Bit-Parallel $GF(2^n)$ Squarer Using Shifted Polynomial Basis

Xi Xiong and Haining Fan

***Abstract*—We present explicit formulae and complexities of bit-parallel shifted polynomial basis (SPB) squarers in finite field $GF(2^n)$s generated by general irreducible trinomials x^n+x^k+1 $(0<k<n)$ and type-II irreducible pentanomials $x^n+x^{k+1}+x^k+x^{k-1}+1$ $(3<k<(n-3)/2)$. The complexities of the proposed squarers match or slightly outperform the previous best results. These formulae can also be used to design polynomial basis Montgomery squarers without any change. Furthermore, we show by examples that XOR gate numbers of SPB squarers are different when different shift factors in the SPB definition, i.e., parameter v in $\{x^{i-v}|0\le i\le n-1\}$, are used. This corrects previous misinterpretation.**

***Index Terms*—Finite field, squarer, polynomial basis, shifted polynomial basis, irreducible polynomial.**

I. Introduction

BESIDES multiplication, squaring is also an important $GF(2^n)$ operation for cryptographic applications. For example, it is crucial to square-and-multiply-based exponentiation and inversion algorithms. When field elements are represented in normal bases, squaring operations are trivial in finite fields of characteristic 2. For this reason, most previous works in this research field focused on polynomial basis squarers. In [1], Piontas proposed the explicit squaring formula for irreducible trinomials x^n+x+1. For an arbitrary irreducible trinomial, Paar et al. and Wu derived complexities of bit-parallel polynomial basis squarers respectively [2], [3] and [4]. In [5], Wu also presented an optimized squarer based on Montgomery algorithm for general irreducible trinomials x^n+x^k+1, where the Montgomery factor x^k are used. For the special type of irreducible pentanomials $x^n+x^{k+1}+x^k+x^{k-1}+1$ $(3<k<(n-3)/2)$, which is known as type-II irreducible pentanomials, Hariri and Reyhani-Masoleh presented a Montgomery squarer using Montgomery factor x^k [6]. For an arbitrary irreducible pentanomial, Park recently presented explicit formulae and complexities of $GF(2^n)$ squarers based on weakly dual basis (WDB) [7].

Besides the above works focusing mainly on bit-parallel squarers, there are also some other related works. In reference [8], Wu and Hasan extended the squaring structure to that of the polynomial basis fourth power. Recently, Järvinen discussed the problem of computing repeated squarings (exponentiations to a power of 2) in $GF(2^n)$ [9].

In this paper, we propose explicit formulae of some bit-parallel squarers based on shifted polynomial basis (SPB). We first consider $GF(2^n)$s generated by general irreducible trinomials x^n+x^k+1 $(0<k<n)$. Unlike previous works on polynomial basis squarers, which presented only explicit formulae for cases $1<k\le n/2$, we present explicit formulae of SPB squarers for all values of k in the range $[1,n-1]$. Then we consider $GF(2^n)$s generated by type-II irreducible pentanomials $x^n+x^{k+1}+x^k+x^{k-1}+1$ $(3<k<(n-3)/2)$. The complexities of these two classes of squarers match or slightly outperform the previous best results. Owing to the equivalent relationship between $GF(2^n)$ Montgomery and SPB multiplication algorithms [10], these formulae can also be used to design Montgomery squarers without any change.

In addition to improvements on XOR gate numbers, another contribution of this work is to show that XOR gate numbers of SPB squarers are different when different shift factors in the SPB definition, i.e., parameter v in $\{x^{i-v}|0\le i\le n-1\}$, are used. Taking $GF(2^n)$ generated by irreducible trinomial x^n+x^k+1 as an example, it had been shown that the two SPB bit-parallel Mastrovito multipliers using different shift factors $v=k$ and $v=k-1$ have the same XOR gate numbers and gate delays [11]. It is then natural to assume that both of these two shift factors may lead to bit-parallel squarers of the same complexities too [6, Section 7]. But this is not true: we will show that XOR gate numbers of the two SPB squarers are indeed different. Please refer to Table I and Table IV for more details.

The remainder of this paper is organized as follows: In Section II, we give a brief review on SPB multipliers. Architectures of bit-parallel SPB squarers for general irreducible trinomials and type-II irreducible pentanomials are presented in Section III and IV, respectively. Finally, concluding remarks are made in Section V.

II. SPB Multipliers

An SPB of $GF(2^n)$ over $GF(2)$ is defined as follows [11]:

Definition 1: Let v be an integer and the ordered set $M=\{x^i|0\le i\le n-1\}$ be a polynomial basis of $GF(2^n)$ over $GF(2)$. The ordered set $x^{-v}M:=\{x^{i-v}|0\le i\le n-1\}$ is called a shifted polynomial basis with respect to M.

Let $f(x)=x^n+x^k+1$ be an irreducible trinomial over $GF(2)$. All elements of $GF(2^n)=GF(2)[x]/(f(x))$ can be represented using an SPB $\{x^{i-v}|0\le i\le n-1\}$. Given two field elements A and B, let $A=x^{-v}\sum_{i=0}^{n-1}a_ix^i$ and $B=x^{-v}\sum_{i=0}^{n-1}b_ix^i$ be their SPB representations. The SPB product $C=x^{-v}\sum_{i=0}^{n-1}c_ix^i$ of A and B can be computed using the following two steps [11].

(i) Perform the conventional polynomial multiplication:

$$S=AB=x^{-2v}\sum_{t=0}^{2n-2}s_tx^t=\sum_{t=-2v}^{2n-2-2v}s_{t+2v}x^t=r_-+r+r_+,$$

Xi Xiong and Haining Fan are with the School of Software, Tsinghua University, Beijing, China. E-mails: xiongxi08@gmail.com, fhn@tsinghua.edu.cn

where $r = \sum_{t=-v}^{n-1-v} s_{t+2v}x^t$, $r_- = \sum_{t=-2v}^{-1-v} s_{t+2v}x^t$, $r_+ = \sum_{t=n-v}^{2(n-1-v)} s_{t+2v}x^t$ and

$$s_t = \sum_{\substack{i+j=t \\ 0\le i,j<n}} a_i b_j = \begin{cases} \sum_{i=0}^{t} a_i b_{t-i} & 0 \le t \le n-1 \\ \sum_{i=t+1-n}^{n-1} a_i b_{t-i} & n \le t \le 2n-2 \end{cases}.$$

(ii) Reduce r_- and r_+ using the following two reduction equations, respectively:

$$x^i = x^{i+k} + x^{i+n}, \tag{1}$$

where $-2v \le i \le -(v+1)$ and

$$x^i = x^{i-n} + x^{i-n+k}, \tag{2}$$

where $n - v \le i \le 2n - 2 - 2v$.

The reduced results $\widetilde{r}_-$ and $\widetilde{r}_+$ are defined as

$$\widetilde{r}_- = \sum_{t=n-2v}^{n-1-v} s_{t+2v-n}x^t + \sum_{t=k-2v}^{k-1-v} s_{t+2v-k}x^t$$

and

$$\widetilde{r}_+ = \sum_{t=k-v}^{k+n-2-2v} s_{t+2v+n-k}x^t + \sum_{t=-v}^{n-2-2v} s_{i+n+2v}x^t.$$

Then the SPB product C of A and B is

$$C = \sum_{i=0}^{n-1} c_i x^{i-v} = \sum_{i=-v}^{n-v-1} c_{i+v}x^i = r + \widetilde{r}_- + \widetilde{r}_+.$$

In order to reduce the total gate delays of bit-parallel SPB quadratic multipliers, reference [11] proved that the best values of shift factor v should be k or $k-1$ for all irreducible trinomials $f(x) = x^n + x^k + 1$. It is natural to assume that these two shift factors are also the best choices for bit-parallel squarers [6, Section 7]. However, our results will show that this is not true. In the following, we will first derive explicit formulae of SPB squarers for the case $v = k$, and then for the case $v = k-1$. Finally, XOR gate complexities of these formulae are compared in Table I. Especially, explicit formulae for the cases $k > \frac{n}{2}$, which have not been published before, will be presented in this section.

III. SPB Squarers for All Irreducible Trinomials

A. Explicit Squarer Formulae for the Case $v = k$

Let $f(x) = x^n + x^k + 1$ be an irreducible trinomial over $GF(2)$, where $0 < k < n$. Let $A = x^{-k}\sum_{i=0}^{n-1} a_i x^i$ be the SPB representation of an arbitrary element in $GF(2^n)$ and define a'_i as that in [4]:

$$a'_i = \begin{cases} a_{\frac{i}{2}} & \text{if } i \text{ is even,} \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The square of A is

$$\begin{aligned} C &= \sum_{i=-k}^{n-k-1} c_{i+k}x^i = A^2 = x^{-2k}\sum_{i=0}^{n-1} a_i x^{2i} \\ &= x^{-2k}\sum_{i=0}^{2n-2} a'_i x^i = \sum_{i=-2k}^{2n-2k-2} a'_{i+2k}x^i \\ &= r_- + r + r_+, \end{aligned} \tag{4}$$

where

$$r = \sum_{i=-k}^{n-k-1} a'_{i+2k}x^i, \qquad r_- = \sum_{i=-2k}^{-k-1} a'_{i+2k}x^i,$$
$$r_+ = \sum_{i=n-k}^{2n-2k-2} a'_{i+2k}x^i = \sum_{i=n-k}^{2n-2k-1} a'_{i+2k}x^i.$$

Please note that $a'_{2n-1} = 0$ in the expression of r_+ by (3).

Then we perform reduction operations using the two reduction equations (1) and (2) and obtain:

$$\begin{aligned} \widetilde{r}_- &= \sum_{i=-2k}^{-k-1} a'_{i+2k}(x^{i+k} + x^{i+n}) \\ &= \sum_{i=-k}^{-1} a'_{i+k}x^i + \sum_{n-2k}^{n-k-1} a'_{i-n+2k}x^i, \end{aligned}$$

and

$$\begin{aligned} \widetilde{r}_+ &= \sum_{i=n-k}^{2n-2k-1} a'_{i+2k}(x^{i-n} + x^{i-n+k}) \\ &= \sum_{i=-k}^{n-2k-1} a'_{i+n+2k}x^i + \sum_{i=0}^{n-k-1} a'_{i+n+k}x^i. \end{aligned}$$

Finally, we get the expression of $C = A^2$:

$$\begin{aligned} C &= \sum_{i=-k}^{n-k-1} c_{i+k}x^i = \widetilde{r}_- + r + \widetilde{r}_+ \\ &= \left(\sum_{i=-k}^{-1} a'_{i+k}x^i + \sum_{i=0}^{n-k-1} a'_{i+n+k}x^i\right) + \sum_{i=-k}^{n-k-1} a'_{i+2k}x^i \\ &\quad + \left(\sum_{i=-k}^{n-2k-1} a'_{i+n+2k}x^i + \sum_{n-2k}^{n-k-1} a'_{i-n+2k}x^i\right). \end{aligned} \tag{5}$$

Comparing the coefficients of x^i in this formula, we may obtain explicit formulae of coordinate c_is for $0 \le i \le n-1$. These formulae are different for there cases of $k = \frac{n}{2}$, $k < \frac{n}{2}$ and $k > \frac{n}{2}$. So we present them separately in the following. Similar to [5], we define the notation "$i \doteq s, \ldots, t$" as

$$i = \begin{cases} s, s+2, s+4, \ldots, t & \text{if } |s| + |t| \text{ is even,} \\ s, s+2, s+4, \ldots, t-1 & \text{otherwise.} \end{cases}$$

i.e., i is from s to t or $t-1$ and increases by 2.

Case 1: $n = 2k$

Close observation of (5) reveals that:

$$c_{i+k} = \begin{cases} a'_{i+k} + a'_{i+n+2k} + a'_{i+2k}, & i\epsilon[-k, -1], \\ a'_{i+n+k} + a'_{i-n+2k} + a'_{i+2k}, & i\epsilon[0, n-k-1]. \end{cases}$$

We can simplify the above expressions by noting that values of a'_is with odd subscripts are zero. Since k is odd and $n = 2k$ is even, we obtain:

$$c_{i+k} = \begin{cases} a'_{i+k}, & i \doteq -k, \ldots, -1, \\ a'_{i+n+2k} + a'_{i+2k}, & i \doteq -k+1, \ldots, -2, \\ a'_{i-n+2k} + a'_{i+2k}, & i \doteq 0, \ldots, n-k-1, \\ a'_{i+n+k}, & i \doteq 1, \ldots, n-k-2. \end{cases}$$

The number of XOR gates required is $(\frac{-2+k-1}{2} + 1) + (\frac{n-k-1-0}{2} + 1) = \frac{n}{2}$ and the gate delay is T_X.

Case 2: $n < 2k$

By comparing the coefficients of x^i in (5), we can obtain the explicit expressions of c_i:

$$c_{i+k} = \begin{cases} a'_{i+k} + a'_{i+n+2k} + a'_{i+2k}, & i\epsilon[-k, n-2k-1], \\ a'_{i+k} + a'_{i-n+2k} + a'_{i+2k}, & i\epsilon[n-2k, -1], \\ a'_{i+n+k} + a'_{i-n+2k} + a'_{i+2k}, & i\epsilon[0, n-k-1]. \end{cases} \tag{6}$$

Since $f(x)$ is irreducible, at least one of n and k should be odd. We now simplify the above expressions in three subcases according to the parities of n and k.

Subcase 2.1: n is even and k is odd.

Based on (3) and (6), we get:

$$c_{i+k} = \begin{cases} a'_{i+n+2k} + a'_{i+2k}, & i \doteq -k+1, \ldots, n-2k-2, \\ a'_{i+k}, & i \doteq -k, \ldots, -1, \\ a'_{i+n+k}, & i \doteq 1, \ldots, n-k-2, \\ a'_{i-n+2k} + a'_{i+2k}, & i \doteq n-2k, \ldots, n-k-1. \end{cases}$$

In this subcase, the SPB squarer requires $\frac{n}{2}$ two-input XOR gates and the gate delay is T_X.

Subcase 2.2: n is odd and k is even.

$$c_{i+k} = \begin{cases} a'_{i+n+2k}, & i \doteq -k+1, \ldots, n-2k-2, \\ a'_{i-n+2k}, & i \doteq n-2k, \ldots, -1, \\ a'_{i+k} + a'_{i+2k}, & i \doteq -k, \ldots, -2, \\ a'_{i+2k}, & i \doteq 0, \ldots, n-k-1, \\ a'_{i+n+k} + a'_{i-n+2k}, & i \doteq 1, \ldots, n-k-2. \end{cases}$$

In this subcase, the SPB squarer requires $\frac{n-1}{2}$ two-input XOR gates and the gate delay is T_X.

Subcase 2.3: both n and k are odd.

$$c_{i+k} = \begin{cases} a'_{i+k} + a'_{i+n+2k}, & i \doteq -k, \ldots, n-2k-2, \\ a'_{i+2k}, & i \doteq -k+1, \ldots, -2, \\ a'_{i+k} + a'_{i-n+2k}, & i \doteq n-2k, \ldots, -1, \\ a'_{i-n+2k}, & i \doteq 1, \ldots, n-k-1, \\ a'_{i+n+k} + a'_{i+2k}, & i \doteq 0, \ldots, n-k-2. \end{cases}$$

In this subcase, the SPB squarer requires $\frac{n+1}{2}$ two-input XOR gates and the gate delay is T_X.

Case 3: $n > 2k$

Careful comparison in (5) shows that:

$$c_{i+k} = \begin{cases} a'_{i+k} + a'_{i+n+2k} + a'_{i+2k}, & i\epsilon[-k, -1], \\ a'_{i+n+k} + a'_{i+n+2k} + a'_{i+2k}, & i\epsilon[0, n-2k-1], \\ a'_{i+n+k} + a'_{i-n+2k} + a'_{i+2k}, & i\epsilon[n-2k, n-k-1]. \end{cases} \tag{7}$$

Similarly, we present three different explicit formulae of c_is according to the parities of n and k in the following.

Subcase 3.1: n is even and k is odd.

From (3) and (7), we can get:

$$c_{i+k} = \begin{cases} a'_{i+k}, & i \doteq -k, \ldots, -1, \\ a'_{i+n+2k} + a'_{i+2k}, & i \doteq -k+1, \ldots, n-2k-2, \\ a'_{i+n+k}, & i \doteq 1, \ldots, n-k-2, \\ a'_{i-n+2k} + a'_{i+2k}, & i \doteq n-2k, \ldots, n-k-1. \end{cases}$$

In this subcase, the SPB squarer requires $\frac{n}{2}$ two-input XOR gates and the gate delay is T_X.

Subcase 3.2: n is odd and k is even.

$$c_{i+k} = \begin{cases} a'_{i+n+2k}, & i \doteq -k+1, \ldots, -1, \\ a'_{i+k} + a'_{i+2k}, & i \doteq -k, \ldots, -2, \\ a'_{i+n+k} + a'_{i+n+2k}, & i \doteq 1, \ldots, n-2k-2, \\ a'_{i+2k}, & i \doteq 0, \ldots, n-k-1, \\ a'_{i+n+k} + a'_{i-n+2k}, & i \doteq n-2k, \ldots, n-k-2. \end{cases}$$

In this subcase, the SPB squarer requires $\frac{n-1}{2}$ two-input XOR gates and the gate delay is T_X.

Subcase 3.3: both n and k are odd.

$$c_{i+k} = \begin{cases} a'_{i+2k}, & i \doteq -k+1, \ldots, -2, \\ a'_{i+k} + a'_{i+n+2k}, & i \doteq -k, \ldots, -1, \\ a'_{i+n+2k}, & i \doteq 1, \ldots, n-2k-2, \\ a'_{i-n+2k}, & i \doteq n-2k, \ldots, n-k-1, \\ a'_{i+n+k} + a'_{i+2k}, & i \doteq 0, \ldots, n-k-2. \end{cases}$$

In this subcase, the SPB squarer requires $\frac{n+1}{2}$ two-input XOR gates and the gate delay is T_X.

In summary, for any $k \in [1, n-1]$, the gate delays of all SPB squarers is T_X when the shift factor is selected as $v = k$. But the XOR gate numbers are different according to the parities of n and k. We list them in the middle row of Table I.

TABLE I
XOR GATE NUMBERS OF SPB SQUARERS FOR IRREDUCIBLE $x^n + x^k + 1$ WHERE $2k \neq n$

	n even, k odd	n odd, k odd	n odd, k even
$v = k$	$n/2$	$(n+1)/2$	$(n-1)/2$
$v = k-1$	$n/2$	$(n-1)/2$	$(n+1)/2$

From Table I, it is clear that the case "n odd, k odd" requires 1 more XOR gate than the case "n odd, k even" when the shift factor v is selected as $v = k$. Is it possible to reduce this number? The answer is yes. If we define v as $v = k-1$, then we can obtain a squarer with only $\frac{n-1}{2}$ XOR gates for the case "n odd, k odd". In fact, for the purpose of obtaining all complexity results, we have also derived explicit formulae for all three cases using the shift factor $v = k-1$. Their complexities are listed in the last row of Table I. But for simplicity, we omit the deriving procedure and present only the explicit formulae for the case "n odd, k odd" in the following.

Case 1: $n < 2k$ **and** $v = k-1$

$$c_{i+k-1} = \begin{cases} a'_{i+k-2} + a'_{i+n+2k-2}, & i \doteq -k+2, \ldots, n-2k, \\ a'_{i+2k-2}, & i \doteq -k+1, \ldots, 0, \\ a'_{i+k-2} + a'_{i-n+2k-2}, & i \doteq n-2k+2, \ldots, -1, \\ a'_{i-n+2k-2}, & i \doteq 1, \ldots, n-k-1, \\ a'_{i+n+k-2} + a'_{i+2k-2}, & i \doteq 2, \ldots, n-k. \end{cases} \tag{8}$$

Case 2: $n > 2k$ **and** $v = k-1$

$$c_{i+k-1} = \begin{cases} a'_{i+2k-2}, & i \doteq -k+1, \ldots, 0, \\ a'_{i+k-2} + a'_{i+n+2k-2}, & i \doteq -k+2, \ldots, -1, \\ a'_{i+n+2k-2}, & i \doteq 1, \ldots, n-2k, \\ a'_{i-n+2k-2}, & i \doteq n-2k+2, \ldots, n-k-1, \\ a'_{i+n+k-2} + a'_{i+2k-2}, & i \doteq 2 \ldots, n-k. \end{cases} \tag{9}$$

TABLE II
COMPARISON OF BIT-PARALLEL SQUARERS FOR IRREDUCIBLE $x^n + x^k + 1$

	$1 < k < n/2$		$n/2 < k < n$	
Proposals	XOR gates	Gate delays	XOR gates	Gate delays
n even, k odd				
[4] PB	$(n+k-1)/2$	$2T_X$	-	-
[5] Montgomery	$\lceil (n-1)/2 \rceil$	T_X	-	-
Proposed ($v = k$)	$\lceil (n-1)/2 \rceil$	T_X	$\lceil (n-1)/2 \rceil$	T_X
n odd, k even				
[4] PB	$(n+k-1)/2$	T_X	-	-
[5] Montgomery	$\lceil (n-1)/2 \rceil$	T_X	-	-
Proposed ($v = k$)	$\lceil (n-1)/2 \rceil$	T_X	$\lceil (n-1)/2 \rceil$	T_X
n odd, k odd				
[4] PB	$(n-1)/2$	$2T_X$	-	-
[5] Montgomery	$(n+1)/2$	T_X	-	-
Proposed ($v = k-1$)	$(n-1)/2$	T_X	$(n-1)/2$	T_X

B. Comparison

Table II compares three different implementations of bit-parallel squarers for irreducible trinomials. For the case "n odd, k odd", we note that the XOR gate complexity of Montgomery squarers of [5, formulae (28) and (29)] is not correct, and it should be $\frac{k-1}{2} + 1 + \frac{n-2-k}{2} + 1 = \frac{n-3}{2} + 2 = \frac{n+1}{2}$. These Montgomery squarers use factor x^k, and they are in fact the same as the SPB squarers we derived using the shift factor $v = k$ because of the equivalent relationship between $GF(2^n)$ Montgomery and SPB multiplication algorithms.

For odd values of n, it can be seen from Table I and Table II that $GF(2^n)$ SPB squarers have the lowest time and space complexities.

C. An Example

As an example, we list explicit formulae of two SPB squarers in $GF(2^7)$, which is generated by $f(x) = x^7 + x^3 + 1$. These two squarers use different shift factors. Clearly, the SPB squarer using $v = k - 1 = 2$ requires 1 less XOR gate than that using $v = k = 3$.

TABLE III
EXPLICIT FORMULAE OF TWO $GF(2^7)$ SPB SQUARERS

$v = 2$	$v = 3$
$c_0 = a_1$	$c_0 = a_0 + a_5$
$c_1 = a_0 + a_5$	$c_1 = a_2$
$c_2 = a_2$	$c_2 = a_1 + a_6$
$c_3 = a_6$	$c_3 = a_3 + a_5$
$c_4 = a_3 + a_5$	$c_4 = a_0$
$c_5 = a_0$	$c_5 = a_4 + a_6$
$c_6 = a_4 + a_6$	$c_6 = a_1$

An interesting property of this example is that $c_2 = a_2$ when the shift factor v is $v = k - 1 = 2$. This means that coefficient c_2 is always unchanged in each exponentiation operation $C = A^{2^i}$, where $i > 0$. In fact, careful observation of (8) and (9) reveals that this property always exists for the case "n odd, k odd" when the shift factor v is $k - 1$, i.e., $c_{k-1} = a_{k-1}$ is always true. Similarly, when the shift factor v is defined as $v = k$, we have $c_k = a_k$ for the case "n odd, k even". Furthermore, this property also exists in some polynomial basis squarers [4].

IV. SPB SQUARERS FOR TYPE II IRREDUCIBLE PENTANOMIALS

A. Architectures

In this section, we present explicit formulae of SPB squarers for type-II irreducible pentanomials $f(x) = x^n + x^{k+1} + x^k + x^{k-1} + 1$. Intuitively, we first consider the shift factor $v = k$. Similar to the trinomial case, the two terms r_- and r_+ in (4) are reduced respectively by the following reduction equations:

$$x^i = x^{i+k-1} + x^{i+k} + x^{i+k+1} + x^{i+n},$$

where $-2k \le i \le -(k+1)$ and

$$x^i = x^{i-n} + x^{i-n+k+1} + x^{i-n+k} + x^{i-n+k-1},$$

where $n - k \le i \le 2n - 2 - 2k$. And the reduced results are:

$$\begin{aligned} \widetilde{r}_- &= \sum_{i=-k-1}^{-2} a'_{i+k+1} x^i + \sum_{i=-k}^{-1} a'_{i+k} x^i \\ &\quad + \sum_{i=-k+1}^{0} a'_{i+k-1} x^i + \sum_{i=-2k+n}^{-k-1+n} a'_{i+2k-n} x^i \end{aligned}$$

and

$$\begin{aligned} \widetilde{r}_+ &= \sum_{i=-k}^{n-2k-1} a'_{i+n+2k} x^i + \sum_{i=1}^{n-k} a'_{i+n+k-1} x^i \\ &\quad + \sum_{i=0}^{n-k-1} a'_{i+n+k} x^i + \sum_{i=-1}^{n-k-2} a'_{i+n+k+1} x^i. \end{aligned}$$

Moreover, $a'_0 x^{-k-1}$ in the first term of $\widetilde{r}_-$ should be reduced again, i.e.,

$$\begin{aligned} \sum_{i=-k-1}^{-2} a'_{i+k+1} x^i &= a'_0 (x^{-2} + x^{-1} + x^0 + x^{n-k-1}) \\ &\quad + \sum_{i=-k}^{-2} a'_{i+k+1} x^i. \end{aligned}$$

And the second term $\sum_{i=1}^{n-k} a'_{i+n+k-1} x^i$ in $\widetilde{r}_+$ equals to $\sum_{i=1}^{n-k-1} a'_{i+n+k-1} x^i$ since $a'_{2n-1} = 0$ by the definition of a'_i in (3).

Therefore, we obtain the following expression of $C = A^2$:

$$\begin{aligned} C &= \sum_{i=-k}^{n-k-1} c_{i+k}x^i = \widetilde{r}_- + r + \widetilde{r}_+ \\ &= \sum_{i=-k}^{-2} a'_{i+k+1}x^i + \sum_{i=-k}^{-1} a'_{i+k}x^i \\ &\quad + \sum_{i=-k+1}^{0} a'_{i+k-1}x^i + \sum_{i=-2k+n}^{-k-1+n} a'_{i+2k-n}x^i \\ &\quad + a'_0(x^{-2} + x^{-1} + x^0 + x^{n-k-1}) \\ &\quad + \sum_{i=-k}^{n-k-1} a'_{i+2k}x^i \\ &\quad + \left(\sum_{i=-k}^{n-2k-1} a'_{i+n+2k}x^i + \sum_{i=1}^{n-k-1} a'_{i+n+k-1}x^i \right. \\ &\quad \left. + \sum_{i=0}^{n-k-1} a'_{i+n+k}x^i + \sum_{i=-1}^{n-k-2} a'_{i+n+k+1}x^i \right). \end{aligned}$$

Comparing the coefficients of x^i in this formula, we can obtain explicit expressions of coordinates c_is for $0 \le i \le n-1$. These expressions are different according to the value of i. For the case $3 < k < (n-1)/2$, these n coordinate formulae can be grouped into ten cases depending on the values of i. We note that the number of cases depends on both k and the field generating irreducible polynomial. Similar to [6] and [7], we consider only the case "n odd" in this work.

Case 1: $i = -k$

$$c_0 = a'_0 + a'_k + a'_{n+k};$$

Case 2: $-k+1 \le i \le -3$

$$c_{i+k} = a'_{i+k+1} + a'_{i+k} + a'_{i+k-1} + a'_{i+2k} + a'_{i+n+2k};$$

Case 3: $i = -2$

$$c_{k-2} = a'_{k-1} + a'_{k-2} + a'_{k-3} + a'_{2k-2} + a'_{n+2k-2} + a'_0;$$

Case 4: $i = -1$

$$c_{k-1} = a'_{k-1} + a'_{k-2} + a'_{2k-1} + a'_{n+2k-1} + a'_{n+k} + a'_0;$$

Case 5: $i = 0$

$$c_k = a'_{k-1} + a'_{2k} + a'_{n+2k} + a'_{n+k} + a'_{n+k+1} + a'_0;$$

Case 6: $1 \le i \le n-2k-2$

$$c_{i+k} = a'_{i+2k} + a'_{i+n+2k} + a'_{i+n+k-1} + a'_{i+n+k} + a'_{i+n+k+1};$$

Case 7: $i = n-2k-1$

$$c_{n-k-1} = a'_{n-1} + a'_{2n-k-2} + a'_{2n-k-1} + a'_{2n-k};$$

Case 8: $n-2k \le i \le n-k-3$

$$c_{i+k} = a'_{i+2k-n} + a'_{i+2k} + a'_{i+n+k-1} + a'_{i+n+k} + a'_{i+n+k+1};$$

Case 9: $i = n-k-2$

$$c_{n-2} = a'_{k-2} + a'_{n+k-2} + a'_{2n-2};$$

Case 10: $i = n-k-1$

$$c_{n-1} = a'_{k-1} + a'_{n+k-1} + a'_{2n-2} + a'_0.$$

These expressions can be further simplified since $a'_i = 0$ when i is odd. Therefore, we have the following explicit formulae of c_{i+k} for the case "n odd, k even":

$c_{i+k} =$

$$\begin{cases} a'_0 + a'_k, & i = -k, \\ a'_{i+k+1} + a'_{i+k-1} + a'_{i+n+2k}, & i \doteq -k+1, \ldots, -3, \\ a'_{i+k} + a'_{i+2k}, & i \doteq -k+2, \ldots, -4, \\ (a'_{k-2} + a'_0) + a'_{2k-2}, & i = -2, \\ (a'_{k-2} + a'_0) + a'_{n+2k-1}, & i = -1, \\ a'_{2k} + a'_{n+k+1} + a'_0, & i = 0, \\ a'_{i+n+2k} + a'_{i+n+k}, & i \doteq 1, \ldots, n-2k-2, \\ a'_{i+2k} + a'_{i+n+k-1} & \\ \quad + a'_{i+n+k+1}, & i \doteq 2, \ldots, n-2k-3, \\ a'_{n-1} + a'_{2n-k-2} + a'_{2n-k}, & i = n-2k-1, \\ a'_{i+2k-n} + a'_{i+n+k}, & i \doteq n-2k, \ldots, n-k-4, \\ a'_{i+2k} + a'_{i+n+k-1} & \\ \quad + a'_{i+n+k+1}, & i \doteq n-2k+1, \ldots, n-k-3, \\ a'_{k-2} + a'_{2n-2}, & i = n-k-2, \\ a'_{n+k-1} + a'_{2n-2} + a'_0 & i = n-k-1. \end{cases}$$

Since term "$(a'_{k-2} + a'_0)$" appears in both cases $i = -1$ and $i = -2$, they can be reused once. Therefore, the SPB squarer requires $\frac{3n+1}{2}$ XOR gates and its gate delay is $2T_X$.

Similarly, for the case "n odd, k odd", we have

$c_{i+k} =$

$$\begin{cases} a'_0 + a'_{n+k}, & i = -k, \\ a'_{i+k+1} + a'_{i+k-1} + a'_{i+2k}, & i \doteq -k+1, \ldots, -4, \\ a'_{i+k} + a'_{i+n+2k}, & i \doteq -k+2, \ldots, -3, \\ (a'_{k-1} + a'_0) + a'_{k-3} + a'_{2k-2}, & i = -2, \\ (a'_{k-1} + a'_0) + a'_{n+2k-1} & \\ \quad + a'_{n+k}, & i = -1, \\ (a'_{k-1} + a'_0) + a'_{2k} + a'_{n+k}, & i = 0, \\ a'_{i+n+2k} + a'_{i+n+k-1} & \\ \quad + a'_{i+n+k+1}, & i \doteq 1, \ldots, n-2k-2, \\ a'_{i+2k} + a'_{i+n+k}, & i \doteq 2, \ldots, n-2k-3, \\ a'_{n-1} + a'_{2n-k-1}, & i = n-2k-1, \\ a'_{i+2k-n} + a'_{i+n+k-1} & \\ \quad + a'_{i+n+k+1}, & i \doteq n-2k, \ldots, n-k-3, \\ a'_{i+2k} + a'_{i+n+k}, & i \doteq n-2k+1, \ldots, n-k-4, \\ a'_{n+k-2} + a'_{2n-2}, & i = n-k-2, \\ (a'_{k-1} + a'_0) + a'_{2n-2} & i = n-k-1. \end{cases}$$

Term "$(a'_{k-1} + a'_0)$" appears in cases $i = -2, -1, 0$ and $n-k-1$, so it can be reused three times. Therefore, the SPB squarer requires $\frac{3n+3}{2}$ XOR gates and its gate delay is $2T_X$.

The above two explicit formulae are derived using the shift factor $v = k$. In fact, we have also obtained explicit formulae using shift factors $k-1$ and $k+1$. The formulae using shift factor $k-1$ are the same as those of Montgomery squarers proposed in [6]. As for the complexities, the gate delays of these three cases are the same, i.e., $2T_X$, but their XOR gate numbers, which are listed in Table IV, are different.

For the case "n odd, k odd", Table IV shows that the shift factor $v = k+1$ leads to an SPB squarer of the minimal

TABLE IV
XOR GATE NUMBERS OF SPB SQUARERS FOR TYPE II IRREDUCIBLE PENTANOMIALS

	n odd, k odd	n odd, k even
$v = k$	$(3n+3)/2$	$(3n+1)/2$
$v = k-1$	$(3n+1)/2$	$(3n+3)/2$
$v = k+1$	$(3n-1)/2$	$(3n+5)/2$

XOR gate number. So we present this explicit formula in the following:

$c_{i+k+1} =$

$$\begin{cases} a'_{i+k+1} + a'_{i+2k+2} + a'_{i+k+3}, & i \doteq -k-1, \ldots, -6 \\ a'_{i+n+2k+2} + a'_{i+k+2}, & i \doteq -k, \ldots, -5, \\ (a'_{k-1} + a'_0) + a'_{k-3} + a'_{2k-2}, & i = -4, \\ a'_{n+2k-1} + a'_{k-1}, & i = -3, \\ a'_{k-1} + a'_{2k}, & i = -2, \\ a'_{n+2k+1} + a'_{n+k+2} + a'_0, & i = -1, \\ a'_{i+n+k+2} + a'_{i+2k+2}, & i \doteq 0, \ldots, n-2k-3, \\ a'_{i+n+2k+2} + a'_{i+n+k+1} & \\ \quad + a'_{i+n+k+3}, & i \doteq 1, \ldots, n-2k-4, \\ a'_{i-n+2k+2} + a'_{i+n+k+1} & \\ \quad + a'_{i+n+k+3}, & i \doteq n-2k-2, \ldots, n-k-5, \\ a'_{i+n+k+2} + a'_{i+2k+2}, & i \doteq n-2k-1, \ldots, n-k-4, \\ (a'_{k-1} + a'_0) + a'_{2n-2}, & i = n-k-3, \\ a'_0 + a'_{n+k} & i = n-k-2. \end{cases}$$

B. Comparison

Table V compares XOR gate numbers of the proposed SPB squarers to those of [6] and [7]. The Montgomery squarer of [6] uses the Montgomery factor x^{k-1}. These squarers have the same gate delays: $2T_X$. But their XOR gate numbers vary slightly.

TABLE V
COMPARISON OF BIT-PARALLEL SQUARERS FOR TYPE II IRREDUCIBLE PENTANOMIALS

	n odd, k odd	n odd, k even
Proposals	XOR gates	XOR gates
[6] Montgomery	$\leq (3n+1)/2$	$\leq (3n+5)/2$
[7] WDB	–	$\leq (3n+3)/2$
Proposed	$(3n-1)/2$	$(3n+1)/2$

V. CONCLUSIONS

We have presented explicit formulae and complexities of bit-parallel SPB squarers in $GF(2^n)$s generated by general irreducible trinomials and type-II irreducible pentanomials. Their complexities match or slightly outperform the previous best results. Owing to the equivalent relationship between $GF(2^n)$ Montgomery and SPB multiplication algorithms, these formulae can also be used to design Montgomery squarers without any change. Contrary to previous speculation, we also show that XOR gate numbers of SPB squarers are different when different SPB shift factors are used.

REFERENCES

[1] M. Piontas, "Algorithm for Squaring in $GF(2^m)$ in Standard Basis," *Electronics Letters*, vol. 25, no. 18, pp. 1262-1263, 1989.

[2] C. Paar, P. Fleischmann and P. Soria-Rodriguez, "Fast Arithmetic for Public-Key Algorithms in Galois Fields with Composite Exponents," *IEEE Trans. Comput.*, vol. 48, no. 10, pp. 1025-1034, 1999.

[3] H. Wu, "Low Complexity Bit-Parallel Finite Field Arithmetic Using Polynomial Basis," *Proc. First Int'l Workshop Cryptographic Hardware and Embedded Systems (CHES)*, LNCS-1717, pp. 280-291, 1999.

[4] H. Wu, "bit-parallel Finite Field Multiplier and Squarer Using Polynomial Basis," *IEEE Trans. Comput.*, vol. 51, no. 7, pp. 750-758, July 2002.

[5] H. Wu, "Montgomery Multiplier and Squarer for a Class of Finite Fields," *IEEE Trans. Comput.*, vol. 51, no. 5, pp. 521-529, May 2002.

[6] A. Hariri and A. Reyhani-Masoleh, "Bit-Serial and Bit-Parallel Montgomery Multiplication and Squaring over $GF(2^m)$," *IEEE Trans. Comput.*, vol. 58, no. 10, pp. 1332-1345, OCT, 2009.

[7] S.M. Park, "Explicit Formulae of Polynomial Basis Squarer for Pentanomials Using Weakly Dual Basis," *Integration, the VLSI journal*, vol. 45, no. 2, pp. 205-210, 2012.

[8] H. Wu and M.A. Hasan, "Efficient Exponentiation of a Primitive Root in $GF(2^m)$," *IEEE Trans. Comput.*, vol. 46, no. 2, pp. 162-172, 1997.

[9] K.U. Järvinen, "On Repeated Squarings in Binary Fields," *Proc. SAC 2009,*, LNCS-5867, pp. 331-349, 2009.

[10] H. Fan and M.A. Hasan, "Relationship between $GF(2^m)$ Montgomery and Shifted Polynomial Basis Multiplication Algorithms," *IEEE Trans. Comput.*, vol. 55, no. 9, pp. 1202-1206, Sept., 2006.

[11] H. Fan and Y. Dai, "Fast bit-parallel $GF(2^n)$ Multiplier for All Trinomials," *IEEE Trans. Comput.*, vol. 54, no. 4, pp. 485-490, Apr., 2005.