Spectral Modular Arithmetic for Binary Extension Fields

Gokay Saldamli
MIS department, Bogazici University
34342 Bebek, Istanbul, Turkey
e-mail: gokay.saldamli@boun.edu.tr

Yoo-Jin Baek
System LSI, Samsung Electronics
Yongin-city, Gyeonggi-do 449-711, S. Korea
e-mail: yoojin.baek@samsung.com

Cetin Kaya Koc
Istanbul Sehir University, Istanbul, Turkey &
University of California, Santa Barbara
e-mail: koc@cs.ucsb.edu

***Abstract*—We describe a method of carrying multiplication in the binary extension fields. The new method fully operates on the Fourier representations of the field elements by successively applying the convolution property and a reduction technique defined on the Fourier coefficients. With some careful parameter selection, the method yields highly parallel architectures for operations involving several field multiplications such as the scalar multiplication calculation of elliptic curve cryptography.**

Keywords-Spectral modular arithmetic, elliptic curve cryptography, polynomial transforms.

I. INTRODUCTION

Spectral techniques for integer multiplication have been known for over a quarter of a century. Because of the overheads related to the backward and forward Fourier transform calculations; these methods (e.g. Schonhage-Strassen [1]) are not effective when operand sizes are small. However, asymptotically superior spectral methods perform significantly better as the operand sizes get larger. Nevertheless; the asymptotic crossovers are still larger than the key sizes of most cryptosystems, so some naive or variations of Karatsuba method [2] having lower complexity are preferred in practice. Similarly, direct use of spectral methods for finite field arithmetic has the same behavior.

In recent studies ([3] [4] [5] [6] [7]), spectral methods are utilized for modular integer and polynomial arithmetic. These studies proposed a modular reduction on the Fourier coefficients. Since the reduction fully works in the spectrum, these methods give highly parallel architectures for performing operations involving several modular multiplications. In this study, we introduce how these ideas can be extended to binary extension fields. After giving a brief introduction to binary extension fields, we formally define the Discrete Fourier Transform (DFT) over a polynomial ring R. We, then, present our main contribution namely spectral modular reduction and multiplication for binary extension fields.

The most essential point of this utilization is to find some suitable DFT domains having acceptable transform lengths for certain principle roots of unity. Unfortunately, if p is small, $\mathbf{Z}_p$ (the ring of integers modulo p) admits very short transform lengths (e.g. $\mathbf{Z}_2$ allows only a transform of length two). One way to overcome this problem is to use some polynomial rings over $\mathbf{Z}_p$ as the domain of DFT. In Sec. 3, we present such suitable spectra. Section 4 is devoted to performance improvements and applications. We conclude our work with some final comments in the last section.

II. SPECTRAL MODULAR ARITHMETIC

A. Binary Finite Fields

Binary extension fields can be represented by the set of polynomials with polynomial addition and multiplication modulo an irreducible polynomial $f(t)$ over GF(2) having degree k. The degree of the polynomial $f(t)$ is also referenced as the degree of the extension. When k is not a prime, the field GF(2^k) is called a *composite field*. Although composite fields enjoy the simplified arithmetic, their usage in practice (particularly in elliptic curve cryptography (ECC)) has some security concerns. In fact, ANSI X9.63 [8] explicitly excludes the use of elliptic curves over composite fields. Therefore, we do not discuss the optimizations making use of these special cases and assume k is arbitrary.

B. Fourier Transform (DFT)

Spectral techniques are widely accepted and used in the field of digital signal processing, hence; most existing notation and concept comes from this theory (see [9][10][11]). We believe that a more appropriate notation permits us to have a better understanding of using the spectral techniques for the computer arithmetic related problems. In this sense, we define the DFT as a map from a polynomial frame to a Fourier ring after introducing these.

Definition 1: Let R be a ring, the set of ordered d-tuples $\mathcal{F}^d = \oplus_{d-1}^{i=0} R$ forms a ring with component-wise addition and multiplication (also called direct sum of rings). For notation purposes, we denote these d-tuples with polynomials (i.e. $(X_1, X_2, \ldots, X_d)$ are written as $X_0+X_1t+ \ldots +X_{d-1}t^{d-1}$). We named the ring $\mathcal{F}^d$ as the *Fourier ring* over R; moreover the elements are called *spectral polynomials* having *spectral coefficients*.

978-1-4244-9515-3/11/$26.00 ©2011 IEEE

Fourier rings can be considered over various ring structures. For instance in [3], R is taken as $\mathbf{Z}_n$ and a construction for modular integer arithmetic is presented. Here we take R as a factor ring, $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$, where $g(\gamma)$ is an arbitrary polynomial in $\mathbf{Z}_2[\gamma]$ (recall that $\mathbf{Z}_2[\gamma]$ represents the ring of polynomials over $\mathbf{Z}_2$ using indeterminate γ).

Remark 1: Since the arithmetic is modulo 2, we add the 2 subscript to our notation and denote the Fourier ring by $\mathcal{F}_2^{\,d}$.

On the other hand, the set of d-tuples $(X_0, X_1, \ldots, X_{d-1})$ forms another ring when it is equipped with the usual polynomial addition and multiplication denoted by $(\mathbf{Z}_2[\gamma])[t]$ using the indeterminate t. The coefficients of the elements of this ring are polynomials in $\mathbf{Z}_2[\gamma]$. We consider a subset of this ring as the domain of the DFT map.

Definition 2: Let d and m be positive integers, we define a *polynomial frame* for $i = 0, 1, \ldots, d\text{-}1$ as

$$\mathbf{Z}_2^{(d,m)} = \{y(t) \in (\mathbf{Z}_2[\gamma])[t] : \deg_t(y(t)) < d \text{ and } \deg_\gamma(y_i(\gamma)) < m$$

Observe that $\mathbf{Z}_2^{(d,m)}$ is a closed set under polynomial addition whereas it is no longer closed under polynomial multiplication (i.e. the product of two elements could be out of $\mathbf{Z}_2^{(d,m)}$ frame). Note that, we define $\mathbf{Z}_2^{(d,m)}$ as a simple subset of $(\mathbf{Z}_2[\gamma])[t]$ without any structure on it. The frame $\mathbf{Z}_2^{(d,m)}$ should not be taken as $R[t]/(f(t)) = (\mathbf{Z}_2[\gamma]/(g(\gamma))[t])/(f(t))$ where $g(\gamma) = \gamma^{m-1}$ defines the reduction on the coefficients and $f(t) = t^{d-1}$ describes the reduction on t. With this description we have a better understanding of the overflows and embeddings to the Fourier rings. Now, we can define the DFT map.

Definition 3: Assume that $\mathbf{Z}_2^{(d,m)}$ is a polynomial frame; $\boldsymbol{F}_2^{\,d}$ is a Fourier ring over $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$ where $g(\gamma)$ is a degree m-1 polynomial over $\mathbf{Z}_2$ and ω is a primitive d-th root of unity in R. The DFT map over R is an invertible set map $DFT_d^{\omega} : \mathbf{Z}_2^{(d,m)} \rightarrow \mathcal{F}_2^{\,d}$ sending $x(t)$ to $X(t)$ defined as follows

$$X_i = DFT_d^{\omega}((x(t)) \coloneqq \sum_{j=0}^{d-1} x_j \omega^{ij} \bmod g(\gamma) \qquad (1)$$

with the inverse

$$x_i = IDFT_d^{\omega}((X(t)) \coloneqq d^{-1} \sum_{j=0}^{d-1} X_j \omega^{ij} \bmod g(\gamma) \qquad (2)$$

where $i = 0, 1, \ldots, d\text{-}1$. Moreover, we write

$$x(t) \quad \xleftrightarrow{DFT} \quad X(t)$$

and say $x(t)$ and $X(t)$ are transform pairs, $x(t)$ is called a *time polynomial* with *time coefficients* and sometimes $X(t)$ is called as the *spectrum* of $x(t)$.

Notice that the coefficient calculations in both (1) and (2) are carried over $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$. Therefore, the summation on the right hand side of (1) is clearly well defined since ω is a principle root of unity. On the other hand, it is not necessarily correct to make the same assumption in (2). First of all, one has to guarantee the existence of the inverse, even which is not sufficient. Pollard [12] mentions that the existence of primitive root d-th of unity and the inverse of d do not guarantee the existence of a DFT over a ring. He adds that a DFT exists in ring R if and only if each quotient field R/M (where M is maximal ideal) possesses a primitive root of unity. If $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$ is taken, the following theorem specifies a more convenient version of this result.

Theorem 1: A d-point DFT defined over $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$ for a primitive root of unity ω, supports circular convolution if and only if the following conditions are satisfied

i. $\omega^d \equiv 1 \bmod g(\gamma)$
ii. $\gcd(d, 2) = 1$, i.e. d is odd
iii. $$\sum_{i=0}^{d-1} \omega^{ir} \equiv \begin{cases} 0 & \text{if } r \neq 0 \bmod d \\ d & \text{if } r = 0 \bmod d \end{cases}$$

Proof : See Sec. 8 of Nussbaumer [13].

Remark that instead of validating the condition (iii) one can simply check whether $\gcd(\omega^{e-1}, g(\gamma)) = 1$ for $e = 1, 2, \ldots, d\text{-}1$ is satisfied or not. For instance in $R = \mathbf{Z}_2[\gamma]/(\gamma^5\text{-}1)$ condition (i) and (ii) are valid since $\omega = \gamma$ had order 5 and $\gcd(5,2) = 1$. However, a 5-point DFT does not exist because condition (iii) fails; $\gcd(\omega - 1, \gamma^5 - 1) = \omega - 1 \neq 1$.

C. Spectral Modular Reduction (SMR)

As we are interested in binary field arithmetic, we start with discussing the relation between the elements of these fields and the polynomial frames using evaluation maps.

Let $x'(\gamma) = x'_0 + x'_1 \gamma + \ldots + x'_{k-1} \gamma^{k-1}$ be an element of $GF(2^k)$ for $x'_i \in GF(2)$, first we want to see $x'(\gamma)$ as an element of $\mathbf{Z}_2^{(d,v)}$. In order to do this we chop $x'(\gamma)$ into d words of length v such that $k < d\,v$, and represent this new polynomial as $x(t) = x_0 + x_1 t + \ldots + x_{d-1} t^{d-1}$ using the indeterminate t. Observe that $x_i = x'_{iv} + x'_{iv+1} \gamma^1 + \ldots + x'_{iv+v-1} \gamma^{v-1}$ for $i = 0, 1, \ldots, d\text{-}1$ and the evaluation of $x(t)$ at $t = \gamma^v$ gives $x'(\gamma)$. Once $\mathbf{Z}_2^{(d,v)}$ representations of the elements of $GF(2^k)$ are computed, these base $t = \gamma^v$ polynomials can be embedded into the DFT machinery.

Definition 4: Let $x'(\gamma)$, $y'(\gamma)$ and $f'(\gamma)$ be polynomials over $GF(2)$ so that $y'(\gamma) \equiv x'(\gamma) \bmod f'(\gamma)$ and $\deg_\gamma(y'(\gamma)) < \deg_\gamma(f'(\gamma))$. If $x(t) \xleftrightarrow{DFT} X(t)$, $y(t) \xleftrightarrow{DFT} Y(t)$ and $f(t) \xleftrightarrow{DFT} F(t)$ for a DFT map where $x(t)$, $y(t)$ and $f(t)$ are evaluation polynomials of $x'(\gamma)$, $y'(\gamma)$ and $f'(\gamma)$ respectively, we call $Y(t)$ as the *spectral modular reduction* of $X(t)$ with respect to $F(t)$.

We present our spectral algorithms as the translations of some time domain algorithms using the properties of DFT such as linearity, convolution and time/frequency shifts (see [4]). Although it is possible to translate a variant of the standard polynomial division, we prefer to present a Montgomery type [14] reduction method.

Algorithm 1: Time domain algorithm for SMR

Let $f(t) \in \mathbf{Z}_2^{(d,u)}$ and $x(t) \in \mathbf{Z}_2^{(d,v)}$ be degree e-1 and d-1 evaluation polynomials of $f'(\gamma)$ and $x'(\gamma)$ such that $d \geq e$, $u = \lfloor v/2 \rfloor$ and $f_0 = 1$ (here we assume $f'_0 = 1$. Note that in case of the least word of $f'(\gamma)$ being not equal to 1, $f(t)$ is taken as an evaluation polynomial of a multiple of $f'(\gamma)$.

Input: evaluation polynomials $x(t)$ and $f(t)$.

Output: $y(t) \equiv x(t)\, t^{(d-e)}$ modulo $f(t)$.

1. $y(t) := x(t)$; $\alpha := 0$;
2. ***for*** $i = 0$ ***to*** d-e
3. $\beta := y_0 + \alpha \bmod \gamma^u$
4. $\alpha := (y_0 + \alpha + \beta)$ ***divide*** γ^u
5. $y(t) := y(t) + \beta f(t)$
6. $y(t) := y(t) + y_0$
7. $y(t) := y(t)/t$
8. ***end for***
9. ***return*** $y(t)$

After giving some related notation, we translate Algorithm 1 to the spectrum using DFT properties.

Notation 1: Assume that ω is a principal d-th root of unity; we let $\gamma(t)$ be the spectral polynomial with coefficients consists of the negative powers of ω i.e. $\Gamma(t) = 1 + \omega^{-1}t + \omega^{-2}t^2 + \ldots + \omega^{-(d-1)}t^{(d-1)}$.

Notation 2:Let x_0 be a constant in $(\mathbf{Z}_2[\gamma])[t]$, the transform of x_0 with respect to a DFT map is given by a spectral polynomial with all coefficients equal to x_0. We denote DFT(x_0) by simply $x_0(t)$. For instance; for $x_0 = 5+\gamma$, $x_0(t) = (5+\gamma)(t) = (5+\gamma) + (5+\gamma)t + \ldots + (5+\gamma)t^{d-1}$ is such a spectral polynomial having degree d-1.

Algorithm 2: Spectral Reduction Algorithm

Assume that $DFT_d^{\omega} : \mathbf{Z}_2^{(d,m)} \rightarrow \mathcal{F}_2^{\,d}$ exists where $\mathcal{F}_2^{\,d}$ is a Fourier ring over $\mathbf{Z}_2[\gamma]/(g(\gamma))$ for a monic polynomial $g(\gamma)$ over $\mathbf{Z}_2$ not necessarily irreducible. Let $f(t)$ and $x(t)$ are evaluation polynomials of $f'(\gamma)$and $x'(\gamma)$ respectively and satisfy $x(t) \xleftarrow{DFT} X(t)$ and $f(t) \xleftarrow{DFT} F(t)$ as described.

Input: $X(t)$ and $F(t)$.

Output: $y(t) \equiv DFT(x(t)\, t^{(e-d)} \bmod f(t))$.

1. $Y(t) := X(t)$; $\alpha := 0$;
2. ***for*** $i = 0$ ***to*** d-e
3. $y_0 := d^{-1}\,(Y_0+Y_1+\ldots+Y_d) \bmod g(\gamma)$
4. $\beta := y_0 + \alpha \bmod \gamma^u$
5. $\alpha := (y_0 + \alpha + \beta)$ ***divide*** γ^u
6. $Y(t) := Y(t) + \beta\, F(t) \bmod g(\gamma)$
7. $Y(t) := Y(t) + (y_0 + \beta)(t) \bmod g(\gamma)$
8. $Y(t) := Y(t) \odot \Gamma(t) \bmod g(\gamma)$
9. ***end for***
10. ***return*** $Y(t)$

Our next step is to prove that Algorithm 1 and 2 agrees.

Theorem 2: Algorithm 1 and 2 agree; in other words there exists a DFT relation between the intermediate and output data in two domains at all times.

Proof: Let $(x(t), X(t))$ and $(f(t), F(t))$ be transform pairs. In Step 4, we compute the least significant coefficient, y_0 of the time polynomial $y(t)$, using the shifting property of DFT (notice that in Algorithm 1, y_0 comes for free). Once y_0 is computed, in Step 5 and 6, the parameters β and carry α are generated.

In Step 7, a β multiple of $F(t)$ is added to $Y(t)$, this updates $Y(t)$ such that $y_0 = 0 \bmod \gamma^u$. In fact, by linearity, this is equivalent to Step 6 of Algorithm 1.

Since carry α is saved in order to add to the consecutive digit in the next run of the loop, a division by t can be performed after eliminating the contribution of y_0 to the spectral polynomial $Y(t)$. Since, Step 7 updates y_0 with $y_0 + \beta$, the computation, $(Y(t)-(y_0+\beta)(t))$, sets zeroth time coefficient of $Y(t)$ to zero (observe that $(y_0+\beta) \in \mathbf{Z}$ is a constant so $(y_0+\beta)(t)$ is a fixed term polynomial, see Notation 2). If it is followed by a $\Gamma(t)$ multiplication, a circular shift is implemented in Steps 8 and 9.

It is still early to conclude that these two algorithms are transform pairs before showing that no overflows exist in the domain $\mathbf{Z}_2^{(d,v)}$. Now, examine how big the degrees of the coefficients get. Initially, $\deg_\gamma(y_i) < v$ for $i = 0, 1, \ldots, d$-1 since $y(t) = x(t)$ is in $\mathbf{Z}_2^{(d,v)}$. As no added value to the coefficients has degree more than v (maximum is attained with $\beta f(t)$ calculation but having $f(t)$ in $\mathbf{Z}_2^{(d,u)}$ and $\deg_\gamma(\beta) < \deg_\gamma(\gamma^u) = u$ imply that $\deg_\gamma(\beta f_i) < u+u=\lfloor v/2 \rfloor + \lfloor v/2 \rfloor < v$), the intermediate values and the output $y(t)$ of the time simulation are in $\mathbf{Z}_2^{(d,v)}$. Therefore, no overflows occur; Algorithm 1 and 2 generate the transform pair $y(t)$ and $Y(t)$ respectively. □

With Algorithm 2 we have completed our primary discussion on spectral modular reduction. Next, we introduce the spectral modular multiplication for polynomials.

D. Spectral Modular Multiplication (SMM)

Convolution and SMR can be combined to harvest a spectral field multiplication algorithm for binary extension fields if a suitable DFT transform exist.

Algorithm 3: Spectral Modular Product

Assume that there exists a DFT map $DFT_d^{\omega} : \mathbf{Z}_2^{(d,m)} \rightarrow \mathcal{F}_2^{\,d}$ where $\mathcal{F}_2^{\,d}$ is a Fourier ring over $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$ for a monic polynomial $g(\gamma)$ over $\mathbf{Z}_2$ not necessarily irreducible. Let $X(t)$, $Y(t)$ and $F(t)$ be transform pairs of $x(t)$, $y(t)$ and $f(t)$ (the evaluation polynomials of $x'(\gamma)$, $y'(\gamma)$ and $f'(\gamma)$ respectively as described in the previous section) respectively where $x(t)$, $y(t)$ in $\mathbf{Z}_2^{(s,u)}$ and $f(t)$ in $\mathbf{Z}_2^{(s+1,u)}$ for $s := \lceil d/2 \rceil$, $u = \lfloor v/2 \rfloor$ and f_0=1.

Input: $X(t)$, $Y(t)$ and $F(t)$; spectral polynomials.

Output: $Z(t) \equiv DFT(z(t))$ where $z(\gamma^u) = z'(\gamma)$ and $z'(\gamma) \equiv x'(\gamma)\, y'(\gamma)\, \gamma^{-du} \bmod f'(\gamma)$,

Procedure SMP($X(t)$, $Y(t)$)

1. $Z(t) := X(t) \odot Y(t)$; $\alpha := 0$;
2. ***for*** $i = 0$ ***to*** d-e
3. $z_0 := d^{-1}\,(Z_0+Z_1+\ldots+Z_d) \bmod g(\gamma)$
4. $\beta := -(z_0 + \alpha) \bmod \gamma^u$
5. $\alpha := (z_0 + \alpha + \beta)$ ***divide*** γ^u
6. $Z(t) := Z(t) + \beta\, F(t) \bmod g(\gamma)$
7. $Z(t) := Z(t) - (z_0 + \beta)(t) \bmod g(\gamma)$
8. $Z(t) := Z(t) \odot \Gamma(t) \bmod g(\gamma)$
9. ***end for***
10. ***return*** $Z(t)$

Algorithm 4: Spectral Modular Multiplication

Assume that $DFT_d^{\omega} : \mathbf{Z}_2^{(d,m)} \rightarrow \mathcal{F}_2^{\,d}$. Let $x'(\gamma)$, $y'(\gamma)$ exists and monic $f'(\gamma)$ be the polynomials over $\mathbf{Z}_2$ such that the degrees of $x'(\gamma)$ and $y'(\gamma)$ are less than $\deg(f'(\gamma)) = k = su$.

Input: Polynomials $x'(\gamma)$, $y'(\gamma)$ and monic $f'(\gamma)$ in GF(2^k).
Output: $z'(\gamma) \equiv x'(\gamma)\, y'(\gamma) \bmod f'(\gamma)$

1. Compute base $t = \gamma^u$ evaluation polynomials $x(t)$, $y(t)$ and $f(t)$ of $x'(\gamma)$, $x'(\gamma)$ and $f'(\gamma)$ such that $f_0 = 1$
2. Compute $\lambda(t)$ where $\lambda'(\gamma) = \gamma^{ud} \bmod f'(\gamma)$.
3. $F(t) := DFT(f(t))$
4. $X^d(t) := DFT(x^d(t))$
5. $Y(t) := DFT(y(t))$
6. $Z(t) := \mathrm{SMP}(X^d(t), Y(t))$
7. $z(t) := IDFT(Z(t))$
8. ***return*** $z'(\gamma)$

Our next step is to prove that Algorithm 4 indeed computes the field multiplication.

Theorem 3: Let $x'(\gamma)$, $y'(\gamma)$ and monic $f'(\gamma)$ be the polynomials over $\mathbf{Z}_2$ such that the degrees of $x'(\gamma)$ and $y'(\gamma)$ are less than $\deg(f'(\gamma))$. Algorithm 4 computes the modular multiplication $z'(\gamma) \equiv x'(\gamma)\, y'(\gamma) \bmod f'(\gamma)$

Proof: Recall that SMP function returns $Z(t)$ where its time polynomial $z(t)$ satisfies $z(t) \equiv x(t)\, y(t)\, t^{-d} \bmod f(t)$. If the time polynomial of Step 7 is examined, one gets

$$IDFT(\mathrm{SMP}(X^d(t), Y(t))) = x(t)\, t^d\, y(t)\, t^{-d} \bmod f(t)$$
$$= x(t)\, y(t) \bmod f(t)$$

since $x^d(t) = x(t)\, t^d \bmod f(t)$. This gives the proof subject to the condition that no overflows occur. Notice that the steps other than Step 7 are usual transform calculations and they do not cause any overflows, hence it suffices to analyze Step 7. Fortunately, SMP generates time polynomials in $\mathbf{Z}_2^{(d,v)}$ and its intermediate values always lies in $\mathbf{Z}_2^{(d,v)}$ frame (see Theorem 2). Therefore Algorithm 4 agrees with its time simulation and computes the modular multiplication. □

III. Suitable Spectrums for Extension Fields

An employment of spectral methods partitions a bigger problem into small pieces and then process on the pieces in a parallel fashion. Notice that the computations in these pieces are carried in the ring, $R = \mathbf{Z}_2[\gamma]/(g(\gamma))$, hence for a proper $g(\gamma)$ selection, spectral methods benefit most.

A. Polynomial Rings with defining binomials

The most convenient choice of $g(\gamma)$ is a binomial. Moreover, if the principal root of unity, ω is chosen as a power of γ, the spectral coefficients are computed only using XORs and circular shifts. Although polynomials rings having defining binomials are good candidates for their simplified arithmetic, they suffer from the short transform lengths. For instance $\gamma^n + 1$ has the linear factor $\Phi_1(\gamma) = \gamma + 1$ for all n, and by Theorem 1 only a transform length of two can be defined over these rings. However, it is possible to overcome such restrictions using the pseudo transforms (PT).

Pseudo number transforms (PNT) are initially defined over subrings of Fermat or Mersenne rings. They support longer transform lengths and benefit the simplified arithmetic of operating in the larger Mersenne or Fermat rings [4]. A similar approach is possible for constructing transforms over polynomial rings. If $g(\gamma) = \gamma^n + 1$ is considered, a nice transform with a longer length can be extracted over a subring defined by a proper factor of $g(\gamma)$.

Example 1: Lets consider the DFT over $R = \mathbf{Z}_2[\gamma]/(\gamma^7+1)$, with the principal root of unity $\omega = \gamma$. Since γ^7+1 has the following factorization

$$\gamma^7+1 = (\gamma+1)\,(\gamma^3+\gamma^2+1)\,(\gamma^3+\gamma+1) = \Phi_1(\gamma)\,\Phi_7(\gamma),$$

the ring R admits transform of lengths at most two but if the ring $R' = \mathbf{Z}_2[\gamma]/(\Phi_7(\gamma))$, we get a 7-point DFT satisfying the convolution property over the ring R'. Beside that one needs a $\Phi_7(\gamma)$ reduction while working in R' which is obviously harder than the arithmetic in R. However, since R' is a subring of R, all calculations can be carried over R with a final $\Phi_7(\gamma)$ reduction whenever necessary.

Remark 2: While embedding the input data to the PT domain, the size of the subring should be considered rather than the size of R. In fact, the most interesting PTs are the ones which enlarge the lengths with minimal shrinkage.

In Table 1, we present parameters for some suitable pseudo transform rings. One can extend the table to an arbitrary $g(\gamma)$ using Theorem 1 meeting the marginal needs of a particular application.

TABLE I. Suitable polynomial rings for an odd prime D

Ring	ω	length
$(z^d+1)/(z+1), (z^{2d}+1)/(z^2+1)$	z	d
$(z^{d^2}+1)/(z^d+1), (z^{2d^2}+1)(z^{2d}+1)$	z	d^2

B. Finite Field Spectrums

If $g(\gamma)$ is irreducible (i.e. the factor ring is a finite field), various arithmetic simplifications can be considered. As binary extension fields can be seen as n-dimensional vector spaces over GF(2): if $\alpha_1, \alpha_2 \ldots, \alpha_n$ is taken as basis set, each element of GF(2^n) can be represented as a linear combination of the elements of this basis set. Among various bases, there are two special types having particular importance. The first one is the canonical polynomial basis $\{1, \alpha, \alpha^2, \ldots, \alpha^{n-1}\}$, made up of powers of a defining (mostly primitive) element α of GF(2^n). The second one is the normal basis of the form $\mathrm{N} = \{\alpha, \alpha^q, \ldots, \alpha^{q^{n-1}}\}$ consists of a normal element $\alpha \in$ GF(q^n) and its conjugates with respect to GF(2).

For every finite field there exists a normal basis, in fact, several such bases may exist for the same field. Those bases having the minimal complexity are called optimal normal bases (ONB). For our purposes type I ONBs have the utmost importance in which the element α is taken as the principal root of unity. Observe that this is the case where the normal basis N and the set of roots of unity (i.e. $\{1, \omega, \omega^2, \ldots, \omega^{n-1}\}$) become set equivalent (not necessarily equal as ordered sets). Therefore, one can change the basis from normal to polynomial basis or vice versa by simply ordering the terms. Unfortunately not all the finite fields have type I ONB; following proposition gives a condition for their existence.

Proposition 1: Suppose $n+1$ is a prime and q is primitive in $\mathbf{Z}_{n+1}$, where q is a prime or prime power. Then the n non-unit $(n+1)$th roots of unity are linearly independent and they form an ONB of GF(q^n) over GF(q).

Proof: See Vanstone et al [15] □

Using the above result, for k = 4, 10 ,12, 18, 28, 36, 52, 58, 60, …, the binary extension field GF(2^k) has type I ONB.

In a normal basis representation, squaring a field element corresponds to a circular shift operation which is well-suited for the realizations of public-key cryptosystems employing repeated square and multiply methods. Nevertheless, these representations could suffer from the complicated bases conversions and field multiplications. Eventually, type I ONB are optimal by giving the simplest conversion and multiplication realizations. Therefore, they initially favor a great interest in realizations of ECC but because of some security concerns, the use of elliptic curves over composite fields (type I ONB only exist in these extensions) is excluded from standards such as ANSI X9.63 [8].

Remark 3: We tend to choose a field having a type I ONB for transform domain. Observe that such a selection is implementation related and does not change any ECC parameter; hence it never jeopardizes the security of the cryptosystem.

IV. Applications and Further Improvements

Parameter selection is quite important for possible ECC realizations and related improvements.

A. Parameter Selection for ECC over Binary Fields

The size of the underlying structure (also defines the key length) is a common security measure for public-key cryptosystems. Standard documents [8] and [16] recommend special curves serving different needs of security levels. Referencing to the key sizes of these curves, in Table II, we tabulate some suitable polynomial rings that admit suitable DFT structures. Note that the first column gives the maximum degree of the defining polynomials, an appropriate less degree polynomial can be used with this selection. For instance, the arithmetic in the prime extension field GF(2^{163}) can be performed using the polynomial transform over $g(\gamma)=(\gamma^{37}+1)/(\gamma+1)$. Notice also that unlike SMM, when SMP is used for ECC, the word size becomes $u \approx v/4$ as a result of successive SMP usage. The readers are referred to [4] for a modification giving a better u value ($u \approx v/2$) which is not included here because of the space limitations.

TABLE II. Standard parameter selection for SMP

degree k ‡	PT ring $g(\gamma)$	DFT d	Root ω	word size u	words s
171	$(\gamma^{37}+1)/(\gamma+1)$†	37	γ	9	19
210	$(\gamma^{41}+1)/(\gamma+1)$	41	γ	10	21
242	$(\gamma^{43}+1)/(\gamma+1)$	43	γ	11	22
288	$(\gamma^{47}+1)/(\gamma+1)$	47	γ	12	24
450	$(\gamma^{59}+1)/(\gamma+1)$ †	59	γ	15	30
578	$(\gamma^{67}+1)/(\gamma+1)$ †	67	γ	17	34

‡ shows the maximum degree of the defining polynomial, an appropriate less degree polynomial can be used with this selection and † shows the domains having type I ONB

V. Conclusion

Our motivation was obtaining a finite field multiplier fully working in the spectrum in order to use the convolution property successively for operations involving several field multiplications. In order to meet this goal, using the linearity and shifting property of DFT, we define a spectral polynomial reduction method. Based on this reduction we describe a binary extension field multiplier in the spectrum.

One essential point of this utilization was to find some suitable DFT domains having long transform lengths. We give a solution to this problem by defining DFT over polynomial rings. After carefully studying the simplest possible Fourier rings, we propose to use pseudo polynomial transforms over rings defined by binomials.

Working fully on the spectrum results in a favorable condition that it provides highly parallel modular arithmetic for both hardware and software realizations of public-key cryptosystems involving modular arithmetic.

Acknowledgment

The authors would like to thank to the anonymous reviewers for their helpful comments. Note that, Gokay Saldamli is partially funded by TUBITAK research project No: 109E180.

References

[1] A. Schonhage and V. Strassen, "Schnelle multiplikation grafer zahlen," *Computing*, vol. 7, pp. 281–292, 1971.

[2] A. Karatsuba and Y. Ofman, "Multiplication of multidigit numbers by automata," *Soviet Physics-Doklady*, vol. 7, pp. 595–596, 1963.

[3] G. Saldamli and C. K. Koc, "Spectral modular arithmetic," in Proceedings of the 18th IEEE Symposium on Computer Arithmetic 2007 *(ARITH'07)*, 2007, pp. 123–132.

[4] G. Saldamli, *Spectral Modular Arithmetic*, Ph.D. thesis, Department of Electrical and Computer Engineering, Oregon State University, May 2005.

[5] G. Saldamli and C. K. Koc, *Spectral Modular Arithmetic for Cryptography*, in *Cryptographic Engineering*, C, . K. Koc,, editor, Springer, January 2009.

[6] S. Baktir, S. Kumar, C. Paar, and B. Sunar, "A state-of-the-art elliptic curve cryptographic processor operating in the frequency domain," *Mobile Networks and Applications (MONET)*, vol. 12, no. 4, pp. 259–270, September 2007.

[7] S. Baktir and B. Sunar, "Optimal Extension Field Inversion in the Frequency Domain," *in Proceedings of International Workshop on the Arithmetic of Finite Fields – WAIFI'08*, LNCS 5130, pages 47–61, Siena, Italy. July 6–9, 2008.

[8] ANSI X9.63, Public Key Cryptography for the Financial Services Industry: Elliptic Curve KeyAgreement and Key Transport Protocols, working draft, October 2000.

[9] R. E. Blahut, *Fast Algorithms for digital signal processing*, Addison-Wesley publishing Company, 1985.

[10] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley publishing Company, 1974.

[11] M. A. Shokrollahi M. Clausen and P. Burgisser, *Algebraic Complexity Theory*, Springer, Berlin, Germany, 1996.

[12] J. M. Pollard, "Implementation of number theoretic transform," *Electronics Letters*, vol. 12, no. 15, pp. 378–379, July 1976.

[13] H. J. Nussbaumer, *Fast Fourier transform and convolution algorithms*, Springer, Berlin, Germany, 1982.

[14] C. K. Koc and T. Acar, "Montgomery multiplication in GF(2^k)," *Designs, Codes and Cryptography*, vol. 14, no. 1, pp. 57–69, Apr. 1998.

[15] S. A. Vanstone R. C. Mullin, I. M. Onyszchuk and R.M. Wilson, "Optimal normal bases in GF(p^k)," *Discrete Applied Math.*, vol. 22, pp. 149–161, 1989.

[16] IEEE, "P1363: Standard specifications for public-key cryptography," November 12, 1999, Draft Version 13.