Yi and Nie *EURASIP Journal on Wireless Communications and Networking* (2018) 2018:93
https://doi.org/10.1186/s13638-018-1117-2

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

High-speed hardware architecture for implementations of multivariate signature generations on FPGAs

Haibo Yi* and Zhe Nie

Abstract

Multivariate signature belongs to Multivariate-Quadratic-Equations Public Key Cryptography (MPKC), which is secure to quantum computer attacks. Compared with RSA and ECC, it is required to speed up multivariate signature implementations. A high-speed hardware architecture for signature generations of a multivariate scheme is proposed in this paper. The main computations of signature generations of multivariate schemes are additions, multiplications, inversions, and solving systems of linear equations (LSEs) in a finite field. Thus, we improve the finite field multiplications via using composite field expression and design a finite field inversion via using binary trees. Besides, we improve solving LSEs in a finite field based on a variant algorithm of Gauss-Jordan elimination and use the XOR gates to compute additions. We implement the high-speed hardware architecture based on the above improvements on an Altera Stratix Field-Programmable Gate Array (FPGA), which shows that it takes only 90 clock cycles and 0.9 μs to generate a multivariate signature. The comparison shows that the hardware architecture is much faster than other implementations.

Keywords: Cryptographic system, Multivariate-Quadratic-Equations Public Key Cryptography (MPKC), Multivariate signature, Field-Programmable Gate Array (FPGA)

1 Introduction

Quantum technology has developed rapidly in recent years. Quantum computer is in a position to attack RSA [1], ECC [2], and other signature algorithms adopted by many chips due to the algorithm by Peter Shor [3]. Therefore, chip security is facing severe threats.

Fortunately, there are a few post-quantum candidates for signature chips, in which multivariate signature is included [4]. Multivariate signature belongs to Multivariate Quadratic Equations Public Key Cryptography (MPKC), which is secure to quantum computer attacks and general computer attacks [5, 6]. MPKC is first proposed by Matsumoto and Imai in the 1980s. During the past 30 years, various schemes of MPKC have been proposed [7–32], which includes Rainbow [28], Unbalanced Oil-Vinegar (UOV) [29], and Tame Transformation Signature (TTS) [30, 31]. Software and hardware implementations of multivariate signature schemes have been one of the topics of many researchers [33–40]. enTTS belongs to the triangular family, which can be viewed as extensions of Tame Transformation Method (TTM) Among the existing enTTS schemes, enTTS(20,28) is believed to be one of the fastest signature schemes, which works with 20 B hashes and 28 B signatures. Compared with the implementations of other public key cryptosystems, e.g., RSA and ECC, we need to speed up multivariate signature generations.

Some previous works of efficient implementations of multivariate signature schemes are as follows:

The work in [33] proposed a fast implementation of SFlash;

The work in [34] presented an efficient public key generation for multivariate cryptosystems;

In [35], a minimized PKC of multivariate schemes on low-resource embedded systems was proposed;

In [36], an efficient implementation of multivariate quadratic systems was presented;

A fast implementation of Rainbow signature generation was proposed in [37];

* Correspondence: haiboyi@126.com
School of Computer Engineering, Shenzhen Polytechnic, Shenzhen, China

A high-speed hardware architecture based on Rainbow signature on Field-Programmable Gate Arrays (FPGAs) was proposed in [38];

For low area design, a multivariate signature FPGA processor was proposed in [39];

The work of [40] was a time-area optimized design, which showed that multivariate cryptosystems are more efficient than ECC.

Among such multivariate signature schemes, enTTS is a Tame-like multivariate public key cryptosystem [32]. Hardware implementations of TTS (including enTTS) signature are mainly proposed in [31, 35, 36, 39, 40]. Most of these implementations are focusing on area optimizations. The main computations during generations of enTTS signature are addition, multiplication, inversion, and solving Systems of Linear Equations (LSEs) in a finite field.

Thus, the main contributions of this paper are as follows. We improve the finite field multiplications via using composite field expression and design a finite field inversion based on binary trees described in [41]. Besides, we improve solving LSEs in a finite field based on the algorithm of Gauss-Jordan elimination and use the XOR gates to compute additions.

We implement the high-speed hardware architecture based on the above improvements on an Altera Stratix FPGA. The comparison shows that the hardware architecture is much faster than other implementations of public key cryptosystems.

We organize the rest of this paper as follows: the algorithm of multivariate scheme is introduced in Section 2; the high-speed hardware architecture for multivariate signature generations is given in Section 3; implementation results of the high-speed hardware architecture on FPGAs and comparisons with related cryptosystems are given in Section 4; Conclusions are summarized in Section 5.

2 Method

This study originates from a need to speed up signature generations of multivariate scheme, since the efficiency of implementations should be improved as a quantum-resistance cryptosystems. Specifically, we propose a high-speed hardware architecture for multivariate scheme through improving finite field multiplications based on composite field expression, finite field inversions based on binary trees and solving LSEs based on the algorithm of Gauss-Jordan elimination.

We implement the high-speed hardware architecture based on the above improvements on an Altera Stratix FPGA and the comparison shows that the hardware architecture is much faster than other implementations.

The multivariate scheme, enTTS is employed to the architecture for hardware implementations of signature generations in a finite field. enTTS belongs to the triangular family, which can be viewed as extensions of Tame Transformation Method (TTM). enTTS is designed with a higher security level than TTS. We illustrate enTTS parameters in Table 1.

We use GF($(2^4)^2$) for implementation of enTTS, which is a composite field of GF(256). We suppose that y denotes the message (20 B) of multivariate scheme and y_0, y_1, …, y_{19} denote each byte from the message, where y_0, y_1, …, y_{19} are elements in $GF((2^4)^2)$. We suppose that x denotes the signature (28 B) and x_0, x_1, …, x_{27} denote each byte of the signature, where x_0, x_1, …, x_{27} are elements in GF($(2^4)^2$).

The construction of this signature scheme uses affine transformation L_1, central map transformation F, and affine transformation L_2.

In order to sign a message, i.e., $y(y_0, y_1, \ldots, y_{19})$, it is required to compute several steps for the following equation:

$$F^{\circ}L_2(x_0, x_1, \ldots, x_{27}) = L_1^{-1}(y_0, y_1, \ldots, y_{19}).$$

First, the following equation is required to solve:

$$\overline{y}(\overline{y}_0, \overline{y}_1, \ldots, \overline{y}_{19}) = L_1^{-1}(y_0, y_1, \ldots, y_{19}).$$

L_1^{-1} is an invertible affine transformation with the following form:

$$\overline{y} = Ay + B.$$

A is a matrix with the size of 20×20, part of private keys of enTTS;

B is a vector with the size of 20, part of private keys of enTTS.

Then, $\overline{y}(\overline{y}_0, \overline{y}_1, \ldots, \overline{y}_{19})$ is the result of affine transformation L_1, where $\overline{y}_0, \overline{y}_1, \ldots, \overline{y}_{19}$ are elements in $GF((2^4)^2)$.

Second, the following equation is required to solve:

$$\overline{x}(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27}) = F^{-1}(\overline{y}_0, \overline{y}_1, \ldots, \overline{y}_{19}).$$

The construction of central map transformation depends on a map with the following representation.

Table 1 Multivariate scheme parameters

Signature scheme	Finite field	Message	Signature	Central map transformation	L_1 transformation	L_2 transformation
enTTS (20,28)	GF($(2^4)^2$)	$y_0, y_1, \ldots, y_{19}$	$x_0, x_1, \ldots, x_{27}$	$F(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27}) = (f_0, f_1, \ldots, f_{19})$.	$\overline{y} = Ay + B$	$x = C\overline{x} + D$

$$F(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27}) = (f_0, f_1, \ldots, f_{19}).$$

We suppose that $x(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27})$ denote the result of central map transformation *F*, where $\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27}$ are elements in $GF((2^4)^2)$. MQ polynomials $f(f_0, f_1, \ldots, f_{19})$ are defined by the following equations:

$$f_{i-8} = \overline{x}_i + \sum\nolimits_{j=1}^{7} p_{ij}\overline{x}_j\overline{x}_{8+((i+j) \mod 9)}, i = 8, 9, \ldots, 16,$$

$$\begin{aligned} f_9 = \overline{x}_{17} &+ p_{17,1}\overline{x}_1\overline{x}_6 + p_{17,2}\overline{x}_2\overline{x}_5 + p_{17,3}\overline{x}_3\overline{x}_4 \\ &+ p_{17,4}\overline{x}_9\overline{x}_{16} + p_{17,5}\overline{x}_{10}\overline{x}_{15} + p_{17,6}\overline{x}_{11}\overline{x}_{14} \\ &+ p_{17,7}\overline{x}_{12}\overline{x}_{13}, \end{aligned}$$

$$\begin{aligned} f_{10} = \overline{x}_{18} &+ p_{18,1}\overline{x}_2\overline{x}_7 + p_{18,2}\overline{x}_3\overline{x}_6 + p_{18,3}\overline{x}_4\overline{x}_5 \\ &+ p_{18,4}\overline{x}_{10}\overline{x}_{17} + p_{18,5}\overline{x}_{11}\overline{x}_{16} + p_{18,6}\overline{x}_{12}\overline{x}_{15} \\ &+ p_{18,7}\overline{x}_{13}\overline{x}_{14}, \end{aligned}$$

$$\begin{aligned} f_{i-8} = \overline{x}_i + p_{i,0}\overline{x}_{i-11}\overline{x}_{i-9} &+ \sum\nolimits_{j=19}^{i} p_{i,j-18}\overline{x}_{2(i-j)}\overline{x}_j \\ &+ \sum\nolimits_{j=i+1}^{27} p_{i,j-18}\overline{x}_{i-j+19}\overline{x}_j, i = 19, 20, \ldots, 27, \end{aligned}$$

p_{ij} are coefficients, part of private key.

The MQ polynomials

$$\overline{y}(\overline{y}_0, \overline{y}_1, \ldots, \overline{y}_{19}) = f(f_0, f_2, \ldots, f_{19})$$

can be divided into three groups:

$$\begin{aligned} &f_i \mid i = 0, 1, \ldots, 8 \\ &f_i \mid i = 9, 10 \\ &f_i \mid i = 11, 12, \ldots, 19. \end{aligned}$$

Similarly, $\overline{x}(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27})$ are divided into four groups:

$$\begin{aligned} &\overline{x}_i \mid i = 0, 1, \ldots, 7 \\ &\overline{x}_i \mid i = 8, 9, \ldots, 16 \\ &\overline{x}_i \mid i = 17, 18 \\ &\overline{x}_i \mid i = 19, 20, \ldots, 27. \end{aligned}$$

The first group variables of $\overline{x}$, i.e., $\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_7$ are randomly chosen and then the first group polynomials of f_i, i.e., $f_0, f_1, \ldots, f_8$ are evaluated.

Then, the second group variables of $\overline{x}$ are $\overline{x}_8, \overline{x}_9, \ldots, \overline{x}_{16}$, and we solve the LSEs on such variables.

Next, we evaluate the second group polynomials of f_i, i.e., f_9, f_{10} and solve the third group variables of $\overline{x}$, i.e., $\overline{x}_{17}, \overline{x}_{18}$.

Then, the third group polynomials of f_i, i.e., $f_{11}, f_{12}, \ldots, f_{19}$ are evaluated and we solve the LSEs on such variables of the fourth group variables of $\overline{x}$, i.e., $\overline{x}_{19}, \overline{x}_{20}, \ldots, \overline{x}_{27}$.

After that, the result of central map transformation *F*, i.e., $x(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27})$ is computed.

Last, we solve the following equations based on the values of $\overline{x}_{19}, \overline{x}_{20}, \ldots, \overline{x}_{27}$.

$$x(x_0, x_1, \ldots, x_{27}) = L_2{}^{-1}(\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_{27}).$$

L_2^{-1} is an invertible affine transformation

$$x = C\overline{x} + D.$$

C is a matrix with the size of 28 × 28, part of private keys of enTTS;

D is a vector with the size of 28, part of private keys of enTTS.

Finally, we have computed the signature of $y(y_0, y_1, \ldots, y_{19})$, which is $x(x_0, x_1, \ldots, x_{27})$.

3 A high-speed hardware architecture for multivariate signature

3.1 Overview of the hardware architecture

We choose enTTS (20,28) scheme described in Section 2 for hardware implementations in a composite field $GF((2^4)^2)$, where the size of message (hash value) is 20 B and the signature size is 28 B.

We illustrate the generation of a multivariate signature in Fig. 1. It can be observed from Fig. 1 that the signature generations of multivariate scheme include seven steps:

(1) Affine transformation L_1.

L_1^{-1} is an invertible affine transformation with the following form.

$$\overline{y} = Ay + B.$$

A is a matrix with the size of 20 × 20.

B is a vector with the size of 20.

It can be observed that L_1^{-1} is performed via matrix-vector multiplications and vector additions, where *A* and *B* are parts of private keys.

(2) Polynomial evaluation (first part *F*)

First, we randomly choose the variables of $\overline{x}_0, \overline{x}_1, \ldots, \overline{x}_7$, i.e., the first group variables of $\overline{x}$.

Second, we evaluate the polynomials of $f_0, f_1, \ldots, f_8$, i.e., the first group polynomials of f_i.

After that, this part of polynomial evaluation is performed via using additions and multiplications in a finite field.

(3) Solving (LSEs) in a finite field

During the signature generations of multivariate scheme, it is required to perform solving LSEs twice with the same matrix of size 9 × 9.

First, for the second group variables of $\overline{x}$, i.e., $\overline{x}_8, \overline{x}_9, \ldots, \overline{x}_{16}$, we solve the LSEs on such variables.

Second, for the fourth group variables of $\overline{x}$, i.e., $\overline{x}_{19}, \overline{x}_{20}$ $, \ldots, \overline{x}_{27}$, we solve the LSEs on such variables.

During this step, solving LSEs is performed via using a variant Gauss-Jordan elimination in a finite field.

(4) Polynomial evaluation (second part *F*)

The third group variables of $\overline{x}$, i.e., $\overline{x}_{17}, \overline{x}_{18}$ are solved by evaluating the second group polynomials of f_i, i.e., f_9, f_{10}.

This part of polynomial evaluation is performed via using additions and multiplications in a finite field;

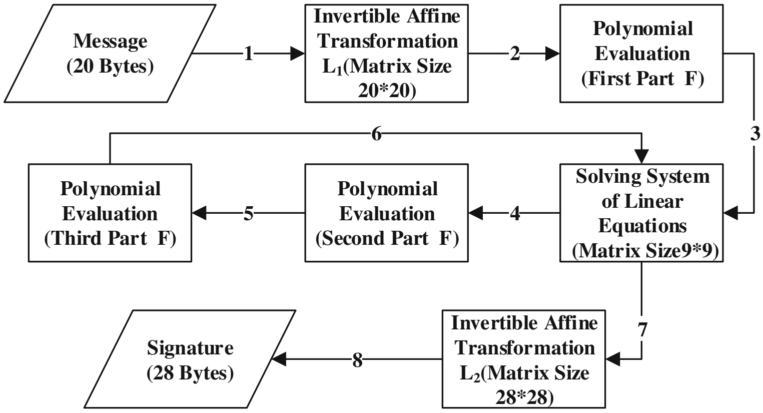

Fig. 1 The generation process of multivariate signatures

(5) Polynomial evaluation (third part F)

We evaluate the third group polynomials of f_i, i.e., f_{11}, f_{12}, …, f_{19}.

This part of polynomial evaluation is performed via using additions and multiplications in a finite field.

(6) Affine transformation L_2: L_2^{-1} is an affine transformation with the following form:

$$x = C\overline{x} + D.$$

C: is a matrix with the size of 28 × 28.

D is a vector with the size of 28.

It can be observed that L_2^{-1} is performed via matrix-vector multiplications and vector additions, where C and D are parts of private keys.

Our hardware architecture for the signature generation of multivariate scheme is depicted in Fig. 2. It can be observed from Fig. 2 that the hardware architecture consists of adders, multipliers, inverter, parallel Gauss-Jordan eliminator, polynomial evaluation, matrix vector multiplication, vector addition, polynomial evaluation, and processor components in a finite field, where only the first four components are computing components and the others are logical components.

3.2 Performance evaluation of irreducible polynomial in composite fields

Irreducible polynomials in composite fields are involved in the additions, multiplications, and other operations during signature generations. Thus, the performance evaluation of the irreducible polynomial in the composite field $\mathrm{GF}((2^4)^2)$ is very critical for the implementation of high-speed hardware architecture of multivariate scheme.

We suppose that $q(x)$ denotes the irreducible polynomial in $\mathrm{GF}((2^4)^2)_{,}$ and it has the following form.

$$q(x) = x^2 + q_1x + q_0.$$

p_1, p_0 are elements in $\mathrm{GF}(2^4)$.

We suppose that $p(x)$ denotes the irreducible polynomial in the subfield of $\mathrm{GF}((2^4)^2)$, i.e., $\mathrm{GF}(2^4)$, and it has the following form:

$$p(x) = x^4 + p_3x^3 + p_2x^2 + p_1x + 1.$$

p_3, p_2, p_1 are bits, i.e., 0 or 1.

The performance of the multiplications and inversions has been evaluated based on such irreducible polynomials, respectively. $q(x) = x^2 + x + 9$ is chosen as the irreducible polynomials in $\mathrm{GF}((2^4)^2)$ and $p(x) = x^4 + x + 1$ is chosen as the irreducible polynomials in the subfield $\mathrm{GF}(2^4)$.

3.3 Finite field adder

Let $a(x) = a_hx + a_l$ and $b(x) = b_hx + b_l$ be the elements in $\mathrm{GF}((2^4)^2)$, where a_h, a_l, b_h, and b_l are elements in $\mathrm{GF}(2^4)$.

Then the addition of $a(x)$ and $b(x)$ can be expressed as

$$a(x) + b(x) =$$
$$(a_hx + a_l) + (b_hx + b_l) =$$
$$(a_h + b_h)x + a_l + b_l.$$

Then, we suppose that c_h, c_l are elements in $\mathrm{GF}(2^4)$, and we can compute their values via the following expressions:

$$c_h = a_h + b_h,$$
$$c_l = a_l + b_l.$$

Thus, $c(x) = c_hx + c_l$ is the addition result of $a(x)$ and $b(x)$.

3.4 Finite field multiplier

Let $a(x) = a_hx + a_l$ and $b(x) = b_hx + b_l$ be the elements in $\mathrm{GF}((2^4)^2)$, where a_h, a_l, b_h, and b_l are elements in $\mathrm{GF}(2^4)$.

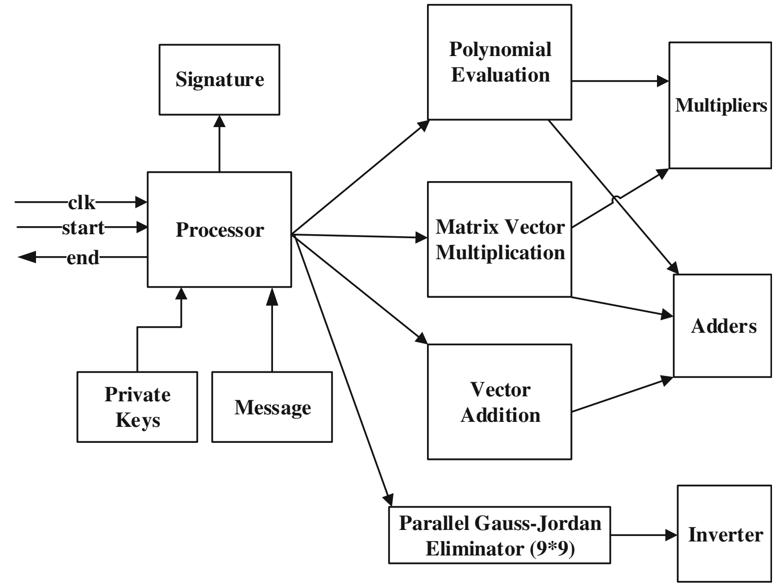

Fig. 2 Hardware architecture of multivariate schemes

Then the multiplication of $a(x)$ and $b(x)$ can be expressed as

$$a(x) \times b(x) =$$

$$(a_h x + a_l)(b_h x + b_l) =$$

$$\left(a_h b_h x^2 + (a_h b_l + a_l b_h)x + a_l b_l\right) \bmod q(x).$$

We perform the polynomial multiplication and reduction module the irreducible polynomial $q(x) = x^2 + x + 9$. We suppose that c_h and c_l are elements in GF(2^4), and we can compute their values via the following expressions:

$$c_h = (a_h + a_l)(b_h + b_l) + a_l b_l,$$

$$c_l = a_l b_l + 9 a_h b_h.$$

It can be observed that the critical path of multiplication of two elements in GF($(2^4)^2$) includes one multiplication, one constant multiplication, and one addition in GF(2^4).

$p(x)$ is the irreducible polynomial in GF(2^4). Let $a(x) = \sum_{i=0}^{3} a_i x^i$ and $b(x) = \sum_{i=0}^{3} b_i x^i$ be elements in $GF(2^4)$, a_i, $b_i \in GF(2)$, and we suppose that

$$c(x) = a(x) \times b(x)(\bmod (p(x))) = \sum_{i=0}^{3} c_i x^i$$

is the multiplication result of two elements, where $c_i \in$ GF(2).

First, we compute v_{ij} for $i = 0, 1, \ldots, 6$and $j = 0, 1, 2, 3$ according to the following equation:

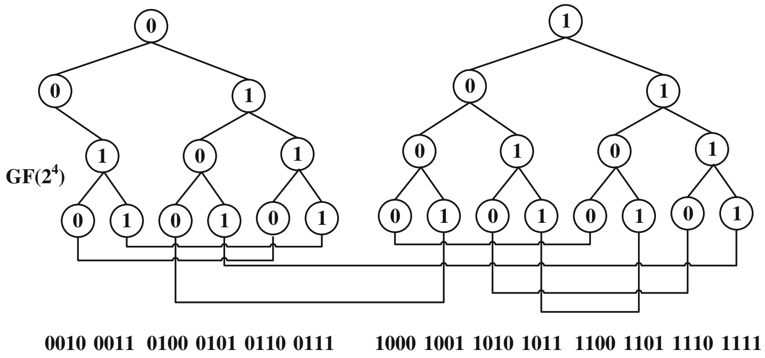

Fig. 3 Multiplicative inversion based on binary trees

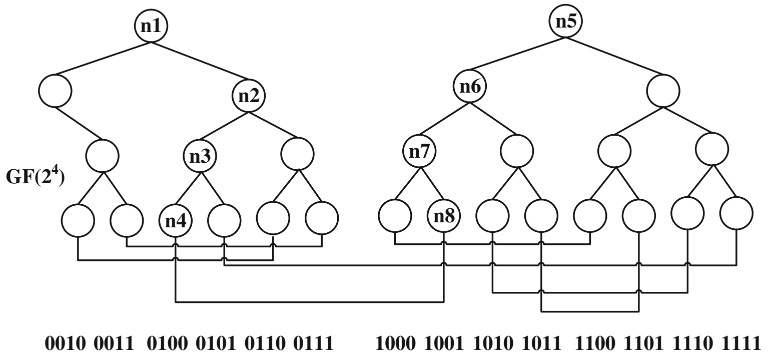

Fig. 4 An example of inversion

$$x^i \mod p(x) = \sum_{j=0}^{3} v_{ij} x^j.$$

Next, we compute $S_i$for $i = 0, 1, \ldots, 6$ by the following equation:

$$S_i = \sum_{j+k=i} a_j b_k.$$

After that, we compute c_i for $i = 0, 1, 2, 3$ by the following equation:

$$c_i = \sum_{j=0}^{6} v_{ji} S_j.$$

Finally, the multiplication result is

$$c(x) = \sum_{i=0}^{3} c_i x^i.$$

3.5 Multiplicative inverter

Let $a(x) = a_h x + a_l$and $b(x) = b_h x + b_l$be the elements in GF($(2^4)^2$), where a_h, a_l, b_h, and $b_l$are elements in GF(2^4).

We suppose that $b(x)$ is the inverse of $a(x)$. Then,

$$b_h = (a_h + a_l)^{-1} a_h b_l,$$
$$b_l = \left(a_l + 9a_h^2 (a_l + a_h)^{-1}\right)^{-1}.$$

We use two binary trees for inversions in subfield GF(2^4), which are illustrated as follows:

Each binary tree has four layers in GF(2^4);

Root nodes are on the third layer;

Each node has at most two child nodes, left node represents value of zero and right node represents value of one;

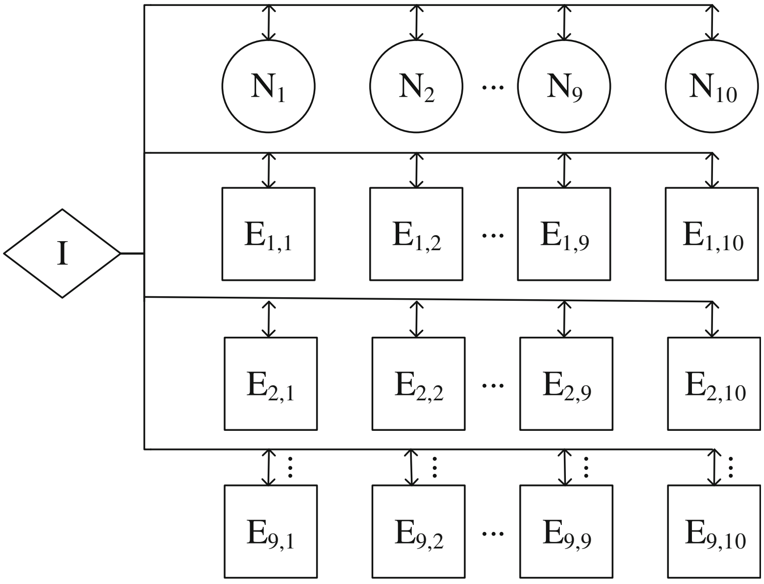

Fig. 5 Parallel Gauss-Jordan eliminator with matrix size 9 × 9

Table 2 FPGA implementation of hardware architecture for multivariate signature generation

Signature scheme	Message size	Signature size	L_1 matrix	L_2 matrix	Systems of linear equations	Clock cycle	Time frequency	Executing time
enTTS(20,28)	20 B	28 B	20 × 20	28 × 28	9 × 9	90	100 MHz	0.9 μs

Each child must either be a leaf or the root of another tree, each node has a father node when it is not a root node;

Each element in a finite field (except $(0000)_2$ and $(0001)_2$) has a unique traversal from root to leaf due to the fact that $(0000)_2$ has no inverse and the inverse of $(0001)_2$ is itself;

Each leaf is linked to another leaf.

Figure 3 depicts the architecture for inversions in $GF(2^4)$.

Example 1. It can be observed from Fig. 4, if it is required to inverse the element $(0100)_2$, we search the binary tree from root nodes to leaf nodes, the path from $n1$ to $n4$ represents $(0100)_2$. $n4$ is linked with $n8$, thus the path from $n5$ to $n8$ represents the inverse of $(0100)_2$, i.e., $(1001)_2$.

3.6 Parallel Gauss-Jordan eliminator

During central map transformation in signature generations, it is required to solve LSEs in a finite field twice with the same matrix size 9 × 9.

We adopt a parallel Gauss-Jordan elimination, which is depicted in Fig. 5. It can solve a LSE with matrix size of 9 × 9. The parallel Gauss-Jordan eliminator solves systems of linear equations with 9 iterations, which is enhanced in the following directions:

First, exclusive adders are used in the parallel Gauss-Jordan elimination based on the design described in Section 3.3;

Second, exclusive multipliers are used in the parallel Gauss-Jordan elimination based on the design described in Section 3.4;

Third, exclusive inverters are used in the parallel Gauss-Jordan elimination based on the design described in Section 3.5.

It can be observed from Fig. 4, I, N_l, and E_{kl} are three kinds of cells in the architecture, where $k = 1, 2, \ldots, 9$ and $l = 1, 2, \ldots, 10$.

The I cell is used for multiplicative inversion in a finite field, which includes an exclusive inverter described in Section 3.5.

The N_l cells are used for normalization of finite field elements, which includes exclusive multipliers described in Section 3.4.

The E_{kl} cells are used for elimination of finite field elements, which includes exclusive adders and multipliers described in Sections 3.3 and 3.4.

In conclusion, the architecture includes one I cell, 9 N_l cells, and 90 E_{lk} cells and solves the LSEs within 9 clock cycles with the matrix size of 9 × 9.

4 Results

In this section, we investigate the performance of the high-speed hardware architecture for multivariate scheme through hardware implementations on an Altera Stratix FPGA. The implementation is programmed in the hardware programming language, Verilog.

The implementation of multivariate signature generations is illustrated in Table 2, where the executing time for a signature generation of enTTS is 0.9 μs, the time frequency is 100 MHz, and the clock cycle is 90. It should be noted that, all of the results from implementations mentioned are extracted after place and route on the Altera Stratix FPGA.

We compare the high-speed hardware architecture with the related implementations of multivariate schemes and other public key schemes, which is depicted in Table 3. The ECC cryptosystems proposed in [2] are efficient implementations and Rainbow cryptosystems proposed in [38] are fast implementations.

It can be observed from Table 3 that the high-speed hardware architecture is much faster than the related implementations of public key cryptosystems.

5 Conclusions

We propose a high-speed cryptographic architecture for hardware implementation of multivariate signature generations in this paper. The main computations of signature generations of multivariate scheme are multiplications, inversions, and solving LSEs in a finite field. Thus, we improve the finite field multiplications via using composite field expression and design a finite field inversion via using binary trees. Besides, we improve solving LSEs in a finite field based on the variant algorithm of Gauss-Jordan elimination.

Table 3 Comparison on public key cryptographic systems

Scheme	Executing Time (μs)	Clock cycle
ECC [2]	41	4100
Rainbow [38]	7.9	198
UOV [40]	5.85	1170
amTTS [40]	2.438	195
enTTS [40]	2.025	162
enTTS (this paper)	0.9	90

We implement the high-speed hardware architecture based on the above improvements on an Altera Stratix FPGA device. The implementation results show that the executing time for a signature generation of multivariate scheme is 0.9 μs, the time frequency is 100 MHz, and the clock cycle is 90. The comparison shows that the hardware architecture is much faster than other implementations.

Abbreviations

ECC: Elliptic Curve Crypto; enTTS: Enhanced Tame Transformation Signature; FPGA: Field-Programmable Gate Array; LSE: Systems of Linear Equations; MPKC: Multivariate Quadratic Equations Public Key Cryptography; MQ: Multivariate Quadratic; PKC: Public Key Cryptography; RSA: Rivest-Shamir-Adleman; TTS: Tame Transformation Signature; UOV: Unbalanced Oil-Vinegar

Acknowledgements

The authors would like to express our cordial thanks to the section editor in charge and the respected reviewers for their time, accurate review of our manuscript, and their invaluable comments on this paper.

Funding

This work is supported by Shenzhen Science and Technology Program under Grant (nos. JCYJ20170306144219159 and JCYJ20160428092427867), Foundation for Distinguished Young Talents in Higher Education of Guangdong, China (no. 2017GkQNCX059), Special Fund for the Development of Strategic Emerging Industries and Future Industries of Shenzhen (no. 20170502142224600), Science and Technology Program of Shenzhen Polytechnic (no. 601722 K20018).

Authors' contributions

HY is the main writer of this paper. He proposed the main idea, designed and implemented the architecture, and drafted the manuscript. ZN gave some important suggestions for the architecture and analyzed the results. Both authors read and approved the final manuscript.

Authors' information

Haibo Yi received the bachelor degree in computer science from Beijing Jiaotong University, China, in 2009 and the PhD from South China University of Technology, China in 2015. Since 2015, he has been with School of Computer Engineering of Shenzhen Polytechnic, as a lecturer. He has published over 20 technical papers. His main research areas are information security, cloud computing, and big data. He is a member of Chinese Association for Cryptologic Research.

Zhe Nie received the B.S. degree in industrial automation from the North China University of Technology, China and the M.S. degree in Computer application technology from Harbin Institute of Technology, China. He joined Shenzhen Polytechnic in 1994 and currently is a professor. His research interests include internet public opinion, artificial intelligence, and computer vision technology.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Received: 29 January 2018 Accepted: 18 April 2018
Published online: 02 May 2018

References

1. M Shand, J Vuillemin, in *Fast implementations of RSA cryptography,* Proceedings of IEEE 11th Symposium on Computer Arithmetic, Windsor, Ont., (IEEE, USA, 1993) pp. 252-259
2. B Ansari, MA Hasan, High-performance architecture of elliptic curve scalar multiplication. IEEE Trans. Comput. **57**(11), 1443–1453 (2008)
3. PW Shor, in *Quantum Entanglement and Quantum Information-Proceedings of Ccast*. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer (1999), pp. 303–332
4. D Bernstein, J Buchmann, E Dahmen, *Post-quantum cryptography* (Springer, Berlin Heidelberg, 2009)
5. J Ding, BY Yang, in *Post-Quantum Cryptography*. Multivariate public key cryptography (2009), pp. 193–241
6. Garey M. R., Johnson D. S. Computers and Intractability: A Guide to the Theory of NP-Completeness. 1986.
7. T Matsumoto, H Imai, in *The Workshop on Advances in Cryptology-Eurocrypt. DBLP*. Public quadratic polynomial-tuples for efficient signature-verification and message-encryption (1988), pp. 419–453
8. J Patarin, in *Advancews in Cryptology - CRYPTO '96, International Cryptology Conference, Santa Barbara, California, USA, August 18–22, 1996, Proceedings*. Asymmetric cryptography with a hidden monomial (Springer, Berlin, 1996), pp. 45–60
9. J Patarin, in *International Conference on the Theory and Applications of Cryptographic Techniques*. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms (Springer, Berlin, Heidelberg, 1996), pp. 33–48
10. Y Tan, S Tang, J Chen, et al., Building a new secure variant of rainbow signature scheme[J]. IET Inf. Secur. **10**(2), 53–59 (2016)
11. Y Tan, S Tang, T Wang, Adding variables variation to rainbow - like scheme to enhance its security level against MinRank attack. Secur. Commun. Netw. **7**(12), 2326–2334 (2015)
12. Patarin J., Courtois N., Goubin L. FLASH, a fast multivariate signature algorithm. **2020**, 298–307 (2001)
13. J Ding, A Petzoldt, Current state of multivariate cryptography. IEEE Secur. Privacy **15**(4), 28–36 (2017)
14. Z Peng, S Tang, Circulant rainbow: A new rainbow variant with shorter private key and faster signature generation. IEEE Access **PP**(99), 1 (2017)
15. J Ding, JE Gower, Inoculating multivariate schemes against differential attacks[J]. Lect. Notes Comput. Sci **2006**, 290–301 (2005)
16. A Diene, J Ding, JE Gower, et al., in *Coding and Cryptography, International Workshop, WCC 2005, Bergen, Norway, March 14–18, 2005. Revised Selected Papers*. Dimension of the linearization equations of the Matsumoto-Imai cryptosystems (DBLP), (Springer, Berlin, 2008) pp. 242–251
17. J Ding, C Wolf, BY Yang, in *International Conference on Practice and Theory in Public-Key Cryptography*. Invertible cycles for multivariate quadratic (MQ) public key cryptography (Springer-Verlag, Berlin, 2007), pp. 266–281
18. S Tsujii, T Itoh, A Fujioka, et al., Public-key cryptosystem based on the difficulty of solving a system of nonlinear equations. Syst. Comput. Japan **19**(2), 10–18 (2010)
19. A Shamir, in *Advances in Cryptology -CRYPTO' 93*. Efficient signature schemes based on birational permutations (Springer, Berlin Heidelberg, 1993), pp. 1–12
20. T Moh, A public key system with signature and master key functions. **27**(5), 2207–2222 (1999)
21. M Kasahara, R Sakai, A construction of public key cryptosystem for realizing Ciphertext of size 100 bit and digital signature scheme (asymmetric cipher) (cryptography and information security). IEICE Trans. Fundamentals Electron. Commun. Comput. Sci. **87-A**(1), 102–109 (2004)
22. LC Wang, FH Chang, Tractable rational map cryptosystem. Iacr Cryptology Eprint Archive (2005), http://eprint.iacr.org/2004/046.pdf
23. M Kasahara, A construction of public-key cryptosystem based on singular simultaneous equations. IEICE Trans. Fundamentals Electron. Commun. Comput. Sci. **88-A**(1), 74–80 (2005)
24. Wolf C., Preneel B. Large superfluous keys in multivariate quadratic asymmetric systems. International Conference on Theory and Practice in Public Key Cryptography. (Springer, Berlin, 2005) pp. 275–287
25. LC Wang, YH Hu, F Lai, et al., in *Public Key Cryptography - PKC 2005, International Workshop on Theory and Practice in Public Key Cryptography, Les Diablerets, Switzerland, January 23–26, 2005, Proceedings*. Tractable rational map signature (DBLP) (Springer, Berlin, 2005), pp. 244–257
26. T Moh, Two new examples of TTM. Iacr Cryptology Eprint Arch. (2007) http://eprint.iacr.org/2007/144
27. J Baena, C Clough, J Ding, in *International Workshop on Post-Quantum Cryptography*. Square-Vinegar Signature Scheme (Springer-Verlag, 2008), pp. 17–30
28. J Ding, D Schmidt, in *Applied Cryptography and Network Security, Third International Conference, ACNS 2005, New York, NY, USA, June 7–10, 2005, Proceedings*. Rainbow, a new multivariable polynomial signature scheme (DBLP) (Springer, Berlin, 2005), pp. 164–175

29. A Kipnis, J Patarin, L Goubin, Unbalanced oil and vinegar signature schemes. Adv. Cryptology Eurocrypt **1592**, 206–222 (1999)
30. JM Chen, BY Yang, in *International Conference on Information Security and Cryptology*. A more secure and efficacious TTS signature scheme (2004), pp. 320–338
31. BY Yang, JM Chen, YH Chen, in *Cryptographic Hardware and Embedded Systems - CHES 2004:, International Workshop Cambridge, Ma, USA, August 11–13, 2004. Proceedings*. TTS: High-speed signatures on a low-cost smart card (DBLP) (Springer, Berlin, 2004), pp. 371–385
32. BY Yang, JM Chen, in *Australasian Conference on Information Security and Privacy*. Building secure Tame-like multivariate public-key cryptosystems: The new TTS (Springer, Berlin, 2005), pp. 518–531
33. ML Akkar, N Courtois, R Duteuil, et al., in *International Workshop on Theory and Practice in Public Key Cryptography: Public Key Cryptography*. A fast and secure implementation of Sflash (Springer, Berlin, 2003), pp. 267–278
34. Wolf C. Efficient Public Key Generation for Multivariate Cryptosystems. 2003.
35. BY Yang, CM Cheng, BR Chen, et al., in *Security in Pervasive Computing*. Implementing minimized multivariate PKC on low-resource embedded systems (Springer, Berlin Heidelberg, 2006), pp. 73–88
36. C Berbain, O Billet, H Gilbert, Efficient implementations of multivariate quadratic systems. Lect. Notes Comput. Sci **4356**, 174–187 (2006)
37. S Balasubramanian, A Bogdanov, A Rupp, et al., in *International Symposium on Field-Programmable Custom Computing Machines*. Fast multivariate signature generation in hardware: The case of rainbow (IEEE Computer Society, USA, 2008), pp. 281–282
38. S Tang, H Yi, J Ding, et al., in *Post-Quantum Cryptography*. High-speed hardware implementation of rainbow signature on FPGAs (Springer, Berlin Heidelberg, 2011), pp. 228–243
39. H Yi, S Tang, Very small FPGA processor for multivariate signatures. Comput. J. **59**(7), 1091–1101 (2016)
40. A Bogdanov, T Eisenbarth, A Rupp, et al., in *Proceeding of the, International Workshop on Cryptographic Hardware and Embedded Systems*. Time-area optimized public-key engines: MQ-cryptosystems as replacement for elliptic curves? (Springer, Berlin, 2008), pp. 45–61
41. H Yi, S Tang, R Vemuri, Fast inversions in small finite fields by using binary trees. Comput. J. **59**(7), 1102–1112 (2016)