Hardware-Software Codesign of RSA for Optimal Performance vs. Flexibility Trade-off

Malik Umar Sharif, Rabia Shahid, Marcin Rogawski, and Kris Gaj
George Mason University
USA

Supported in part by NIST/U.S. Department of Commerce under Grant No. 60NANB15D058
Primary Designers & Co-Authors

Malik Umar Sharif  Rabia Shahid  Marcin Rogawski

PhD Students in the Cryptographic Engineering Research Group (CERG) at GMU

Former PhD Student Cadence Design Systems San Jose, CA
Cryptography at Crossroads

Traditional Cryptography

- RSA
- Elliptic Curve Cryptosystems (existing standards)

Post-Quantum Cryptography

- Hash-based
- Code-based
- Lattice-based
- Multivariate (emerging standards)

Transition Period

- Attacks using quantum computers
- Trusted resistance

Complete collapse

Trusted resistance

Limited trust
Solutions for the Transition Period

Traditional Scheme (e.g., RSA)  Post-Quantum Scheme (e.g., NTRU)

Maximum flexibility with the choice of parameters and key sizes

Hardware acceleration crucial because of high-computational complexity
Why RSA?

The oldest and most trusted public key scheme

Baseline for evaluation of post-quantum cryptosystems

Simple description:
Encryption and decryption equivalent to modular exponentiation, $Y = X^E \text{ mod } N$

Shamir
Rivest
Adleman
MIT, 1977
Basic Operations of RSA

Modular Exponentiation: \( Y = X^E \mod N \)

Modular Multiplication: \( C = A \cdot B \mod N \)

Typical operand sizes: 512-2048 bits
Our Platform – Zynq-7000 & ZedBoard

Processing System (PS) – ARM based Microprocessor System

Software in C based on RELIC (Efficient Library for Cryptography)
- Free
- Optimized for embedded systems

Programmable Logic (PL) – a 28nm Artix-7-based reconfigurable logic

Hardware in VHDL based on architecture by Orup-Suzuki
- DSP-unit based
- Optimized for maximum clock frequency
Design Options

- **Software in C**
  - Mod Exp
  - Mod Mul
  - Most Flexible
  - Least Efficient

- **Hardware in VHDL**
  - Mod Exp
  - Mod Mul
  - Best Balanced

- **Software in C**
  - Mod Exp
  - Mod Mul
  - Most Efficient
  - Least Flexible
Features of our Solution

Three modular exponentiation schemes:

- Left-to-Right (L2R),
- Right-to-Left (R2L)
- Sliding Window (SLID)

selected at run time

Four operand sizes:

- 512, 1024, 1536, 2048 bits

selected at run time

Maximum flexibility and scalability
Speed-up vs. Software Based on RELIC

Left-to-Right

Right-to-Left

Sliding Window

512 1024 1536 2048
Future Work

• Implementation of selected post-quantum cryptographic algorithms on Zynq using the similar software/hardware co-design approach

• Implementation of the hardware portions using High-Level Synthesis

http://cryptography.gmu.edu

Poster: 3:30-4:15pm