SERIAL VS. PARALLEL ELLIPTIC CURVE CRYPTO PROCESSOR DESIGNS

Adnan Abdul-Aziz Gutub\textsuperscript{1}, Abdul-Aziz Tabakh\textsuperscript{2}, Ayed Al-Qahtani\textsuperscript{2} and Alaaeldin Amin\textsuperscript{2}

\textsuperscript{1}Umm Al-Qura University (UQU) - Makkah 21955, Saudi Arabia
\textsuperscript{2}King Fahd University of Petroleum and Minerals (KFUPM) - Dhahran 31261, Saudi Arabia

ABSTRACT

Point conversion in Elliptic Curve Cryptography (ECC) arithmetic suffer the costly complex division operation. Several methods proposed to overcome this division by multiple multiplication steps using chosen projective coordinates systems. This work adopts Aladdin Modulo Multiplier algorithm (an improvement to Montgomery modulo multiplier) in its design to compare both serial and parallel ECC multiplier hardware for its computations. The investigation focuses on tradeoffs between time and area showing the benefit of different ECC designs, i.e. using serial and parallel Aladdin modulo multipliers.

KEYWORDS

Elliptic curve cryptography, serial & parallel implementation, modulo multiplier architecture, Aladdin multiplier, projective coordinates, data security.

1. INTRODUCTION

Cryptography systems have been growing in importance recently as a method for improving data security. Most of the public key cryptosystems (such as RSA, ElGamal, and Elliptic curve cryptosystems) (Gutub & Khan 2012, Gutub 2010, Gutub 2010-IJS, Tawalbeh \textit{et al.} 2010) use modular multiplication in encryption and decryption operations with a large modulus (ranging 100–1000 bits). Elliptic curve cryptography (ECC) had proven its high security with lowest number of modulus bits leading faster and simpler computations compared to others (Black \textit{et al.} 1999, Gutub \textit{et al.} 2011, Tawalbeh \textit{et al.} 2009). The most complex arithmetic operation within ECC is its point conversion based on division. Several methods have been proposed to convert this division operation to multiple multiplication steps using a chosen projective coordinates system (Gutub \textit{et al.} 2007, Gutub & Ibrahim 2003). In this work, we adopted an improved algorithm for modulo multiplication named Aladdin modulo multiplier to implement serial and parallel ECC hardware.

The ECC arithmetic is described to contain several elliptic curve point doubling and elliptic curve point adding operations (Black \textit{et al.} 1999, Gutub & Tahhan 2008). The traditional way of calculating the ECC doubling and adding operations is performed in sequence and can be improved using specific parallelism (Gutub 2003, Gutub & Ibrahim 2003, Gutub \textit{et al.} 2011, Gutub & Khan 2012), which is further enhanced in this work adopting Aladdin multipliers. The proposed parallel algorithms in the summarized papers (Gutub \textit{et al.} 2007, Gutub & Ibrahim 2003) are evaluated and compared to this proposed work using VHDL. The investigation focused on the performance tradeoffs relating time (speed) and area (space) of these methods. The results exploited parallelism in order to boosting up the ECC operation speed.

The paper is organized as follows. Section 2 gives a brief introduction to ECC. Then, Section 3 describes Aladdin multiplier as the improved basic block adopted to perform modulo multiplication in this ECC design. Parallel multiplications within ECC research is introduced in Section 4. Section 5 gives an idea of the complete ECC architecture designed and tested. Then, Section 6 details the experimental results and findings. Finally, Section 7 concludes this research work.
Table 3. Cost Performance Analysis

<table>
<thead>
<tr>
<th>Focus</th>
<th>Figure of Merit</th>
<th>Serial Design</th>
<th>Parallel Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>Both Same</td>
<td>AT</td>
<td>2,039,797.5</td>
<td>1,724,380</td>
</tr>
<tr>
<td>Time</td>
<td>AT2</td>
<td>39,776,051.25</td>
<td>8,621,900</td>
</tr>
<tr>
<td>Area</td>
<td>AT2</td>
<td>2.13373×10^11</td>
<td>5.94697×10^11</td>
</tr>
</tbody>
</table>

7. CONCLUSION

ECC parallel and serial multiplication hardware are modeled using the improved Aladdin modulo multiplier algorithm. The study compared the designs through VHDL implementation that investigated the tradeoffs regarding time and area. The analysis showed how exploiting parallelism can boost up the speed on the cost of 30% extra hardware area. The work also proofed that the multipliers worth investigation, since they occupy the largest portion of the design, i.e. 35% of the serial design and 42% of the parallel design are consumed by multipliers. Parallelization may require more hardware, more registers and multipliers, to give the high performance needed in today applications.

ACKNOWLEDGEMENT

Thanks to King Fahd University of Petroleum and Minerals (KFUPM) for partially hosting this research. Thanks also to Umm Al-Qura University (UQU) for collaborative moral support toward these achievements.

REFERENCES