ELLIPTIC-CURVE CRYPTOGRAPHIC ARCHITECTURES FOR
SYSTEM-ON-CHIP BASED ON FIELD PROGRAMMABLE GATE ARRAYS

ARIF IRWANSYAH

UNIVERSITI TEKNOLOGI MALAYSIA
ABSTRACT

Elliptic curve cryptography (ECC) is an alternative mechanism for implementing public-key cryptographic system. The main reason for the attractiveness of ECC in data security systems is the fact that significantly smaller parameters are needed as compared to other competitive systems, but with equivalent levels of security. This thesis presents the design exploration of elliptic-curve cryptographic architectures for Field Programmable Gate Arrays (FPGA)-based System-on-Chip (SoC). The architectures explored include tightly-coupled custom logic and loosely-coupled coprocessor. The ECC hardware is designed and parameterized for key sizes of 163, 193, and 233 bits. The designs are described in Verilog and VHDL. A demonstration application prototype is developed in which an Elliptic Curve Digital Signature Algorithm (ECDSA) system is combined with a hybrid encryption cryptosystem in one SoC implementation. This application prototype is used in the verification of the designs. Experimental results show that, while utilizing less logic, tightly-coupled architecture improves the execution time of point multiplication operation by about 50% as compared to the loosely-coupled coprocessor. For point addition operation execution time, the tightly-coupled architecture offers 56% improvement as compared to the loosely-coupled coprocessor. The benchmarking of the design with other existing ECC tightly coupled hardwares shows that the design is about fourteen times faster in terms of clock cycles.
Kriptografi lengkung bujur (ECC) merupakan alternatif untuk melaksanakan mekanisma sistem kriptografi kunci-awam. Tarikan utama ECC untuk sistem keselamatan data ialah parameter yang diperlukan lebih kecil jika dibandingkan dengan sistem yang lain, untuk tahap keselamatan yang setara. Tesis ini mempersembah penjelajahan reka bentuk ECC untuk seni bina Sistem atas-Cip (SoC) berasaskan tata susunan get boleh diatur cara medan (FPGA). Seni bina yang dikaji termasuk gandingan ketat logik langganan dan gandingan longgar kompemproses. Perkakasan ECC ini direkabentuk dan diparameterkan untuk kunci bersaiz 163, 193 dan 233 bit. Reka bentuk ini diperihalkan dalam bahasa Verilog dan VHDL. Satu aplikasi prototaip telah dibina di mana algoritma tandatangan digital lengkung bujur (ECDSA) digabungkan dengan kriptosistem enkripsi hibrid dalam satu SoC. Prototaip aplikasi ini telah digunakan dalam penentusahan reka bentuk. Hasil kajian menunjukkan bahawa penggunaan get logik yang sedikit, seni bina gandingan ketat memperbaiki waktu pelaksanaan operasi pendaraban titik sebanyak 50% jika dibandingkan dengan gandingan longgar. Bagi operasi pertambahan titik pula, waktu perlaksanaan untuk seni bina gandingan ketat diperbaiki sebanyak 56%. Penanda aras reka bentuk ini jika dibandingkan dengan perkakasan gandingan ketat ECC yang lain menunjukkan bahawa reka bentuk ini 14 kali ganda lebih pantas dari segi kitar jam.
TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECLARATION</td>
<td></td>
<td>ii</td>
</tr>
<tr>
<td>DEDICATION</td>
<td></td>
<td>iii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td></td>
<td>iv</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td></td>
<td>v</td>
</tr>
<tr>
<td>ABSTRAK</td>
<td></td>
<td>vi</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td></td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td></td>
<td>xi</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td></td>
<td>xiii</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td></td>
<td>xvii</td>
</tr>
<tr>
<td>LIST OF APPENDICES</td>
<td></td>
<td>xix</td>
</tr>
</tbody>
</table>

1 INTRODUCTION
1.1 Background and Motivation 1
1.2 Research Objectives 3
1.3 Scope of Work 4
1.4 Overview of Research Methodology 4
1.5 Research Contribution 6
1.6 Thesis Organization 6

2 LITERATURE REVIEW AND BACKGROUND
2.1 Previous Work 8
2.2 Hardware Accelerator in Embedded System on Chip Design 12
2.3 Tightly Coupled Hardware in Nios II Platform 14
### 3 THEORY AND ALGORITHM

#### ELLIPTIC CURVE CRYPTOGRAPHY

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Cryptography in Data Security</td>
<td>19</td>
</tr>
<tr>
<td>3.2</td>
<td>Elliptic Curve Cryptography – An Introduction</td>
<td>20</td>
</tr>
<tr>
<td>3.3</td>
<td>Theory of Finite Fields</td>
<td>23</td>
</tr>
<tr>
<td>3.4</td>
<td>Finite Field Arithmetic</td>
<td>25</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Field Addition</td>
<td>26</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Field Multiplication</td>
<td>26</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Field Squaring</td>
<td>29</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Field Inversion</td>
<td>30</td>
</tr>
<tr>
<td>3.5</td>
<td>Elliptic Curve Arithmetic over $F_2^m$</td>
<td>33</td>
</tr>
<tr>
<td>3.6</td>
<td>Montgomery Point Multiplication in Projective Coordinate</td>
<td>36</td>
</tr>
<tr>
<td>3.7</td>
<td>Elliptic Curve Scheme</td>
<td>38</td>
</tr>
<tr>
<td>3.7.1</td>
<td>ECDH Key Agreement Algorithm</td>
<td>38</td>
</tr>
<tr>
<td>3.7.2</td>
<td>Elliptic Curve Digital Signature Algorithm (ECDSA)</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>EC-AES Hybrid Encryption Algorithm</td>
<td>41</td>
</tr>
<tr>
<td>3.8</td>
<td>Summary</td>
<td>42</td>
</tr>
</tbody>
</table>

### 4 DESIGN OF ECC HARDWARE ACCELERATOR

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>ECC Domain Parameter</td>
<td>43</td>
</tr>
<tr>
<td>4.2</td>
<td>ECC System Design Exploration</td>
<td>44</td>
</tr>
<tr>
<td>4.3</td>
<td>Design of ECC Processor</td>
<td>48</td>
</tr>
<tr>
<td>4.3.1</td>
<td>ECC Field Arithmetic Level Coprocessor (LC-F)</td>
<td>48</td>
</tr>
<tr>
<td>4.3.2</td>
<td>ECC Point Arithmetic Level Coprocessor (LC-P)</td>
<td>53</td>
</tr>
<tr>
<td>4.4</td>
<td>Design of ECC TC-hardware</td>
<td>55</td>
</tr>
<tr>
<td>4.4.1</td>
<td>ECC Field Arithmetic Level TC-hardware (TC-F)</td>
<td>55</td>
</tr>
<tr>
<td>4.4.2</td>
<td>ECC Point Arithmetic Level TC-hardware (TC-P)</td>
<td>60</td>
</tr>
<tr>
<td>4.5</td>
<td>Summary</td>
<td>62</td>
</tr>
</tbody>
</table>
5  ECC BASED HYBRID ENCRYPTION AND DIGITAL SIGNATURE CRYPTOSYSTEMS

5.1 Elliptic Curve Cryptosystem Scheme 63
5.2 Embedded Software Development of ECHEDSC 66
5.3 Hardware Development of ECC-based Security Scheme 68
  5.3.1 Elliptic Curve Cryptography TC-Hardware Custom Instruction 69
  5.3.2 SHA-1 Hash Function Coprocessor 71
  5.3.3 Modular Arithmetic Processor (MAP) 75
  5.3.4 AES 78
  5.3.5 Pseudo Random Number Generator 81
5.4 Summary 82

6  DESIGN VERIFICATION, TEST AND PERFORMANCE ANALYSIS

6.1 Tests Consideration 83
6.2 Test Verification of ECC Hardware Accelerator 84
  6.2.1 Test Verification of ECC Field Arithmetic Level Hardware Accelerator 84
  6.2.2 Test Verification of ECC Point Arithmetic Level Hardware Accelerator 86
6.3 Resource Utilization 88
6.4 ECC Hardware Performance 90
  6.4.1 Performance in Field Arithmetic Level 91
  6.4.2 Performance in Point Arithmetic Level 92
  6.4.3 Performance Comparison of ECC Arithmetic Level 93
6.5 Benchmarking 94
6.6 Tests in Elliptic Curve Cryptosystem 95
  6.6.1 SHA-1 Verification Test 95
  6.6.2 AES-256 Verification Test 96
6.6.3 MAP-233 Verification Test
6.7 ECDSA and ECAES-Hybrid Encryption Test
6.7.1 ECDSA Verification Test
6.7.2 EC-AES Hybrid Encryption Verification Test
6.7.3 Timing Performance
6.8 Tests in Demonstration Application Prototype
6.8.1 Demonstration Application System View
6.8.2 e-Cheque GUI Application Test
6.10 Summary

7 CONCLUSIONS
7.1 Concluding Remarks
7.2 Future Work

REFERENCES

Appendices A – E
CHAPTER 1

INTRODUCTION

This thesis presents the design exploration of alternative elliptic-curve cryptographic architectures for FPGA-based System-on-chip (SoC). This chapter covers the background, problem statement, research objectives, scope of work, overview of research methodology, the significance and contribution of the research, and finally the thesis organization.

1.1 Background and Motivation

The need for secure communications has led to a high demand for cryptographic service such as encrypted communications between unfamiliar hosts over insecure channels (such as the Internet). With the rapid deployment of applications like online banking, stock trading and corporate remote access, recent years have seen an explosive growth in the amount of sensitive data exchanged over the Internet. These days, an increasing number of Internet hosts are battery-powered, wireless, handheld devices with restricted memory, CPU, latency and bandwidth. For these constrained environments, besides security, it is very important to consider about the speed performance and logic cost.

Elliptic Curve Cryptography (ECC) is a public key mechanism, independently proposed by Miller (1986) and Kobitz (1987). ECC offers secure and efficient solutions for the new communication technologies. It requires fewer bits than the well known RSA (Rivest, Shamir, Adleman) for similar level of security.
For example, a 163-bit ECC key provides a level of security equivalent to a 1024-bit RSA key (Certicom, 2000). As a result of using smaller key sizes, it is possible to achieve higher speeds, and at the same time use less power, bandwidth and storage (Juliato et al., 2007). Nowadays, this technology is well accepted in the industry and the academic communities and has been the subject of several standards such as ANSI X9.62 and IEEE P1363.

There are likely to be two groups of devices which will participate in public key cryptography application, such as secure mobile and wireless environments: servers and end devices (Weimerskirch et al., 2003). The server platform are less likely to problems due to the availability of high-speed processors and extensive memory space. However, end devices are often lever restricted regarding computing power, memory for software code, RAM size and energy supply. The implementation of public key cryptography on end devices or embedded systems, for example smart card and mobile phone, requires fast computation, small memory and high energy efficiency.

Typically, application running on an embedded system platform can be executed either as a firmware running on embedded processor or a specialized hardware unit that purposes as a coprocessor. The implementation of public key cryptography in embedded systems performs slowly in software. However, this approach is economical in logic cost and is flexible to change in algorithm. Software implementations of cryptographic algorithms often spend the majority of their execution time in a few performance-critical code sections (Hankerson et al., 2004). Typical examples of such code sections are the inner loops of long integer arithmetic operations needed in public-key cryptography. The hardware-based accelerators are often the solution reaching an acceptable performance-cost ratio (Meurice et al., 2007).

Two key factors influence the performance of embedded system with hardware accelerator (Batina et al., 2006) are the communication interface between processor and co-processor; and the boundary between hardware and software in terms of hardware/software partitioning.
A conventional technique to enhance the performance cryptographic operations in an embedded system is to off-load the computational heavy sections of an algorithm into a dedicated hardware accelerator. An alternative architecture is to extend the functionality of an embedded processor by applying a tightly-coupled custom logic, and utilizing it to define new custom instructions set (Tillich and Großschädl, 2006). This thesis explores the potential use of tightly-coupled custom logic in elliptic curve cryptographic operations.

1.2 Research Objectives

The objectives of this thesis are as follows:

1. To propose alternative architectures for ECC hardware accelerator that enhances the embedded processor by using tightly-coupled custom logic and extending the instruction set.

2. To improve ECC hardware accelerator performance by eliminating communication bottleneck between processor and hardware accelerator.
1.3 Scope of Work

1. The architectures explored include a tightly-coupled custom logic (TC-H) and a loosely-coupled coprocessor (LC-H). The architectures are designed in Verilog HDL and VHDL code in 163-bit, 193-bit and 233-bit field size, with the digit size (or degree of parallelism) of 32-bit.

2. Polynomial basis representations have been chosen as the basis of binary field arithmetic of ECC. In this work, we utilize the recommended elliptic curve domain parameters from Certicom (2000).

3. The implementation of the ECC hardware accelerator must fit into Altera Stratix EP1S40F780C5 FPGA development board and the running frequency is 50 MHz. The embedded processor is Altera Nios II CPU soft core.

4. The evaluation of the explored architecture design is limited to speed performance and resource utilization.

1.4 Overview of Research Methodology

As illustrated in Figure 1.1, this research work is divided into two phases. The first phase is the literature review and design specification. Then the work continues with design of ECC hardware accelerator, both in TC-H and LC-H architectures. This involves the hardware/software partitioning in ECC finite field arithmetic level and ECC point arithmetic level. Hardware test verification and performance comparison between tightly-coupled hardware and loosely-coupled coprocessor architectures is also evaluated here. Constraints of speed and hardware resource are taken into considerations.
The second phase is to develop an elliptic curve cryptosystem to validate the design correctness and functionality of the proposed ECC TC-hardware. This ECC embedded systems utilize Nios II soft-core embedded processor core (Altera, 2007), 233-bit Modular Arithmetic Processor (MAP), SHA-1 (Secure Hash Algorithm) cryptographic hash processor core, 256-bit AES encryption processor core and the proposed ECC TC-hardware. Elliptic Curve – AES (EC-AES) Hybrid Encryption and Elliptic Curve Digital Signature Algorithm (ECDSA) (Certicom, 2000a) demonstration application prototype was implemented in this work.

For the development of our cryptosystem, we choose the Altera FPGA hardware system based on Nios II 32-bit RISC embedded processor as the prototyping platform. This development system contains a powerful and flexible IDE, the Quartus II SOPC Builder that facilitates the tasks of extending Nios II instruction set to include custom instructions, and also the creation of a tightly-coupled custom logic.

![Figure 1.1 System Design Flow](image-url)
1.5 **Research Contribution**

The contributions of this thesis are as follows:

1. Introduce design exploration and performance comparison between tightly-coupled custom logic and loosely coupled coprocessor for ECC field arithmetic and point arithmetic.

2. Implement elliptic curve cryptography in FPGA-based embedded system modules which are applied in EC-AES Hybrid Encryption and ECDSA schemes.

3. Develop an application demonstration prototype that implements an e-cheque application as secure document through insecure network. This is used to validate the ECC architectures.

1.6 **Thesis Organization**

The thesis is organized into seven chapters. The first chapter introduces the research motivation, research objectives, research scope, overview of research methodology, research contribution, and the thesis organization.

Chapter 2 reviews the background of the research. Related works are presented. Summary of the literature review is given to clarify the research rationale.

Chapter 3 presents the brief introduction of the mathematical concepts of finite fields and elliptic curves. Various design styles of hardware accelerator to implement elliptic curve arithmetic are described. The ECDH, EC-AES Hybrid Encryption, and ECDSA scheme are discussed in this chapter.
REFERENCES


