ADVANCED ENCRYPTION STANDARD (AES) COPROCESSOR

LIM JOO SONG

UNIVERSITI TEKNOLOGI MALAYSIA

ADVANCED ENCRYPTION STANDARD (AES) COPROCESSOR

LIM JOO SONG

A project report submitted in partial fulfillment of the
requirements for the award of the degree of
Master of Engineering (Electrical - Computer & Microelectronic System)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JUNE 2014

ABSTRACT

The purpose of this project is to design a high throughput Advanced Encryption Standard (AES) coprocessor using SystemVerilog Hardware Description Language (HDL). AES coprocessor has been widely used to offload the compute intensive cryptography tasks from the main processor due to its efficiency and performance compare to the pure software solution. Conventional AES core design using iterative loop approach is not optimized for high throughput operation. Therefore pipelined architecture is the most recommend method for high throughput design. However most of the high throughput designs tend to reliance on vendor specific features to boost the performance. This has caused the design becomes non-generic and low portability. A bottom-up approach has being used to design the coprocessor. A non-pipelined AES coprocessor was being built first as the baseline design. Then different types of pipelined implementation were being explored for possible adoption. The throughput, size and potential enhancement are the main criteria being evaluated. It was discovered that the Full Outer-Round pipelined architecture is the most efficient design in term of throughput and size. From the analysis of the Full Outer-Round pipelined model, a novel method called Single Datapath Dual Output (SDDO) has been proposed to double the throughput of the pipelined coprocessor. Simulation results demonstrated that SDDO architecture is able to double the throughput of the pipelined design while only requires 7% of additional resource to implement.

ABSTRAK

Projek ini bertujuan mereka betuk kopemproses *Advanced Encryption Standard* (AES) yang berthroughput tinggi dengan menggunakan SystemVerilog HDL. Kopemproses AES telah digunakan secara meluas untuk mengambil alih tugas pengiraan intensif kriptogafi daripada pemproses utama kerana ia mempunyai kecekapan dan prestasi yang lebih tinggi daripada kaedah menggunakan preisian. Kebiasaanya reka bentuk AES yang menggunakan kaedah interaktif tidak dapat menghasilkan *throughput* yang tinggi. Justeru itu, seni bina *pipeline* adalah kaedah yang biasa disyorkan untuk tujuan membina reka bentuk yang dapat menghasilkan *throughput* yang tinggi. Walau bagaimana pun, kebanyakan reka bentuk yang berkemampuan tinggi biasanya bergantung kepada ciri-ciri istimewa yang ditawarkan oleh vendor tertentu untuk menjana prestasi yang tinggi. Ini menyebabkan reka bentuk tersebut tidak generik dan mempunyai kemudahalihan yang rendah. Kaedah yang digunakan untuk mereka bentuk kopemproses dalam projek ini adalah kaedah *bottom-up*, dimana kopemproses AES yang tidak berpipeline dibina dahulu untuk tujuan menjadikan reka bentuk asas. Selepas itu, berbagai jenis reka bentuk pipeline dikaji dengan terliti. Jenis kriteria yang diselidik termasuk throughput, saiz dan keupayaan peningkatan. Hasil daripada penyelidikan tersebut menunjukkan reka bina pipeline Full Outer-Round adalah lebih cekap dari segi throughput and saiz. Selain daripada itu, usaha dalam membina modal Full Outer-Round pipline telah hasilkan satu kaedah baru yang dikenali sebagai Single Datapath Dual Output (SDDO) , diman ia dapat mengandakan throughput daripada kopemproses pipeline. Keputusan simulasi menunjukkan kaedah SDDO memang dapat mengandakan throughput reka bina pipeline dengan hanya memerlukan penambahan sumber sebanyak 7% sahaja.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AES	-	Advanced Encryption Standard
NIST	-	National Institute of Standard and Technology
XOR	-	Exclusive OR operation
LUT	-	Look Up Table
RAM	-	Random Access Memory
FPGA	-	Field Programmable Gate Array
HDL	-	Hardware Description Language
Gbps	-	Gigabit per second
GF(x)	-	Galois Field of x element
SBox	-	Substitution-box
BRAM	-	Block RAM
ROM	-	Read-only Memory
MHz	-	Megahertz
DMA	-	Direct Memory Access
CPU	-	Center Processing Unit
RTL	-	Register Transfer Level
CLB	-	Configurable Logic Block
CAM	-	Content Addressable Memory
DFG	-	Data Flow Graph
CU	-	Control Unit

DU	-	Datapath Unit
KeyGen	-	Key Generator
ASM	-	Algorithmic State Machines
Mop/s	-	Mega-operation per second
CPD	-	Critical Path Delay
SDDO	-	Single Datapath Dual Output

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Advanced Encryption Standard (AES) is security encryption standard selected by the United State National Institute of Standard and Technology (NIST) in 2001 for electronic data protection purpose [1]. The AES algorithm can be implemented in software, firmware, hardware or combination design. However, since the AES algorithm is compute intensive, the hardware implementation is the preference design due to performance and efficiency. Study shows the hardware implementation of AES algorithm is 60 times faster than the pure software approach [2]. Therefore the AES coprocessor is being widely used to offload the compute intensive cryptography tasks from the main processor.

In today environment, digital communication and data exchange becomes the essential part of modern daily lives. Everyone regardless of social and economy status will be direct or indirect impacted by how secure the data get exchanged in the communication network especially through the internet. The amount of data in the internet transection is growing exponentially due to increase demand on media content and online social networking activities. It is estimated by 2016, annual global IP traffic is forecasted to be 1.3 zeta-bytes [3]. Hence, a high throughput AES coprocessor is needed to accelerate the processing of the electronic data in real time.

AES is a subset of the Rijndael cipher developed by by Joan Daemen and Vincent Rijmen [4]. The Rijndael cipher allows data block and cipher key length in the multiple of 32 bits with a minimum of 128 bits and maximum of 256 bits. AES limits the data block length to 128 bits only and cipher key length of 128, 192 and 256 bits. AES is symmetric block cipher where a similar key is being used to perform both encryption and decryption on a fixed length block to block basis.

1.2 Keyword and Definition

The following definitions and Keyword are used throughout the document:

AES	Advanced Encryption Standard
Array	A collection of identical entities.
Affine Transformation	A transformation through the multiplication by a matrix and followed by the addition of a vector.
Bit	Binary digit having a value of 0 or 1.
Block	An array of bytes in a fixed length group. In this document, the Block size is 16 bytes.
Byte	A group of eight bits.
Cipher	Sequence of transformations that converts a plaintext to cipher text using a Cipher Key. It is also known as Encryption process.
Cipher Key	Secret key that is used to encrypt or decrypt the plain text and cipher text through the cipher process.
Cipher text	Result from the Cipher. Input to the Inverse Cipher.
Inverse Cipher	Sequence of transformations that converts cipher text to plain text using a Cipher Key. In this document it is known as Decryption process.
Key Expansion	Routine used to generate Round Keys from the Cipher Key.

Plain text	Data input to the Cipher function. Result from the Inverse Cipher
Rijndael	Cryptopgraphic algorithm developed by Joan Daemen and Vincent Rijmen.
Round Key	Series of keys that are derived from the Cipher Key using the Key Expansion routine. The keys are applied during Encryption and Decryption
State	Intermediate result from the Cipher. Can be presented as a rectangular array of bytes with four rows and columns.
S-box	Non-linear substitution table used in bytes substitution transformations and Key Expansion routine to perform one-to-one substitution of a byte value.
Word	Group of 32 bits that is treated either as a single entity consist of 4 bytes.
AddRoundKey	Transformation process in which a Round Key is added to the State using XOR operation.
InvMixColumns	Transformation step during the Inverse Cipher in which the process is inverse of the MixColumns.
InvShiftRows	Transformation step during the Inverse Cipher in which the process is inverse of the ShiftRows.
InvSubBytes	Transformation step during the Inverse Cipher in which the process is inverse of the SubBytes
MixColumns	Transformation step during the Cipher in which all the columns of the State are mixed in a certain way to produce new columns.
Rcon	Round constant value.
RotWord	Process of performing a cyclic permutation on a 4 bytes word during Key Expansion.
ShiftRows	Transformation step during the Cipher in which the last three rows of the State is cyclically shifted by different offsets.
SubBytes	Transformation step during the Cipher in which each of

	the State bytes is substituted by byte value from the S-box.
SubWord	Process of taking a 4 bytes input word and applies an S-box to each of the 4 bytes to produce an output word during Key Expansion.
XOR	Exculsive-OR operation.

1.3 Problem Statement

During encryption and decryption, AES algorithm applies four different byte-oriented transformations: SubBytes, ShiftRows, MixColumns and AddRoundKey in one round execution of the algorithm. A complete operation consists of applying several round of execution, depending on the Cipher Key size. Key length of 128 bits is known as AES-128, 192 bits is AES-192 and 256 bits is known as AES-256. Table 1.1 below shows the number of round required by the AES specification [1].

Table 1.1: Key-Block-Round Relation [1]

	Key Length, N_k (words of 4 bytes)	Block Size, N_b (words of 4 bytes)	Number of Rounds, N_r
AES-128	4 (128 bits)	4 (16 bytes)	10
AES-192	6 (192 bits)	4 (16 bytes)	12
AES-256	8 (256 bits)	4 (16 bytes)	14

AES encryption and decryption process involves at least 10 round of repeated transformation steps. In order to achieve that process, conventional AES processing core design normally employs iterative loop approach which is not optimized for high throughput operation. For high throughput design, pipelined architecture is well known to be the main design of choice.

The AES decryption process is mainly applying the encryption steps in the reserve order using different set of Round Key. This seems to require two different set of process sequence to perform the decryption steps. Due to this reason, most of the high throughput designed AES core only implements the encryption sequence. However, the AES algorithm allows the implementation of the Equivalent Inverse Cipher that employs the same sequence of transformation similar to the Cipher using the inverse operation. Given the minimum hardware resource overhead required for such encryption-decryption dual function design, it is desire to have a single high throughput design which combining both encryption and decryption function.

Since the AES was introduced in 2009, there has been a lot of study on how to design and implement fast and efficient AES coprocessor, especially on the FPGA device. However, most of the design is very device and vendor specific. The intention is to utilize the vendor specific feature such as Look-Up Table (LUT) logic and RAM Block to cut down the latency of the processing in order to improve the throughput. This has limited the portability of the design. A pure logic design will be preferable as it will avoid the re-design of the system when migrating from one vendor device to another.

1.4 Objective

The objective of the project is to design a high throughput Advanced Encryption Standard (AES) coprocessor with pipelining architecture. The coprocessor will support both encryption and decryption function in a single core design. The design will be as generic as possible using pure logic to allow for future portability across different technology.

1.5 Scope

The scope of the project is to design an AES128 coprocessor with pipelining capability. It involves re-designing the basic non-pipelined AES128 core from previous work to establish the baseline for performance evaluation, then enhancing the design with pipelined architecture to obtain a high throughput AES core. The design will be modeled with SystemVerilog HDL and simulate using Altera's Model-Sim program. Lastly, analysis will be done on the performance of the proposed pipelined architecture in term of throughput and area.

1.6 Expected Result

A System Verilog model of a high throughput pipelined architecture AES128 coprocessor which has throughput of more than 1.0Gbps (Gigabit per second), using pure combination logic entity. The design should be efficient when performing encryption/decryption with minimum overhead from data block to data block.

REFERENCES

1. National Institute of Standard and Technology (NIST) *Federal Information Processing Standards Publication 197*. November, 2001.

2. Hakhamaneshi, B. *A hardware implementation of the Advanced Encryption Standard (AES) algorithm using SystemVerilog*. Project Report, California State University of Sacramento; 2009.

3. Shetty, S. *Cisco's VNI Forecast Projects the Internet Will Be Four Times as Large in Four Years*. Press Release, CISCO, May 2012.

4. Joan Daemen, Vincent Rijmen. "AES Rpoposal: Rijndael", Version 2, National Institute of Standard and Technology, September, 2003.

5. Biglari M. et al. Maestro: A High Performance AES Encryption/Decryption System. *Proceedings of the 2013 17th CSI International Symposium on Computer Architecture and Digital Systems (CADS)*. October 30-31, 2013. Tehran, Iran: IEEE, 2013. 145-148.

6. Kshirsagar, R.V and Vyawahare, M.V. FPGA Implementation of High speed VLSI Architectures for AES Algorithm. *Proceedings of the 2012 Fifth International Conference on Emerging Trends in Engineering and Technology (ICETET)*. November 5-7, 2012. Himeji, Japan: IEEE, 2012. 239-242.

7. Borkar A. M. et al. FPGA Implementation of AES Algorithm. *Proceedings of the 3rd International Conference on Electronics Computer Technology (ICECT)*. April 8-10, 2011. Kanyakumari, India: IEEE, 2011. Vol3 401-405.

8. Zhang Y. and Wang. X. Pipelined implementation of AES encryption based on FPGA. *Proceedings of the 2010 IEEE International Conference on Information Theory and Information Security (ICITIS*) . Dec 17-19, 2010. Beijing. China: IEEE, 2010. 170-173.

9. Jyrwa, B. and Paily, R. An Area-Throughput Efficient FPGA implementation of Block Cipher AES algorithm. *Proceedings of 2009 International Conference on Advances in Computing, Control, & Telecommunication Technologies (ACT '09).* Dec 28-29, 2009. Trivandrum, India:IEEE 328-332.

10. Chih-Peng Fan et al. FPGA implementations of high throughput Sequential and Fully pipelined AES algorithm. *International Journal of Electrical Engineering*, 2008. Vol 15,No 6. 447-455.

11. Kaur, Swinder and Vig, R. Efficient Implementation of AES Algorithm in FPGA Device. *Proceedings of 2007 International Conference on Conference on Computational Intelligence and Multimedia Applications*. Dec 13-15, 2007. Sivakasi, Tamil Nadu: IEEE 179-187.

12. Nalini, C et al. An FPGA Based Performance Analysis of Pipelining and Unrolling of AES Algorithm. *Proceedings of International Conference on Advanced Computing and Communications (ADCOM 2006).* Dec 20-23, 2006. Surathkal: IEEE 477-482.

13. Kotturi, D. et al. AES Crypto Chip Utilizing High-Speed Parallel Pipelined Architecture. *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS 2005).* May 23-26, 2005. IEEE 4653-4656 Vol.5.

14. Sever, R et al. A High speed fpga Implementation of the Rijndael Algorithm. *Proceedings of the EUROMICRO Systems on Digital System Design (DSD'04).* Aug 31-Sept 3, 2004:IEEE, 358-362.

15. Saqib, N. A. et al. AES Algorithm Implementation—An efficient approach for Sequential and Pipeline Architectures. *Proceedings of the Fourth Mexican International Conference on Computer Science (ENC'03)*. Sept 8-12, 2003:IEEE 126 – 130.

16. Sklavos, N et al. Architectures and VLSI Implementations of the AES-Proposal Rijndael. *IEEE Transactions on Computers 2002*. Vol. 51, Issue 12, 1454-1459.

17. Good, T. and Benaissa, M. AES on FPGA from the fastest to the smallest. *Proceedings of 7th International Workshop in Cryptographic Hardware and Embedded Systems – CHES 2005*. Aug 29 - Sept 1, 2005. 427-440.

18. Arulpaniandi. *Design of AES Encryption Core*. Project Journal Report, University Technology Malaysia; 2004.