社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2007－SLDM－130 (11)
2007／5／11

楕円曲線暗号に適したGF(2^m)上のSIMD型MSD乗算器の設計

奈良 竜太† 清水 一範† 小原 俊逸†† 戸川 望† 柳澤 政生†
大附 辰夫†

† 早稲田大学基幹理工学研究科
†† 早稲田大学理工学研究科
〒 169-8555 東京都新宿区大久保 3-4-1 Tel: 03-3209-3211(5775), Fax: 03-3208-7439
E-mail: †nara@togawa.cs.waseda.ac.jp

あらまし 楕円曲線暗号ハードウェアを実装する場合，用途に合わせて複数の鍵長を処理することが求められる．Digit-serial 乗算器は楕円曲線暗号を構成する GF(2^m) 上の乗算を高速に処理することができる．しかし，digit-serial 乗算器はデータ長が固定された数値を扱うことに適しており，複数の鍵長を扱う暗号システムには向いていなかった．そこで本稿では楕円曲線暗号に適した GF(2^m) 上の SIMD 型 MSD 乗算器を提案する．digit-serial 乗算器の一つである MSD 乗算器をデータ長に合わせて SIMD 演算で乗算を並列処理することにより，楕円曲線スカラー乗算を高速処理することができる．また，NIST が推奨する 5 つのデータ長について提案乗算器で処理することができるため，5 種類の MSD 乗算器を使用した場合に対し処理速度が同程度で比較した場合，面積を最大 1/3 まで削減することができる．また短い鍵長に対し SIMD 演算することで 2 つの乗算を同時に処理することができるため，従来の MSD 乗算器と比較し最大で約 2 倍の処理速度を得ることができる．

キーワード $GF(2^m)$, digit-serial 乗算器，MSB (most significant bit) 乗算器，MSD (most significant digit) 乗算器，SIMD, 楕円曲線暗号，公開鍵暗号

An SIMD MSD Multiplier based on variable GF(2^m) for Elliptic Curve Cryptosystems

Ryuta NARA†, Kazunori SHIMIZU†, Shunitsu KOHARA††, Nozomu TOGAWA†, Masao YANAGISAWA†, and Tatsuo OHTSUKI†

† Grad. of Fundamental Science and Engineering, Waseda University
†† Dept. of Computer Science,
Waseda University 3-4-1 Okubo, Shinjuku, Tokyo 169-8555, Japan
Tel: +81-3-3209-3211(5775), Fax: +81-3-3208-7439
E-mail: †nara@togawa.cs.waseda.ac.jp

Abstract Originally elliptic curve cryptosystem (ECC) hardware are often required to operate variable key length. Digit-serial multipliers for ECC enable the hardware to accelerale the finite field operation. However, the lack of flexibility of digit-serial multipliers is major challenge for building the ecc hardware which operates variable key length. In this paper, we propose a SIMD MSD multiplier based on variable GF(2^m) for ECC. Adjusting the parallellizm of the SIMD MSD multiplier according to the field length enables us to accelarate the ecc scalar multiplication throughput. The proposed multiplier operates 5 types of field length which are recommended by NIST, where 2 multiplications can be operated simultaneously for the small field length. Implementation results show that the proposed multiplier reduces the hardware area by up to 1/3 compared to the same throughput. while achieving up to about 2 times multiplication throughput compared to the conventional multipliers for the variable field length.

Key words $GF(2^m)$, digit-serial multiplier, most significant bit (MSB) multiplier , most significant digit (MSD) multiplier, SIMD, elliptic curve cryptosystem, public key cryptosystem

1. まえがき

楕円曲線暗号 [4], [8] とは公開鍵暗号方式の一種で，現在最も普及している公開鍵暗号方式である RSA 暗号 [12] よりも高い安全性を有している．公開鍵暗号は安全性は鍵長に依存するが，楕円曲線暗号の場合，鍵長 160 ビットでの安全性が RSA 暗号の鍵長 1024 ビットの安全性と同等とされている．RSA 暗号よりも少ないデータ量で実装できるため，楕円曲線暗号はより高速，低面積，低消費電力な公開鍵暗号 LSI が実装できる．そのため RSA 暗号が実装できないリソースの限られた用途に適しているとされている．

楕円曲線暗号は有限体の演算で構成されており，主に素体 $GF(p)$ と 2 の拡大体 $GF(2^m)$ が使われる．$GF(2^m)$ は加算が XOR で計算できるため桁上がりがなく，乗算はシフト演算と XOR で計算できる．桁上がりによる遅延の増加がないため実装する際には $GF(2^m)$ が有利である．また楕円曲線暗号の演算量の 90%以上は乗算が占めているため，暗号処理の高速化には乗算器の改善が有効である．

楕円曲線暗号に適した $GF(2^m)$ 上の乗算器は複数提案されている．bit-serial 乗算アルゴリズム [2] は基本的な $GF(2^m)$ 上の乗算アルゴリズムであり，簡単に設計ができる．しかし，1 サイクルあたりの処理量が小さく高速処理が難しいため，楕円曲線暗号 LSI に適した乗算器ではない．楕円曲線暗号 LSI に適した digit-serial 乗算アルゴリズムは [14] がある．1 サイクルに複数ビットを処理することで処理量を増やしたアルゴリズムで，法 $f(z)$ を制限することで Reduction 演算を容易にし，XOR 加算をツリー構造で構成することで遅延の増加を抑えている．また，処理するビットサイズを変えることにより処理量と面積のトレードオフを選択することができるため，bit-serial 乗算器よりも拡張性が高い．また，1 サイクルあたりの処理量が大きいため，低い動作周波数でも高い処理性能を維持できる [11]．並列性を高め，遅延を減らすことで高速化する手法 [5] などが提案されている．[3] は Massey-Omura 乗算アルゴリズム [7] を用いている．しかし，サイクル数あたりの処理量が小さく，クロック周波数を上げにくい回路構造のため高速化が難しいという欠点がある．[13] は Montgomery 乗算アルゴリズム [9] を使用している．しかし，1 サイクルあたりの処理量が小さいため，高速処理するためには動作周波数を高くしなければならず，消費電力の面で不利である．また入力するデータサイズに対し面積が指数的に増えてしまうため 1 サイクルあたりの処理量を増やすには限界がある．

本稿では楕円曲線暗号に適した GF(2^m) 上の SIMD 型 MSD 乗算器を提案する．digit-serial 乗算器の一つである MSD 乗算器をデータ長に合わせて SIMD 演算で乗算を並列処理することにより，楕円曲線スカラー乗算を高速処理することができる．また，NIST が推奨する 5 つのデータ長について提案乗算器で処理することができるため，5 種類の MSD 乗算器を使用した場合に対し処理速度が同程度で比較した場合，面積を最大 1/3 まで削減することができる．また短い鍵長に対し SIMD 演算することで 2 つの乗算を同時に処理することができるため，従来の MSD 乗算器と比較し最大で約 2 倍の処理速度を得ることができる．MSD 乗算器は [10] で提案した手法を元にしている．

本稿は以下のように構成される．2 章では MSD アルゴリズムについて説明する．3 章では [10] で提案された MSB 乗算器をベースにした MSD 乗算器について述べる 4 章では SIMD 型 MSD 乗算器を提案し,, 面積と遅延の見積もりについて説明する．5 章では提案乗算器の論理合成結果について示す．

Input: $A = \sum_{i=0}^{m-1} a_i z^i, a_i \in GF(2)$
Input: $B = \sum_{i=0}^{m-1} b_i z^i, b_i \in GF(2)$
Output: $S = A \cdot B \bmod f(z)$
$f(z) = z^m + \sum_{i=0}^{m-1} c_i z^i,\ c_i \in GF(2)$

1: $S \leftarrow 0$
2: for $i = m-1$ to 0 do
  2.1: $S \leftarrow A \cdot B_i + S$
  2.2: $S \leftarrow S \cdot z \bmod f(z)$
3: Return(S)

図 1　MSB アルゴリズム.

Input: $A = \sum_{i=0}^{m-1} a_i z^i, a_i \in GF(2)$
Input: $B = \sum_{i=0}^{d-1} B_i z^{Di}$, where B_i is as in (1)
Output: $S = A \cdot B \bmod f(z)$
$f(z) = z^m + \sum_{i=0}^{m-1} c_i z^i$,where $c_i \in GF(2)$

1: $S \leftarrow 0$
2: for $i = \lceil m/D \rceil - 1$ to 0 do
  2.1: $S \leftarrow A \cdot B_i + S$
  2.2: $S \leftarrow T \cdot z^D \bmod f(z)$
3: Return(S)

図 2　MSD アルゴリズム.

2. digit-serial 乗算

GF(2^m) の要素 A と B とし，乗算 $S = A \cdot B \bmod f(z)$ を演算する方法は様々に提案されている．基本的な手法としては m ビット×1 ビットずつ処理する bit-serial アルゴリズム [2] がある．bit-serial アルゴリズムの一つである B の最上位ビットから処理する MSB 乗算アルゴリズムを図 1 に示す．

一方，digit-serial アルゴリズム [14] は複数のビット同時に処理する．同時に処理するデータ量 D を digit サイズと定義する．B を D で分割したとき，digit の数は $d = \lceil m/D \rceil$ となる．このとき $B = \sum_{i=0}^{d-1} B_i z^{Di}$ となり，B_i は式 (1) となる．

$$B_i = \sum_{j=0}^{D-1} b_{Di+j} z^j, 0 \leqq i \leqq d-1, b_i \in GF(2) \qquad (1)$$

従って，digit-serial アルゴリズムにおける最上位 Digit から処理する MSD アルゴリズムは図 2 のように表すことができる．

3. MSB 乗算器をベースとした MSD 乗算器

本章では，[10] で提案した MSB 乗算器をベースにした MSD 乗算器について述べる．MSB 乗算器を D 個を直列に接続することで MSD 乗算器を実現する．MSB 乗算器のように 1 ビットシフトするごとに Reduction 演算を行ない，その出力を次

の MSB 乗算器に伝えることを D だけ繰り返すことで処理を行う．そのためデータ長は D に依存せず常に m ビットにすることができ，面積の増加率を抑えることができる．従来の手法 [5], [14] では，乗算部と Redution 部が分かれており，どちらも D に依存するため提案手法よりも回路が複雑になる．また，乗算部と Reduction 部をレジスタで分離しているため 1 回の乗算に必要なクロックサイクル数が提案手法より多い．図 3 に提案 digit-serial 乗算アルゴリズムを示す．2.1 の for ループ部が MSB-first 型乗算器である．乗算，Reduction 演算を digit サイズ D ごと処理するのではなく 1 ビットごとに処理している．m=163，D=4 の時のブロック図を図 4 に示す．

クロックサイクル数と出力位置は digit サイズ D から計算する．各パラメータの求め方は以下の通りである．

- 1 回の乗算に必要なクロックサイクル数 $= \lceil m/D \rceil$
- 出力 $= m\%D$

m%n : m を D で除算したときの余り

3.1 リダクション演算

任意の法 $f(z)$ が扱える汎用性を高めた digit-serial 乗算器は D ビットのシフトを行うと D 回リダクション演算を行う必要があるため，単純に XOR 加算部をツリー構造にすることができない．そこで，[14] では法 $f(z)$ に対し，2 つの条件をつけることでシフト量に関わらずリダクション演算を一度で計算できるようにし，リダクション演算部をツリー構造に変換可能にしている．

[条件 1] 法 $f(z)$ は $p(z) = z^m + p_k z^k + \sum_{j=0}^{k-1} p_j z^j$ $(k < m)$ で表される．また，$t \leqq m-1-k$ としたとき z^{m+t} は次の式で表現される．

$$z^{m+t} \bmod p(z) = (z^m \bmod p(z)) \cdot z^t = p_k z^{k+t} + \left(\sum_{j=0}^{k-1} p_j z^{j+t} \right) \quad (2)$$

[条件 2] *digit* サイズ D は $D \leqq m-k$ を満たす．このとき式 *2* は *1* 度のリダクション演算で処理できる．

条件 1 は法 $f(z)$ の 2 番目に大きい項の次数が k であることを示す．条件 2 を満たす D を選ぶことで，D ビットシフトしても 1 回のリダクション演算で済むため，ツリー構造に変形可能な XOR によるリダクション演算部を構成することができる．digit サイズを大きくしても遅延の増加率を $O(\log(D))$ に抑えることができる．

反面，digit サイズの範囲は $D \leqq m-k$ になるため，k が m に近くなるほど D の範囲が狭くなる．法 $f(z)$ が任意である場合，$k = m-1$ となるので，この条件の下では 1 サイクルで乗算を行う bit-parallel 型しか構成できない．D の範囲を大きく取るためには $k \ll m$ でないとならないため，法 $f(z)$ の自由度は低い．

楕円曲線暗号を実装する場合，安全かつ実装に有利なパラメータを NIST が公表している（表 1）．NIST パラメータに従えば，D の範囲は十分であると言える．

3.2 面積と遅延の見積もり

提案手法の面積と遅延の見積もりについて述べる．乗算に必要な回路は AND ゲート部，乗算と Reduction 演算を行う XOR ゲート部，途中結果を保存する Accumulator 部で構成されている．AND ゲート部は m ビットの A と D ビットの B_i との AND を取るのでゲート数は mD である．同様に加算を行う XOR ゲートも mD となる．Reduction 演算を行う XOR ゲートは $f(x)$ の項数 k に依存し，kD となる．従って XOR ゲートは $(m+k)D$ になる．Accumulator 部は m ビットのレジスタである．表 2 に面積の見積もり値を示す．文献 [5], [14] の手法は乗算部の出力が $m+D$ となるためレジスタのサイズが大きくなる．

表 1 NIST 推奨パラメータと digit サイズ D の範囲．

$f(z)$	D
$f(z) = z^{163} + z^7 + z^6 + z^3 + 1$	$\leqq 156$
$f(z) = z^{233} + z^{74} + 1$	$\leqq 159$
$f(z) = z^{283} + z^{12} + z^7 + z^5 + 1$	$\leqq 271$
$f(z) = z^{409} + z^{87} + 1$	$\leqq 322$
$f(z) = z^{571} + z^{10} + z^5 + z^2 + 1$	$\leqq 561$

Input: $A = \sum_{i=0}^{m-1} a_i z^i, a_i \in GF(2)$
Input: $B = \sum_{i=0}^{d-1} B_i z^{Di}$, where B_i is as in (1)
Output: $S = A \cdot B \bmod f(z)$
$f(z) = z^m + \sum_{i=0}^{m-1} c_i z^i$,where $c_i \in GF(2)$

1: $S \leftarrow 0$
2: for $i = \lceil m/D \rceil - 1$ to 0 do
  2.1: for $j = D-1$ to 0 do
    2.1.1: $S \leftarrow A \cdot (B_i)_j + S$
    2.1.2: if($i == 0$ and $j == m\%D$)
      Return(S)
    2.1.3: $S \leftarrow S \cdot z \bmod f(z)$

図 3 MSD 乗算アルゴリズム．

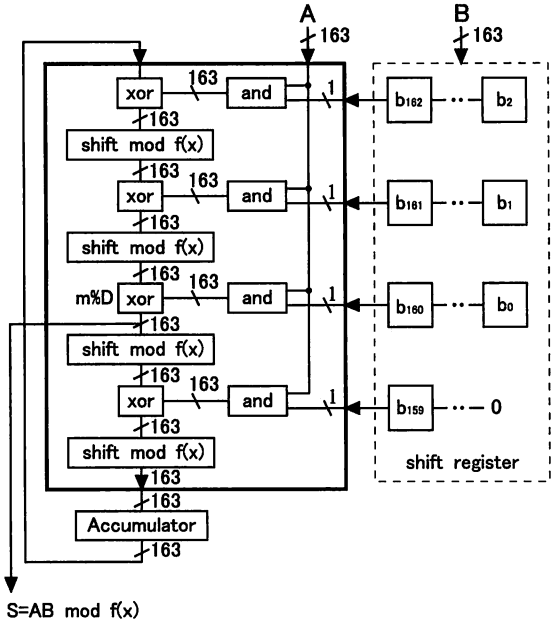

図 4 MSD 乗算器のブロック図 ($GF(2^{163}), D = 3$).

表 2　面積の見積もり.

	# XOR	# AND	# FF
文献 [5]	$(m+k-2)D+(k-1)(D-1)$	$(m+k-1)D+(k-1)(D-1)$	$3m+D-2$
文献 [14]	$(m+k+1)D+(k+1)(D-1)$	$(m+k)D+(k+1)(D-1)$	$2m+D+k$
提案手法	$(m+k)D$	mD	m

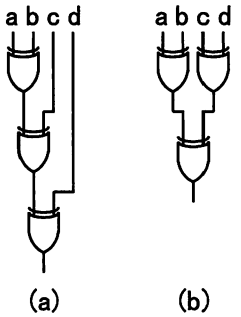

図 5　XOR ゲートの構造 (a) アレイ型 (b)2 分木型.

表 3　サイクル数と遅延の見積もり.

	サイクル数	遅延
文献 [5]	$\lceil m/D \rceil+1$	$1\Delta_{AND}+\lceil \log_2(\lceil D/2 \rceil+1) \rceil \Delta_{XOR}$
文献 [14]	$\lceil m/D \rceil+1$	$1\Delta_{AND}+\lceil \log_2(D+1) \rceil \Delta_{XOR}$
提案手法	$\lceil m/D \rceil$	$1\Delta_{AND}+\lceil \log_2(k(D-1)) \rceil \Delta_{XOR}$

遅延の見積もりは D に対する出力 $S_i (i \in \{0,1,\ldots,m-1\})$ を計算するのに必要な項 $\sum_{i=0}^{m-1}\sum_{j=0}^{m-1} a_i b_j$ の最大数から求める．$a_i b_j$ は AND で計算するが，依存関係はないので AND ゲート 1 つ分の遅延で済む．XOR ゲートは MSB 乗算器を並べただけでは遅延は digit サイズ D に比例して増加してしまうが，図 5 に示すように XOR のアレイ構造を 2 分木構造に再構成し，遅延を対数に抑えることができる．そのため XOR ゲートの遅延は出力までに加算される項数の対数で求めることができる．表 3 にサイクル数と遅延の見積もりを示す．k は法 $f(z)$ の項数である．NIST 推奨パラメータ [1] の場合，$k=2,4$ である．

4. GF(2^m) 上の SIMD 型 MSD 乗算器

本章では GF(2^m) 上の SIMD 型 MSD 乗算器を提案する．m ビット MSB 乗算器の場合，データを左詰にし，Reduction 演算をマルチプレクサを使って選択できるようにすれば $m \geq n$ を満たす n ビット以下の乗算を計算することができる．例えば表 1 に示す値のうち，$m=571$ ビットの MSB 乗算器なら $m \geq n$ を満たす $n=231,233,283,409$ を計算することができる．法 $f(z)$ はマルチプレクサを用いて選択する．[10] で述べているように MSB 乗算器は容易に MSD 乗算器を構成できるので MSD 乗算器も同様に $m=571$ ビットなら $n=163,233,283,409$ を計算することができる．

しかし，$m-n$ ビットには 0 ビットを代入するので，その分のデータは何も処理をしていない．そこで $m-n$ ビット分を有効に使うために SIMD 型にすることを提案する．すなわち $m-n \geq n$ を満たす場合，n を並列に乗算を行う．

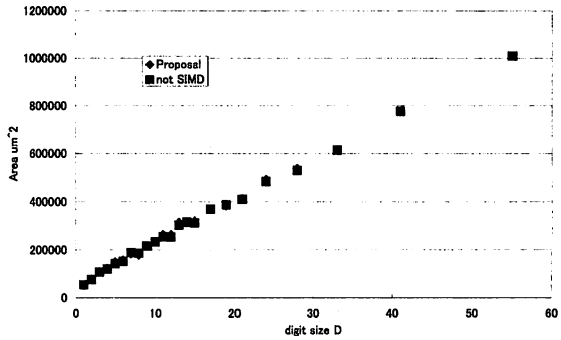

図 6　digit サイズ D と面積の関係.

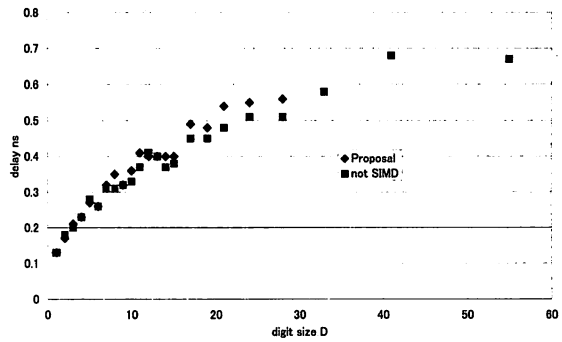

図 7　digit サイズ D と遅延の関係.

5. 論理合成結果

NIST 推奨パラメータ [1]（表 1）に基づいた 5 種類のパラメータに対応した SIMD 型 MSD 乗算器を設計した．$m=571$ ビットとし，571 ビット，409 ビットは 1 並列，163 ビット，233 ビット，283 ビットについて 2 並列の SIMD 乗算を行うよう設計した．STARC90nm プロセス用ライブラリを使用し，DesignComplier W-2004.12-SP2(注1)で合成した．提案手法と従来の SIMD 型ではない乗算器との比較を行った．D に対する面積のグラフを図 6，D に対する遅延のグラフを図 7 にそれぞれ示す．面積は SIMD 型にしてもほとんど変わりがないことが分かる．遅延は SIMD 型にしたほうが若干大きいことが分かる．

図 8 は $m=163$ の場合の処理時間で比較した図である．SIMD 型にしたほうは 2 並列で演算するため処理時間を半分にして見積もり値を出している．この図から $m=163$ において提案乗算器の方が処理時間が短くて済むという結果が得られた..

(注1)：本研究は東京大学大規模集積システム設計教育研究センターを通し、シノプシス株式会社の協力で行われたものである。

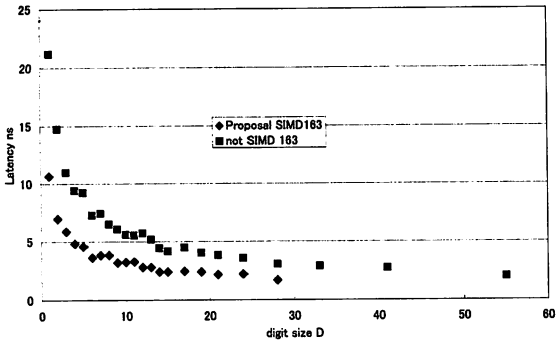

図 8　D と処理時間 (m=163) の関係.

6. むすび

本稿では，楕円曲線暗号に用いる GF(2^m) 上の SIMD 型 MSD 乗算器について提案した．MSD 乗算器を複数のパラメータで乗算を行えるよう設計し，SIMD 型にすることで $m = 163$ において乗算を高速に行えることを示した．今後の研究課題は，提案乗算器を用いた楕円曲線暗号ハードウェアを実装することである．

文　献

[1] *IEEE Standard Specifications for Public-Key Cryptography*, IEEE Std. 1363-2000.

[2] T. Beth and D. Gollmann, "Algorithm engineering for public key algorithms." *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 458–465, 1989.

[3] M. Ernst, S. Klupsch, O. Hauck and S. A. Huss, "Rapid prototyping for hardware accelerated elliptic curve publickeycryptosystems," *12th Int'l Workshop Rapid System Prototyping (RSP 2001)*, pp. 24–29, June 2001.

[4] N. Koblitz, "Elliptic Curve Cryptosystems," *Math. Computation*, vol. 48, pp. 203–209, 1987.

[5] S. Kumar, T. Wollinger and C. Paar, "Optimum Digit Serial $GF(2^m)$ Multipliers for Curve-Based Cryptography," *IEEE TRANSACTIONS ON COMPUTERS*, vol. 55, no. 10, pp. 1306–1311, Oct. 2006.

[6] J. Lopez and R. Dahab, "Fast multiplication on elliptic curves over $GF(2^m)$ without precomputation," *Cryptographic Hardware and Embedded Systems - CHES'99, Springer-Verlag, Lecture Notes in Computer Science 1717*, pp. 316–327, August, 1999.

[7] J. L. Massey and J. K. Omura, "Computational method and apparatus for finite field arithmetic," US Patent No. 4,587,627, to OMNET Assoc., Sunnyvale CA, Washington, D.C.: Patent and Trademark Office, 1986.

[8] V. Miller, "Uses of Elliptic Curves in Cryptography," *Advances in Cryptology, Proc. CRYPTO '85*, H.C. Williams, ed., pp. 417–426, 1986.

[9] P. L. Montgomery, "Modular multiplication without trial division," *Math. Computing*, vol. 44, no. 170, pp. 519–521, April 1985.

[10] 奈良 竜太, 小原 俊逸, 清水 一範, 戸川 望, 池永 剛, 柳澤 政生, 後藤 敏, 大附 辰夫, "楕円曲線暗号向け $GF(2^m)$ 上の Digit-Serial 乗算器の設計, " *VLSI* 設計技術研究会, 電子情報通信学会, VLD2006-85〜93, pp. 21–35, January 2007.

[11] G. Orlando and C. Paar, "A high-performance reconfigurable elliptic curve processor for $GF(2^m)$," *Cryptographic Hardware and Embedded Systems (CHES 2000)*, pp. 41–56, August 2000.

[12] R.L. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Comm. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[13] A. Satoh and K. Takano, "A Scalable Dual-Field Elliptic Curve Cryptographic Processor," *IEEE TRANSACTIONS ON COMPUTERS*, vol. 52, no. 4, pp. 449–460, April, 2003.

[14] L. Song and K.K. Parhi, "Low Energy Digit-Serial/Parallel Finite Field Multipliers," *J. VLSI Signal Processing*, vol. 19, no. 2, pp. 149–166, June. 1998.