Review of Elliptic Curve Processor Architectures

Ibrahim Hazmi
PhD Candidate - University of Victoria
24 August 2015

Coauthors: Fan Zhou, Fayez Gebali, and Turki al-Somani
Outline

Introduction
- Motivation
- ECC Pyramid
- ECC Implementation

ECP Design Parameters
- ECC Processor
- Over view of ECP Design Parameters

ECP Architectures in the Literature
- Choice of Fields and Bases
- Choice of Algorithms and Coordinate Systems
- Parallelism, and Countermeasure against attacks
- Control option, and Hardware Platforms
- Key size, Operating Clock speed, and Area*Delay

Conclusion
Why ECC?

**ECC**

1. **Stronger Encryption**
   - Shorter key than RSA
   - 256-bit ECC = 3072-bit RSA
   - 10k times harder to crack than RSA 2048
   - Meets NIST recommendations

2. **Efficient Performance**
   - Handles more requests per second with lower CPU utilization
   - Uses less server power
   - ECC performance is expected to improve over time as the industry optimizes for ECC as it did for RSA

3. **Highly Scalable**
   - Large SSL deployments w/out additional hardware
   - Uses fewer resources with Lower costs

4. **Future of Crypto Tech**
   - Viable for many years
   - Built for Internet of things to come
   - Supports billions of new devices
   - Ideal for Open Networks

**ECC: 10,000 times harder to break than RSA keys**

“We believe in constantly furthering web security, which is why Chrome supports Elliptic Curve Digital Signature Algorithm (ECDSA) on all modern operating systems,”

Adam Langley, Software Engineer Google, 2013.

Source: Symantec Internal Research and Testing. Andrew Horbury, Symantec’s Algorithm Agility, March 2013
What is ECC?

- Public-key Cryptosystem based on the algebraic structure of elliptic curves over finite fields
- Based on the difficulty of the Discrete Logarithm Problem (DLP) over a group of points.
- ECC Pyramid
ECC Implementation

**HW-SW Co-Design:**
The Lower 3 Levels of the pyramid can be implemented in Hardware:
1. Finite Field Accelerator
2. Point Arithmetic Accelerator
3. Scalar Mul. Accelerator

**ECC Processor**
All ECC Layers can be implemented purely in SW or in the form of HW-SW Co-Design

---

**Introduction**

**ECC Implementation**

Upper Layer: The kP algorithm
- Representations of Scalar k (Binary, Ternary, NAF, Bit_Grouping)
  - The Binary Method
  - Addition Subtraction
  - Montgomery
  - Comb
  - Window

Middle Layer: Point Addition and Doubling & Coordinates Systems
- Affine Coordinates
- Projective Coordinates
  - Standard
  - Lopez-Dahad
  - Jacobians

Lower Layer: Finite Field Arithmetic
- Binary Field (GF_{2^m})
  - Multiplication Squaring
  - Addition Subtraction
  - Division Inversion
- Prime Field (GF_p)
  - Multiplication Squaring
  - Addition Subtraction
  - Division Inversion
ECC Implementation

Finite Field Arithmetic in Point Arithmetic

Field and Point Arithmetic in Scalar Mult. Algorithm

Example of a Scalar Multiplier Over Binary Field

Introduction

Lopez-Dahab Point Doubling Algorithm
ECC Processor (ECP) is a Cryptographic Hardware Processor that accelerates the intensive computations of ECC Scalar Multiplication.

MC: Main Controller
AUC: Arithmetic Unit Controller
AU: Arithmetic Unit
## ECP Design Parameters

### Design Parameters

- **A.** Choice of Fields and Bases
- **B.** Choice of Algorithms and Coordinate Systems
- **C.** Parallelism, and Countermeasure against attacks
- **D.** Control option, and Hardware Platforms
- **E.** Key size, Operating Clock speed, Area*Delay

The parameters were recorded in:

**A table looks like:**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field/Basis</td>
<td>Field Algorithms</td>
<td>Coordinate System</td>
<td>SM Algorithm</td>
<td>Parallelism</td>
</tr>
</tbody>
</table>
ECP Architectures in the Literature

Choice of Fields and Bases

Fields:
- Binary Field
- Prime Field
- Dual Field

Examples of Bases:
- Polynomial
- Normal GNB
- Optimal ONB
ECP Architectures in the Literature

Choice of Field Algorithms

Multipliers:
- Interleaved Multiplication
- Karatsuba Multiplication
- Sunar-Koc Multiplier
- Montgomery Modular Multiplication
- Cellular Automata Multiplier

Inversion:
- EEA-Based
- Almost Inverse
- Binary Euclid.
- Montgomery
- FLT-Based
- Itoh-Tsuji
# ECP Architectures in the Literature

## Choice of Coordinate System

<table>
<thead>
<tr>
<th>Coordinate System</th>
<th>General Addition</th>
<th>General Addition, Mixed Coordinates</th>
<th>Doubling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affine Coordinate</td>
<td>V+M</td>
<td>-</td>
<td>V+M</td>
</tr>
<tr>
<td>Standard Projective</td>
<td>13M</td>
<td>12M</td>
<td>7M</td>
</tr>
<tr>
<td>Jacobian Projective</td>
<td>14M</td>
<td>10M</td>
<td>5M</td>
</tr>
<tr>
<td>Lopez-Dahab Projective</td>
<td>14M</td>
<td>8M</td>
<td>4M</td>
</tr>
</tbody>
</table>

M = Multiplication  
V = Division

- Affine and Projective Coordinates: 41.2%
- Affine Coordinates: 14.7%
- Projective Coordinates: 23.5%
- Lopez-Dahab Projective Coordinates: 20.6%
## ECP Architectures in the Literature

### Choice of SM Algorithm

<table>
<thead>
<tr>
<th>Unknown Point Multiplication</th>
<th>Fixed Point Multiplication</th>
<th>Multiple Point Multiplication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left-to-Right and Right-to-Left Binary Methods</td>
<td>Fixed-Based Comb method (with 2 Tables)</td>
<td>Simultaneous MPM (Shamir’s trick)</td>
</tr>
<tr>
<td>Montgomery Point Multiplication</td>
<td>Fixed-Based Comb method</td>
<td>Joint Sparse Form (JSF)</td>
</tr>
<tr>
<td>Non-Adjacent Form (NAF)</td>
<td>Fixed-Based NAF Windowing</td>
<td>Interleaving with NAFs</td>
</tr>
<tr>
<td>Window and Sliding Window</td>
<td>Fixed-Based Windowing</td>
<td></td>
</tr>
</tbody>
</table>

### Frequency Table

<table>
<thead>
<tr>
<th>SM Algorithm</th>
<th>Frequency</th>
<th>Frequency %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Montgomery Multiplication</td>
<td>22</td>
<td>52.38%</td>
</tr>
<tr>
<td>Left-to-Right Binary Method</td>
<td>8</td>
<td>19.05%</td>
</tr>
<tr>
<td>NAF</td>
<td>5</td>
<td>11.90%</td>
</tr>
<tr>
<td>JSF</td>
<td>2</td>
<td>4.76%</td>
</tr>
<tr>
<td>Slide Window Algorithm</td>
<td>1</td>
<td>2.38%</td>
</tr>
<tr>
<td>Interlaeving with NAF</td>
<td>1</td>
<td>2.38%</td>
</tr>
<tr>
<td>Double-Add Algorithm</td>
<td>3</td>
<td>7.14%</td>
</tr>
</tbody>
</table>
ECP Architectures in the Literature

Parallelism, and Countermeasure

- Parallel: 65.8%
- Not mentioned: 34.2%

Timing Attack: 43.5%
Power Analysis Attack: 17.4%
Power and Timing Attacks: 34.8%
Side Channel Attack: 34.8%
ECP Architectures in the Literature

Control option and HW Platforms

Controllers:
- Microprogrammed
- Finite State Machine (FSM)

HW Platforms:
- FPGA
- Xilinx
- Altera
- Others
- ASIC

Microprogrammed: 28.9%
FSM Controller: 71.1%
ASIC: 11.6%
FPGA: Xilinx: 23.3%
FPGA: Altera: 65.1%
### ECP Architectures in the Literature

**Key size, Clock speed, Area*Delay**

<table>
<thead>
<tr>
<th>Key Size</th>
<th>Frequency</th>
<th>Frequency %</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>83</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>89</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>113</td>
<td>4</td>
<td>5.80%</td>
</tr>
<tr>
<td>131</td>
<td>2</td>
<td>2.90%</td>
</tr>
<tr>
<td>160</td>
<td>2</td>
<td>2.90%</td>
</tr>
<tr>
<td><strong>163</strong></td>
<td><strong>21</strong></td>
<td><strong>30.43%</strong></td>
</tr>
<tr>
<td>167</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>173</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>191</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>192</td>
<td>4</td>
<td>5.80%</td>
</tr>
<tr>
<td>193</td>
<td>3</td>
<td>4.35%</td>
</tr>
<tr>
<td>224</td>
<td>3</td>
<td>4.35%</td>
</tr>
<tr>
<td><strong>233</strong></td>
<td><strong>7</strong></td>
<td><strong>10.14%</strong></td>
</tr>
<tr>
<td><strong>256</strong></td>
<td><strong>6</strong></td>
<td><strong>8.70%</strong></td>
</tr>
<tr>
<td>283</td>
<td>2</td>
<td>2.90%</td>
</tr>
<tr>
<td>384</td>
<td>2</td>
<td>2.90%</td>
</tr>
<tr>
<td>409</td>
<td>1</td>
<td>1.45%</td>
</tr>
<tr>
<td>521</td>
<td>4</td>
<td>5.80%</td>
</tr>
<tr>
<td>571</td>
<td>2</td>
<td>2.90%</td>
</tr>
</tbody>
</table>

- The operating frequency varies significantly from 300 kHz to 2 GHz.
- The Area-Delay product varies from the minimum of (0.002 Slice*sec) to (112.1 Slice*sec).
Conclusion

- ECC has stronger encryption, more efficient performance, greater scalability over RSA
- ECC can be implemented purely in SW or in the form of HW-SW Co-Design
- Binary field with Polynomial representation is preferred.
- Interleaved multipliers and FLT Inverters are dominating the field arithmetic.
- Montgomery scalar multiplier is the most used algorithm, with the dominance of Lopez-dahab Projective Coordinate system.
- Parallelism and pipelining are more demanded for the desired efficiency.
Countermeasure against side-channel attacks is prioritized.

FSM is dominant preferred control approach

Xilinx FPGAs have the highest demand among the other hardware platforms

163-bit key size has the most appearance in the literature

The operating frequency varies significantly from 300 kHz to 2 GHz.

The Area-Delay product varies from the minimum of (0.002 Slice*sec) to (112.1 Slice*sec)
Thank you