LOW LATENCY CIPHER FOR HARDWARE-BASED MEMORY PROTECTION UNIT FOR APPLICATION IN SYSTEM-ON-CHIP ROOT-OF-TRUST

SITI NADHIRAH BINTI ZAINURIN

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Master of Engineering (Computer and Microelectronic Systems)

School of Electrical Engineering
Faculty of Engineering
Universiti Teknologi Malaysia

FEBRUARY 2021

DEDICATION

This project report is specially dedicated to my mother, Mazlinah binti Osman, who has been my source of strength and inspiration when I thought of giving up. It is also dedicated to my father, Zainurin bin Sapuan, who has taught me the meaning of patience in facing hardship. Not to forget, to my family members, Siti Juwairiah, and Siti Maisarah for all their overflowing supports. I thank the Almighty Allah for all His blessings, for the supportive and kind people He surrounds me.

ACKNOWLEDGEMENT

In the name of Allah, the Most Gracious, and the Most Merciful.

Alhamdulillah, very thankful to Allah, for granting me the health and strength in completing this final year project. I would like to express my deepest appreciation to all those who provided me the possibility to complete this final year project successfully. A special gratitude I give to my supervisor, PM. Ir. Dr. Muhammad Nadzir Bin Marsono, who always monitoring, giving guidance and invaluable advice throughout this project. His constructive comments and thoughtful ideas had contributed a lot of positive outcomes to this project.

Furthermore, I would like to express my deepest gratitude to my beloved parents, and family who had always supported and encouraged me through this tough journey in UTM. Their endless love and motivation have brought me the strength to complete this master program. To all my dearest fellow friends especially Nurfaizah, Ain Insyirah, Nur Afiqah, Jeevan Sirkunan and Intel's colleagues, a big thank you for their endless support and encouragement and for putting colors in my life, may Allah bless you all. Lastly, I wish to express my sincere thanks to all those who had contributed in completing my final year project. Thank you very much.

ABSTRACT

The goal of trusted computing is to guarantee the behaviour of the software running on the devices. Memory Protection Unit (MPU) secures accesses between main memory, and caches which plays an important role in the System-on-Chip Root-of-Trust. There are several cipher algorithms such as SIMON, PRESENT, PRINCE, SPECK, and KATAN are suitable for MPU. These ciphers have been analyzed in different metrics such as area, power, throughput, latency, and others. Latency is one of the critical parameters that need to be considered when selecting the MPU cipher. The PRINCE algorithm has the lowest latency based on previous studies to fulfill the main MPU design requirement. The objective of this research is to synthesize and implement the PRINCE cipher architecture for the MPU to secure sensitive data exchange with an emphasis on low latency. For the hardware-based memory protection targeted for Field Programmable Gate Array (FPGA), the PRINCE cipher using three pipeline stage is designed at the register transfer level (RTL) using Verilog and verified by dynamic simulation using the Xillinx ISE. As the additional MPU structure between the main memory and cache increased the memory access latency, we validate the latency result based on memory performance model to quantify the overall performance with the additional five clock cycle encryption latency. The validation result for *libquantum* and *gcc* application showed 0.64% and 0.29% execution time overhead due to increased memory, respectively. Besides, the resource overhead decreased as pipeline design was implemented. The impact of additional latency cipher on the execution time has low significant on the overall performance and produced more impact if the Last Level Cache (LLC) miss ratio is high as *libquantum* application has higher LLC miss ratio compared to *gcc* application.

ABSTRAK

Tujuan pengkomputeran yang dipercayai adalah untuk memberi jaminan kepada pengguna mengenai tingkah laku perisian pada peranti. Unit Perlindungan Memori (MPU) menjamin perlindungan akses antara ingatan utama dan cache yang memainkan peranan penting dalam Sistem-atas-Chip "Root-of-Trust". Terdapat beberapa algoritma sifer seperti SIMON, PRESENT, PRINCE, SPECK dan KATAN yang sesuai untuk MPU. Sifer ini telah dianalisis berdasarkan ukuran sukatan yang berbeza seperti luas, kuasa, daya hasil, kependaman dan pelbagai lagi. Kependaman adalah salah satu parameter penting yang perlu dipertimbangkan semasa memilih sifer MPU. Tambahan pula, berdasarkan kajian sebelum ini, algoritma PRINCE mempunyai kependaman terendah dalam memenuhi syarat reka bentuk utama MPU. Objektif penyelidikan ini adalah untuk mensintesis dan melaksanakan seni bina sifer PRINCE bagi MPU untuk menjamin pertukaran data sensitif pada kependaman yang rendah. Untuk perlindungan ingatan yang disasarkan untuk Tatasusunan Boleh Diaturcara Medan (FPGA), sifer PRINCE menggunakan tiga talian paip dirancang pada tingkat pemindahan daftar (RTL) menggunakan Verilog dan dikaji dengan simulasi dinamik menggunakan Xillinx ISE. Oleh kerana penambahan struktur MPU antara ingatan utama dan cache akan meningkatkan kependaman akses ke ingatan, kami mengkaji hasil kependaman berdasarkan model prestasi ingatan sebelumnya untuk mengukur prestasi keseluruhan selepas penambahan kependaman penyulitan sebanyak lima kitaran. Hasil pengesahan untuk aplikasi *libquantum* dan *gcc* iaitu 0.64% dan 0.29% telah memberi penambahan masa pelaksanaan masing-masing disebabkan penambahan kependaman ingatan. Kesan penambahan kependaman sifer mempunyai signifikan yang rendah terhadap prestasi keseluruhan dan menghasilkan lebih banyak kesan jika nisbah kehilangan tahap cache terakhir tinggi. Hal ini kerana aplikasi *libquantum* mempunyai kadar kehilangan tahap cache terakhir yang lebih tinggi berbanding dengan aplikasi *gcc*.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AES	-	Advanced Encryption Standard
ASIC	-	Application Specific Integrated Circuit
CAN	-	Controller Area Network
CPU	-	Central Processing Unit
FPGA	-	Field Programmable Gate Array
IC	-	Integrated Circuit
IoT	-	Internet of Things
LLC	-	Last Level Cache
LRD	-	Low Resource Device
MC	-	Memory Controller
MPU	-	Memory Protection Unit
RFID	-	Radio Frequency Identification
RISC	-	Reduced Instruction Set Computer
RoT	-	Root-of-Trust
RTL	-	Register Transfer Level
SoC	-	System-on-Chip
WSN	-	Wireless Sensor Network

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Research Background

System-on-Chip (Soc) is an integrated circuit that integrates all components of the computer or other electronic system into a single chip. SoC may contains one core or multi-processor cores which implements multiprocessing in a single physical package. The security function is the heart of hardware Root-of-Trust (RoT) which offers good protection for overall SoC system performance. In other words, hardware RoT is the security foundation of SoC operations that contains the keys used for cryptographic functions and act as security architecture [1], [2], [3]. In the most modern processor, there are Memory Protection Unit (MPU) that has been built in order to protect and avoid illegal memory access. The function of MPU is to secure accesses between main memory, and caches, thus plays an important role in hardware-based memory protection as shown in Figure 1.1.

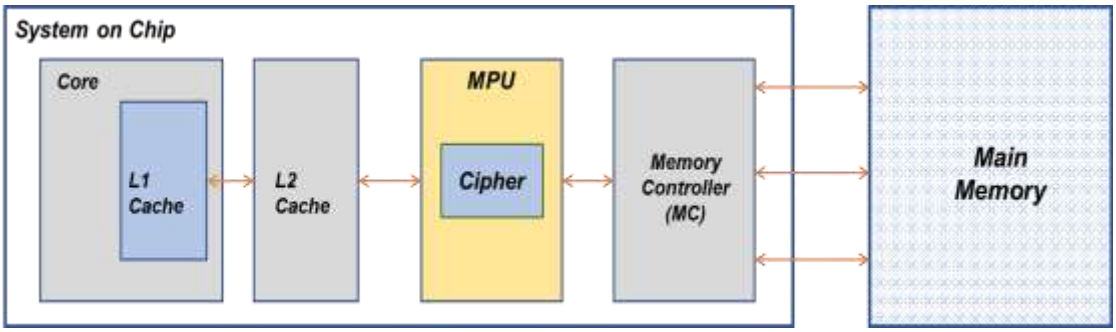

Figure 1.1 Architecture Overview [4]

Cipher is an algorithm located inside the MPU for performing encryption or decryption in cryptography. When entering or leaving the cache, data will be decrypted and encrypted transparently. The system requires cipher to prevent physical type of attacks to memory to prevent attackers from physical access to memory. Thus, memory encryption is one of the standard techniques. Over the last few years, the area of lightweight cryptography with particularly low latency cipher has drawn considerable attention especially applications with real-time requirements [5], [6]. The suitable low-latency cipher is needed to be implemented for MPU to prevent the degradation of overall SoC system performance. Based on the previous research, there are several ciphers such as PRESENT, KATAN, RECTANGLE, SIMON, PRINCE and SPECK that have been evaluated their performance metrics in terms of area, latency, throughput, power and other metrics [7]. PRINCE cipher is one of the lightweight block ciphers is optimized in latency when implemented in hardware while SIMON cipher is optimized with respect to area.

1.2 Problem Statement

Latency is one of the critical parameters when selecting the cipher. The MPU which is located between LLC and the memory controller that controlled accesses from main memory has their own amount of latency consumption. The amount of latency depends on how SoC architecture has been designed. Thus, different processor consumes different value of latency. Next, by adding other components that consume more latency will impact the overall performance of the system. Before designing the MPU, we need to perform analysis on the low latency cipher architecture design and evaluate the performance on memory transfer. Therefore, the choice of low latency cipher is important as the memory access latency impacts the overall SoC system performances while maintaining the security architecture Root of Trust. In precise, as SoC is no longer limited as hard Application Specific Integrated Circuit (ASIC), but also as soft cores, the impact of adding MPU on Field Programmable Gate Array (FPGA) based SoC need to be quantified.

1.3 Research Objective

The aim for this project is to investigate the effect of low latency cipher algorithm design for memory protection of memory access transfer which targeted for SoC on FPGA. In other words, this project also wants to quantify the performance application with additional encryption latency as CPU performance is highly connected to memory performance. The objectives are listed as below:

(a) To synthesize the design of PRINCE cipher architecture for the MPU to secure sensitive data exchange with an emphasis on low latency.
(b) To implement the PRINCE block cipher in Field Programmable Gate Array (FPGA) on Xillinx ISE using Verilog at RTL level.
(c) To validate and analyze the latency performance of the PRINCE cipher design based on memory performance model.

1.4 Scope of Project

The scope of this study includes the investigation on the low latency PRINCE cipher algorithm that is suitable for MPU SoC application. In order to achieve the objectives of the project, few scopes are included in this project. The scopes are listed as below:

(a) Synthesize three pipeline stages of PRINCE cipher Verilog code from Open Source GitHub using FPGA Vivado 2019.2 version.
(b) Verify the PRINCE cipher design which emphasis on latency parameter.
(c) Validate the PRINCE cipher design using memory performance model from previous research work for *libquantum* and *gcc* application.

REFERENCES

[1] GlobalPlatform and Inc, “GlobalPlatform Security Task Force Root of Trust Definitions and Requirements,” 2017.

[2] S. I. R. Andrew Elias, “Understanding Hardware Roots of Trust.” [Online]. Available: https://www.synopsys.com/designware-ip/technical-bulletin/understanding-hardware-roots-of-trust-2017q4.html. [Accessed: 26-Jan-2021].

[3] Rambus Press, “Hardware Root of Trust - Everything you need to know | Rambus,” 10-Sep-2019. [Online]. Available: https://www.rambus.com/blogs/hardware-root-of-trust/. [Accessed: 26-Jan-2021].

[4] M. M. Wong, J. Haj-Yahya, and A. Chattopadhyay, “SMARTS: Secure memory assurance of RISC-V trusted SoC,” *ACM Int. Conf. Proceeding Ser.*, no. December, 2018.

[5] S. S. Dhanda, B. Singh, and P. Jindal, *Lightweight Cryptography: A Solution to Secure IoT*, vol. 112, no. 3. Springer US, 2020.

[6] Cryptrec, “CRYPTREC Cryptographic Technology Guideline (Lightweight Cryptography),” *CRYPTREC Light. Cryptogr. Work. Gr.*, no. March, pp. 1–127, 2017.

[7] P. Maene, “Lightweight Roots of Trust for Modern Systems-on-Chip,” no. October, 2019.

[8] R. Pan, G. Peach, Y. Ren, and G. Parmer, “Predictable virtualization on memory protection unit-based microcontrollers,” *Proc. IEEE Real-Time Embed. Technol. Appl. Symp. RTAS*, pp. 62–74, 2018.

[9] I. B. P. Instructions, “Arm Cortex-M7 Processor Datasheet.”

[10] T. Unterluggauer, M. Werner, and S. Mangard, “MEAS: memory encryption and authentication secure against side-channel attacks,” *J. Cryptogr. Eng.*, vol. 9, no. 2, pp. 137–158, 2019.

[11] “Memory Hierarchy - Cache Memory.” [Online]. Available: https://sites.google.com/site/cachememory2011/memory-hierarchy. [Accessed: 10-Feb-2021].

[12] M. Girija, P. Manickam, and M. Ramaswami, “Comprehensive analysis on lightweight cryptographic algorithms for low resource devices,” *Test Eng. Manag.*, vol. 81, no. 11–12, pp. 3747–3760, 2019.

[13] S. P. Jadhav, “Towards Light Weight CryptographySchemes for Resource ConstraintDevices in IoT,” *J. Mob. Multimed.*, vol. 15, no. 1, pp. 91–110, 2020.

[14] B. T. Hammad, N. Jamil, M. E. Rusli, M. R. Z’Aba, and I. T. Ahmed, “Implementation of lightweight cryptographic primitives,” *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 20, pp. 5571–5586, 2017.

[15] W. J. Buchanan, S. Li, and R. Asif, “Lightweight cryptography methods,” *J. Cyber Secur. Technol.*, vol. 1, no. 3–4, pp. 187–201, 2017.

[16] A. H. A. Al-Ahdal and N. K. Deshmukh, “A Systematic Technical Survey Of Lightweight Cryptography On Iot Environment.”

[17] N. Saqib and U. Iqbal, “Security in wireless sensor networks using ECC,” *2016 IEEE Int. Conf. Adv. Comput. Appl. ICACA 2016*, no. November, pp. 270–274, 2017.

[18] J. Eterovic, M. Cipriano, E. Garcia, and L. Torres, “Lightweight Cryptography in IIoT the Internet of Things in the Industrial Field,” in *Communications in Computer and Information Science*, 2020, vol. 1184 CCIS, pp. 335–353.

[19] P. Maene, J. Gotzfried, T. Muller, R. De Clercq, F. Freiling, and I. Verbauwhede, “Atlas: Application Confidentiality in Compromised Embedded Systems,” *IEEE Trans. Dependable Secur. Comput.*, vol. 16, no. 3, pp. 415–423, 2019.

[20] S. Ghosh, R. Misoczki, L. Zhao, and M. R. Sastry, “Lightweight block cipher circuits for automotive and iot sensor devices,” *ACM Int. Conf. Proceeding Ser.*, vol. Part F1285, 2017.

[21] S. Abed, R. Jaffal, B. J. Mohd, and M. Alshayeji, “FPGA modeling and optimization of a SIMON lightweight block cipher,” *Sensors (Switzerland)*, vol. 19, no. 4, 2019.

[22] P. Maene and I. Verbauwhede, “Single-cycle implementations of block ciphers,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9542, pp. 131–147, 2016.

[23] N. X. P. Semiconductors, “AN12527 LPC55Sxx PRINCE Real-time Data

Encryption," 2020.

[24] Engineer and Ambitiously, "FPGA Fundamentals," *NATIONAL INSTRUMENTS CORP*, 2020. [Online]. Available: https://www.ni.com/en-us/innovations/white-papers/08/fpga-fundamentals.html.

[25] D. M. Khalil-Hani, *Volume 3: Digital System Design with FPGA RTL Optimization and C-Synthesis with Vivado HLS*, 3rd ed. UTM Book Series in Electronics & Computer Engineering, 2020.

[26] S. I. Workshop and D. Hutchison, *LNCS 8162 - Lightweight Cryptography for Security and Privacy*, no. May. 2013.

[27] Y. Zou and L. Li, "FPGA optimal implementation of PRINCE against power analysis attacks," *Proc. Sci.*, vol. 2017-Decem, no. 14, pp. 1–7, 2017.

[28] Y. A. Abbas, R. Jidin, N. Jamil, M. R. Z'aba, M. E. Rusli, and B. Tariq, "Implementation of PRINCE algorithm in FPGA," *Conf. Proc. - 6th Int. Conf. Inf. Technol. Multimed. UNITEN Cultiv. Creat. Enabling Technol. Through Internet Things, ICIMU 2014*, pp. 1–4, 2015.

[29] Y. A. Abbas, R. Jidin, N. Jamil, and M. R. Z'aba, "Lightweight PRINCE algorithm IP core for securing GSM messaging using FPGA," *Res. J. Inf. Technol.*, vol. 8, no. 1–2, pp. 17–28, 2016.

[30] J. Jean, I. Nikolić, T. Peyrin, L. Wang, and S. Wu, "Security analysis of PRINCE," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8424 LNCS, pp. 92–111, 2014.

[31] Y. A. Abbas, R. Jidin, N. Jamil, M. R. Z'aba, M. E. Rusli, and B. Tariq, "Implementation of PRINCE algorithm in FPGA," *Conf. Proc. - 6th Int. Conf. Inf. Technol. Multimed. UNITEN Cultiv. Creat. Enabling Technol. Through Internet Things, ICIMU 2014*, no. October, pp. 1–4, 2015.

[32] Sebastien-riou, "Prince-c-ref," *GitHub, Inc*, 2016. [Online]. Available: https://github.com/sebastien-riou/prince-c-ref/blob/master/log.txt.

[33] J. Strömbergson, "The Prince lightweight block cipher in Verilog," *GitHub, Inc.*, 2020. [Online]. Available: https://github.com/secworks/prince.

[34] J. Strömbergson, "Some Notes on the Lightweight Block Cipher PRINCE," 2020. [Online]. Available: https://www.assured.se/2020/04/24/some-notes-on-the-lightweight-block-cipher-prince/.

[35] A. Navarro-Torres, J. Alastruey-Benedé, P. Ibáñez-Marín, and V. Viñals-Yúfera, "Memory hierarchy characterization of SPEC CPU2006 and SPEC

CPU2017 on the Intel Xeon Skylake-SP," *PLoS One*, vol. 14, no. 8, 2019.

[36] "SPEC - CPU Benchmark Suites." [Online]. Available: https://www.spec.org/cpu/. [Accessed: 28-Jan-2021].

[37] T. K. Prakash and L. Peng, "Performance characterization of spec cpu2006 benchmarks on intel core 2 duo processor," *ISAST Trans. Comput. Softw. Eng*, vol. 2, no. 1, pp. 36–41, 2008.

[38] M. Oh, J. Choi, S. J. Cho, J. Kim, C. Youn, and W. Chung, "Analyzing and modeling the impact of memory latency and bandwidth on application performance," *Proc. ACM Symp. Appl. Comput.*, pp. 1095–1101, 2018.