknet.com.pk (Ph: +92-333-2228300) P N Engineering College, Habib Rehmatullah Road, Karachi 75350, Pakistan

nassar.ikram@gmail.com (Ph: +92-333-5241464) PO Box 2855, GPO, Islamabad, Pakistan

ABSTRACT

Fast and secure implementations of cryptographic algorithms are essential for the realisation of any real time communication system. Cryptographic transformations are computationally intensive and therefore, strongly influence the performance of enabling crypto device. The choice of development platforms for embedding cryptographic applications is made considering the power and speed besides obviously the cost. With the ever increasing computational power vis-à-vis decreasing costs, reconfigurable devices like Field Programmable Gate Array (FPGAs) offer viable avenue for embedding cryptographic applications. Because of outperforming merits which distinguish FPGAs from other development platforms, these are increasing being deployed in a number of applications. This paper focuses on efficient implementation of secure wireless paradigm. As an aftermath of published weaknesses and vulnerabilities in Wired Equivalent Privacy (WEP), IEEE 802.11 interim security solution based on Temporal Key Integrity Protocol (TKIP) accommodates the existing WEP hardware by upgrading the software or firmware. However, the proposed long-term security solution is based on Advanced Encryption Standard (AES) CCM (Counter Mode, Cipher Block Chaining, Message Authentication Code), which definitely entails hardware upgradation. Through this work, manifestation of cryptographic implementation required to address the long term security solution for Robust Secure Wireless Network (RSN) based on fast, efficient and low power FPGA has been demonstrated. Computational intensive processes are therefore, offloaded from the main processor thus enabling achievement of secure high speed wireless connectivity. The Design utilizes low cost and low power Spartan-3 FPGA, producing a throughput of 2699 Mpbs using 10 Block RAM, 481 Slices and throughput per area of 5.6 Mbps/Slice.

KEY WORDS

AES, CCMP, FPGA, Cryptography, Wireless Security, WEP, TKIP, RSN

HARDWARE IMPLEMENTATION OF AES-CCM FOR ROBUST SECURE WIRELESS NETWORK

1 INTRODUCTION

Cryptography is a fundamental component of any secure system seeking protection of sensitive information. Cryptographic transformations are computationally intensive and therefore, strongly influence the performance of enabling secure device. The choice of development platforms for embedding cryptographic applications is made considering the power and speed besides obviously the cost. With the ever increasing computational power vis-à-vis decreasing costs, reconfigurable devices like Field Programmable Gate Arrays (FPGAs) offer viable avenue for embedding cryptographic applications. Because of outperforming merits which distinguish FPGAs from other development platforms, these are increasing being deployed in a number of applications. With growing shift towards wireless networks as a result of enhanced speeds that have become possible in recent times, its security is becoming an area of active research. Wireless technology has become an essential ingredient of today's corporate networks and therefore, its security merits due treatment. This paper focuses on efficient implementation of secure wireless paradigm. As an aftermath of published weaknesses and vulnerabilities in Wired Equivalent Privacy (WEP), IEEE 802.11 interim security solution based on Temporal Key Integrity Protocol (TKIP) accommodates the existing WEP hardware by upgrading the software or firmware. WEP and TKIP are based on the RC4 algorithm. However, the proposed long-term security solution is based on Advanced Encryption Standard (AES) CCM (Counter Mode, Cipher Block Chaining, Message Authentication Code) which definitely entails hardware upgradation. Through this work, manifestation of cryptographic implementation required to address the long term security solution for Robust Secure Wireless Network (RSN) based on fast, efficient and low power FPGA has been demonstrated. The CCMP, however, is computational intensive and overloads the main processor affecting the speed. Implementation of CCMP on separate, dedicated platform e.g. FPGAs results in offloading of main processor from crypto functions with realisation of greater speeds for the network.

Section 2 defines new security paradigm for wireless networks. Section 3 gives a brief summary of AES and section 4 deals with CCM. Section 5 presents the system architecture of implementation. Our implementation results with comparison from earlier implementations are contained in Section 6. Section 7 concludes the paper setting directions for further work.

2 NEW SECURITY PARADIGM FOR WIRELESS NETWORK

The IEEE standard 802.11 introduced Wired Equivalent Privacy (WEP) for Wireless LANs. The prime objective of WEP is to defend confidentiality of data from eavesdroppers. Other objectives include guarding against covert modification (i.e., integrity) and provision of access control. WEP utilizes the RC-4 encryption algorithm. However, grave flaws were exposed with WEP's intended security goals by researchers from the University of California at Berkeley and Zero Knowledge Systems [1] after which it was realized that a strong security mechanism was needed. The RC-4 encryption algorithm was not necessarily weak, it was the key exchange that compromised WEP. The new security mechanism thus defined thereafter [2] uses two main developments; Wi-Fi Protected Access (WPA) and Robust Security Network (RSN). WPA/TKIP provides important data encryption enhancements to address WEP vulnerabilities which include a per-packet key mixing function, a Message Integrity Check (MIC) that prevents packet forgeries, an extended initialisation vector with sequencing rules, and finally a re-keying mechanism. The other development i.e. RSN uses dynamic negotiation of authentication and encryption algorithms. The authentication is based on 802.1x and Extensible Authentication Protocol (EAP) built around AES

encryption algorithm. Together, these implementations provide a framework for string user authentication.

Deployment of RSN on legacy WLAN client devices and Access Points, however, is a challenge as the existing hardware does not offer the computational power required to support RSN's computationally intensive cryptographic implementation. In RSN the security protocol built around AES is called Counter mode with CBC-MAC Protocol or CCMP. CCMP defines a set of rules that use the AES block cipher to enable the encryption and protection of IEEE 802.11 frames of data, it has little resemblance to WEP and is the strongest security solution in development for IEEE 802.11i. CCMP encrypts data at the MPDU (MAC Protocol Data Unit) level as shown in Fig 1. [2]

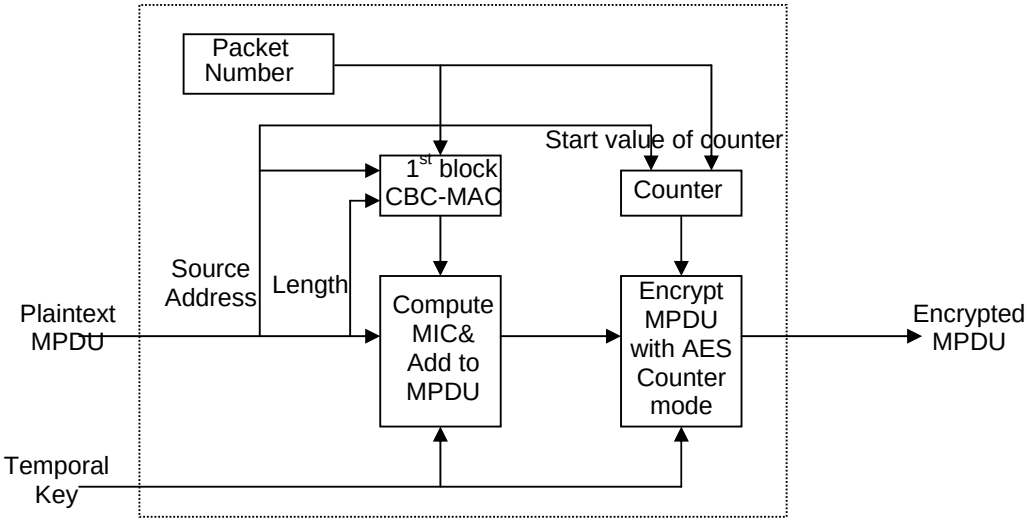

Figure 1. CCMP Encryption Block

3 AES ENCRYPTION ALGORITHM

Rijndael Algorithm [3] has been selected as the new Advanced Encryption Standard Algorithm [4] by the National Institute of Standards and Technology (NIST) [5]. AES is a symmetric block cipher having variable key and fix data length. The key lengths can be independently chosen as 128, 192 or 256 bits which result in 10, 12 and 14 rounds of operation respectively. The data length is however fixed to 128 bits. The input data can be considered as a matrix with four rows and four columns called *state*. Each element of the matrix is composed of eight bits. The AES algorithm has four basic transformations.

3.1 SubBytes Transformation

A nonlinear transformation applied to the elements of the matrix. This first step in each round is a simple substitution that operates independently on each byte of *state* using S-box. The byte, s[*i*,*j*] become s`[*i*,*j*] through a defined substitution table.

3.2 ShiftRows Transformation

A cyclically shift operation with constant offsets, applied to the rows of the matrix. This second step in each round is permutation of rows by left circular shift; the first (leftmost, high order) *i* elements of row *i* are shifted around to the end (rightmost, low order).

3.3 MixColumns Transformation

The third step is a complex transformation on the columns of *state* under which the four elements of each column are multiplied by a polynomial, essentially diffusing each element of the column over all four element of that column.

3.4 AddRoundKey Transformation

This performs XOR operation on the round key, which is obtained from the initial key by a key expansion procedure.

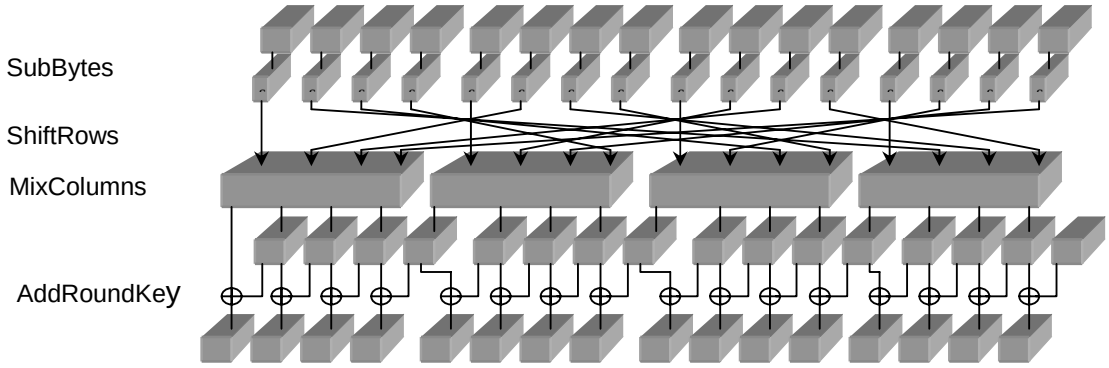

Figure 2. Block diagram of the encryption procedure

The encryption flow starts with the addition of the initial key to the plaintext. Then the iteration continues for (Nr – 1) rounds, where Nr is the total no of rounds. Fig. 2 shows the block diagram of the encryption procedure.

4 CCM PROTOCOL

CCM is a new mode of operation that combines two existing modes of block ciphers i.e., the Counter Mode (CTR) and Cipher Block Chaining – Message Authentication Code (CBC-MAC). Descriptions of these as well as other modes of block cipher operation are given at [6]. CCM uses encryption algorithm to generate encrypted and authenticated data at the same time [7]. CCM mode was created especially for use in IEEE 802.11i RSN, but it is applicable to other systems as well. It is intended for packet environment with no attempt to accommodate streams.

5 SYSTEM ARCHITECTURE AND IMPLEMENTATION

Hardware implementation of AES can provide either high performance or low cost solution for specific application. One of the efficient implementation is based on pipelined AES architecture, albeit offering throughputs greater than 20 Gbps, it is not a viable candidate for the applications involving feedbacks in mode of operation e.g. CCM. Our implementation is targeted to be efficient, minimize real estate on the chip and be low on power. To achieve these objectives and to provide support for the feedback mode, sequential implementation of AES algorithm has been used. The design architecture is shown in Fig. 3 with four AES transformations indicated. Utilising reusable hardware functions, duplication has been avoided. Implementation of byte permutation and bitwise addition modulo 2 (XOR) functions on FPGA is easy. However, the MixColumns function that involves matrix multiplication in GF (2^8) and Byte Substitution consume more then 75% of FPGA resources. Implementations of four AES transformations are explained in succeeding subparagraphs.

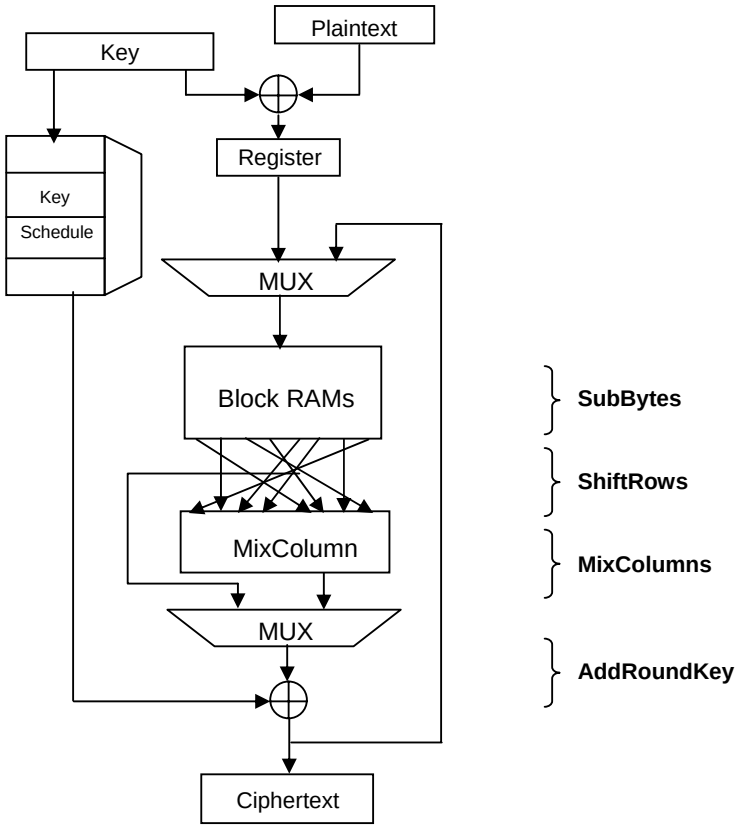

Figure 3. Design Architecture

5.1 SubBytes Step

Implemented as a simple Look Up Table (LUT) which obviously uses a large number of gates. Efficiency entails deployment of large embedded memories for S-box. Each S-box uses 2048 bits of memory and allows 8 bit processes. To process 128-bits, 16 S-boxes are needed. Implementation of 16 S-boxes for AES requires 32kb memory. Dedicated embedded memory blocks are ideal for implementing S-box. Our design uses the special feature of Xilinx Spatan-3 FPGA [8] offering

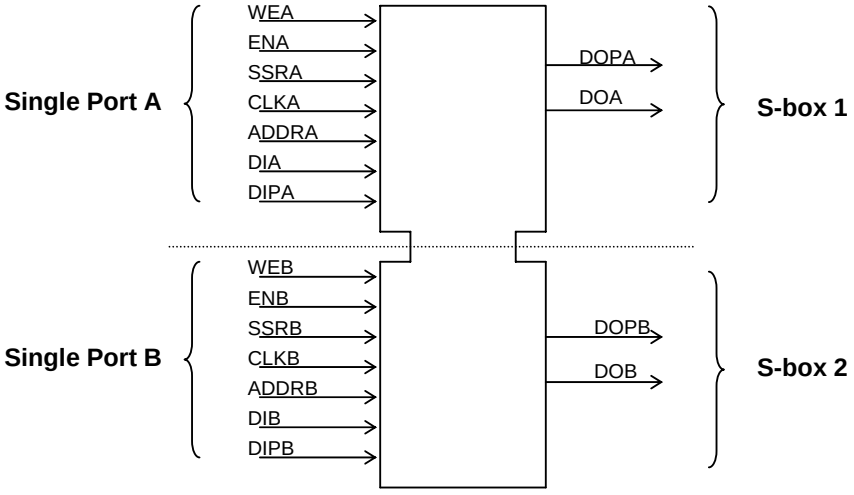

Figure 4. One Block RAM becomes two independent Single-Port RAMs

multiple block RAM memories, organized in columns. Each block RAM contains 18kb of fast static RAM [9].

The Xilinx Spartan-3 xc3s200pq208-5 has 12 block RAMs. These block RAMs can either be used as Single or Dual port RAM. In Spartan-3 FPGA, a Dual port RAM can also be configured as two separate Single port RAMs as shown in Fig. 4. One S-box can be implemented per block RAM only, therefore utilising each Dual port RAM block as two single port RAM, 16 S-boxes are realisable in only 8 block RAMs.

5.2 ShiftRows Step

The ShiftRows transformation can be expressed as an arrangement of the matrix using an address expression for each element. The address expression calculates row dependant circular shift of rows. The circular shift operation uses routing only with no dedicated hardware resource required.

5.3 MixColumns Step

The MixColumns transformation maps one column of the input *state* to a new column *state*. The transformation is based on a four-byte input. Our design uses the mathematical properties of multiplication by a fixed constant in GF(2^8), optimally utilizing combinational logic circuit. The optimized hardware implementation considers its four byte input as a polynomial over GF(2^8) and is capable of performing a multiplication of the input with the constant polynomial. This zero clock cycle parallelism reduces the entire MixColumn transformation to combinational logic, efficiently exploited by us in our implementation.

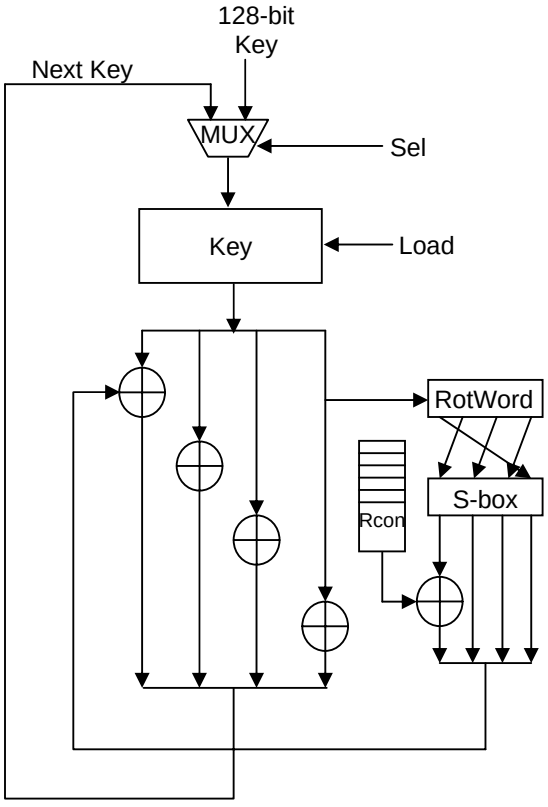

Figure 5. Key Expansion

5.4 AddRoundKey Step

The key scheduling, presented in Fig. 5 uses on the fly key generation. The initial key is 128-bit and XORing of previous column expands the round keys from initial key. For columns that are in multiple of four, the process involves Shift operation (Rot), byte substitution (S-box) and a round constant (Rcon) addition. The round keys are expanded as required for each round. On the fly key generation method produces the keys needed for the round at every clock cycle and does not store all the keys in the register, thus decreasing the resources needed for overall design. This technique is efficient when only encryption is required as is the case with CCMP (AES-CTR and CBC-MAC).

6 IMPLEMENTATION RESULTS AND COMPARISON

The system has been designed in Verilog HDL. ModelSim 6.0 has been used as functional simulator and Xilinx ISE Foundation 6.2i used for performing logic synthesis, mapping, placing and routing. Xilinx XST tool was optimised for area and high effort optimisation for synthesis. The targeted device was Spartan-3 xc3s200pq208-5 with detailed specifications at [8].

Our design occupies 481 Slices (25% of total offered) and 10 Block RAM. Working at frequency of 231.97 MHz, throughput of 2699 Mbps has been achieved. The data is accepted after every 11-clock cycle. A comparison of our results has been carried out with other sequential implementations for 128 bit AES in Table 1. It is clear from the table that our work stands out on account of being efficient implementation with minimized inter routing delays, on low-cost and low-power Spartan-3 device utilising fewer Slices and exhibiting higher speed of 2699 Mbps achieving throughput per area of 5.6 Mbps/Slice.

Table 1. Comparison with other AES implementations.

Implementation	K. Vu [10]	A. J. Elbirt [11]	Dandalise [12]	S. M. Farhan [13]	F. Standaert [14]	**Our Work**
Platform Device	Xilinx 2s200pq208-5	Xilinx xcv1000bg560-4	Xilinx xcv1000	Xilinx x2v1000	Xilinx xcv3200e	Xilinx xc3s200pq208-5
Available Slices	2035	3528	5673	336	542	481
Throughput (Mbps)	--	294.2	353	53	1450	**2699**
Block RAMs	--	0	0	2	10	10
Frequency	43.337MHz	25.3MHz	---	110MHz	119MHz	231.97 MHz
Throughput/Area (Mbps/Slices)	--	0.083	0.062	0.157	2.67	**5.6**

7 CONCLUSION

This paper present an efficient and low power FPGA implementation of AES block cipher to satisfy the security requirements of today's Robust Secure Wireless Network (RSN). Our implementation, characterised by high throughput, low-cost and low-power consumption, is a candidate option for practical use in improving the speed, efficiency and processing power of CCMP. Offering encryption rate of 2699 Mbps for CCMP, it not only meets the current IEEE 802.11 operating data

rates of 54Mbps and 108Mbps (Super G Mode), but also the high speed requirement of emerging wireless standard like IEEE 802.11n which will support a data rate of 500 Mbps.

Future work includes further reduction in the area and power by using quarter of round approach and making certain other possible design improvements as well as using different optimization techniques.

REFERENCES

1. Borisov Nikita, Goldberg Ian and Wagner David, "Security of the WEP Algorithm" http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html University of California at Berkeley, February 2001.
2. J. Edney, W.A. Arbaugh, "Real 802.11i Security Wi-Fi Protected Access and 802.11i", Addison-Wesley, August 2003.
3. J. Daemen and V. Rijmen, "AES Proposal: Rijndael, AES algorithm submission", September 3, 1999, available: http://www.nist.gov/CryptoToolkit.
4. "Draft FIPS for the AES", available from: http://csrc.nist.gov/encryption.aes, February 2001.
5. National Institute of Standards and Technology http://csrc.nist.gov.
6. Modes of Operation for Symmetric Key Block Cipher available at http://csrc.nist.gov/ CryptoToolkit/modes/.
7. M. Dworkin, " Recommendation for Block Cipher Modes of Operation: The CCM Mode For Authentication and Confidentiality", NIST special Publication 800-38C. May 2004. http://csrc.nist.gov/publications/nistpubs/800-38C/SP800-38C.pdf.
8. Xilinx. Spartan-3 Field Programmable Gate Array data sheets available at http://www. xilinx.com/spartan3
9. Xilinx "Using Block RAM in Spartan-3 FPGAs" available at http://www.xilinx.com/xapp/ xapp463.pdf
10. K. Vu, D. Zier, "FPGA Implementation AES for CCM Mode Encryption Using Xilinx Spartan-II." ECE-679, Oregon State University, Spring 2003.
11. A.J. Elbirt, W. Yip, B. Chetwynd and C.Par, " An FPGA Implementation and Performance Evaluation of the AES Block Cipher Candidate Algorithm Finalists", Third AES Candidate Conference, April 2000.
12. Dandalis, V.K. Prasanna and J.D.P. Rolim, "A Comparative Study of AES Final Candidates Using FPGAs.", Cryptographic Hardware and Embedded Systems 2000, Workshop, CHES 2000, Worcester, MA, August 17-10, 2000.
13. S.M. Farhan, H. Jamal and M. Rahmatullah, "High Data Rate 8-bit crypto-processor." ISSA 2004 enabling tomorrow Conference, 30 June – 2 July 2004, Gallagher Estate, Midrand, SOUTH AFRICA.
14. F.X. Standaert, G. Rouvoy, J.J. Quisquater, J.D. Legat, "Efficient Implementation of Rijndael Encryption in Reconfigurable Hardware: Improvements and Design Tradeoff." Cryptographic Hardware and Embedded Systems Workshop (CHES) 2003, LNCS vol 2779, pp 334-350, Springer-Verlag.