

# Modular Multiplication in the Residue Number System

A DISSERTATION SUBMITTED TO  
THE SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING  
OF THE UNIVERSITY OF ADELAIDE

BY

**Yinan KONG**

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR  
THE DEGREE OF DOCTOR OF PHILOSOPHY

July 2009

# Declaration of Originality

Name: Yinan KONG

Program: Ph.D.

This work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text.

I give consent to this copy of my thesis, when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

The author acknowledges that copyright of published works contained within this thesis (as listed below) resides with the copyright holder/s of those works.

Signature:

Date:



# Acknowledgments

My supervisor, Dr Braden Jace Phillips, is an extremely hard working and dedicated man. So, first and foremost, I would like to say “thank you” to him, for his critical guidance, constant sustainment and role modelling as a supervisor. It is my true luck that I have been able to work with him for these years. This has been a precious experience which deserves my cherishing throughout my whole life.

I am also grateful to Associate Professor Cheng-Chew Lim and Dr Alison Wolff for their guidance through important learning phases of this complex technology. Throughout the course of my study I have received considerable help from my colleagues Daniel Kelly and Zhining Lim, who have been, and will remain, my great friends.

Acknowledgement is given to the Australian Research Council (ARC) as this work has been supported by the ARC Discovery Project scheme.

Thanks to the support from my family and friends. I have been relying on you throughout my candidature. Thanks are due to Mum, Dad, Ranran, Zhaozhao and Jingdong JU, who flew over 5000 miles to take care of me. I would definitely not have been able to get to this point without your encouragement. You are the real pearls lying on the bottom of my mind.

Mother, thank you for giving birth to me as well as cultivating me through those tough years. This thesis has your sweat in it.

My love, BEN YA, you are my greatest inspiration. Thank you for all you have done for me.

Yinan KONG  
November 2008



# Abstract

*Public-key cryptography* is a mechanism for secret communication between parties who have never before exchanged a secret message. This thesis contributes arithmetic algorithms and hardware architectures for the modular multiplication  $Z = A \times B \pmod{M}$ . This operation is the basis of many public-key cryptosystems including RSA and Elliptic Curve Cryptography. The *Residue Number System* (RNS) is used to speed up long word length modular multiplication because this number system performs certain long word length operations, such as multiplication and addition, much more efficiently than positional systems.

A survey of current modular multiplication algorithms shows that most work in a positional number system, e.g. binary. A new classification is developed which classes these algorithms as Classical, Sum of Residues, Montgomery or Barrett. Each class of algorithm is analyzed in detail, new developments are described, and the improved algorithms are implemented and compared using FPGA hardware.

Few modular multiplication algorithms for use in the RNS have been published. Most are concerned with short word lengths and are not applicable to public-key cryptosystems that require long word length operations. This thesis sets out the hypothesis that each of the four classes of modular multiplication algorithms possible in positional number systems can also be used for long word length modular multiplication in the RNS; moreover using the RNS in this way will lead to faster implementations than those which restrict themselves to positional number systems. This hypothesis is addressed by developing new Classical, Sum of Residues and Barrett algorithms for modular multiplication in the RNS. Existing Montgomery RNS algorithms are also discussed.

The new Sum of Residues RNS algorithm results in a hardware im-

plementation that is novel in many aspects: a highly parallel structure using short arithmetic operations within the RNS; fully scalable hardware; and the fastest ever FPGA implementation of the 1024-bit RSA cryptosystem at 0.4 ms per decryption.

# Publications

1. Yinan Kong and Braden Phillips, “Fast Scaling in the Residue Number System”, accepted by IEEE Transactions on VLSI Systems in December 2007.
2. Yinan Kong and Braden Phillips, “Simulations of modular multipliers on FPGAs”, Proceedings of the IASTED Asian Conference on Modelling and Simulation, Beijing, China, Oct. 2007, pp. 11281131.
3. Yinan Kong and Braden Phillips, “Comparison of Montgomery and Barrett modular multipliers on FPGAs”, 40th Asilomar Conference on Signals, Systems and Computers. Pacific Grove, CA, USA: IEEE, Piscataway, NJ, USA, Oct. 2006, pp. 16871691.
4. Yinan Kong and Braden Phillips, “Residue number system scaling schemes”, in Smart Structures, Devices, and Systems II, ser. Proc. SPIE, S. F. Al-Sarawi, Ed., vol. 5649, Feb. 2005, pp. 525536.
5. Yinan Kong and Braden Phillips, “A classical modular multiplier for RNS channel operations”, The University of Adelaide, CHiPTec Tech. Rep. CHIPTEC-05-02, November 2005.
6. Yinan Kong and Braden Phillips, “A Montgomery modular multiplier for RNS channel operations”, The University of Adelaide, CHiPTec Tech. Rep. CHIPTEC-05-02, November 2005.



# Publications in Submission

1. Yinan Kong and Braden Phillips, “Modular Reduction and Scaling in the Residue Number System Using Multiplication by the Inverse”, submitted to IEEE Transactions on VLSI Systems in November 2008.
2. Yinan Kong and Braden Phillips, “Low latency modular multiplication for public-key cryptosystems using a scalable array of parallel processing elements”, submitted to 19th IEEE Computer Arithmetic in October 2008.
3. Braden Phillips and Yinan Kong, “Highly Parallel Modular Multiplication in the Residue Number System using Sum of Residues Reduction”, submitted to Journal of Applicable Algebra in Engineering, Communication and Computing in June 2008.
4. Yinan Kong and Braden Phillips, “Revisiting Sum of Residues Modular Multiplication”, submitted to International Journal of Computer Systems Science and Engineering in May 2008.

# Nomenclature

$\langle X \rangle_M$	The operation $X \bmod M$ .
$D$	The dynamic range of a RNS.
$M$	The modulus of a modular multiplication, typically $n$ bits.
$m_i$	The $i$ th RNS channel modulus.
$N$	The number of RNS channels.
$n$	The wordlength of $M$ .
$w$	The RNS channel width.
$\lceil X \rceil$	The ceiling of $X$ . The smallest integer greater than or equal to $X$ .
$\lfloor X \rfloor$	The floor of $X$ . The largest integer smaller than or equal to $X$ .
BE	Base Extension.
CRT	Chinese Remainder Theorem.
DSP	Digital Signal Processing.
ECC	Elliptic Curve Cryptography.
LUC	Look-Up Cycle.
LUT	Look-Up Table.
LUT	Look-Up Table

MRS Mixed Radix Number System.

QDS Quotient Digit Selection.

RNS Residue Number System.

RSA RSA Cryptography.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Thesis Outline . . . . .	2
1.2	Contribution . . . . .	4
<b>2</b>	<b>Background</b>	<b>7</b>
2.1	Residue Number Systems . . . . .	8
2.1.1	RNS Representation . . . . .	8
2.1.2	Conversion between RNS and Positional Number Systems . . . . .	9
2.1.3	RNS Arithmetic . . . . .	14
2.1.4	Moduli Selection . . . . .	15
2.1.5	Base Extension . . . . .	16
2.2	RSA Public-Key Cryptography . . . . .	18
2.2.1	Public-Key Cryptography . . . . .	18
2.2.2	The RSA Cryptosystem . . . . .	19
2.2.3	Exponentiation . . . . .	20
<b>3</b>	<b>Four Ways to Do Modular Multiplication in a Positional Number System</b>	<b>25</b>
3.1	Introducing Four Ways to Do Modular Multiplication . . . . .	26
3.1.1	Classical Modular Multiplication . . . . .	27
3.1.2	Sum of Residues Modular Multiplication . . . . .	29
3.1.3	Barrett Modular Multiplication . . . . .	30
3.1.4	Montgomery Modular Multiplication . . . . .	31

3.2	Reinvigorating Sum of Residues . . . . .	33
3.2.1	Tomlinson’s Algorithm . . . . .	33
3.2.2	Eliminating the Carry Propagate Adder . . . . .	35
3.2.3	Further Enhancements . . . . .	37
3.2.4	High Radix . . . . .	40
3.2.5	Summary of the Sum of Residues Modular Multiplication . . . . .	40
3.3	Bounds of Barrett Modular Multiplication . . . . .	42
3.3.1	Bound Deduction . . . . .	42
3.3.2	Performance at Different Word Lengths . . . . .	45
3.4	Montgomery Modular Multiplication on FPGA . . . . .	47
3.4.1	Separated Montgomery Modular Multiplication Algorithm . . . . .	47
3.4.2	High Radix Montgomery Algorithm . . . . .	47
3.4.3	Bound Deduction . . . . .	54
3.4.4	Interleaved vs. Separated Structure . . . . .	57
3.4.5	Trivial Quotient Digit Selection . . . . .	58
3.4.6	Quotient Digit Pipelining . . . . .	62
3.5	Modular Multiplications within RNS Channels . . . . .	69
<b>4</b>	<b>Four Ways to Do Modular Multiplication in the Residue Number System</b>	<b>71</b>
4.1	Short Word Length RNS Modular Multiplications using Look-Up Tables . . . . .	73
4.1.1	RNS Modular Reduction, Scaling and Look-Up Tables	74
4.1.2	Existing RNS Scaling Schemes using Look-Up Tables .	75
4.1.3	A New RNS Scaling Scheme using Look-Up Tables . .	88
4.2	RNS Classical Modular Multiplication . . . . .	93
4.2.1	The Core Function . . . . .	93
4.2.2	A Classical Modular Multiplication Algorithm in RNS using the Core Function . . . . .	97
4.2.3	Examples . . . . .	105

---

4.3	RNS Sum of Residues Modular Multiplication . . . . .	109
4.3.1	Sum of Residues Reduction in the RNS . . . . .	109
4.3.2	Approximation of $\alpha$ . . . . .	110
4.3.3	Bound Deduction . . . . .	115
4.3.4	The RNS Sum of Residues Modular Multiplication Al- gorithm . . . . .	117
4.4	RNS Barrett Modular Multiplication . . . . .	121
4.4.1	RNS Modular Reduction using Barrett Algorithm . . .	121
4.4.2	The Algorithm . . . . .	127
4.5	RNS Montgomery Modular Multiplication . . . . .	133
4.5.1	Montgomery Modular Reduction in RNS . . . . .	133
4.5.2	The Algorithm . . . . .	136
4.5.3	A Comparison between RNS Barrett and Montgomery Modular Multiplication . . . . .	136
<b>5</b>	<b>Implementation of RNS Sum of Residues Modular Multipli- cation</b>	<b>141</b>
5.1	A Scalable Structure for Sum of Residues Modular Multipli- cation in RNS . . . . .	142
5.1.1	A 4-Channel Architecture . . . . .	143
5.1.2	The Scaled Architecture for Modular Multiplication . .	144
5.2	Implementation Results . . . . .	149
<b>6</b>	<b>Conclusion and Future Perspectives</b>	<b>153</b>
6.1	Conclusion . . . . .	154
6.2	Future Perspectives . . . . .	154
	<b>Bibliography</b>	<b>157</b>

*CONTENTS*

---

# List of Figures

2.1	Conversion from the RNS to the Mixed Radix Number System (MRS) . . . . .	12
3.1	An Architecture for Tomlinson's Modular Multiplication Algorithm [Tomlinson89] . . . . .	34
3.2	Modified Sum of Residues Modular Multiplier Architecture . .	36
3.3	$n$ -bit Carry Save Adders . . . . .	37
3.4	New Sum of Residues Modular Multiplier Architecture . . . .	38
3.5	An Example for New Sum of Residues Modular Multiplication $n = 4, A = (1111)_2, B = (1011)_2$ and $M = (1001)_2$ . . . . .	39
3.6	New High-Radix Sum of Residues Modular Multiplier Architecture . . . . .	41
3.7	Delays and Areas of Improved Barrett Modular Multiplication Algorithm . . . . .	46
3.8	Delays of Montgomery Modular Multiplication at Different Radices. . . . .	54
3.9	Delays and Areas of Interleaved and Separated Montgomery Modular Multiplication at Different Radices . . . . .	59
3.10	Shortest Delays of Separated Montgomery Multiplication Algorithm using Trivial QDS at $n$ from 12 to 32 . . . . .	62
3.11	Shortest Delays of Separated Montgomery Multiplication Algorithm using Quotient Digit Pipelining at $n$ from 13 to 32 . .	67



3.12 Shortest Delays of Separated Montgomery Multiplication Algorithm with & without Quotient Digit Pipelining at $n$ from 13 to 32 . . . . .	68
3.13 Delays and Areas of Four Classes of Modular Multipliers in Binary . . . . .	70
4.1 RNS Scaling using Look-Up Tables. . . . .	75
4.2 The 3 Modulus RNS Scaler . . . . .	77
4.3 RNS Scaling using Estimation for Channels $S + 1 \leq i \leq N$ . .	80
4.4 RNS Scaling using a Decomposition of the CRT (two redundant channels with channel $m_i$ ) . . . . .	83
4.5 RNS Scaling using Parallel Base Extension (BE) Blocks . . . .	85
4.6 Parallel Architecture to Perform $Y = \lfloor \frac{X}{M} \rfloor \approx \sum_{i=1}^N f(x_i)$ . . . .	86
4.7 One Channel of a Parallel Residue Arithmetic Process using Memories with Addressing Capacity = $r$ . . . . .	87
4.8 A New Architecture to Perform Short Word length Scaling in RNS . . . . .	90
4.9 A Typical Core Function $C(X)$ : $D = 30030$ ; $C(D) = 165$ . . . .	95
4.10 The $h$ Least Significant Bits to be Maintained during the Modular Multiplication $\langle x_i C(\sigma_i) \rangle_{2^h}$ . . . . .	102
4.11 An End-Around Adder to Compute $\langle \theta_i + \theta_{i+1} \rangle_{2^k-1}$ . . . . .	126
4.12 Architecture for Scaling by 2 in RNS . . . . .	128
4.13 High-Radix Architecture for Scaling by $2^l$ in RNS . . . . .	129
4.14 Delay of Channel-Width Multiplication and Addition modulo a General Modulus and Multiplication and Addition modulo $2^k - 1$ used for RNS Barrett Algorithm against RNS Channel Width $w$ . . . . .	132
4.15 Delay of Barrett and Montgomery RNS Modular Multiplication against RNS Channel Width $w$ . . . . .	138
5.1 A 4-Channel RNS Modular Multiplier . . . . .	145

5.2	A 4-Channel Modular Multiplier Block Suitable for a Scalable Array of Blocks . . . . .	146
5.3	A 12-Channel Modular Multiplier Built from a $3 \times 3$ Array of 4-Channel Blocks . . . . .	148

*LIST OF FIGURES*

---

# List of Tables

2.1	The Pre-Computed Constants of RNS $\{3, 5, 7\}$ . . . . .	9
2.2	An Example of Right-to-Left Modular Exponentiation . . . . .	23
3.1	Montgomery Modular Reduction Process at Radix 2 . . . . .	49
3.2	Montgomery Modular Reduction Process at Radix 4 . . . . .	50
3.3	Montgomery Modular Reduction Process at Radix 8 ( $l = \lceil \frac{n}{k} \rceil$ ) . . . . .	52
3.4	Montgomery Modular Reduction Process at Radix 8 ( $l = \lfloor \frac{n}{k} \rfloor$ ) . . . . .	53
3.5	The $k$ -boundary of Montgomery Reduction for $n$ from 12 to 32 . . . . .	61
3.6	Shortest Delays of Separated Montgomery Multiplication Algorithm using Trivial QDS at $n$ from 12 to 32 . . . . .	61
3.7	$k$ and $d$ in Quotient Digit Pipelining of Montgomery Algorithm for $n$ from 12 to 25 . . . . .	66
3.8	Shortest Delays of Separated Montgomery Multiplication Algorithm using Quotient Digit Pipelining at $n$ from 13 to 32 . . . . .	67
4.1	A Comparison of 6 Short Word length RNS Scaling Schemes using LUTs . . . . .	92
4.2	An Example of Pre-Computed Parameters for the Classical Modular Reduction Algorithm in RNS using the Core Function . . . . .	106
4.3	An Example of the Classical Modular Reduction Algorithm in RNS using the Core Function for a case in which $\lfloor \text{Mag}(X) \rfloor \geq 1107$ . . . . .	1107
4.4	An Example of the Classical Modular Reduction Algorithm in RNS using the Core Function for a case in which $\lfloor \text{Mag}(X) \rfloor < 1108$ . . . . .	1108

4.5	The Maximum Possible $N$ against $w$ in RNS Sum of Residues Reduction . . . . .	113
5.1	The Pre-Computed Constants for the RNS Sum of Residues Modular Multiplication Algorithm in Algorithm 4.2 . . . . .	144