Springer Series on Signals and Communication Technology

SIGNALS AND COMMUNICATION TECHNOLOGY

Multimedia Database Retrieval

A Human-Centered Approach P. Muneesawang and L. Guan ISBN 0-387-25627-X

Broadband Fixed Wireless Access

A System Perspective M. Engels and F. Petre ISBN 0-387-33956-6

Distributed Cooperative Laboratories

Networking, Instrumentation, and Measurements F. Davoli, S. Palazzo and S. Zappatore (Eds.) ISBN 0-387-29811-8

The Variational Bayes Method in Signal Processing

V. Šmídl and A. Quinn ISBN 3-540-28819-8

Topics in Acoustic Echo and Noise Control

Selected Methods for the Cancellation of Acoustical Echoes, the Reduction of Background Noise, and Speech Processing E. Hänsler and G. Schmidt (Eds.) ISBN 3-540-33212-x

EM Modeling of Antennas and RF **Components for Wireless Communication** Systems

F. Gustrau, D. Manteuffel ISBN 3-540-28614-4

Interactive Video Methods and Applications

R. I Hammoud (Ed.) ISBN 3-540-33214-6

ContinuousTime Signals

Y. Shmaliy ISBN 1-4020-4817-3

Voice and Speech Quality Perception

Assessment and Evaluation U. Jekosch ISBN 3-540-24095-0

Advanced ManMachine Interaction

Fundamentals and Implementation K.-F. Kraiss

ISBN 3-540-30618-8

Orthogonal Frequency Division Multiplexing for Wireless Communications

Y. (Geoffrey) Li and G.L. Stüber (Eds.) ISBN 0-387-29095-8

Circuits and Systems **Based on Delta Modulation** Linear, Nonlinear and Mixed Mode Processing

ISBN 3-540-23751-8

Functional Structures in Networks

AMLn-A Language for Model Driven Development of Telecom Systems T. Muth ISBN 3-540-22545-5

RadioWave Propagation for Telecommunication Applications

ISBN 3-540-40758-8 H. Sizun

Electronic Noise and Interfering Signals Principles and Applications
G. Vasilescu ISBN 3-540-40741-3

D.G. Zrilic

The Family of International Standards for Digital Video Broadcasting, 2nd ed. U. Reimers ISBN 3-540-43545-X

Digital Interactive TV and Metadata

Future Broadcast Multimedia A. Lugmayr, S. Niiranen, and S. Kalli ISBN 3-387-20843-7

Adaptive Antenna Arrays

Trends and Applications S. Chandran (Ed.) ISBN 3-540-20199-8

Digital Signal Processing with Field Programmable Gate Arrays U. Meyer-Baese ISBN 3-540-21119-5

Neuro-Fuzzy and Fuzzy Neural Applications in Telecommunications

P. Stavroulakis (Ed.) ISBN 3-540-40759-6

SDMA for Multipath Wireless Channels Limiting Characteristics

and Stochastic Models I.P. Kovalyov ISBN 3-540-40225-X

Digital Television

A Practical Guide for Engineers W. Fischer ISBN 3-540-01155-2

Speech Enhancement

J. Benesty (Ed.) ISBN 3-540-24039-X

Multimedia Communication Technology

Representation, Transmission and Identification of Multimedia Signals J.R. Ohm ISBN 3-540-01249-4

Francisco Rodríguez-Henríquez N.A. Saqib A. Díaz-Pèrez Çetin Kaya Koç

Cryptographic Algorithms on Reconfigurable Hardware



Francisco Rodríguez-Henríquez Arturo Díaz Pérez

Departamento de Computación Centro de Investigación y de Estudios Avanzados del IPN Av. Instituto Politécnico Nacional No. 2508 Col. San Pedro Zacatenco. CP 07300 México, D.F. MEXICO

Nazar Abbas Saqib Centre for Cyber Technology and Spectrum Management (CCT & SM) National University of Sciences and Technology (NUST) #295, Street 35, F-11/3, Islamabad-44000 Pakistan

Çetin Kaya Koç Oregon State University Corvallis, OR 97331, USA & Istanbul Commerce University Eminönü, Istanbul 34112, Turkey

Cryptographic Algorithms on Reconfigurable Hardware

Library of Congress Control Number: 2006929210

ISBN 0-387-33883-7 e-ISBN 0-387-36682-2 ISBN 978-0-387-33883-5

Printed on acid-free paper.

© 2006 Springer Science+Business Media, LLC

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now know or hereafter developed is forbidden.

The use in this publication of trade names, trademarks, service marks and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed in the United States of America.

987654321

springer.com

Dedication

A mi esposa Nareli y mi hija Ana Iremi, por su amor y estoica paciencia; A mis padres y hermanos, por compartir las mismas esperanzas. Francisco Rodríguez-Henríquez

To Afshan (wife), Fizza (daughter), Ahmer (son) and Aashir (son), I love you all.

Nazar A. Saqib

To Mary, Maricarmen and Liliana, my wife and daughters, my love will keep alive for you all.

Arturo Díaz-Pérez

With my love to Laurie, Murat, and Cemre. Çetin K. Koç

Contents

Lis	st of	Figures XIII
Lis	st of	Tables
Lis	st of	AlgorithmsXX
Αc	rony	msXXIII
Pr	eface	xxv
1	Inti	roduction
	1.1	Main goals 1
	1.2	Monograph Organization
	1.3	Acknowledgments
2	A E	Brief Introduction to Modern Cryptography 7
	2.1	Introduction 8
	2.2	Secret Key Cryptography
	2.3	Hash Functions
	2.4	Public Key Cryptography 12
	2.5	Digital Signature Schemes
		2.5.1 RSA Digital Signature
		2.5.2 RSA Standards
		2.5.3 DSA Digital Signature
		2.5.4 Digital Signature with Elliptic Curves 19
		2.5.5 Key Exchange
	2.6	A Comparison of Public Key Cryptosystems
	2.7	Cryptographic Security Strength
	2.8	Potential Cryptographic Applications
	2.9	Fundamental Operations for Cryptographic Algorithms 29

VIII Contents

	2.10	Design Alternatives for Implementing Cryptographic	
		Algorithms	31
	2.11	Conclusions	32
3	Doo	onfigurable Hardware Technology	35
3	3.1	Antecedents	36
	$\frac{3.1}{3.2}$	Field Programmable Gate Arrays	38
	0.4	3.2.1 Case of Study I: Xilinx FPGAs	39
		3.2.2 Case of Study II: Altera FPGAs	44
	3.3	FPGA Platforms versus ASIC and General-Purpose	44
	ა.ა	Processor Platforms	48
		3.3.1 FPGAs versus ASICs	48
	0.4	3.3.2 FPGAs versus General-Purpose Processors	49
	3.4	Reconfigurable Computing Paradigm	50
		3.4.1 FPGA Programming	52
		3.4.2 VHSIC Hardware Description Language (VHDL)	52
		3.4.3 Other Programming Models for FPGAs	53
	3.5	Implementation Aspects for Reconfigurable Hardware Designs	53
		3.5.1 Design Flow	53
		3.5.2 Design Techniques	55
		3.5.3 Strategies for Exploiting FPGA Parallelism	58
	3.6	FPGA Architecture Statistics	59
	3.7	Security in Reconfigurable Hardware Devices	61
	3.8	Conclusions	62
4	Mat	thematical Background	63
-	4.1	Basic Concepts of the Elementary Theory of Numbers	63
	1.1	4.1.1 Basic Notions	64
		4.1.2 Modular Arithmetic	67
	4.2	Finite Fields	70
		4.2.1 Rings	70
		4.2.2 Fields	70
		4.2.3 Finite Fields	70
		4.2.4 Binary Finite Fields	71
	4.3	Elliptic curves	73
		4.3.1 Definition	73
		4.3.2 Elliptic Curve Operations	74
		4.3.3 Elliptic Curve Scalar Multiplication	76
	4.4	Elliptic Curves over $GF(2^m)$	77
		4.4.1 Point Addition	78
		4.4.2 Point Doubling	78
		4.4.3 Order of an Elliptic Curve	79
		4.4.4 Elliptic Curve Groups and the Discrete Logarithm	
		Problem	79
		4.4.5 An Example	79

				Contents	IX
	4.5	Point	Representation		. 82
		4.5.1	Projective Coordinates		
		4.5.2	López-Dahab Coordinates		
	4.6	Scala	Representation		
		4.6.1	Binary Representation		. 85
		4.6.2	Recoding Methods		. 85
		4.6.3	ω -NAF Representation		. 87
	4.7	Concl	usions		
5	Pri	me Fi	nite Field Arithmetic		. 89
	5.1		ion Operation		
		5.1.1	Full-Adder and Half-Adder Cells		
		5.1.2	Carry Propagate Adder		
		5.1.3	Carry Completion Sensing Adder		
		5.1.4	Carry Look-Ahead Adder		
		5.1.5	Carry Save Adder		
		5.1.6	Carry Delayed Adder		
	5.2	Modu	ılar Addition Operation		
		5.2.1	Omura's Method		
	5.3	Modu	ılar Multiplication Operation		
		5.3.1	Standard Multiplication Algorithm		
		5.3.2	Squaring is Easier		
		5.3.3	Modular Reduction		
		5.3.4	Interleaving Multiplication and Reduction .		
		5.3.5	Utilization of Carry Save Adders		
		5.3.6	Brickell's Method		
		5.3.7	Montgomery's Method		. 116
		5.3.8	High-Radix Interleaving Method		
		5.3.9	High-Radix Montgomery's Method		. 124
	5.4	Modu	dar Exponentiation Operation		. 124
		5.4.1	Binary Strategies		. 125
		5.4.2	Window Strategies		
		5.4.3	Adaptive Window Strategy		. 129
		5.4.4	RSA Exponentiation and the Chinese Rem	ainder	
			Theorem		. 132
		5.4.5	Recent Prime Finite Field Arithmetic Designation		
		~ .	FPGAs		
	5.5	Concl	lusions	• • • • • • • • • • • • • • • • • • • •	. 138
6			inite Field Arithmetic		
	6.1		Multiplication		
		6.1.1	Classical Multipliers and their Analysis		
		6.1.2	Binary Karatsuba-Ofman Multipliers		
		6.1.3	Squaring		
		6.1.4	Reduction		. 152

37	a , ,
X	Contents

		6.1.5 Modular Reduction with General Polynomials	156
		6.1.6 Interleaving Multiplication	
		6.1.7 Matrix-Vector Multipliers	
		6.1.8 Montgomery Multiplier	
		6.1.9 A Comparison of Field Multiplier Designs	
	6.2	Field Squaring and Field Square Root for Irreducible Trinomia	
	0.2	6.2.1 Field Squaring Computation	
		6.2.2 Field Square Root Computation	
		6.2.3 Illustrative Examples	
	6.3	Multiplicative Inverse	
	0.0	6.3.1 Inversion Based on the Extended Euclidean Algorithm	
		6.3.2 The IToh-Tsujii Algorithm	
		6.3.3 Addition Chains	
		6.3.4 ITMIA Algorithm	
		6.3.5 Square Root ITMIA	
		6.3.6 Extended Euclidean Algorithm versus Itoh-Tsujii	. 119
		Algorithm	191
		6.3.7 Multiplicative Inverse FPGA Designs	192
	6.4	Other Arithmetic Operations	
	0.4	6.4.1 Trace function	
		6.4.2 Solving a Quadratic Equation over $GF(2^m)$	
		6.4.3 Exponentiation over Binary Finite Fields	
	6.5	Conclusions	
	0.0	Concretions	100
7	\mathbf{Rec}	configurable Hardware Implementation of Hash	
	Fun	ctions	189
	7.1	Introduction	
	7.2	Some Famous Hash Functions	. 191
	7.3	MD5	
		7.3.1 Message Preprocessing	
		7.3.2 MD Buffer Initialization	
		7.3.3 Main Loop	
		7.3.4 Final Transformation	
	7.4	SHA-1, SHA-256, SHA-384 and SHA-512	
		7.4.1 Message Preprocessing	
		7.4.2 Functions	. 204
		7.4.3 SHA-1	. 205
		7.4.4 Constants	
		7.4.5 Hash Computation	
	7.5	Hardware Architectures	
		7.5.1 Iterative Design	
		7.5.2 Pipelined Design	212
		7.5.3 Unrolled Design	. 212
			. 212 213

	7.7	Conclusions	. 220
8		neral Guidelines for Implementing Block Ciphers in	
		GAs	
	8.1	Introduction	
	8.2	Block Ciphers	
		8.2.1 General Structure of a Block Cipher	
		8.2.2 Design Principles for a Block Cipher	. 224
		8.2.3 Useful Properties for Implementing Block Ciphers in	
		FPGAs	
	8.3	The Data Encryption Standard	. 232
		8.3.1 The Initial Permutation (IP $^{-1}$)	
		8.3.2 Structure of the Function f_k	
		8.3.3 Key Schedule	
	8.4	FPGA Implementation of DES Algorithm	
		8.4.1 DES Implementation on FPGAs	
		8.4.2 Design Testing and Verification	. 240
		8.4.3 Performance Results	. 240
	8.5	Other DES Designs	. 240
	8.6	Conclusions	. 244
9		chitectural Designs For the Advanced Encryption ndard	. 245
	9.1	Introduction	. 245
	9.2	The Rijndael Algorithm	. 247
		9.2.1 Difference Between AES and Rijndael	
		9.2.2 Structure of the AES Algorithm	
		9.2.3 The Round Transformation	
		9.2.4 ByteSubstitution (BS)	
		9.2.5 ShiftRows (SR)	
		9.2.6 MixColumns (MC)	
		9.2.7 AddRoundKey (ARK)	
		9.2.8 Key Schedule	
	9.3	AES in Different Modes	
		9.3.1 CTR Mode	
		9.3.2 CCM Mode	
	9.4	$\label{eq:main_equation} \textbf{Implementing AES Round Basic Transformations on FPGAs} .$	
		9.4.1 S-Box/Inverse S-Box Implementations on FPGAs	. 260
		9.4.2 MC/IMC Implementations on FPGA	
		9.4.3 Key Schedule Optimization	
	9.5	AES Implementations on FPGAs	
		9.5.1 Architectural Alternatives for Implementing AES	
		9.5.2 Key Schedule Algorithm Implementations	. 273
		9.5.3 AES Encryptor Cores - Iterative and Pipeline	
		Approaches	. 276

		9.5.4	AES Encryptor/Decryptor Cores- Using Look-Up
			Table and Composite Field Approaches for S-Box 278
		9.5.5	AES Encryptor/Decryptor, Encryptor, and Decryptor
			Cores Based on Modified MC/IMC
		9.5.6	Review of This Chapter Designs
	9.6	Perfor	rmance
		9.6.1	Other Designs
	9.7	Concl	usions
10	Elli	ptic C	curve Cryptography291
	10.1	Introd	duction
	10.2	Hessia	an Form
	10.3	Weier	strass Non-Singular Form
		10.3.1	Projective Coordinates
		10.3.2	The Montgomery Method
	10.4		lel Strategies for Scalar Point Multiplication 300
			menting scalar multiplication on Reconfigurable Hardware 302
		10.5.1	Arithmetic-Logic Unit for Scalar Multiplication 303
		10.5.2	Scalar multiplication in Hessian Form
		10.5.3	Montgomery Point Multiplication
			Implementation Summary
	10.6	Kobli	tz Curves
			The τ and τ^{-1} Frobenius Operators
			$\omega \tau$ NAF Scalar Multiplication in Two Phases
		10.6.3	Hardware Implementation Considerations
	10.7		and-Add Algorithm for Scalar Multiplication317
		10.7.1	Efficient Elliptic Curve Arithmetic
			Implementation
		10.7.3	Performance Estimation
	10.8	Perfor	rmance Comparison
	10.9	Concl	usions
Re	feren	ces	329
Ind	lex		

List of Figures

2.1	A Hierarchical Six-Layer Model for Information Security	
	Applications	8
2.2	Secret Key Cryptography	10
2.3	Recovering Initiator's Private Key	11
2.4	Generating a Pseudorandom Sequence	12
2.5	Public Key Cryptography	12
2.6	Basic Digital Signature/Verification Scheme	13
2.7	Public key cryptography Main Primitives	14
2.8	Diffie-Hellman Key Exchange Protocol	24
2.9	Elliptic Curve Variant of the Diffie-Hellman Protocol	25
3.1	A Taxonomy of Programmable Logic Devices	38
3.2	Xilinx Virtex II Architecture	40
3.3	Xilinx CLB	41
3.4	Slice Structure	42
3.5	VirtexE Logic Cell (LC)	42
3.6	CLB Configuration Modes	42
3.7	Stratix Block Diagram	45
3.8	Stratix LE	46
3.9	Design flow	54
3.10	Hardware Design Methodology	56
	2-bit Multiplixer Using (a) Tristate Buffer. (b) LUT	57
3.12	Basic Architectures for (a) Iterative Looping (b) Loop Unrolling	58
3.13	Round-pipelining for (a) One Round (b) n Rounds	59
4.1	Elliptic Curve Equation $y^2 = x^3 + ax + b$ for Different a and b .	73
4.2	Adding two Distinct Points on an Elliptic curve $(Q \neq -P)$	74
4.3	Adding two Points P and Q when $Q = -P$	75
4.4	Doubling a Point P on an Elliptic Curve	75
4.5	Doubling $P(x, y)$ when $y = 0$	76

XIV List of Figures

4.6	Elliptic Curve Scalar Multiplication kP , for $k=6$ and for the Elliptic Curve $y^2 = x^3 - 3x + 3 \dots$. 77
4.7	Elements in the Elliptic Curve of Equation (4.15)	81
5.1	Full-Adder and Half-Adder Cells	
5.2	Carry Propagate Adder	
5.3	Carry Completion Sensing Adder	
5.4	Detecting Carry Completion	
5.5	Carry Look-Ahead Adder	
5.6	Carry Save Adder	
5.7	Carry Delayed Adder	
5.8	High-Radix Interleaving Method	
5.9	Partitioning Algoritm	. 130
6.1	Binary Karatsuba-Ofman Strategy	. 148
6.2	Karatsuba-Ofman Multiplier $GF(2^{191})$. 150
6.3	Programmable Binary Karatsuba-Ofman Multiplier	
6.4	Squaring Circuit	
6.5	Reduction Scheme	
6.6	Pentanomial Reduction	
6.7	A Method to Reduce k Bits at Once	
6.8	$\alpha \cdot A(\alpha)$ Multiplication	
6.9	LSB-First Serial/Parallel Multiplier	
	Finite State Machine for the Binary Euclidean Algorithm	
6.11	Architecture of the Itoh-Tsujii Algorithm	. 182
7.1	Hash Function	. 190
7.2	Requirements of a Hash Function	
7.3	Basic Structure of a Hash Function	
7.4	MD5	
7.5	Message Block = $32 \times 16 = 512$ Bits	. 195
7.6	Auxiliary Functions in Reconfigurable Hardware (a) F(X,Y,Z)	
	(b) $G(X,Y,Z)$ (c) $H(X,Y,Z)$ (d) $I(X,Y,Z)$	
7.7	One MD5 Operation	
7.8	Padding Message in SHA-1 and SHA-256	
7.9	Padding Message in SHA-384 and SHA-512	. 204
7.10	Implementing SHA-1 Auxiliary Functions in Reconfigurable Hardware	วกร
7 11	$\Sigma_0, \Sigma_1, \sigma_0$, and σ_1 in Reconfigurable Hardware	
	Single Operation for SHA-1	
	Single Operation for SHA-256	
	Iterative Approach for Hash Function Implementation	
	Hash Function Implementation (a) Unrolled Design (b)	. 411
1.10	Combining k Stages	212
7 16	A Mixed Approach for Hash Function Implementation	213

	List of Figures	XV
8.11	General Structure of a Block Cipher Same Resources for 2,3,4-in/1-out Boolean Logic in FPGAs Three Approaches for the Implementation of S-Box in FPGAs Permutation Operation in FPGAs Shift Operation in FPGAs Iterative Design Strategy Pipeline Design Strategy Sub-pipeline Design Strategy DES Algorithm DES Implementation on FPGA Functional Simulation Timing Verification	228 229 229 230 231 231 231 234 239 241
9.1 9.2 9.3 9.4	Basic Structure of Rijndael Algorithm	249 250
9.5 9.6 9.7	MixColumns Operates at Columns of the State Matrix ARK Operates at Bits of the State Matrix	252 253
$9.8 \\ 9.9$	Authentication and Verification Process for the CCM Mode Encryption and Decryption Processes for the CCM Mode	257 258
9.11	S-Box and Inv. S-Box Using Same Look-Up Table	262
	Composite Fields	269
$9.16 \\ 9.17$	Loop Unrolling Design Strategy Pipeline Design Strategy Sub-pipeline Design Strategy	271 272
9.18 9.19	Sub-pipeline Design Strategy with Balanced Stages KGEN Architecture	$272 \\ 274$
9.22	Key Schedule for a Fully Pipeline Encryptor Core	
9.24	with Modified IMC	277
$9.26 \\ 9.27$	S-Box and Inv S-Box Using (a) Different MI (b) Same MI Data Path for Encryption/Decryption	279 280
9.28 9.29	Block Diagram for 3-Stage MI Manipulation	

XVI List of Figures

9.30	$GF(2^2)^2$ and $GF(2^2)$ Multipliers
9.31	Gate Level Implementation for x^2 and λx
9.32	AES Algorithm Encryptor/Decryptor Implementation282
9.33	The Data Path for Encryptor Core Implementation
9.34	The Data Path for Decryptor Core Implementation
	Hierarchical Model for Elliptic Curve Cryptography293
10.2	Basic Organization of Elliptic Curve Scalar Implementation303
10.3	Arithmetic-Logic Unit for Scalar Multiplication on FPGA
	Platforms
10.4	An illustration of the τ and τ^{-1} Abelian Groups (with m an
	Even Number)
10.5	A Hardware Architecture for Scalar Multiplication on the
	NIST Koblitz Curve K-233316
10.6	Point Halving Scalar Multiplication Architecture
	Point Halving Arithmetic Logic Unit
	Point Halving Execution
	Point Addition Execution
	OPoint Doubling Execution

List of Tables

2.1	A Comparison of Security Strengths (Source: [258])
2.2	A Few Potential Cryptographic Applications
2.3	Primitives of Cryptographic Algorithms (Symmetric Ciphers) 30
2.4	Comparison between Software, VLSI, and FPGA Platforms 31
3.1	FPGA Manufacturers and Their Devices
3.2	Xilinx FPGA Families Virtex-5, Virtex-4, Virtex II Pro and
	Spartan 3E
3.3	Dual-Port BRAM Configurations
3.4	Altera Stratix Devices 45
3.5	Comparing Cryptographic Algorithm Realizations on different
	Platforms
3.6	High Level FPGA Programming Software 53
4.1	Elements of the field $F = GF(2^4)$, Defined Using the Primitive
	Trinomial of Eq. ((4.12))
4.2	Scalar Multiples of the Point P of Equation $(4.16) \dots 82$
4.3	A Toy Example of the Recoding Algorithm
4.4	Comparing Different Representations of the Scalar $k \dots 88$
5.1	Modular Exponentiation Comparison Table
5.2	Modular Exponentiation: Software vs Hardware Comparison
	Table
6.1	The Computation of $C(x)$ Using Equation (6.5)
6.2	Space and Time Complexities for Several $m = 2^k$ -bit Hybrid
	Karatsuba-Ofman Multipliers
6.3	Fastest Reconfigurable Hardware $GF(2^m)$ Multipliers 165
6.4	Most Compact Reconfigurable Hardware $GF(2^m)$ Multipliers 166
6.5	Summary of Complexity Results

XVIII List of Tables

6.6	Irreducible Trinomials $P(x) = x^m + x^n + 1$ of Degree	
	$m \in [160, 571]$ Encoded as $m(n)$, with m a Prime Number	. 171
6.7	Squaring matrix M of Eq. (6.40)	. 172
6.8	Square Root Matrix M^{-1} of Eq. (6.41)	. 173
6.9	Square and Square Root Coefficient Vectors	. 174
6.10	$\beta_i(a)$ Coefficient Generation for $m-1=192$. 180
	$\gamma_i(a)$ Coefficient Generation for m -1=192	
6.12	BEA Versus ITMIA: A Performance Comparison	. 183
	Design Comparison for Multiplicative Inversion in $GF(2^m)$	
7.1	Some Known Hash Functions	
7.2	Bit Representation of the Message M	
7.3	Padded Message (M)	
7.4	Message in Little Endian Format	
7.5	Initial Hash Values in Little Endian Format	
7.6	Auxiliary Functions for Four MD5 Rounds	. 197
7.7	Four Operations Associated to Four MD5 Rounds	. 198
7.8	Round 1	. 199
7.9	Round 2	. 199
7.10	Round 3	. 200
7.11	Round 4	. 200
7.12	Final Transformation	. 201
	Comparing Specifications for Four Hash Algorithms	
	Initial Hash Values for SHA-1	
	Initial Hash Values for SHA-256	
	Initial Hash Values for SHA-384	
7.17	Initial Hash Values for SHA-512	. 205
	SHA-256 Constants	
	SHA-384 & SHA-512 Constants	
	MD5 Hardware Implementations	
7.21	Representative SHA-1 hardware Implementations	. 216
7.22	Representative RIPEMD-160 FPGA Implementations	. 217
7.23	Representative SHA-2 FPGA Implementations	. 218
7.24	Representative Whirlpool FPGA Implementations	. 219
	V. D	
8.1	Key Features for Some Famous Block Ciphers	. 227
8.2	Initial Permutation for 64-bit Input Block	
8.3	E-bit Selection	
8.4	DES S-boxes	
8.5	Permutation P	
8.6	Inverse Permutation	
8.7	Permuted Choice one PC-1	
8.8	Number of Key Bits Shifted per Round	
8.9	Permuted Choice two (PC-2)	. 238
Q 10	Tost Voctors	940

	List of Tables	XIX
8.11	DES Comparison: Fastest Designs	242
	DES Comparison: Compact Designs	
	DES Comparison: Efficient Designs	
	TripleDES Designs	
9.1	Selection of Rijndael Rounds	248
9.2	A Roadmap to Implemented AES Designs	273
9.3	Specifications of AES FPGA implementations	
9.4	AES Comparison: High Performance Designs	286
9.5	AES Comparison: Compact Designs	
9.6	AES Comparison: Efficient Designs	
9.7	AES Comparison: Designs with Other Modes of Operation	288
10.1	$GF(2^m)$ Elliptic Curve Point Multiplication Computational	
	Costs	302
10.2	Point addition in Hessian Form	305
10.3	Point doubling in Hessian Form	305
	kP Computation, if Test-Bit is '1'	
10.5	kP Computation, If Test-Bit is '0'	307
10.6	Design Implementation Summary	308
	Parallel López-Dahab Point Doubling Algorithm	
10.8	Parallel López-Dahab Point Addition Algorithm	319
10.9	Operations Supported by the ALU Module	. 323
	OCycles per Operation	
10.11	1Fastest Elliptic Curve Scalar Multiplication Hardware Designs	326
10.12	2Most Compact Elliptic Curve Scalar Multiplication Hardware	
	Designs	326
10.13	BMost Efficient Elliptic Curve Scalar Multiplication Hardware	
	Designs	327

List of Algorithms

2.1	RSA Key Generation
2.2	RSA Digital Signature
2.3	RSA Signature Verification
2.4	DSA Domain Parameter Generation
2.5	DSA Key Generation
2.6	DSA Signature Generation 20
2.7	DSA Signature Verification
2.8	ECDSA Key Generation
2.9	ECDSA Digital Signature Generation
2.10	ECDSA Signature Verification
4.1	Euclidean Algorithm (Computes the Greatest Common Divisor) 65
4.2	Extended Euclidean Algorithm as Reported in [228] 69
4.3	Basic Doubling & Add algorithm for Scalar Multiplication 85
4.4	The Recoding Binary algorithm for Scalar Multiplication 86
4.5	ω -NAF Expansion Algorithm
5.1	The Standard Multiplication Algorithm
5.2	The Standard Squaring Algorithm
5.3	The Restoring Division Algorithm
5.4	The Nonrestoring Division Algorithm
5.5	The Interleaving Multiplication Algorithm
5.6	The Carry-Save Interleaving Multiplication Algorithm 110
5.7	The Carry-Save Interleaving Multiplication Algorithm Revisited 113
5.8	Montgomery Product
5.9	Montgomery Modular Multiplication: Version I
	Montgomery Modular Multiplication: Version II
5.11	Specialized Modular Inverse
	Montgomery Modular Exponentiation
	Add-and-Shift Montgomery Product
	Binary Add-and-Shift Montgomery Product
5.15	Word-Level Add-and-Shift Montgomery Product 124
5.16	MSB-First Binary Exponentiation

XXII LIST OF ALGORITHMS

5.17	LSB-First Binary Exponentiation	127
	MSB-First 2^k -ary Exponentiation	
5.19		
6.1	$mul2^{k}(C, A, B)$: $m = 2^{k}n$ -bit Karatsuba-Ofman Multiplier	
6.2	$mulgen_{\cdot}d(C,A,B)$: m -bit Binary Karatsuba-Ofman Multiplier . 1	149
6.3	Constructing a Look-Up Table that Contains All the 2^k	
	Possible Scalars in Equation (6.23)	157
6.4	Generating a Look-Up Table that Contains All the 2^k Possible	
	Scalars Multiplications $S \cdot P \dots \dots \dots$	158
6.5	Modular Reduction Using General Irreducible Polynomials	159
6.6	LSB-First Serial/Parallel Multiplier	161
6.7	Montgomery Modular Multiplication Algorithm	164
6.8	Binary Euclidean Algorithm	176
6.9	Itoh-Tsujii Multiplicative Inversion Addition-Chain Algorithm . :	179
6.10	Square Root Itoh-Tsujii Multiplicative Inversion Algorithm	181
6.11	MSB-first Binary Exponentiation	185
6.12	Square root LSB-first Binary Exponentiation	186
6.13	Squaring and Square Root Parallel Exponentiation	187
10.1	Doubling & Add algorithm for Scalar Multiplication: MSB-First:	295
10.2	Doubling & Add algorithm for Scalar Multiplication: LSB-First	295
10.3	Montgomery Point Doubling	297
10.4	Montgomery Point Addition	298
10.5	Montgomery Point Multiplication	299
10.6	Standard Projective to Affine Coordinates	299
10.7	$\omega \tau$ NAF Expansion[133, 132]	312
10.8	$\omega \tau \text{NAF Scalar Multiplication [133, 132]} \dots$	313
10.9	$\omega \tau$ NAF Scalar Multiplication: Parallel Version	314
	$0\omega au$ NAF Scalar Multiplication: Hardware Version \dots	
10.1	$1\omega au$ NAF Scalar Multiplication: Parallel HW Version	315
	2Point Halving Algorithm	
10.13	3Half-and-Add LSB-First Point Multiplication Algorithm	321

Acronyms

AES Advanced Encryption Standard

AF Affine Transformation

ANSI American National Standard Institute API Application Programming Interface

ARK Add Round Key

ASIC Application Specific Integrated Circuit

ATM Automated Teller Machine BEA Binary Euclidean Algorithm

BRAMs Block RAMs

BS Byte Substitution

CBC Cipher Block Chaining

CCM Counter with CBC-MAC

CCSA Carry Completion Sensing Adder

CDA Carry Delayed Adder

CFB Cipher Feedback mode

CLB Configurable Logic Block

CPA Carry Propagate Adder

CPLDs Complex PLDs

CRT Chinese Remainder Theorem

CSA Carry Save Adder

CTR Counter mode

DCM Digital Clock Managers

DEA Data Encryption Algorithm

DES Data Encryption Standard

DSA Digital Signature Algorithm

DSS Digital Signature Standard

ECB Electronic Code Book

ECC Elliptic Curve Cryptography

ECDLP Elliptic Curve Discrete Logarithmic Problem

ECDSA Elliptic Curve Digital Signature Algorithm

ETSI European Telecommunications Standards Institute

FIPS Federal Information Processing Standards

FLT Fermat's Little Theorem

FPGAs Field Programmable Gate Arrays

XXIV

GAL Generic Array Logic GSM Global System for Mobile Communications HDLs Hardware Description Languages IAF Inverse Affine Transformation IARK Inverse Add Round Key IBS Inverse Byte Substitution IEEE Institute of Electrical and Electronics Engineers Π Iterative Looping IMC Inverse Mix Column IOBsInput/Output Blocks IOEs Input/Output Elements IPSec Internet Protocol Security ISE Xilinx Integrated Software Environment ISO International Organization for Standardization ISR Inverse ShiftRow ITMIA Itoh-Tsujii Multiplicative Inverse Algorithm International Telecommunication Union JTAG Joint Test Action Group KOM Karatsuba-Ofman Multiplier LABs Logic Array Blocks LCLogic Cell LEs Logic Elements MAC Message Authentication Code MRC Mixed-Radix Conversion NAF Non-Adjacent Form NFS Number Field Sieve NIST National Institute of Standards and Technology NZWS Nonzero Window State Output Feedback mode OFB Programmable Array Logic PAL PC-1 Permuted Choice One PC-2 Permutated Choice Two PDAs Portable Digital Assistants PKCS Public Key Cryptography Standard PLAProgrammable Logic Array PLDs Programmable Logic Devices SRC Single-Radix Conversion SSLSecure Socket Layer TDEA Triple DEA TNAF τ -adic NAF

VHDL Very-High-Speed Integrated Circuit Hardware Description Language

VLSI Very Large Scale Integration WEP Wired Equivalent Privacy ZWS Zero Window State

Preface

Cryptography provides techniques, mechanisms, and tools for private and authenticated communication, and for performing secure and authenticated transactions over the Internet as well as other open networks. It is highly probable that each bit of information flowing through our networks will have to be either encrypted and decrypted or signed and authenticated in a few years from now. This infrastructure is needed to carry over the legal and contractual certainty from our paper-based offices to our virtual offices existing in the cyberspace. In such an environment, server and client computers as well as handheld, portable, and wireless devices will have to be capable of encrypting or decrypting and signing or verifying messages. That is to say, without exception, all networked computers and devices must have cryptographic layers implemented, and must be able to access to cryptographic functions in order to provide security features. In this context, efficient (in terms of time, area, and power consumption) hardware structures will have to be designed, implemented, and deployed. Furthermore, general-purpose (platform-independent) as well as special-purpose software implementing cryptographic functions on embedded devices are needed. An additional challenge is that these implementations should be done in such a way to resist cryptanalytic attacks launched against them by adversaries having access to primary (communication) and secondary (power, electromagnetic, acoustic) channels.

This book, among only a few on the subject, is a fruit of an international collaboration to design and implement cryptographic functions. The authors, who now seem to be scattered over the globe, were once together as students and professors in North America. In Oregon and Mexico City, we worked on subjects of mutual interest, designing efficient realizations of cryptographic functions in hardware and software.

Cryptographic realizations in software platforms can be used for those security applications where the data traffic is not too large and thus low encryption rate is acceptable. On the other hand, hardware methods offer high speed and bandwidth, providing real-time encryption if needed. VLSI (also known as ASIC) and FPGAs are two distinct alternatives for implementing

cryptographic algorithms in hardware. FPGAs offer several benefits for cryptographic algorithm implementations over VLSI, as they offer flexibility and fast time-to-market. Because they are reconfigurable, internal architectures, system parameters, lookup tables, and keys can be changed in FPGAs without much effort. Moreover, these features come with low cost and without sacrificing efficiency.

This book covers computational methods, computer arithmetic algorithms, and design improvement techniques needed to obtain efficient implementations of cryptographic algorithms in FPGA reconfigurable hardware platforms. The concepts and techniques introduced in this book pay special attention to the practical aspects of reconfigurable hardware design, explain the fundamental mathematics behind the algorithms, and give comprehensive descriptions of the state-of-the-art implementation techniques. The main goal pursued in this book is to show how one can obtain high-speed cryptographic implementations on reconfigurable hardware devices without requiring prohibitive amount of hardware resources.

Every book attempts to take a still picture of a moving subject and will soon need to be updated, nevertheless, it is our hope that engineers, scientists, and students will appreciate our efforts to give a glimpse of this deep and exciting world of cryptographic engineering. Thanks for reading our book.

May 2006

F. Rodríguez-Henríquez, Nazar A. Saqib, A. Díaz-Pérez, and Çetin K. Koç