FPGA Implementation and Performance Evaluation of AES-CCM Cores for Wireless Networks

Ignacio Algredo-Badillo, Claudia Feregrino-Uribe, René Cumplido, Miguel Morales-Sandoval
Department of Computer Science, INAOE
Luis Enrique Erro 1, Puebla, México
{algredobadillo, cferegrino, rcumplido, mmorales}@inaoep.mx

Abstract

Reconfigurable architectures are important elements on the design of software radios. Nowadays, diverse platforms are being developed to support multiple tasks; these platforms are designed specially for the different layers of the OSI (Open System Interconnection) reference model. Specifically, the security architectures described in the MAC sublayer should be evaluated, which are based on cryptographic algorithms that require high computational costs. In this work, two proposed AES-CCM hardware architectures for the IEEE 802.11i-2004 and IEEE 802.16e-2005 standards are implemented in diverse FPGA devices to examine implementation costs and performance evaluation. The results presented in this work will be used for designing and developing a reconfigurable platform with software-radio applications, which will include the high-performance AES-CCM hardware architectures meeting the specifications of the IEEE 802.11i-2004 and IEEE 802.16e-2005 standards.

1 Introduction

There are many types of networks that have been widely developed, but the wireless ones have grown significantly more due to their mobility, that is an increasingly important feature for users. The wireless networks use different set of rules or protocols for governing the communication among diverse devices, and each network has applications that can use different protocols. Ideally, a device should operate in the diverse applications of the different wireless networks. This last idea is conceptualized by using software radios, which have several configurations for operating in different communication networks. Software radios have changed according to the technology advances, where is possible to find a basic radio architecture, which has a key element that configures the radio to operate in the different networks [1]. Considering the OSI model, the main development of these radios is focused on the lower layers, which are implemented in hardware. Security is a key element for using software radios, because these can enter to different wireless networks and use the air like transmission medium, being vulnerable to possible data transmission attacks. Several security architectures have been standardized for different networks, such as the IEEE 802.11i-2004 for WLANs (Wireless Local Area Networks) and the IEEE 802.16e-2005 for WMANs (Wireless Metropolitan Area Networks), operating on the MAC (Medium Access Control) sublayer.

In cryptography, diverse types of algorithms are focused on offering different security services. Thus, confidentiality, authenticity, integrity, and non-repudiation services are provided by symmetric and asymmetric algorithms and hash functions [2]. Recently, the CCM mode is defined and used in security schemes for wireless communication networks, such as in the IEEE 802.11i-2004 and IEEE 802.16e-2005 standards. The AES-CCM algorithm executes two related processes: generation-encryption and decryption-verification. This work presents the design and implementation of hardware architectures for the generation-encryption process, which are well suited to be integrated in a complete transmission platform. The hardware architectures are implemented in diverse FPGA devices and the implementation costs and performance evaluation are examined.

The rest of this paper is organized as follows: Section 2 revises the security architectures of the IEEE 802.11i and IEEE 802.16e standards. Section 3 describes the AES-CCM algorithm. Section 4 presents the details of the proposed AES-CCM hardware architectures. Section 5 presents the FPGA implementations and results. Section 6 shows comparisons against related work and finally, Section 7 gives the conclusion of this work.

978-0-7695-3474-9/08 $25.00 © 2008 IEEE
DOI 10.1109/ReConFig.2008.54

2 Security Architectures

Security architectures of the standards IEEE 802.11i-2004 and IEEE 802.16e-2005 standards establish a set of cryptographic algorithms to provide different security services. The AES-CCMP (AES-CCM Protocol) provides data confidentiality, integrity, and replay-attack protection, operating on the MAC Protocol Data Unit (MPDU), see Fig. 1 [3]. MPDU contains several fields, including, for example, the payload, AAD (Additional Authentication Data), and the header of the MAC sublayer. The output of the AES-CCMP algorithm are the ciphered data and the Message Integrity Code (MIC). A temporal key (TK) is required for every ciphering session.

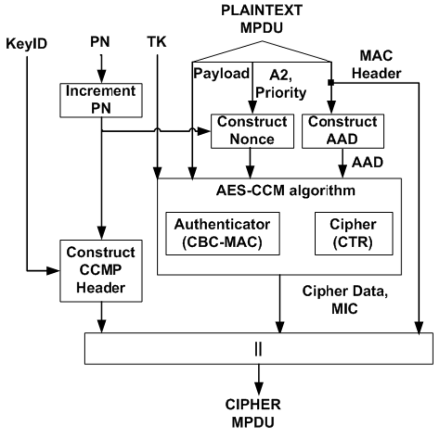

Figure 1. Security architecture based on the AES-CCMP for IEEE 802.11i-2004 standard.

In the same way, in the IEEE 802.16e-2005 security scheme [4], see Fig. 2, data are protected by ciphering the information or plaintext payload, and by providing a value for the message integrity, but this security architecture sets to zero the length of the AAD field. Details of the AES-CCM algorithm are described in the next section.

2.1 The AES-CCM Algorithm

The parameters of the AES-CCM Algorithm are defined in the NIST CCM specification [5]. In IEEE 802.11i-2004 and IEEE 802.16e-2005, these parameters shall be fixed to specific values: i) the number of bytes in the Message Authentication Code field shall be set to eight, and ii) the size of the length field (bit string representation of the octet length of the payload) shall be set to two. The difference is that the IEEE 802.11i-2004 standard establishes AAD with a maximum length of 32 bytes, whereas in the IEEE 802.16e-2005 standard the length of the AAD shall be set to zero. AES-CCM operates on several fields, which are formatted for providing 128-bit data blocks, for example, BXs are blocks from the payload, AAD, and nonce value, whereas CBs are blocks from the nonce value and the length of the payload. It outputs the ciphertext payload and the MIC value that are used together with modified generic MAC headers to build the cipher MPDU.

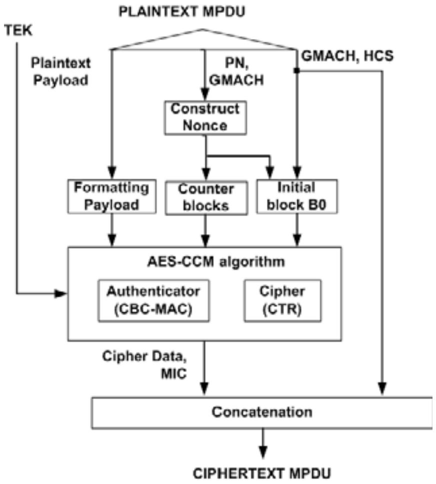

Figure 2. Security Architecture of the IEEE 802.16e-2005 Standard.

AES-CCM is based on two operation modes: CBC-MAC (Cipher Block Chaining - Message Authentication Code) and CTR (Counter), see Fig. 3. CBC-MAC process is applied to the BX blocks to generate a T value whereas CTR mode is applied to the T, BX and CB blocks to obtain the MIC value and the ciphertext payload (cipher MPDU).

In general, AES-CBC-MAC takes the first block and ciphers it using AES. An XOR operation is performed by using the previous result with second block, and this result is ciphered. This process is applied for the remaining blocks. CBC-MAC works sequentially and it cannot be parallelized. The final result is the MIC value (T), which is used in the AES-CTR process.

When ciphering two identical input blocks, CTR mode produces different cipher blocks, which is based on a nonce value rather than starting it from a fixed value. This mode provides authentication by adding extra capabilities. Some properties of CTR is that ciphering can be done in parallel, decryption is the same process as encryption, and the message does not need to be broken into an exact number of blocks [5].

AES-CBC-MAC and AES-CTR are constituted by a common algorithm, and they have AES block cipher as elemental part. This is a symmetric block cipher that can process data blocks of 128 bits and it uses cipher keys of 128, 192, and 256 bits [6]. All AES processing in

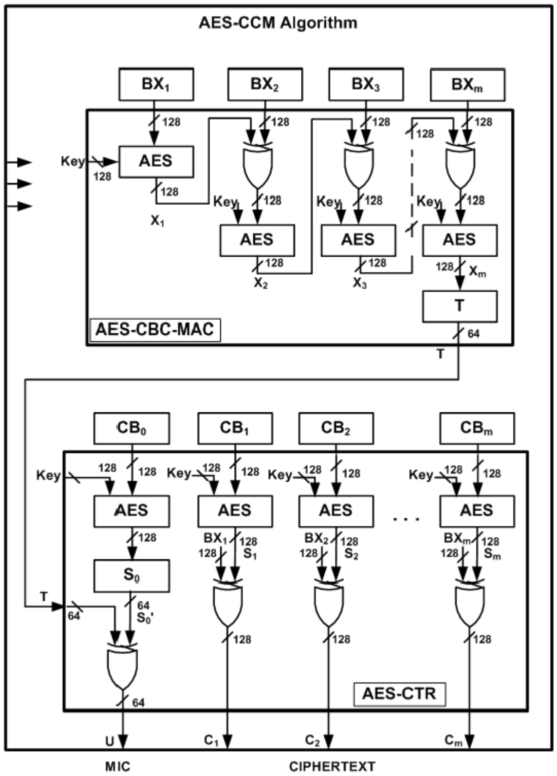

Figure 3. Block diagram of the AES-CCM algorithm.

CCM encryption uses AES with a 128-bit key and a 128-bit block size. AES executes an initial round followed by ten rounds with four main operations: i) byte-to-byte substitution (SubByte), ii) rotation of rows (ShiftRow), iii) mixing of columns (MixColumn), and iv) addition of round key (AddRoundKey).

The ciphertext is produced after these rounds. Other operation is key expansion, which computes a key schedule or a 128-bits key in each round. The non-linear byte substitution and key expansion operations require S-box substitution, where one byte is substituted and determined by the intersection of the row and the column.

The use of cryptographic algorithms requires computing complex operations, which may result in system bottlenecks in applications that transmit great amounts of data. For future data transmissions such as in the new wireless networks [7], data is transmitted at 1 Gbps for applications such as high quality TV, movies in DVD, and great amount of digital files using personal computers, among others. So, it is necessary to reach such speeds, and this can be achieved with hardware architectures for software radios with high performance and high throughput/area ratio of the algorithms. The next section shows the proposed AES-CCM hardware architectures.

3 AES-CCM hardware architectures

A careful analysis of the AES-CCM algorithm allowed exploiting parallelization of some processes and designing highly specialized processing modules in order to achieve the highest throughput when compared against related works. The design methodology of the AES-CCM architectures is based on a straightforward architecture, which is balanced and parallelized to decrease the critical path (increased performance). Firstly, the specialized modules are designed, parallelizing the data buses and processes such as CTR, CBC-MAC, and S-boxes computations. After, the paths formed by the combinational elements (i.e. multiplexors and gates) and sequential elements (i.e. registers and memories) are balanced, moving these last ones between the combinational elements. The aim is to reduce the critical path without increasing the latency. The highly-parallelized architectures designed are AESCCMiv1 (see Fig. 4) and AESCCMev1 (see Fig. 5), for IEEE 802.11i-2004 and IEEE 802.11e-2005, respectively. In these architectures, the data buses and processes are parallelized, computing two AES processes in different modes of operation. These processes require the data buses of the payload and key at the same time.

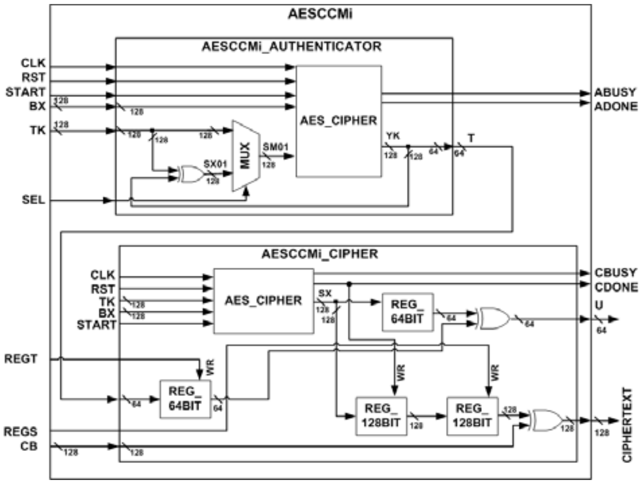

Figure 4. Block diagram of the AESCCMi module.

An additional analysis was made to decrease both the critical path and used hardware resources, improving efficiency. It is reached by using a common module. This module is selected by analyzing the two AES processes used in the AES-CTR and AES-CBC-MAC, where the generation key (module AES_GenKey, see Fig. 6) produces the same

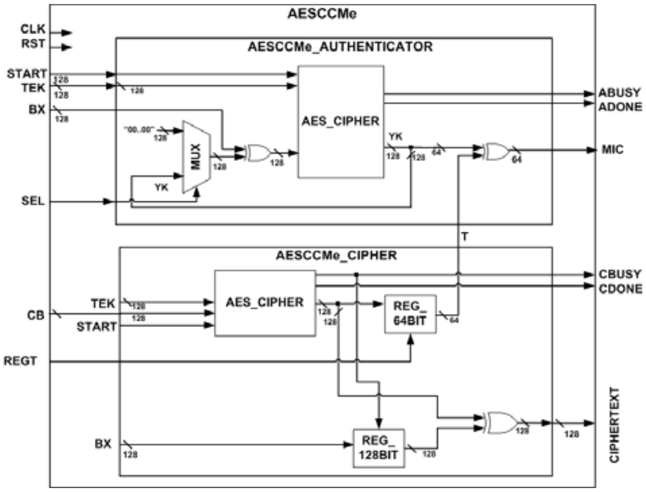

Figure 5. Block diagram of the AESCCMe module.

key. AES_Cipher has a compact and iterative architecture with a high performance, more details in [8].

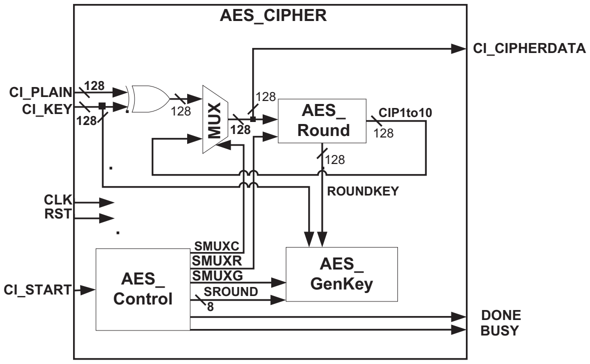

Figure 6. Block diagram of the proposed AES hardware architecture

Considering these AES_GenKey common module, the new designs of the architectures are AESCCMiv2 and AESCCMev2. In each standard, the proposed block diagram is the same for the architectures versions 1 and 2, but the key expansion of each AES_Cipher is the common module. The AESCCMi architecture has differences against the AESCCMe architectures because of the AAD field, see Section 2, requiring more registers to store data blocks.

4 Implementation results

Synthesis results for the proposed AES-CCM hardware architectures are presented in this section. For the purpose of validation and prototyping, these architectures were synthesized, mapped, placed and routed for the Xilinx's Viterx-4 and Spartan-3 devices using ISE 9.2 design tools. The implemented architecture was simulated and verified considering real-time operation condition by using the design conformance test data, provided with the IEEE 802.11i standard.

Two metrics are considered for evaluating these architectures and comparing them against previous works. These metrics are throughput and implementation efficiency. The throughput is computed by Eq. 1, measured in bits per second (bps), where Clock_frequency is obtained by implementing in the different FPGA technologies.

$$Throughput = \frac{(Data\_input) \times (Clock\_frequency)}{Clock\_cycles} \tag{1}$$

The other metric is the implementation efficiency, see Eq. 2, and it is a measure of this type of cryptographic hardware implementations, which is defined as the ratio between the reached throughput and the number of slices that each implementation consumes (bps/slice) [9].

$$Efficiency = \frac{Throughput}{Area} \tag{2}$$

To set values of Data_input and Clock_cycles, it is highlighted that computing MIC and cipherdata is executed in parallel by the proposed AES-CCM architectures, and several details are considered for computing the throughput: 1) in these security architectures, the message has a maximum size of 1024 bytes, thus 64 data blocks of 128 bits are obtained, 2) AAD has a maximum size of 32 bytes for IEEE 802.11i-2004, and 0 bytes for IEEE 802.11e-2005, thus 2 data blocks are constituted for the first standard, and 3) there is an initial data block, which is formed by using the Nonce value, see Section 2.

For the AESCCMi hardware architectures, versions 1 and 2, these process 67 data blocks (64 for the message, 2 for AAD and 1 for the initial data block), but the initial data block is considered overhead, so only 66 data blocks have effective bits, i. e., (66) (128 bits) = 8448 bits = Data_input. These data blocks for authentication (initial block, AAD, and message) and ciphering (CBs and message) are processed in parallel, so, it is necessary to process 67 data blocks, requiring (67) (10 clock cycles) = 670 clock cycles. The 10 clock cycles are used for the ciphering process by the AES_Cipher modules. An extra clock cycle is used for loading the initial data block, whereas the next data blocks are loaded during the processing of the previous data block. So, Clock_cycles = 671 clock cycles.

For the AESCCMe architectures, versions 1 and 2, due to the length of AAD is zero, Data_input = (64) (128 bits) = 8192 bits, and Clock_cycles = 651 clock cycles.

In Table 1, FPGA implementation costs are shown for the proposed AESCCMi and AESCCMe architectures. The versions 2 use fewer hardware resources than versions 1, due to the common module. Further, implementations in Virtex-4 use more hardware resources than the implementations in Spartan-3. This FPGA hardware resource requirement enables to develop a reconfigurable architecture with a basic structure for ciphering in the IEEE 802.11i-2004 and IEEE 802.16e-2005 security architectures. The reconfigurable architecture can be analyzed by implementing the versions 1, and after, by implementing the version 2, because they have a close number of hardware elements.

Table 1. FPGA implementation costs of the AES-CCM hardware architectures.

Architecture	Virtex-4	Spartan-3	BRAM&IOB
AESCCMiv1	1533 Slices 2707 LUTs	1309 Slices 2603 LUTs	20 BRAMs 586 IOBs
AESCCMiv2	1200 Slices 1995 LUTs	1041 Slices 1661 LUTs	18 BRAMs 586 IOBs
AESCCMev1	1453 Slices 2702 LUTs	1105 Slices 2058 LUTs	20 BRAMs 582 IOBs
AESCCMev2	1333 Slices 2477 LUTs	951 Slices 1754 LUTs	18 BRAMs 582 IOBs

In general, if more hardware resources are necessary, then the critical paths can be larger. The period is obtained by implementing the architecture in a given FPGA device. Clock_frequency is the inverse of the period, which is obtained from the reports of the ISE tool. Using this tool, several Virtex-4 and Spartan-3 FPGA devices were considered, and based on Eq. 1 and Eq. 2, efficiency and throughput results are shown in Fig. 7. Results of implementing the architectures in Spartan-3 devices were obtained in order to make a fair comparison with other works that are based on these devices. Results of implementing the architectures in Virtex-4 devices were obtained in order to show the benefits of the proposed approach using state of the art FPGAs in their different versions.

In the case of the performance evaluation, the throughput of the AES-CCM hardware architectures varies conforming to the used devices (see Fig. 7). In general, implementations on Virtex-4 series LX report higher throughput than the implementations in the other series. Comparing hardware architecture versions 1 against versions 2, in most of the cases, the throughput of the seconds is improved because they require fewer hardware resources, reporting a smaller critical path. These two last characteristics, resources and critical path, enable an improved efficiency, see Fig. 7.

For each FPGA family, Virtex-4 or Spartan-3, efficiency of all implementations is very similar, and versions 2 have better efficiency than versions 1.

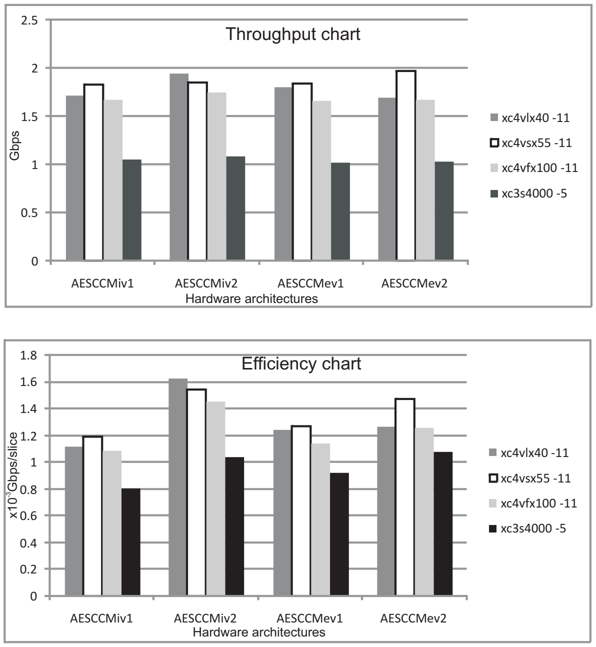

Figure 7. Throughput results.

The architectures in Virtex-4 report an improved performance than the ones in Spartan-3, which is reached by a better technology, and in this point, it is important to highlight that the key element is the architectural design, that enables to increase the throughput when it is compared against related work. In the next section, these comparisons are made, using a same FPGA technology and showing the advantages of the parallelized and balanced iterative architectures.

5 Comparisons

Comparison results of the AESCCM hardware implementation are shown in Table 2. The main goal in the hardware implementations is the highest throughput, which is reached by the AESCCMiv2 architecture in both FPGA families. Considering efficiency, the AESCCMiv2 implementation in Spartan-3 achieves lower efficiency when compared against [11], which presents higher efficiency at 247 MHz, using less hardware resource requirements with a lower throughput. However, the AESCCMiv2 achieves 58.6% higher throughput with a slower clock frequency of 86.34 MHz. This metric of the efficiency is based on the slices, but considering BRAMs, the same situation occurs, because [10] uses more memories and [11] reports few hardware resources. The proposed AESCCMiv2 architecture implemented in Virtex-4 device achieves the highest throughput and the highest efficiency.

Table 2. AES-CCM hardware implementations

Work	Device	Clock Frequency (MHz)	Slices	BRAM	Throughput (Gbps)	Efficiency x10^{-3} (Gbps/slices)
[10]	XC3S4000	100.07	2154	106	1.051	0.488
[11]	XC3S50	247.00	487	4	0.687	1.411
[12]	ASIC	36.00	-	-	0.800	-
This work	XC3S4000	86.34	1041	18	1.087	1.044
AESCCMiv2	XC4VLX40	152.42	1200	18	1.951	1.626

6 Conclusions

The hardware design methodology used in this work aimed to obtain an iterative hardware architecture with high throughput and efficiency, resulting in a balanced implementation. In most of the cases, it was shown that highly-parallelized architectures have lower throughput and efficiency, which are improved by architectures with common modules. The proposed AESCCMi and AESCCMe hardware architectures have similar hardware resource requirements and report comparable throughput and efficiency, which enable to design a reconfigurable architecture for cryptographic processing in software-radio applications for the IEEE 802.11i-2004 and IEEE 802.16e-2005 security architectures, accelerating computation of the AES-CCM algorithm.

References

[1] Center for Software Defined Radio. Software defined radio: Terms, trends and perspectives. White paper, 2007. http://www.csdr.ck.

[2] B. Schneier. Applied Cryptography. JohnWiley Sons, NY, 1996.

[3] LAN/MAN Standards Committee. Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE Std 802.11i-2004. IEEE Computer Society, 2004.

[4] LAN/MAN Standards Committee of the IEEE Computer Society and the IEEE Microwave Theory and Techniques Society. Part 16: Air interface for fixed and mobile broadband wireless access systems. IEEE Std 802.16e-2005, IEEE Standard for Local and Metropolitan Area Networks, 2006.

[5] M. Dworkin. Recommendation for block cipher modes of operation: The CCM mode for authentication and confidentiality. NIST Special Publication 800-38C., 2004.

[6] FIPS, Announcing the advanced encryption standard (AES). Federal Information Processing Standards Publication 197 (FIPS-197), 2001.

[7] ICT-Centre. Multi gigabit millimeter wave wireless. innovative ICT transforming australian industries. Federal Information Processing Standards Publication 197 (FIPS-197), 2001.

[8] Algredo-Badillo I., Feregrino-Uribe C., Cumplido-Parra R., Design and Implementation of an FPGA-Based 1.452- Gbps Non-pipelined AES Architecture, ICCSA 2006, Lecture Notes in Computer Science 3982, pp. 446-455, Springer-Verlag, 2006.

[9] P. Kitsos. Hardware implementations for the ISO/IEC 18033-4:2005 Standard for Stream Ciphers. Journal of Signal Processing, 3(1):6673, 2006.

[10] E. López-Trejo, F. Rodríguez-Henr, and A. Díaz-Pérez. An efficient FPGA implementation of CCM using AES. In Proc. of the 8th International Conference on Information Security and Cryptology (ICISC05), volume 3935 of Lecture Notes in Computer Science, pages 208215. Springer, 2005.

[11] A. Aziz and N. Ikram. An FPGA-based AES-CCM crypto core for IEEE 802.11i architecture. International Journal of Networks Security, 5(2):224232, 2007.

[12] Elliptic Semiconductor Inc. CLP-20 high performance AES-CCM core. Datasheet, 2008. http://www. ellipticsemi.com.