Low Complexity Bit Parallel Architectures for Polynomial Basis Multiplication over $GF(2^m)$

Arash Reyhani-Masoleh, *Member*, *IEEE*, and M. Anwar Hasan, *Senior Member*, *IEEE*

Abstract—Representing the field elements with respect to the polynomial (or standard) basis, we consider bit parallel architectures for multiplication over the finite field $GF(2^m)$. In this effect, first we derive a new formulation for polynomial basis multiplication in terms of the *reduction* matrix $\mathbf{Q}$. The main advantage of this new formulation is that it can be used with any field defining irreducible polynomial. Using this formulation, we then develop a generalized architecture for the multiplier and analyze the time and gate complexities of the proposed multiplier as a function of degree m and the *reduction* matrix $\mathbf{Q}$. To the best of our knowledge, this is the first time that these complexities are given in terms of $\mathbf{Q}$. Unlike most other articles on bit parallel finite field multipliers, here we also consider the number of signals to be routed in hardware implementation and we show that, compared to the well-known Mastrovito's multiplier, the proposed architecture has fewer routed signals. In this article, the proposed generalized architecture is further optimized for three special types of polynomials, namely, equally spaced polynomials, trinomials, and pentanomials. We have obtained explicit formulas and complexities of the multipliers for these three special irreducible polynomials. This makes it very easy for a designer to implement the proposed multipliers using hardware description languages like VHDL and Verilog with minimum knowledge of finite field arithmetic.

Index Terms—Finite or Galois field, Mastrovito multiplier, all-one polynomial, polynomial basis, trinomial, pentanomial and equally-spaced polynomial.

1 INTRODUCTION

With the rapid expansion of the Internet and wireless communications, more and more digital systems are becoming increasingly equipped with some form of cryptosystems to provide various kinds of data security. Many such cryptosystems rely on computations in very large finite fields and require fast computations in the fields [14], [2]. Finite field arithmetic operations are also used in error control coding [11], [16], VLSI testing [6], [27], and digital signal processing [5]. Among the basic arithmetic operations over the finite field $GF(2^m)$, addition is easily realized using m two-input XOR gates, while multiplication is costly in terms of gate count and time delay. The other operations of finite fields, such as exponentiation, division, and inversion can be performed by repeated multiplications [21], [26], [1], [7]. In order to satisfy the high speed requirements of many such applications, there is a need to develop an efficient architecture for finite field multiplication which is suitable for VLSI implementation. In this paper, a new general bit parallel structure for the polynomial basis multiplication which is applicable to all types of irreducible binary polynomials is proposed.

1.1 Summary of Previous Work

The earliest parallel polynomial basis (PB) multiplier over $GF(2^m)$ was suggested by Bartee and Schneider [3]. Depending on the irreducible polynomial, this implementation requires as many as $m^3 - m$ two-input adders over $GF(2)$ (i.e., XOR gates) [4]. Because of its high circuit complexity and lack of regularity, it is often advantageous to use other hardware structures to implement the multiplier [16]. In [13], [12], Mastrovito has proposed an algorithm along with its hardware architecture (hereafter referred to as the Mastrovito algorithm/multiplier) for PB multiplication. Sunar and Koc [24] have presented a new formulation for the Mastrovito algorithm using trinomials and have shown that $m^2 - 1$ XOR and m^2 AND gates are sufficient to implement the multiplier. In [8], Halbutogullari and Koc have generalized the approach of Sunar and Koc and have found a method for constructing the Mastrovito multiplier for arbitrary irreducible polynomials. This method considers general as well as special classes of irreducible polynomials such as trinomials, all-one polynomials (AOPs) and equally spaced polynomials (ESPs). So far, for these special polynomials, the XOR gate count and time delay of the Halbutogullari-Koc algorithm appear to be the lowest. In [28], Zhang and Parhi propose a systematic method to design the Mastrovito multiplier. Moreover, they extend the method to systematically design the modified Mastrovito multiplication scheme proposed in [23]. They also present new results of the complexities of the Mastrovito multiplier for two classes of irreducible pentanomials.

Unlike Mastrovito's method, a $GF(2^m)$ multiplication can also be performed by a straightforward polynomial multiplication followed by modular reduction. This approach has been used in a number of papers. For

- *A. Reyhani-Masoleh is with the Centre for Applied Cryptographic Research, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. E-mail: areyhani@math.uwaterloo.ca.*
- *M.A. Hasan is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N21 3G1. E-mail: ahasan@ece.uwaterloo.ca.*

Manuscript received 21 July 2003; accepted 15 Jan. 2004.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-0090-0703.

0018-9340/04/$20.00 © 2004 IEEE Published by the IEEE Computer Society

example, in [25], Wu considered irreducible trinomials as reduction polynomials and showed that a modular multiplication operation in $GF(2^m)$ can be performed with $(\omega - 1)(m-1)$ bit additions, where ω is the Hamming weight of the irreducible polynomial. In hardware implementation, its multiplication operations can be realized with m^2 AND and $(m-1)^2 + (\omega - 1)(m-1)$ XOR gates. Recently, Rodriguez-Henriquez and Koc in [20] proposed a PB multiplier for special case of pentanomials and have obtained its time delay and gate count. Although they have referred to it as the Mastrovito multiplier, their architecture is different from the original Mastrovito multiplier and uses the two steps of multiplication separately.

1.2 Scope of Our Work

In this paper, we present a new formulation for polynomial basis multiplication and then a generalized bit-parallel hardware architecture. We consider the time delay and gate count of the proposed multiplier as a function of degree m and the *reduction* matrix $\mathbf{Q}$. Using the $\mathbf{Q}$ matrix, the complexities of multipliers based on special reduction polynomials, namely: 1) trinomials, 2) ESPs, and 3) two classes of pentanomials are obtained. We also present explicit formulas for multiplication for the above three special classes. These formulas maximize the number of intermediate signals that are reused. These formulas can be easily coded using hardware description languages such as VHDL or Verilog to implement an optimized multiplier. These codings can be done by a hardware designer without running an algorithm for precomputation or even having any knowledge of finite field arithmetic. In this paper, we also show that, for general irreducible polynomials, both the time delay and gate count of the proposed structures are, overall, lower than those available in the literature. Furthermore, these architectures have fewer routed signals and are suitable for VLSI implementation.

The organization of this paper is as follows: In Section 2, polynomial basis multiplication over $GF(2^m)$ and the Mastrovito multiplier in particular are considered. The new architecture and its complexities are introduced in Section 3. In Sections 4, 5, 6, and 7, optimized multiplication schemes using irreducible equally spaced polynomials, generic polynomials, trinomials, and pentanomials are respectively considered and comparisons between our architectures and other PB multipliers are made. Finally, conclusions are given in Section 8.

2 Polynomial Basis Multiplication over $GF(2^m)$

Let $P(x) = x^m + \sum_{i=0}^{m-1} p_i x^i$ be a monic irreducible polynomial over $GF(2)$ of degree m, where $p_i \in GF(2)$ for $i = 0, 1, \cdots, m-1$. Let $\alpha \in GF(2^m)$ be a root of $P(x)$, i.e., $P(\alpha) = 0$. Then, the set $\{1, \alpha, \alpha^2, \cdots, \alpha^{m-1}\}$ is referred to as the polynomial or standard basis and each element of $GF(2^m)$ can be written with respect to the polynomial basis (PB). Let A be an element in $GF(2^m)$, then the representation of A w.r.t. the PB is $A = \sum_{i=0}^{m-1} a_i \alpha^i$, $a_i \in \{0, 1\}$, where a_is are the coordinates. For convenience, these coordinates will be denoted in vector notation[1] as $\mathbf{a} = [a_0, a_1, a_2, \cdots, a_{m-1}]^T$, where T denotes the transposition. Using this vector notation, the representation of A can be written as $A = \boldsymbol{\alpha}^T \mathbf{a}$, where $\boldsymbol{\alpha} = [1, \alpha, \alpha^2, \cdots, \alpha^{m-1}]^T$. Let S be the binary polynomial of degree not more than $2m-2$ obtained by the direct multiplication of the PB representations of any two elements A and B of $GF(2^m)$, i.e.,

$$S = \left(\sum_{i=0}^{m-1} a_i \alpha^i\right)\left(\sum_{j=0}^{m-1} b_j \alpha^j\right) = \sum_{k=0}^{2m-2} s_k \alpha^k, \qquad (1)$$

where

$$s_k = \sum_{i+j=k} a_i b_j, \quad 0 \le i, j \le m-1, \ 0 \le k \le 2m-2. \qquad (2)$$

Then, the product $C = A \cdot B$ can be obtained by the following modulo reduction:

$$C \triangleq \sum_{i=0}^{m-1} c_i \alpha^i \equiv S \mod P(\alpha). \qquad (3)$$

Using (1) and (3), the product coordinates, i.e., c_is, are obtained in terms of a_is, b_is, and the irreducible polynomial $P(x)$. In [12], Mastrovito shows that these coordinates can be calculated using a matrix equation as follows:

$$\mathbf{c} = \mathbf{F}\mathbf{b}, \qquad (4)$$

where $\mathbf{b} = [b_0, b_1, \cdots, b_{m-1}]^T$ and $\mathbf{c} = [c_0, c_1, \cdots, c_{m-1}]^T$ are the vectors associated with B and C, respectively. The exact definition of the *product matrix* $\mathbf{F} = [f_{i,j}]_{i,j=0}^{m-1}$ can be found in [13].

Remark 1. *Matrix* $\mathbf{F}$ *is unique and depends on the multiplicand A and the irreducible polynomial $P(x)$.*

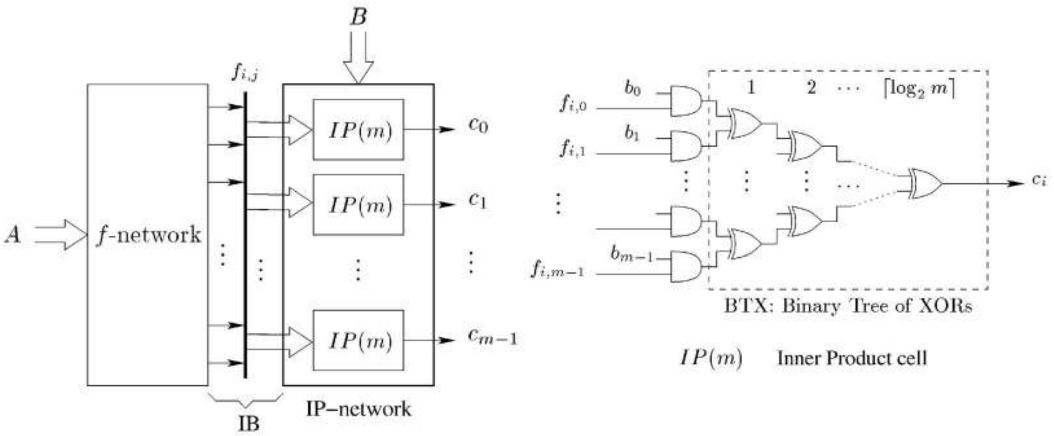
Using (4), an architecture for the Mastrovito multiplier is shown in Fig. 1a, which basically consists of two blocks, namely, *f*-network and IP-network. The *f*-network generates the entries of the product matrix $\mathbf{F}$. The IP-network performs the matrix-vector multiplication as shown in (4) and consists of m inner product units, each generating one coordinate of C, i.e.,

$$c_i = [f_{i,0}, f_{i,1}, \cdots, f_{i,m-1}][b_0, b_1, \cdots, b_{m-1}]^T, \ 0 \le i \le m-1.$$

In Fig. 1a, block $IP(m)$ corresponds to an inner product unit which has two input vectors of m elements each. Assuming that only two-input logic gates are used, $IP(k)$ for $k > 0$, requires k AND gates and $k-1$ XOR gates and has a gate delay of $T_A + \lceil \log_2 k \rceil T_X$, where T_A and T_X correspond to the delays due to an AND and an XOR gate respectively (see Fig. 1b where $k = m$).

In Fig. 1a, there are two buses: the coordinates of B and the interconnection bus IB which contains the coordinates of A. The interconnection bus IB carries the elements $f_{i,j}$ of $\mathbf{F}$ from the *f*-network to the IP-network. The number of lines on IB depends on the irreducible polynomial $P(x)$ and varies between $2m-1$ (for trinomials) and $\frac{m(m+1)}{2}$ (for AOPs) [13].

[1] In this paper, vectors and matrices are shown with small and capital bold faces, respectively.



(a) (b)

Fig. 1. (a) Architecture of the Mastrovito multiplier over $GF(2^m)$. (b) Details of $IP(m)$.

3 An Efficient Multiplication Scheme

In this section, we first give a new formulation for multiplication over $GF(2^m)$. Using this formulation, we then present a bit parallel architecture for the multiplier. At the end, we give upper bounds of the space and time complexities of the architecture.

3.1 New Formulation

Definition 1 [12]. *The* reduction matrix $\mathbf{Q}$ *is an* $m-1$ *by* m *binary matrix which is obtained from*

$$\boldsymbol{\alpha}^{\uparrow} \equiv \mathbf{Q}\boldsymbol{\alpha} \ (\mathrm{mod}\ P(\alpha)), \tag{5}$$

where $\boldsymbol{\alpha}^{\uparrow} = [\alpha^m,\ \alpha^{m+1},\ \cdots,\ \alpha^{2m-2}]^T$.

Remark 2. *For each irreducible* $P(x)$, *the reduction matrix* $\mathbf{Q}$ *is unique.*

In order to present our new multiplication scheme, we introduce the following two Toeplitz matrices

$$\mathbf{L} \triangleq \begin{bmatrix} a_0 & 0 & 0 & 0 & \cdots & 0 \\ a_1 & a_0 & 0 & 0 & \cdots & 0 \\ a_2 & a_1 & a_0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{m-2} & a_{m-3} & \cdots & a_1 & a_0 & 0 \\ a_{m-1} & a_{m-2} & \cdots & a_2 & a_1 & a_0 \end{bmatrix},$$
$$\mathbf{U} \triangleq \begin{bmatrix} 0 & a_{m-1} & a_{m-2} & \cdots & a_2 & a_1 \\ 0 & 0 & a_{m-1} & \cdots & a_3 & a_2 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & a_{m-1} & a_{m-2} \\ 0 & 0 & \cdots & 0 & 0 & a_{m-1} \end{bmatrix}, \tag{6}$$

where a_is are the coordinates of A. Note that $\mathbf{L}$ is an $m \times m$ lower triangular matrix and $\mathbf{U}$ is an $(m-1) \times m$ upper triangular matrix. Now, define the following two vectors which are functions of A and B:

$$\mathbf{d} = \mathbf{L}\mathbf{b}, \tag{7}$$

$$\mathbf{e} = \mathbf{U}\mathbf{b}. \tag{8}$$

Then, we can state the following theorem, which is the key step toward the development a new architecture for the PB multiplication in $GF(2^m)$.

Theorem 1. *Let* C *be the product of* A *and* $B \in GF(2^m)$. *Then,*

$$\mathbf{c} = \mathbf{d} + \mathbf{Q}^T\mathbf{e}, \tag{9}$$

where $\mathbf{Q}$, $\mathbf{d}$, *and* $\mathbf{e}$ *are defined in (5), (7), and (8) respectively.*

Proof. In vector notation, (1) can be written as

$$S = \boldsymbol{\alpha}^{\uparrow\uparrow^T}\mathbf{s}, \tag{10}$$

where

$$\boldsymbol{\alpha}^{\uparrow\uparrow} = [1,\ \alpha,\ \cdots,\ \alpha^{2m-2}]^T = \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\alpha}^{\uparrow} \end{bmatrix}$$

and $\mathbf{s} = [s_0,\ s_1, \cdots,\ s_{2m-2}]^T$. Using (5), we have

$$\boldsymbol{\alpha}^{\uparrow\uparrow} \equiv \begin{bmatrix} \mathbf{I}_m \\ \mathbf{Q} \end{bmatrix}\boldsymbol{\alpha},$$

where $\mathbf{I}_m$ is the m by m unity matrix. Note that $d_k = s_k,\ 0 \le k \le m-1,$ and $e_l = s_{l+m},\ 0 \le l \le m-2,$ then

$$\mathbf{s} = \begin{bmatrix} \mathbf{d} \\ \mathbf{e} \end{bmatrix} = \begin{bmatrix} \mathbf{L} \\ \mathbf{U} \end{bmatrix}\mathbf{b}$$

and, using (10), C is obtained as

$$\begin{aligned} C &\equiv S\ (\mathrm{mod} P(\alpha)) \\ &= \left(\begin{bmatrix} \mathbf{I}_m \\ \mathbf{Q} \end{bmatrix}\boldsymbol{\alpha}\right)^T \begin{bmatrix} \mathbf{L} \\ \mathbf{U} \end{bmatrix}\mathbf{b} = \boldsymbol{\alpha}^T\left([\mathbf{I}_m \quad \mathbf{Q}^T]\begin{bmatrix} \mathbf{L} \\ \mathbf{U} \end{bmatrix}\right)\mathbf{b} \\ &= \boldsymbol{\alpha}^T(\mathbf{L} + \mathbf{Q}^T\mathbf{U})\mathbf{b}. \end{aligned} \tag{11}$$

Since $C = \boldsymbol{\alpha}^T\mathbf{c}$, (11) yields (9) and the proof is complete.□

3.2 Architecture

Using the formulation presented in the previous section, an architecture for polynomial basis multiplication over $GF(2^m)$ is shown in Fig. 2. This structure is hereafter

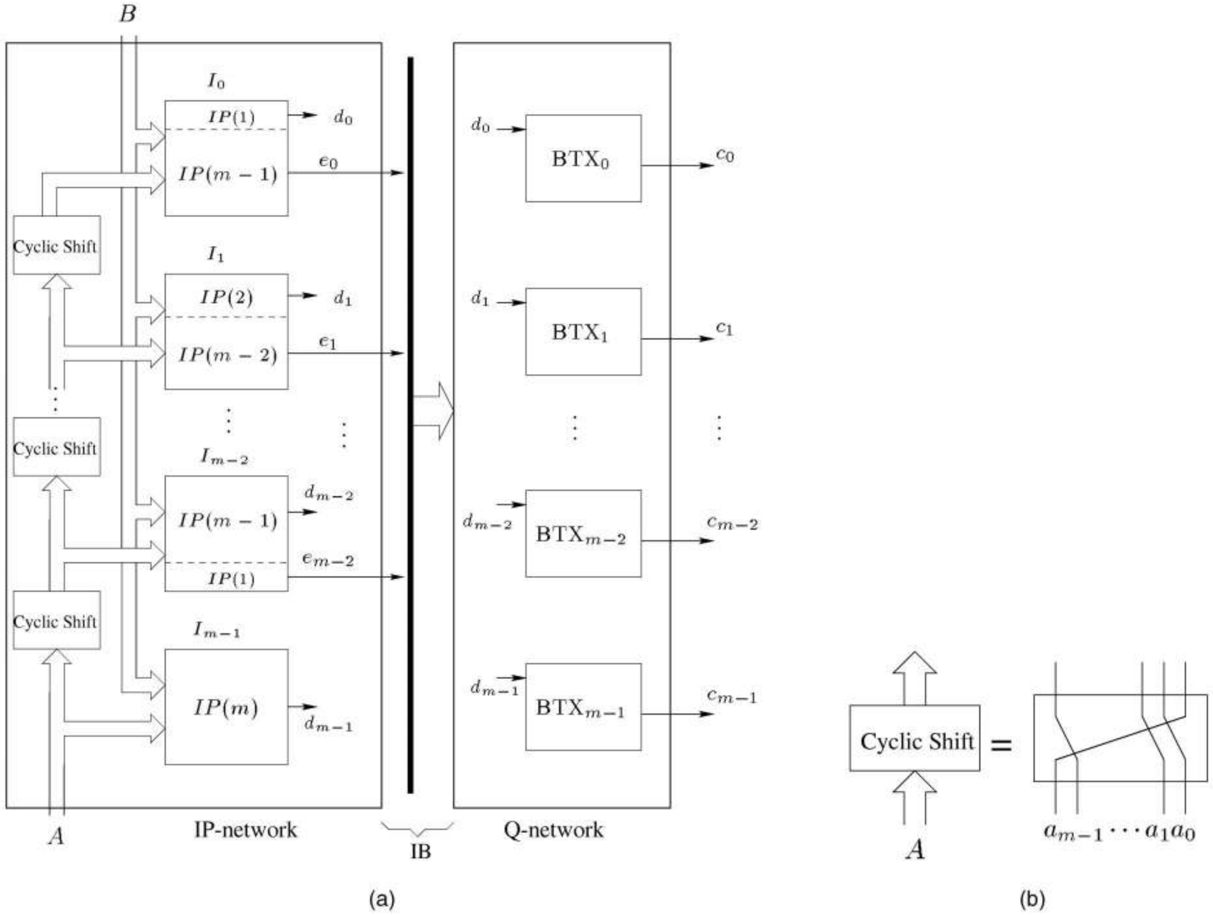


Fig. 2. (a) Architecture of the LCBP multiplier over $GF(2^m)$. (b) Detail of cyclic shift block.

referred to as the low complexity bit parallel (LCBP) multiplier. It is divided into two parts: IP-network and **Q**-network. The IP-network, which has m blocks (denoted as $I_0,\ I_1,\ \cdots,\ I_{m-1}$), generates vectors $\mathbf{d}$ and $\mathbf{e}$ in accordance with (7) and (8). For $0 \leq i \leq m-2$, block I_i consists of two inner product cells, namely, $IP(i+1)$ and $IP(m-i-1)$; however, the last block I_{m-1} consists of only one such cell, namely, $IP(m)$.

In Fig. 2, the **Q**-network takes $\mathbf{d}$ and $\mathbf{e}$ as inputs and generates $\mathbf{c}$. It consists of m binary trees of XOR gates ($\mathrm{BTX}_{0\cdots m-1}$). The number of XOR gates in $\mathrm{BTX}_i,\ 0 \leq i \leq m-1,$ is equal to the number of 1s in the ith column of the **Q** matrix. It is noted that the number of lines on the interconnection bus IB is fixed and is equal to the number of e_js, i.e., $m-1$. In Fig. 2a, there are three buses, A, B, and IB, and the number of lines on these buses is $3m-1$.

In order to illustrate the new multiplier structure, we consider the finite field of $GF(2^4)$ constructed by the irreducible polynomial $P(x) = x^4 + x^3 + 1$. For this field, the circuit diagram based on the new multiplier structure is shown in Fig. 3. The total number of XOR gates of this figure can be reduced by reusing signals. This is considered later in this paper for special irreducible polynomials.

3.3 Complexities

For the LCBP multiplier structure shown in Fig. 2, we now give its complexities, in terms of gate counts and time delay due to gates. For this purpose, let $\mathbf{q}_j,\ 0 \leq j \leq m-1$, be the jth column of the reduction matrix, i.e.,

$$\mathbf{Q} = [\mathbf{q}_0,\ \mathbf{q}_1,\ \cdots,\ \mathbf{q}_{m-1}]$$

and $H(\mathbf{q}_j)$ be the Hamming weight (i.e., the number of 1s) of $\mathbf{q}_j$. We denote θ as the maximum Hamming weight of a column of **Q**, i.e.,

$$\theta = \max\{H(\mathbf{q}_j) :\ \ 0 \leq j \leq m-1\} \tag{12}$$

and $H(\mathbf{Q})$ as the Hamming weight of **Q**, i.e.,

$$H(\mathbf{Q}) = \sum_{j=0}^{m-1} H(\mathbf{q}_j). \tag{13}$$

Now, consider the IP-network of the multiplier in Fig. 2. Since each I_i for $0 \leq i \leq m-2$ has m AND and $(m-2)$ XOR gates and I_{m-1} has m AND and $(m-1)$ XOR gates, the IP-network has a total of m^2 AND gates and $(m-1)(m-2)+m-1=(m-1)^2$ XOR gates. For the **Q**-network, using (13), one can determine the maximum number of XOR gates needed as $H(\mathbf{Q})$.

To determine the time complexity of this architecture, we need to consider the time delays due to gates of the IP as well as **Q**-networks. Using (7) and (8), the delays for $d_j,\ 0 \leq j \leq m-1$, and $e_i,\ 0 \leq i \leq m-2,$ are given as

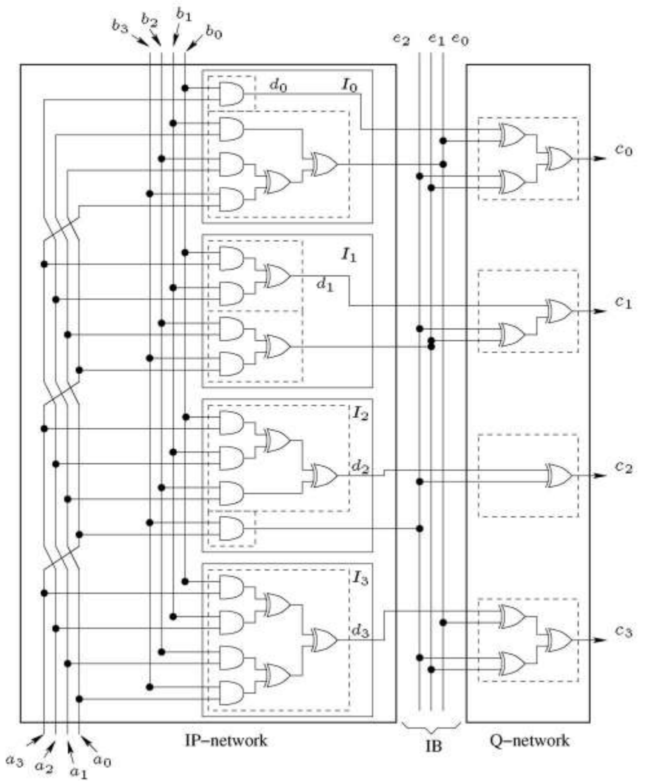


Fig. 3. Architecture for the $GF(2^4)$ multiplier with $P(x) = x^4 + x^3 + 1$.

$$T(d_j) = T_A + \lceil \log_2 (j+1) \rceil T_X, \qquad 0 \leq j \leq m-1, \quad (14)$$

$$T(e_i) = T_A + \lceil \log_2 (m-i-1) \rceil T_X, \quad 0 \leq i \leq m-2, \quad (15)$$

respectively. In the IP-network, the maximum gate delay is due to the I_{m-1} cell and is equal to $T_A + \lceil \log_2 m \rceil T_X$. Using (12), it is not difficult to see that the maximum gate delay in the **Q**-network is $\lceil \log_2 (\theta + 1) \rceil T_X$. In the worst case, a signal will have maximum delays of both the IP and **Q**-networks. Thus, an upper bound for the time delay of the entire multiplier structure is $T_C \leq T_A + (\lceil \log_2 m \rceil + \lceil \log_2 (\theta + 1) \rceil) T_X$. The following theorem summarizes the above results on the complexities of the proposed multiplier structure.

Theorem 2. *For the LCBP multiplier, the number of two-input AND gates is*

$$N_A = m^2 \quad (16)$$

and the number of XOR gates and time delay due to gates are upper bounded by

$$N_X \leq (m-1)^2 + H(\mathbf{Q}), \quad (17)$$

$$T_C \leq T_A + (\lceil \log_2 m \rceil + \lceil \log_2 (\theta + 1) \rceil) T_X. \quad (18)$$

The above theorem gives upper bounds for the number of XOR gates and time delay. However, the exact values can be obtained by designing a multiplier which is either highly space efficient or very fast. In order to minimize the number of XOR gates, the intermediate signals can be reused. This is illustrated in the following example.

3.4 An Example

We consider the field $GF(2^7)$ defined by the irreducible polynomial $P(x) = x^7 + x^5 + x^3 + x + 1$ for which gate and time complexities have been reported in [8]. For this irreducible polynomial, one has

$$\mathbf{Q} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}. \quad (19)$$

Since $H(\mathbf{Q}) = 20$ and $\theta = 4$, then, using (17) and (18), the upper bounds of XOR gate count and time delay are $N_X \leq (7-1)^2 + 20 = 56$ and

$$T_C \leq T_A + (\lceil \log_2 7 \rceil + \lceil \log_2 5 \rceil) T_X = T_A + 6T_X,$$

respectively.

Substituting (19) into Theorem 1, the coordinates of the product $C = AB$ over $GF(2^7)$ can be obtained as

$$\begin{aligned} c_0 &= d_0 + e_0 + e_2 \\ c_1 &= (d_1 + e_0) + (e_1 + (e_2 + e_3)) \\ c_2 &= (d_2 + e_4) + (e_1 + (e_2 + e_3)) \\ c_3 &= (d_3 + e_0) + (e_3 + (e_4 + e_5)) \\ c_4 &= d_4 + (e_1 + (e_4 + e_5)) \\ c_5 &= d_5 + e_0 + e_5 \\ c_6 &= d_6 + e_1, \end{aligned} \quad (20)$$

where d_j, $0 \leq j \leq 6$ and e_i, $0 \leq i \leq 5$ are from (7) and (8), respectively. Note that the brackets in (20) show the order of modulo two addition which defines the position of XOR gates in the **Q**-network. Since we reuse partial sums $(e_1 + (e_2 + e_3))$ and $(e_4 + e_5)$ in (20), for the realization of (20), 17 XOR gates are needed in the **Q**-network and the total number of XOR gates of the entire multiplier is $(7-1)^2 + 17 = 53$. Also, since the time delays of d_j, $0 \leq j \leq 6$ and e_i, $0 \leq i \leq 5$ are $T_A + \lceil \log_2 (j+1) \rceil T_X$ and $T_A + \lceil \log_2 (6-i) \rceil T_X$, respectively, the time delay of the entire multiplier is $T_C = T_A + 5T_X$.

In the following sections, we attempt to minimize the number of XOR gates for special irreducible polynomials, namely, equally spaced polynomials, trinomials, and pentanomials. The LCBP multipliers for the above-mentioned irreducible polynomials are achieved by properly defining some intermediate signals and then reusing them as much as possible. We start with equally spaced polynomials which are very structured and will help us present the remaining special cases with fewer difficulties.

4 Multipliers Using Equally Spaced Polynomials

Defintion 2. *A polynomial*

$$P(x) = x^{ns} + x^{(n-1)s} + \cdots + x^s + 1, \quad (21)$$

over $GF(2)$, with $ns = m$, is called an equally spaced polynomial (denoted as s-ESP) of degree m.

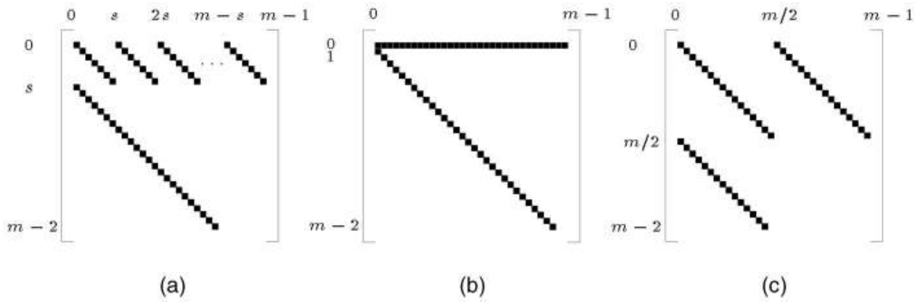


Fig. 4. Graphical representation of the locations of nonzero entries of $\mathbf{Q}$ for s-ESP $P(x) = x^{ns} + x^{(n-1)s} + \cdots + x^s + 1$, $m = ns$. (a) $1 < s < \frac{m}{2}$. (b) $s = 1$ (AOP). (c) $s = \frac{m}{2}$ (trinomial).

TABLE 1
Comparison of Related s-ESP-Based Polynomial Basis Multipliers

Reference	#AND	#XOR	Time delay
Itoh-Tsujii [10]	$(m+s)^2$	$(m+s)^2 - s$	$T_A + (\lceil \log_2 m \rceil + \lceil \log_2 (m+s+1) \rceil) T_X$
Hasan et al. [9]	m^2	$m^2 + m - 2s$	$T_A + \left(\frac{m}{s} + \lceil \log_2 m \rceil\right) T_X$
Mastrovito [12, 13]	m^2	$\frac{2s+1}{2s} m^2 - \frac{3}{2} m$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$
Halbutogullari-Koc [8]	m^2	$m^2 - s$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$
Zhang-Parhi [28]	m^2	$m^2 - s$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$
Presented Here	m^2	$m^2 - s$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$

An s-ESP is a self-reciprocal polynomial. In (21), both n and s are integers and $1 \le s \le \frac{m}{2}$. When $s = 1$, we have 1-ESP and it is the same as the all-one polynomial (AOP). The latter has the highest Hamming weight among all polynomials of degree m. On the other hand, $s = \frac{m}{2}$ results in the least Hamming weight irreducible polynomial (i.e., trinomial) of degree m.

Theorem 3. *For an s-ESP based LCBP multiplier over* $GF(2^m)$, *the gate counts, time delay, and number of lines on the buses are* $N_A = m^2$, $N_X = m^2 - s$, $T_C = T_A + (1 + \lceil \log_2 m \rceil) T_X$, *and* $N_L = 2m + s$, *respectively.*

Proof. When α is a root of the s-ESP in (21), we have

$$\alpha^{m+i} = \begin{cases} \alpha^i + \alpha^{s+i} + \cdots + \alpha^{(n-1)s+i}, & 0 \le i < s, \\ \alpha^{i-s}, & s \le i \le m-2. \end{cases} \tag{22}$$

Using (22), the reduction matrix $\mathbf{Q}$ is obtained as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{I}_s & \mathbf{I}_s & \cdots & \mathbf{I}_s \\ \mathbf{I}_{m-s-1} & & & \mathbf{0}_{s+1} \end{bmatrix}, \tag{23}$$

where $\mathbf{I}_j$ is the $j \times j$ unity matrix and $\mathbf{0}_{s+1}$ is a zero matrix which has $m - s - 1$ rows and $s + 1$ columns.

The graphical representations of $\mathbf{Q}$ in (23) for different values of s are shown in Fig. 4. In this figure, nonzero entries of $\mathbf{Q}$ are shown with the small squares.

In order to obtain exact expressions for N_X and T_C, first we attempt to obtain the coordinates of C. Using (23) into Theorem 1, one can write

$$c_j = d'_j + e_{j \bmod s}, \quad 0 \le j \le m-1, \tag{24}$$

where

$$d'_j = \begin{cases} d_j + e_{j+s} & 0 \le j \le m-s-2, \\ d_j & m-s-1 \le j \le m-1. \end{cases} \tag{25}$$

Thus, using (24) and (25), the exact XOR gate count for an s-ESP based multiplier is $N_X = m^2 - s$. Referring to Fig. 2, note that the gate delays to generate d_j, $0 \le j \le m-1$, and e_i, $0 \le i \le m-2$, are $T_A + \lceil \log_2 (j+1) \rceil T_X$, and $T_A + \lceil \log_2 (m-i-1) \rceil T_X$, respectively. Thus, d'_j of (25) can be generated with a maximum delay of $T_A + \lceil \log_2 m \rceil T_X$. Although, this changes the architecture for the LCBP multiplier slightly, now each c_j, $0 \le j \le m-1$, has a maximum delay of $T_A + (1 + \lceil \log_2 m \rceil) T_X$.

It is worth mentioning that the resultant number of bus lines on IB reduces from $m - 1$ to s. This corresponds to e_0 up to e_{s-1} as used in (24). It is noted that e_j for $s \le j \le m-2$ is not considered as a bus line because it is used only once in the multiplication formulations, i.e., (24) and (25). Thus, the total number of lines on the buses for the multiplier is $2m + s$. □

Table 1 compares the proposed ESP-based multiplier with a number of existing multipliers of the same kind. As seen in the table, our gate count and time delay match the best ones available in the literature.

5 Extension to More Generic Polynomials

Here, we consider irreducible polynomials of the form $P(x) = x^m + x^{k_t} + \cdots + x^{k_2} + x^{k_1} + 1$, where $1 \le k_1 < k_2 < \cdots < k_t \le \frac{m}{2}$. The Hamming weight of $P(x)$ is $t + 2$ and the degree of the second leading term is less than or equal to $\frac{m}{2}$. All five binary fields recommended by NIST for ECDSA can be constructed by such irreducible polynomials [15].

In order to apply the general formulation stated in Section 3 to these polynomials, first we obtain the corresponding $\mathbf{Q}$ matrix. Note that all the rows of the $\mathbf{Q}$ matrix are the PB representations of α^{m+i}, $0 \le i \le m-2$, where α is a root of $P(x)$. Since $P(\alpha) = 0$, then $\alpha^m = 1 + \alpha^{k_1} + \alpha^{k_2} + \cdots + \alpha^{k_t}$. Thus, row 0 of $\mathbf{Q}$ has 1s in

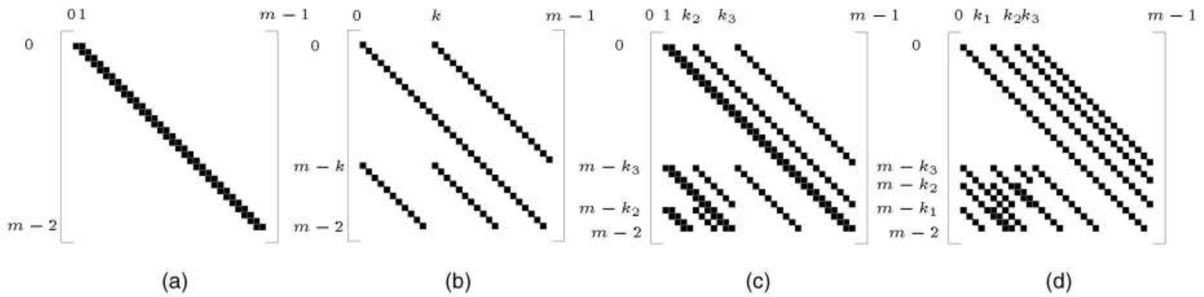


Fig. 5. Graphical representations of the reduction matrix $\mathbf{Q}$ for a trinomial ($t=1$): (a) $k=k_1=1$, (b) $1<k<\frac{m}{2}$ (see Fig. 4c for $k_1=\frac{m}{2}$), and a pentanomial ($t=3$): (c) $k_1=1$, (d) $1<k_1<k_2<k_3\leq\frac{m}{2}$.

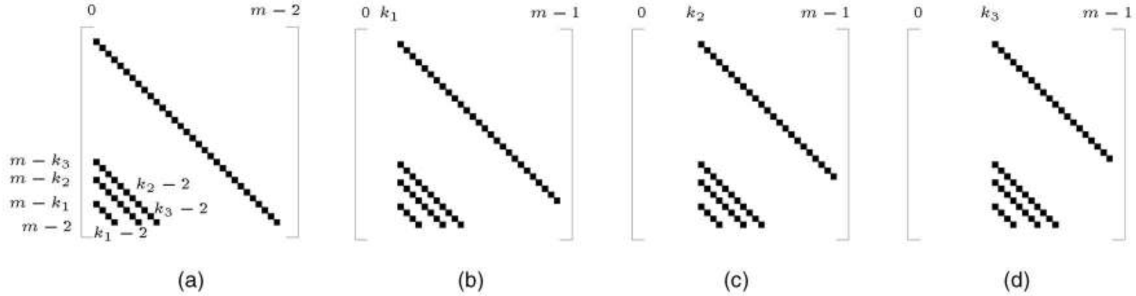


Fig. 6. Graphical representations of submatrices of $\mathbf{Q}=\mathbf{Q}_0+\mathbf{Q}_1+\mathbf{Q}_2+\mathbf{Q}_3$ for pentanomials $P(x)=x^m+x^{k_3}+x^{k_2}+x^{k_1}+1$, where $1<k_1<k_2<k_3\leq\frac{m}{2}$ (see Fig. 5d for $\mathbf{Q}$). (a) $\mathbf{Q}_0$, (b) $\mathbf{Q}_1$, (c) $\mathbf{Q}_2$, (d) $\mathbf{Q}_3$.

these $t+1$ columns: 0, k_1, k_2, $\cdots$, k_t. The consecutive rows of this matrix can be obtained by using a linear feedback shift register (LFSR). As a result, the number of 1s in rows 0 to $m-k_t-1$ is $t+1$. For the purpose of illustration, the $\mathbf{Q}$ matrices for $t=1$ and $t=3$ (i.e., trinomials and pentanomials, respectively) are shown in Fig. 5.

As shown above, row i, $0\leq i\leq m-k_t-1$ of $\mathbf{Q}$ has $t+1$ 1s which correspond to the $t+1$ segmented lines. The last column of $\mathbf{Q}$ contains 1 in rows $i=m-k_j-1$, $j=t,\cdots,2,1$. When a row ends with a 1, the following row originates new $t+1$ lines in columns: 0, k_1, k_2, up to k_t, provided that there are no previous lines that pass these columns. If there exists a previous line that passes the column of k_j, $1\leq j\leq t$, then the previous line terminates in column k_j-1 and no new line originates from column k_j due to XORing of two lines. This happens in row $\frac{m}{2}$ and column $\frac{m}{2}$ in Fig. 4c for trinomials when $k_1=\frac{m}{2}$. This is also the case for pentanomials where $t=3$ and it is shown in Fig. 5c and Fig. 5d for $k_1=1$ and $1<k_1\leq\frac{m}{2}$, respectively.

We divide the lines of $\mathbf{Q}$ into $t+1$ sets (see Fig. 6 for $t=3$) such that $\mathbf{Q}=\mathbf{Q}_0+\mathbf{Q}_1+\mathbf{Q}_2+\cdots+\mathbf{Q}_t$ where nonzero entries of $\mathbf{Q}_i$, $0\leq i\leq t$ start from column k_i (assume that $k_0=0$). It is noted that the last nonzero entry of submatrix $\mathbf{Q}_i$, $1\leq i\leq t$ is in column $m-1$, whereas the one in $\mathbf{Q}_0$ is in column $m-2$. Moreover, the number of 1s in each column of $\mathbf{Q}_i$, $0\leq i\leq t$ is at most $t+1$ if $k_1>1$ and t if $k_1=1$.

Theorem 4. *The number of XOR gates, time delay, and the number of lines on the buses of the multiplier based on the irreducible polynomial $P(x)=x^m+x^{k_t}+\cdots+x^{k_2}+x^{k_1}+1$, $1\leq k_1<k_2<\cdots<k_t\leq\frac{m}{2}$ are $N_X=(m+t)(m-1)$,*

$$T_C = T_A+\left(\lceil\log_2(t+1)\rceil+\left\lceil\log_2\left(\left\lceil\frac{t}{2}\right\rceil+1\right)\right\rceil+\lceil\log_2(m-1)\rceil\right)T_X,$$

and $N_L=3m+k_t-k_1-2$, respectively.

Proof. Let us denote $\mathbf{e}^{(i)}=[e_0^{(i)},\ e_1^{(i)},\cdots,\ e_{m-1}^{(i)}]^T=\mathbf{Q}_i^T\mathbf{e}$, $0\leq i\leq t$, then, using Theorem 1, we can obtain the coordinates of C as

$$\mathbf{c}=\mathbf{d}+\mathbf{e}^{(0)}+\mathbf{e}^{(1)}+\mathbf{e}^{(2)}+\cdots+\mathbf{e}^{(t)}. \tag{26}$$

First, let us assume that $k_1\neq 1$. Using $\mathbf{Q}_0$ (see Fig. 6a for $t=3$), the elements of $\mathbf{e}^{(0)}$ can be written as follows:

$$e_j^{(0)} = \begin{cases} e_j+e_{j+m-k_t}+\cdots+e_{j+m-k_2} & \text{if } 0\leq j\leq k_1-2 \\ +e_{j+m-k_1}, & \\ e_j+e_{j+m-k_t}+\cdots+e_{j+m-k_2} & \text{if } k_1-1\leq j\leq k_2-2 \\ \vdots & \vdots \\ e_j+e_{j+m-k_t} & \text{if } k_{t-1}-1\leq j\leq k_t-2 \\ e_j & \text{if } k_t-1\leq j\leq m-2 \\ 0 & \text{if } j=m-1. \end{cases} \tag{27}$$

For $0\leq j\leq k_t-2$ the total number of XOR gates to form $e_j^{(0)}$s, is

TABLE 2
Comparison of Related Polynomial Basis Multipliers for $P(x) = x^m + x^{k_t} + \cdots + x^{k_2} + x^{k_1} + 1$, $1 \leq k_1 < k_2 < \cdots < k_t \leq \frac{m}{2}$

Reference	#AND	#XOR	Time delay
Zhang-Parhi [28]	m^2	$(m+t)(m-1)$	$T_A + (2t + \lceil \log_2 m \rceil) T_X$
Presented here	m^2	$(m+t)(m-1)$	$T_A + (\lceil \log_2(t+1) \rceil + \lceil \log_2(\lceil \frac{t}{2} \rceil + 1) \rceil + \lceil \log_2(m-1) \rceil) T_X$

$$\begin{aligned} N_1 &= t(k_1 - 1) + (t-1)(k_2 - k_1) + \cdots + k_t - k_{t-1} \\ &= \sum_{i=1}^{t} k_i - t. \end{aligned}$$

Let $T(e_j^{(0)})$ denote the time delay due to the gates to generate $e_j^{(0)}$. As seen in (27), the longest delay is due to $e_0^{(0)} = e_0 + e_{m-k_t} + \cdots + e_{m-k_2} + e_{m-k_1}$, i.e., $T(e_j^{(0)}) \leq T(e_0^{(0)})$. In order to reduce this delay, we first add any two terms except c_0, e.g., $e_{m-k_j} + e_{m-k_i}$, $1 \leq i, j \leq t$, $i \neq j$. Then, add these $\lceil \frac{t}{2} \rceil$ terms/signals to c_0 using a binary tree of XOR gates. Since $T(e_j) = T_A + \lceil \log_2(m-j-1) \rceil T_X$, then

$$\begin{aligned} T(e_{m-k_j} + e_{m-k_i}) &\leq T_X + T(e_{m-k_t}) \\ &= T_A + (1 + \lceil \log_2(k_t - 1) \rceil) T_X \\ &\leq T_A + \lceil \log_2(m-1) \rceil T_X, \end{aligned}$$

where the last inequality is due to $k_t \leq \frac{m}{2}$. Thus, we have

$$T(e_j^{(0)}) \leq \begin{cases} T_A + (\lceil \log_2(\lceil \frac{t}{2} \rceil + 1) \rceil \\ +\lceil \log_2(m-1) \rceil) T_X, & \text{if } 0 \leq j \leq k_t - 2 \\ T_A + \lceil \log_2(m-1) \rceil T_X & \text{if } k_t - 1 \leq j \leq m-2. \end{cases} \tag{28}$$

By reusing the terms $e_j^{(0)}$s, the coordinates of $\mathbf{e}^{(i)}$, for $1 \leq i \leq t$, can be obtained as

$$e_j^{(i)} = \begin{cases} 0, & \text{if } 0 \leq j \leq k_i - 1 \\ e_{j-k_i}^{(0)} & \text{otherwise.} \end{cases} \tag{29}$$

Equations (29) and (26) result in the following:

$$c_j = d_j + \begin{cases} e_j^{(0)} & \text{if } 0 \leq j \leq k_1 - 1 \\ e_j^{(0)} + e_j^{(1)} & \text{if } k_1 \leq j \leq k_2 - 1 \\ \vdots & \vdots \\ e_j^{(0)} + e_j^{(1)} + \cdots + e_j^{(t-1)} & \text{if } k_{t-1} \leq j \leq k_t - 1 \\ e_j^{(0)} + e_j^{(1)} + \cdots + e_j^{(t)} & \text{if } k_t \leq j \leq m-2 \\ e_j^{(1)} + e_j^{(2)} + \cdots + e_j^{(t)} & \text{if } j = m-1. \end{cases} \tag{30}$$

To realize (30) in hardware, one requires

$$\begin{aligned} N_2 &= m + (k_2 - k_1) + 2(k_3 - k_2) + \cdots + (t-1)(k_t - k_{t-1}) \\ &\quad + t(m - k_3 - 1) + t - 1 \\ &= (t+1)m - \sum_{i=1}^{t} k_i - 1 \end{aligned}$$

XOR gates. Thus, the total number of XOR gates needed for the multiplier is $(m-1)^2 + N_1 + N_2 = (m+t)(m-1)$.

To obtain the time delay of the proposed multiplier, we assume a binary tree of XOR gates for each coordinate in (30). For $j \notin [k_t, m-2]$, it can be seen from (30) that $T_C \leq \lceil \log_2(t+1) \rceil T_X + T(e_0^{(0)})$ and the proof is complete by using (28).

Now, we need only obtain the time delay of c_js for $k_t \leq j \leq m-2$. For $j \in [k_t, m-2],$, if we form $c_j = (d_j + e_j^{(0)}) + e_j^{(1)} + e_j^{(2)} + \cdots + e_j^{(t)}$ such that $d_j + e_j^{(0)}$ is calculated first, then

$$\begin{aligned} T(d_j + e_j^{(0)}) &\leq T_A + (1 + \lceil \log_2(m-1) \rceil) T_X \\ &\leq T_A + \left(\left\lceil \log_2(\left\lceil \frac{t}{2} \right\rceil + 1) \right\rceil + \lceil \log_2(m-1) \rceil \right) T_X. \end{aligned} \tag{31}$$

Also, using (29) and (28), one can see $T(e_j^{(t)}) \leq T_A + (\lceil \log_2(\lceil \frac{t}{2} \rceil + 1) \rceil + \lceil \log_2(m-1) \rceil) T_X$ which implies that

$$\begin{aligned} T_C \leq T_A + \Bigg(\lceil \log_2(t+1) \rceil + \left\lceil \log_2 \left(\left\lceil \frac{t}{2} \right\rceil + 1 \right) \right\rceil \\ + \lceil \log_2(m-1) \rceil \Bigg) T_X \end{aligned}$$

and the proof is complete.

In addition to the three buses shown in Fig. 2, now there will be another bus in the middle of the $\mathbf{Q}$-network for signals $e_j^{(0)}$, $0 \leq j \leq k_t - 2$. Also note that the signal e_j, $0 \leq j \leq k_1 - 1$, is used once in (27) and (30). Thus, the total number of bus lines is $3m + k_t - k_1 - 2$. □

Corollary 1. *For $k_1 = 1$ and $t > 1$, the time delay would reduce to*

$$T_A + \left(\lceil \log_2(t+1) \rceil + \left\lceil \log_2 \left\lceil \frac{t}{2} \right\rceil \right\rceil + \lceil \log_2(m-1) \rceil \right) T_X.$$

For this special case of irreducible polynomials, our multiplier has the same gate complexities with shorter time delay compared to the Mastrovito multiplier reported in [28]. This comparison is shown in Table 2.

6 TRINOMIALS

Let $P(x) = x^m + x^k + 1$ be an irreducible trinomial generating $GF(2^m)$. Trinomial $P(x)$ has only three nonzero coefficients and (for $m > 1$) no binary irreducible polynomial can have any fewer nonzero coefficients. Since low Hamming weight polynomials can potentially reduce the space and time complexities of a finite field multiplier, irreducible trinomials have drawn significant attention in the past. Reference [22] lists an irreducible trinomial for every degree m ($\leq 10,000$) for which such a polynomial exists.

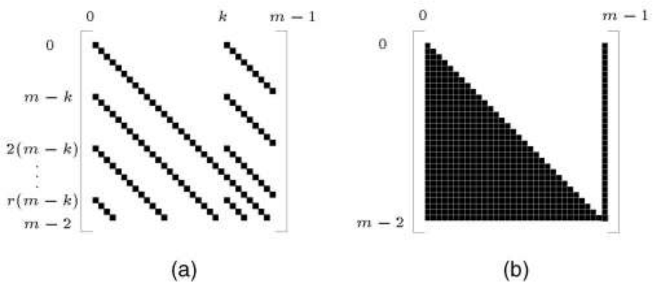


Fig. 7. Graphical representations of the reduction matrix $\mathbf{Q}$ for trinomial $P(x) = x^m + x^k + 1$. (a) $\frac{m}{2} < k < m-1$, $r = \lfloor \frac{m-2}{m-k} \rfloor$, (b) $k = m-1$ (see Fig. 5a, Fig. 5b, and Fig. 4c for $k = 1$, $1 < k < \frac{m}{2}$, and $k = \frac{m}{2}$, respectively).

Now, we derive $\mathbf{Q}$ for the trinomial to obtain the complexities of the LCBP multiplier. The graphical representations of the locations of the nonzeros of $\mathbf{Q}$ for irreducible trinomials with $k = 1$ and $1 < k < \frac{m}{2}$ have already been shown in Fig. 5a and Fig. 5b, respectively. Similarly, for $\frac{m}{2} < k < m$ and $k = m-1$, $\mathbf{Q}$ can be obtained and their graphical representation of the locations of the nonzeros is shown in Fig. 7a and Fig. 7b, respectively. Now, using the representation of $\mathbf{Q}$, we can state the following theorem.

Theorem 5. *The number of XOR gates and the time delay of the LCBP multiplier based on the trinomial $x^m + x^k + 1$ are*

$$N_X = \begin{cases} m^2 - \frac{m}{2}, & \text{for } k = \frac{m}{2} \\ m^2 - 1, & \text{otherwise} \end{cases}$$

and

$$T_C = \begin{cases} T_A + (2 + \lceil \log_2(m-1) \rceil) T_X, & \text{for } 1 \leq k < \frac{m}{2}, \\ T_A + (1 + \lceil \log_2 m \rceil) T_X, & \text{for } k = \frac{m}{2}, \\ T_A + \big(1 + \lfloor \frac{m-2}{m-k} \rfloor \\ \quad + \lceil \log_2 \big(m - 1 - \lfloor \frac{k-2}{m-k} \rfloor (m-k)\big) \rceil \big) T_X, & \text{for } \frac{m}{2} < k < m. \end{cases}$$

Proof. Based on the results obtained in Section 5, one can obtain the time delay and the number of XOR gates by substituting $t = 1$ in Theorem 4, for $1 \leq k < \frac{m}{2}$. Since the trinomial with $k = \frac{m}{2}$, i.e., $P(x) = x^m + x^{\frac{m}{2}} + 1$, is an $\frac{m}{2}$-ESP, the complexities of the multiplier based on this type of trinomial can also be obtained from Theorem 3 with $s = \frac{m}{2}$.

Below, we discuss trinomials with $k > \frac{m}{2}$.

Case: $\frac{m}{2} < k \leq m-1$.

Using Fig. 7a and Theorem 5, one can generate the coordinates of C as

$$c_j = d_j + \begin{cases} e'_j & \text{for } 0 \leq j \leq k-2 \\ e_{k-1} & \text{for } j = k-1 \\ e_j + e'_{j-k} & \text{for } k \leq j \leq 2k-2 \\ e_j + e_{j-k} & \text{for } 2k-1 \leq j \leq m-2 \\ e_{m-k-1} & \text{for } j = m-1. \end{cases} \tag{32}$$

where e'_j can be obtained recursively from $j = k-2$ down to $j = 0$ as follows:

$$e'_j = \begin{cases} e_j + e_{j+m-k}, & \text{for } k-2 \geq j \geq 2k-m-1, \\ e_j + e'_{j+m-k,} & \text{for } 2k-m-2 \geq j \geq 0. \end{cases} \tag{33}$$

These require the same number of XOR gates as in the case of $1 \leq k < \frac{m}{2}$, which is $m^2 - 1$. Also, the time delay of the multiplier is determined by

$$c_k = ((d_k + e_k) + (e_0 + (e_{m-k} \\ + \cdots + (e_{(r-1)(m-k)} + e_{r(m-k)}) \cdots))),$$

where $r = \lfloor \frac{m-2}{m-k} \rfloor$. Thus, $T_C = (r+1)T_X + T(e_{(r-1)(m-k)})$ and, using (15), the total time delay of the multiplier is

$$T_C = T_A \\ + \left(1 + \left\lfloor \frac{m-2}{m-k} \right\rfloor + \left\lceil \log_2 \left(m - 1 - \left\lfloor \frac{k-2}{m-k} \right\rfloor (m-k) \right) \right\rceil \right) T_X.$$

Note that for, $k = m-1$, (33) becomes

$$e'_j = \begin{cases} \sum_{i=j}^{m-2} e_i, & \text{for } 0 \leq j \leq m-2, \\ e'_0, & \text{for } j = m-1, \end{cases}$$

which requires the same number of XOR gates and the corresponding delay is $T_A + mT_X$. □

In [25], a trinomial based multiplier for the cases of $1 \leq k \leq \frac{m}{2}$ has been proposed. For these values of k, the above results match those reported in [25]. Table 3 compares the presented multiplier with other trinomial-based multipliers. As shown in this table, the proposed multiplier has the same gate complexities as the Mastrovito multiplier. For $k = 1$, the proposed multiplier has a time delay which is longer by T_X than the Mastrovito multiplier. However, for the other values of k, i.e., $1 < k < m$, it has the same or shorter delay compared to the Mastrovito multiplier.

To reduce the time delay of the Mastrovito multiplier for $\frac{m}{2} < k < m$, a hybrid tree structure is used in [28]. One can also use a similar technique to the proposed multiplier by applying a hybrid tree to generate e'_j in (33).

If one attempts to apply Theorem 5 to the multiplier in Fig. 3, the $\mathbf{Q}$-network should be modified by reusing signals (e'_0, e'_1, e'_2) instead of signals (e_0, e_1, e_2). The coordinates of C can be obtained as $c_j = d_j + e'_j$, $0 \leq j \leq 3$, where $e'_0 = e'_3 = e_0 + e'_1$, $e'_1 = e_1 + e_2$, $e'_2 = e_2$.

7 Special Classes of Pentanomials

A polynomial with five nonzero coefficients, i.e., $P(x) = x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1$, where

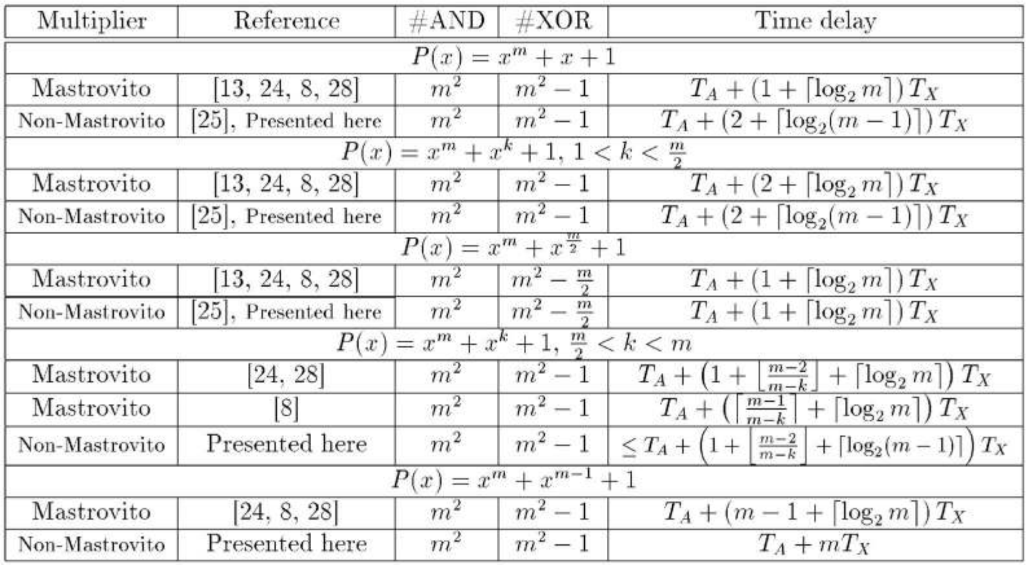
$$1 \leq k_1 < k_2 < k_3 \leq m-1,$$

is called a *pentanomial* of degree m. The nonzero constant term is due to the irreducible property needed to define the representation of the field. In terms of the values of k_is, the pentanomials can be divided into a number of different classes. Below we consider two special classes of irreducible pentanomials as proposed in [28].

7.1 Class 1: $k_3 \leq \frac{m}{2}$

For this class of irreducible pentanomial where $k_3 \leq \frac{m}{2}$, one can apply $t = 3$ to the complexity results we have presented in Section 5. This yields the following:

TABLE 3
Comparison of Related Polynomial Basis Multipliers Based on Trinomials



Multiplier	Reference	#AND	#XOR	Time delay
$P(x) = x^m + x + 1$				
Mastrovito	[13, 24, 8, 28]	m^2	$m^2 - 1$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$
Non-Mastrovito	[25], Presented here	m^2	$m^2 - 1$	$T_A + (2 + \lceil \log_2 (m-1) \rceil) T_X$
$P(x) = x^m + x^k + 1,\ 1 < k < \frac{m}{2}$				
Mastrovito	[13, 24, 8, 28]	m^2	$m^2 - 1$	$T_A + (2 + \lceil \log_2 m \rceil) T_X$
Non-Mastrovito	[25], Presented here	m^2	$m^2 - 1$	$T_A + (2 + \lceil \log_2 (m-1) \rceil) T_X$
$P(x) = x^m + x^{\frac{m}{2}} + 1$				
Mastrovito	[13, 24, 8, 28]	m^2	$m^2 - \frac{m}{2}$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$
Non-Mastrovito	[25], Presented here	m^2	$m^2 - \frac{m}{2}$	$T_A + (1 + \lceil \log_2 m \rceil) T_X$
$P(x) = x^m + x^k + 1,\ \frac{m}{2} < k < m$				
Mastrovito	[24, 28]	m^2	$m^2 - 1$	$T_A + \left(1 + \left\lfloor \frac{m-2}{m-k} \right\rfloor + \lceil \log_2 m \rceil\right) T_X$
Mastrovito	[8]	m^2	$m^2 - 1$	$T_A + \left(\left\lceil \frac{m-1}{m-k} \right\rceil + \lceil \log_2 m \rceil\right) T_X$
Non-Mastrovito	Presented here	m^2	$m^2 - 1$	$\leq T_A + \left(1 + \left[\frac{m-2}{m-k} \right] + \lceil \log_2 (m-1) \rceil\right) T_X$
$P(x) = x^m + x^{m-1} + 1$				
Mastrovito	[24, 8, 28]	m^2	$m^2 - 1$	$T_A + (m - 1 + \lceil \log_2 m \rceil) T_X$
Non-Mastrovito	Presented here	m^2	$m^2 - 1$	$T_A + m T_X$

Corollary 2. *The gate counts and time delay of the multiplier for the pentanomial* $P(x) = x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1$, *where* $k_1 < k_2 < k_3 \leq \frac{m}{2}$, *are*

$$N_A = m^2,$$
$$N_X = m^2 + 2m - 3,$$
$$T_C = \begin{cases} T_A + (3 + \lceil \log_2 (m-1) \rceil) T_X, & \text{if } k_1 = 1 \\ T_A + (4 + \lceil \log_2 (m-1) \rceil) T_X, & \text{otherwise,} \end{cases}$$

and the number of lines on the buses is $N_L = 3m + k_3 - k_1 - 2$.

The number of XOR gates can be reduced if we choose a pentanomial such that $k_1 = k_3 - k_2$. Toward this, let us introduce the following set of intermediate terms/signals:

$$e'_j = e_{j+m-k_3} + e_{j+m-k_2}, \quad 0 \leq j \leq k_2 - 2. \tag{34}$$

Equation (34) can be used to generate $e_j^{(0)}$, $0 \leq j \leq k_2 - 2$, by substituting $t = 3$ in (27) as follows:

$$e_j^{(0)} = \begin{cases} e_j + e'_j + e_{j+m-k_1}, & \text{if } 0 \leq j \leq k_1 - 2 \\ e_j + e'_j & \text{if } k_1 - 1 \leq j \leq k_2 - 2 \\ e_j + e_{j+m-k_3} & \text{if } k_2 - 1 \leq j \leq k_3 - 2 \\ e_j & \text{if } k_3 - 1 \leq j \leq m - 2 \\ 0 & \text{if } j = m - 1. \end{cases} \tag{35}$$

The total number of XOR gates needed to generate $e_j^{(0)}$s (see (35)) is $N_1 = k_1 + k_2 + k_3 - 3$ in which (34) contributes $k_2 - 1$. Also, the maximum delay due to gates in (35) is

$$T(e_j^{(0)}) \leq \begin{cases} T_A + (2 + \lceil \log_2 (m-1) \rceil) T_X & \text{if } 0 \leq j \leq k_1 - 2 \\ T_A + (1 + \lceil \log_2 (m-1) \rceil) T_X & \text{if } k_1 - 1 \leq j \leq k_3 - 2 \\ T_A + \lceil \log_2 (m-1) \rceil T_X & \text{if } k_3 - 1 \leq j \leq m - 1. \end{cases} \tag{36}$$

Lemma 1. *With symbols defined as above, one has*

$$e_j^{(0)} + e_j^{(1)} = e'_{j+k_2-m}, \text{ for } m - k_2 \leq j \leq m - 2,$$
$$e_j^{(2)} + e_j^{(3)} = e_{j-k_2}^{(0)} + e_{j-k_2}^{(1)}, \text{ for } k_3 \leq j \leq m - 1.$$

Proof. Since $k_3 \leq \frac{m}{2}$, one can easily verify that, for all js, $k_3 - 1 \leq j - k_1$ (and, hence, $k_3 - 1 \leq j$). Thus, using (35) and (29), one can simply obtain

$$\begin{aligned} e_j^{(0)} + e_j^{(1)} &= e_j^{(0)} + e_{j-k_1}^{(0)} \\ &= e_j + e_{j-k_1} \\ &= e'_{j+k_2-m}, \text{ for } m - k_2 \leq j \leq m - 2. \end{aligned}$$

Similarly, the second equation can be proven by using (35), (29), and $k_2 = k_3 - k_1$ as follows:

$$\begin{aligned} e_j^{(2)} + e_j^{(3)} &= e_{j-k_2}^{(0)} + e_{j-k_3}^{(0)} \\ &= e_{j-k_2}^{(0)} + e_{j+k_1-k_3}^{(1)} \\ &= e_{j-k_2}^{(0)} + e_{j-k_2}^{(1)}, \text{ for } k_3 \leq j \leq m - 1. \end{aligned}$$

□

Let us represent $e_j^{(01)}$, $0 \leq j \leq m - 1$, as the elements of $(\mathbf{Q}_0 + \mathbf{Q}_1)^T \mathbf{e}$, where $\mathbf{Q}_0$ and $\mathbf{Q}_1$ are shown in Fig. 6a and Fig. 6b, respectively. Then, substituting $t = 3$ in the general case given in (30) and using the above lemma, we can obtain the coordinates of $C = AB$ as follows:

$$c_j = d_j + e_j^{(01)} + e_{j-k_2}^{(01)}, \ 0 \leq j \leq m - 1, \tag{37}$$

where $e_{j-k_2}^{(01)} = 0$ for $j < k_2$, and

$$e_j^{(01)} = \begin{cases} e_j^{(0)} & \text{if } 0 \leq j \leq k_1 - 1 \\ e_j^{(0)} + e_j^{(1)} & \text{if } k_1 \leq j \leq m - k_2 - 1 \\ e'_{j+k_2-m} & \text{if } m - k_2 \leq j \leq m - 2 \\ e_j^{(1)} & \text{if } j = m - 1. \end{cases} \tag{38}$$

As seen in (38), one has to realize $e_j^{(0)} + e_j^{(1)}$ for all $k_1 \leq j \leq m - k_2 - 1$, which requires $m - k_2 - k_1$ XOR gates.

TABLE 4
Maximum Time Delays of the Signals, where $t(i)$, $0 \leq i \leq 4$, Represents the Time Delay of $T_A + (i + \lceil \log_2(m-1) \rceil)T_X$, Numbers inside Square Brackets Are for $k_1 = 1$, and x to Indicate whether $e_j^{(01)}$ or $e_{j-k_2}^{(01)}$ Is to Be Added First to d_j

j	$e_j^{(0)}$	$e_j^{(1)}$	$e_j^{(01)}$	$e_{j-k_2}^{(01)}$	$d_j + x$	c_j
$0 \leq j \leq k_1 - 1$	$t(2), [t(1)]$	-	$t(2), [t(1)], x$	-	$t(3)$	$t(3)$
$k_1 \leq j \leq k_2 - 1$	$t(1)$	$t(2), [t(1)]$	$t(3), [t(2)], x$	-	$t(4), [t(3)]$	$t(4), [t(3)]$
$k_2 \leq j \leq k_3 - 1$	$t(1)$	$t(2), [t(1)]$	$t(3), [t(2)]$	$t(2), [t(1)], x$	$t(3), [t(2)]$	$t(4), [t(3)]$
$k_3 \leq j \leq k_3 + k_1 - 1$	$t(0)$	$t(1)$	$t(2), x$	$t(3), [t(2)]$	$t(3)$	$t(4)$
$k_3 + k_1 \leq j \leq m - k_2 - 1$	$t(0)$	$t(0)$	$t(1), x$	$t(3), [t(2)]$	$t(2)$	$t(4), [t(3)]$
$m - k_2 \leq j \leq m - 1$	$t(0)$	$t(0)$	$t(1), x$	$t(3), [t(2)]$	$t(2)$	$t(4), [t(3)]$
$j = m - 1$	-	$t(0)$	$t(1), x$	$t(1)$	$t(2)$	$t(3)$

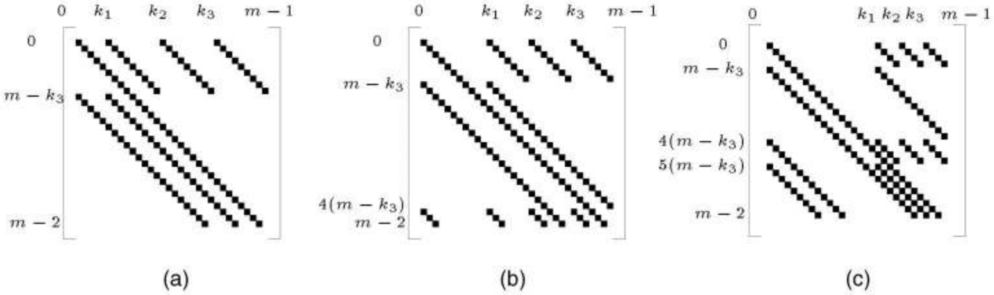


Fig. 8. Graphical representations of the reduction matrix **Q** for class 2 pentanomials $P(x) - x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1$, where $m - k_3 = k_3 - k_2 = k_2 - k_1 = s$. (a) $\frac{m-1}{4} \leq s \leq \frac{m-1}{3}$ or $1 \leq k_1 \leq s+1$ (see Fig. 4a for $k_1 = s$), (b) $\frac{m-1}{5} \leq s < \frac{m-1}{4}$ or $s+1 < k_1 \leq 2s+1$, (c) $\frac{m-1}{8} \leq s < \frac{m-1}{5}$ or $2s+1 < k_1 \leq 5s+1$.

Once $e_j^{(01)}$'s are obtained, then (37) requires $2m - k_2$ XOR gates. Thus, the total number of XOR gates needed for the multiplier is

$$(m-1)^2 + N_1 + m - k_2 - k_1 + 2m - k_2 = m^2 + m + k_1 - 2.$$

Due to the reuse of terms e'_j, $0 \leq j \leq k_2 - 1$, and $e_j^{(0)} + e_j^{(1)}$, $k_1 \leq j \leq m - k_2 - 1$, additional lines needed on the bus in the **Q**-network are $(k_2 - 1)$ and $(m - k_1 - k_2)$, respectively. Thus, the total number of lines on the buses is increased to $4m + k_2 - k_1 - 3$.

To obtain the time delay of the proposed multiplier, we use Table 4 which shows the maximum delay of the signals in (37) for the given ranges of j in each row. In this table, the parameter $t(i)$, $0 \leq i \leq 4$, represents the time delay of $t(i) = T_A + (i + \lceil \log_2(m-1) \rceil)T_X$ and the numbers inside square brackets are for $k_1 = 1$. Also, x determines whether $e_j^{(01)}$ or $e_{j-k_2}^{(01)}$ is to be added to d_j first to obtain c_j. For example, using the fifth row of this table, c_{k_3} can be obtained as $c_{k_3} = (d_{k_3} + e_{k_3}^{(01)}) + e_{k_1}^{(01)}$. In each row of this table, the delays are obtained for the first digit of the given range. This is because, as j increases, the time delays of the used signals in each row of this table decreases. As seen in this table, the maximum delay of the multiplier is $T_A + (4 + \lceil \log_2(m-1) \rceil)T_X$. For $k_1 = 1$, only one signal, i.e., c_{k_3}, has the delay of $T_A + (4 + \lceil \log_2(m-1) \rceil)T_X$. One can reduce this delay to $T_A + (3 + \lceil \log_2(m-1) \rceil)T_X$ if only c_{k_3} is realized as $c_{k_3} = ((d_{k_3} + e_j^{(0)}) + e_j^{(1)}) + e_{k_3-k_2}^{(01)}$ by using one extra XOR gate.

Based on the above results, we can state the following:

Theorem 6. *The gate counts and time delay of the multiplier based on the pentanomial $P(x) = x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1$, where $k_1 < k_2 < k_3 \leq \frac{m}{2}$ and $k_3 - k_2 = k_1$ are*

$$N_A = m^2,$$

$$N_X = \begin{cases} m^2 + m & \text{if } k_1 = 1 \\ m^2 + m + k_1 - 2 & \text{otherwise,} \end{cases}$$

$$T_C = \begin{cases} T_A + (3 + \lceil \log_2(m-1) \rceil)T_X, & \text{if } k_1 = 1 \\ T_A + (4 + \lceil \log_2(m-1) \rceil)T_X, & \text{otherwise,} \end{cases}$$

and the number of lines on the buses is $N_L = 4m + k_2 - k_1 - 3$.

Remark 3. *To verify that class 1 irreducible pentanomials exist, we have used a Maple*Ⓣⓜ program for $m \in [160, 600]$ *and have found that at least one such irreducible pentanomial exists for every m in the range of 160 to 600. This is of interest to elliptic curve cryptosystem designers. In order to minimize the number of XOR gates of the multiplier, we have obtained irreducible pentanomials such that k_1 is minimum. These are shown in Tables 5 and 6 in [18]. As can be seen from these tables, k is less than or equal to 6 for any m in the above mentioned range.*

It is noted that the pentanomial presented in [20] is the special case of $k_1 = 1$.

7.2 Class 2: $m - k_3 = k_3 - k_2 = k_2 - k_1 = s$, $\frac{m-1}{8} \leq s \leq \frac{m-1}{3}$

We refer to polynomials $P(x) = x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1$, where $1 \leq k_1 < k_2 < k_3 \leq m - 1$ and $m - k_3 = k_3 - k_2 = k_2 - k_1 = s$ as class 2 type pentanomials. Similar to the other special irreducible polynomials, here we first obtain the corresponding reduction matrix. Then, the coordinates and complexities of the multiplier can be obtained. Based on the values of s (or $k_1 = m - 3s$), we can divide the reduction matrix into different forms. Here, only three of them are presented. These **Q** matrices for $\frac{m-1}{8} \leq s \leq \frac{m-1}{3}$ (or $1 \leq k_1 \leq 5s + 1$) are shown in Fig. 8. Based on this figure, we can state the following theorem.

Theorem 7. *The gate counts and the time delay of the multiplier for the pentanomial $P(x) = x^m + x^{m-s} + x^{m-2s} + x^{m-3s} + 1$, for $\frac{m-1}{8} \le s \le \frac{m-1}{3}$ are $N_A = m^2$,*

$$N_X = \begin{cases} m^2 + m - s - 1, & if\ \frac{m-1}{4} \le s \le \frac{m-1}{3} \\ m^2 + 2m - 5s - 2 & if\ \frac{m-1}{5} \le s < \frac{m-1}{4} \\ m^2 + m - 2 & if\ \frac{m-1}{8} \le s < \frac{m-1}{5} \end{cases}$$

$$T_C = \begin{cases} T_A + (3 + \lceil \log_2(m-1) \rceil) T_X, & if\ \frac{m-1}{5} \le s \le \frac{m-1}{3} \\ T_A + (4 + \lceil \log_2(m-1) \rceil) T_X, & otherwise, \end{cases}$$

and

$$N_L = \begin{cases} 4m - 2, & if\ \frac{m-1}{4} \le s \le \frac{m-1}{3} \\ 5m - 4s - 3 & if\ \frac{m-1}{5} \le s < \frac{m-1}{4} \\ 5k_3 - 3 & if\ \frac{m-1}{8} \le s < \frac{m-1}{5}. \end{cases}$$

Proof. Case I: $1 \le k_1 \le s + 1$, $\frac{m-1}{4} \le s \le \frac{m-1}{3}$.

Using (9) and Fig. 8a, one can compute the coordinates of C as

$$c_j = d_j + \begin{cases} e_j + e_{j+s} & \text{if } 0 \le j \le k_1 - 1 \\ e_{j-k_1} + e_j + e_{j-k_1+s} + e_{j+s} & \text{if } k_1 \le j \le k_2 - 1 \\ e_{j-k_2} + e_j + e_{j-k_1+s} + e_{j+s} & \text{if } k_2 \le j \le k_3 - 2 \\ e_{j-k_2} + e_j + e_{j-k_1+s} & \text{if } j = k_3 - 1 \\ e_{j-k_3} + e_j + e_{j-k_1+s} & \text{if } k_3 \le j \le k_1 + k_3 - 2 \\ e_{j-k_3} + e_j & \text{if } k_1 + k_3 - 1 \le j \le m - 2 \\ e_{j-k_3} & \text{if } j = m - 1. \end{cases} \tag{39}$$

In order to reduce the number of XOR gates needed for implementing (39), one can precompute

$$\begin{aligned} e'_j =& \quad e_j + e_{j+s}, & \text{for } 0 \le j \le k_2 - 1, \\ e''_{j-k_2} =& \quad e_{j-k_2} + e_{j+s}, & \text{for } k_2 \le j < k_3 - 1. \end{aligned}$$

The precomputation requires a total of $k_3 - 1$ XOR gates with a maximum time delay of $T_A + (1 + \lceil \log_2(m-1) \rceil) T_X$. Then, by reusing the first s terms of e'_js and all signals of e''_{j-k_2}s, i.e., $e'_0,\ e'_1,\ \cdots,\ e'_{s-1}, e''_0,\ e''_1,\ \cdots,\ e''_{k_3-2}$, one can simplify (39) as

$$c_j = d_j + \begin{cases} e'_j & \text{if } 0 \le j \le k_1 - 1 \\ e'_j + e'_{j-k_1} & \text{if } k_1 \le j \le k_2 - 1 \\ e''_{j-k_2} + e_j + e_{j-k_1+s} & \text{if } k_2 \le j \le k_3 - 2 \\ e_{j-k_2} + e_j + e_{j-k_1+s} & \text{if } j = k_3 - 1 \\ e''_{j-k_3} + e_{j-k_1+s} & \text{if } k_3 \le j \le k_1 + k_3 - 2 \\ e''_{j-k_3} & \text{if } k_1 + k_3 - 1 \le j \le m - 2 \\ e_{j-k_3} & \text{if } j = m - 1. \end{cases} \tag{40}$$

Equation (40) requires $m + (k_2 - k_1) + 2(k_3 - k_2 - 1) + 2 + (k_1 - 1) = m + 2k_3 - k_2 - 1 = 2m - 1$ XOR gates with a time delay of $2T_X$. Thus, the total number of XOR gates required for the whole multiplier is

$$(m-1)^2 + k_3 - 1 + 2m - 1 = m^2 + k_3 - 1 = m^2 + m - s - 1$$

with a time delay of $T_C = T_A + (3 + \lceil \log_2(m-1) \rceil) T_X$. The number of lines on the buses has now increased by the number of reused e'_js and e''_{j-k_2}s, i.e., $3m - 1 + s + k_3 - 1 = 4m - 2$.

It is noted that, for the special case of $k_1 = 1$, (40) should be modified by simply removing the fifth line with condition $k_3 \le j \le k_1 + k_3 - 2$. This does not affect the complexities of the whole multiplier structure.

Case II: $s + 1 < k_1 \le 2s + 1$, $\frac{m-1}{5} \le s < \frac{m-1}{4}$.

By comparing Fig. 8b with Fig. 8a, one can see that the $\mathbf{Q}$ matrix in this case has four more small lines at the bottom of Fig. 8b. This results in more terms in the representations of the coordinates of C. In order to be consistent with the previous case and to use (40), one can introduce the following terms:

$$e'_j = \begin{cases} e_j + e_{j+4s} + e_{j+s} & \text{for } 0 \le j \le m - 2 - 4s \\ e_j + e_{j+s} & \text{for } m - 1 - 4s \le j \le k_2 - 1, \end{cases} \tag{41}$$

and

$$e''_{j-k_2} = \begin{cases} e_{j-k_2} + e_{j-k_2+4s} + e_{j+s} & \text{for } k_2 \le j \le k_2 + m - 2 - 4s \\ e_{j-k_2} + e_{j+s} & \text{for } k_2 + m - 1 - 4s \le j < k_3 - 1. \end{cases} \tag{42}$$

These new terms cause Case II to require $m - 1 - 4s$ more XOR gates than Case I. Note that the following terms: $e_j + e_{j+4s}$, $0 \le j \le m - 2 - 4s$, are common between (41) and (42). Thus, using (40) with new e'_js and e''_{j-k_2}s, i.e., (41) and (42), we have a total number of XOR gates as $m^2 + k_3 - 1 + m - 1 - 4s = m^2 + 2m - 5s - 2$ and the total number of lines on the buses is $4m - 2 + m - 1 - 4s = 5m - 4s - 3$. The maximum delays in (41) and (42) are due to e'_0 and e''_0, respectively, and are equal to $T_A + (2 + \lceil \log_2(m-1) \rceil) T_X$ each. Compared to Case I, this delay is increased by T_X, however, for an implementation similar to Case I, one can obtain $T_C = T_A + (3 + \lceil \log_2(m-1) \rceil) T_X$.

Case III: $2s + 1 < k_1 \le 5s + 1$ $\frac{m-1}{8} \le s < \frac{m-1}{5}$.

Let us introduce

$$e'_j = e_j + e_{j+4s},\ \text{for } 0 \le j \le m - 2 - 4s,$$

which requires $m - 1 - 4s$ XOR gates and a delay of $T_A + (1 + \lceil \log_2(m-1) \rceil) T_X$. Let $\mathbf{Q}_0$ be a submatrix which contains all four lines starting from column 0 in Fig. 8c. Then, the coordinates of $\mathbf{e}^{(0)} = \mathbf{Q}_0^T \mathbf{e}$ can be obtained as

$$e_j^{(0)} = \begin{cases} e'_j + e'_{j+s} & \text{if } 0 \le j \le m - 2 - 5s \\ e'_j + e_{j+s} & \text{if } m - 1 - 5s \le j \le m - 2 - 4s \\ e_j + e_{j+s} & \text{if } m - 1 - 4s \le j \le k_3 - 2 \\ e_j & \text{if } k_3 - 1 \le j \le m - 2 \\ 0 & \text{if } j = m - 1, \end{cases} \tag{43}$$

which requires $k_3 - 1$ XOR gates and a maximum time delay of $T_A + (2 + \lceil \log_2(m-1) \rceil) T_X$. Thus, using Fig. 8c and (9), the coordinates of C can be obtained as

TABLE 5
Comparison of Related Pentanomial-Based Multipliers

Reference	Special Case	#XOR	Time delay
$P(x) = x^m + x^{k_3} + x^{k_2} + x^{k_1} + 1,\, 1 < k_1 < k_2 < k_3 \leq \frac{m}{2}$			
[28]	$k_1 \geq 1$	$m^2 + 2m - 3$	$T_A + (6 + \lceil \log_2 m \rceil)\, T_X$
LCBP	$k_1 > 1$	$m^2 + 2m - 3$	$T_A + (4 + \lceil \log_2 (m-1) \rceil)\, T_X$
LCBP	$k_1 = 1$	$m^2 + 2m - 3$	$T_A + (3 + \lceil \log_2 (m-1) \rceil)\, T_X$
LCBP	$k_3 - k_2 = k_1$	$m^2 + m + k_1 - 2$	$T_A + (4 + \lceil \log_2 (m-1) \rceil)\, T_X$
[20]	$k_3 - k_2 = k_1 = 1$	$m^2 + m + 2k_2$	$T_A + (3 + \lceil \log_2 (m-1) \rceil)\, T_X$
LCBP	$k_3 - k_2 = k_1 = 1$	$m^2 + m$	$T_A + (3 + \lceil \log_2 (m-1) \rceil)\, T_X$
LCBP,[20]	$k_i = i$	$m^2 + m$	$T_A + (3 + \lceil \log_2 (m-1) \rceil)\, T_X$
$P(x) = x^m + x^{m-s} + x^{m-2s} + x^{m-3s} + 1$			
[28]	$1 \leq s \leq \frac{m-1}{3}$	$m^2 + 4m - 5s - 5$	$T_A + \left(\lfloor \frac{d}{4} \rfloor + 4 + \lceil \log_2 (m-1) \rceil\right) T_X$
[28]	$s \leq \frac{m-1}{3}$	$\geq m^2 + 2.33m - 7$	$\geq T_A + (4 + \lceil \log_2 (m-1) \rceil)\, T_X$
LCBP	$\frac{m-1}{8} \leq s \leq \frac{m-1}{3}$	$\leq m^2 + m$	$\leq T_A + (4 + \lceil \log_2 (m-1) \rceil)\, T_X$

TABLE 6
Values $m \in [160, 600]$ and s such that Polynomial $P(x) = x^m + x^{m-s} + x^{m-2s} + x^{m-3s} + 1,\, 1 \leq s \leq \frac{m-1}{3}$ Is Irreducible

161, 20	166, 43	167, 44	169, 45	170, 53	172, 57	175, 53	178, 49
182, 27	185, 48	191, 40	193, 40	194, 29	196, 43	199, 55	202, 49
209, 67	212, 35	214, 47	215, 64	217, 51	218, 69	220, 71	223, 63
233, 53	236, 77	238, 55	239, 27	241, 57	242, 49	244, 37	247, 55
250, 49	253, 69	257, 72	260, 75	263, 31	265, 46	266, 73	268, 81
271, 71	274, 69	278, 91	281, 33	284, 77	286, 71	287, 72	289, 28
292, 85	295, 61	302, 87	305, 34	308, 5	310, 31	313, 78	314, 5
316, 45	319, 89	322, 85	329, 93	332, 81	337, 94	340, 55	343, 53
346, 21	350, 99	353, 86	358, 19	359, 97	362, 85	364, 99	367, 57
370, 77	377, 112	380, 111	382, 27	383, 45	385, 81	386, 101	388, 53
391, 121	394, 45	401, 83	404, 113	406, 83	407, 112	409, 29	412, 49
415, 84	418, 73	422, 91	425, 78	428, 35	431, 77	433, 124	436, 55
439, 130	446, 51	449, 105	455, 139	457, 147	458, 85	460, 147	463, 83
470, 107	473, 91	476, 47	478, 119	479, 125	481, 77	484, 35	487, 131
490, 73	494, 159	497, 76	500, 135	503, 159	505, 58	506, 161	508, 133
511, 167	514, 149	518, 135	521, 163	524, 119	526, 143	527, 160	529, 124
532, 177	538, 65	545, 141	550, 119	551, 80	553, 153	556, 91	559, 175
566, 91	569, 164	574, 187	575, 143	577, 184	580, 79	583, 151	590, 31
593, 169	596, 91	599, 70					

$$c_j = d_j + \begin{cases} e_j^{(0)} & \text{if } 0 \leq j \leq k_1 - 1 \\ e_j^{(0)} + e_{j-k_1}^{(0)} & \text{if } k_1 \leq j \leq k_2 - 1 \\ e_j^{(0)} + e'_{j-k_2} + e'_{j+s-k_1} & \text{if } k_2 \leq j \leq k_3 - 1 \\ e_j^{(0)} + e'_{j-k_3} + e'_{j+s-k_1} & \text{if } k_3 \leq j \leq k_1 + m - 2 - 5s \\ e_j^{(0)} + e'_{j-k_3} + e_{j+s-k_1} & \text{others.} \end{cases} \tag{44}$$

To implement (44), one requires $3m - k_1 - k_2 - 1$ XOR gates with the time delay of $2T_X$. Thus, the total number of XOR gates and time delay of the multiplier are

$$(m-1)^2 + (m - 1 - 4s) + (k_3 - 1) + (3m - k_1 - k_2 - 1) \\ = m^2 + m - 2$$

and $T_A + (4 + \lceil \log_2 (m-1) \rceil) T_X$, respectively. Also, similar to the previous cases, one can obtain the number of lines on the buses as $5k_3 - 3$. □

A comparison of our newly obtained gate counts and delays as presented above with those of existing ones for pentanomial based multiplier is shown in Table 5. As seen in this table, for class 1 pentanomials with $k_3 - k_2 = k_1$, the proposed multiplier is faster than [28] and has fewer XOR gates. This proposed special case of class 1 covers the case of pentanomials reported in [20], where $k_1 = 1$. Compared to the multiplier proposed in [20], the proposed multiplier for the special case of $k_1 = k_3 - k_2 = 1$ has $2k_2$ fewer XOR gates and matches the ones proposed in [20] which uses $k_1 = 1$ and $k_2 = 2$. Also, for class 2 pentanomials, our multiplier is either faster than or has the same gate delay and has at least $1.33m - 7$ fewer XOR gates than the multiplier reported in [28].

Remark 4. *Using Maple*Ⓣⓜ, we have found that there exist 147 values of m, *as shown in Table 6, where* $m \in [160, 600]$ *such*

TABLE 7
Comarison of the Numbers of Bus Lines of Fig. 2 with that of the Mastrovito Multiplier

Multipliers	# Lines on the buses			
	trinomial	s-ESP	pentanomial	generic
Mastrovito [13]	$3m-1$	$\frac{m(m-s)}{2s}+2m$	$5m-3$	$(t+2)(m-1)+2$
Presented here	$3m-1$	$2m+s$	$\leq 4m+k_2$	$3m+k_t-k_1-2$

that polynomial $P(x)=x^m+x^{m-s}+x^{m-2s}+x^{m-3s}+1$, $1 \leq s \leq \frac{m-1}{3}$ *is irreducible. Among them, only 23 have* $1 \leq s < \frac{m-1}{8}$.

8 Concluding Remarks

In this paper, new bit parallel polynomial basis multipliers over $GF(2^m)$ have been proposed. Time and space complexities of such a multiplier heavily depend on the field defining irreducible polynomials. Based on a number of important classes of irreducible polynomials, we have given an exact complexity analysis of the multiplier. In general, our results match or outperform the previously known best results in similar classes. We have also presented exact formulations for the coordinates of the multiplier output. Such formulations are expected to be useful to efficiently implement the multiplier using hardware description languages, such as VHDL and Verilog, without having much knowledge of finite field arithmetic.

Moreover compared to the well-known Mastrovito multiplier, the architectures discussed here have fewer number of lines on the buses. This is shown in Table 7. Fewer number of lines on the buses can be advantageous for VLSI implementation, especially for cryptographic applications where m is usually very large.

Acknowledgments

The authors would like to thank the reviewers for their comments. The work has been supported in part by an NSERC postdoctoral fellowship to A. Reyhani-Masoleh. Preliminary versions of this article can be found in [17] and [19].

References

[1] G.B. Agnew, T. Beth, R.C. Mullin, and S.A. Vanstone, "Arithmetic Operations in $GF(2^m)$," *J. Cryptology,* vol. 6, pp. 3-13, 1993.

[2] G.B. Agnew, R.C. Mullin, and S.A. Vanstone, "An Implementation of Elliptic Curve Cryptosystems over $F_{2^{155}}$," *IEEE J. Selected Areas in Comm.,* vol. 11, no. 5, pp. 804-813, June 1993.

[3] T.C. Bartee and D.I. Schneider, "Computation with Finite Fields," *Information and Computers,* vol. 6, pp. 79-98, Mar. 1963.

[4] E.R. Berlekamp, *Algebraic Coding Theory.* McGraw-Hill, 1968.

[5] R.E. Blahut, *Fast Algorithms for Digital Signal Processing.* Addison-Wesley, 1985.

[6] T.A. Gulliver, M. Serra, and V.K. Bhargava, "The Generation of Primitive Polynomials in $GF(q)$ with Independent Roots and Their Application for Power Residue Codes, VLSI Testing and Finite Field Multipliers Using Normal Bases," *Int'l J. Electronics,* vol. 71, no. 4, pp. 559-576, 1991.

[7] J.H. Guo and C.L. Wang, "Systolic Array Implementation of Euclid's Algorithm for Inversion and Division in $GF(2^m)$," *IEEE Trans. Computers,* vol. 47, no. 10, pp. 1161-1167, Oct. 1998.

[8] A. Halbutogullari and C.K. Koc, "Mastrovito Multiplier for General Irreducible Polynomials," *IEEE Trans. Computers,* vol. 49, no. 5, pp. 503-518, May 2000.

[9] M.A. Hasan, M.Z. Wang, and V.K. Bhargava, "Modular Construction of Low Complexity Parallel Multipliers for a Class of Finite Fields $GF(2^m)$," *IEEE Trans. Computers,* vol. 41, no. 8, pp. 962-971, Aug. 1992.

[10] T. Itoh and S. Tsujii, "Structure of Parallel Mutipliers for a Class of Fields $GF(2^m)$," *Information and Computation,* vol. 83, pp. 21-40, 1989.

[11] R. Lidl and H. Niederreiter, *Introduction to Finite Fields and Their Applications.* Cambridge Univ. Press, 1994.

[12] E.D. Mastrovito, "VLSI Designs for Multiplication over Finite Fields $GF(2^m)$," *Proc. Sixth Symp. Applied Algebra, Algebraic Algorithms, and Error Correcting Codes (AAECC-6),* pp. 297-309, July 1988.

[13] E.D. Mastrovito, "VLSI Architectures for Computation in Galois Fields," PhD thesis, Linkoping Univ., Linkoping, Sweden, 1991.

[14] A.J. Menezes, I.F. Blake, X. Gao, R.C. Mullin, S.A. Vanstone, and T. Yaghoobian, *Applications of Finite Fields.* Kluwer Academic, 1993.

[15] Nat'l Inst. of Standards and Technology, *Digital Signature Standard,* FIPS Publication 186-2, Jan. 2000.

[16] I.S. Reed and X. Chen, *Error-Control Coding for Data Networks.* Kluwer Academic, 1999.

[17] A. Reyhani-Masoleh and M.A. Hasan, "A New Efficient Architecture of Mastrovito Multiplier over $GF(2^m)$," *Proc. 20th Biennial Symp. Comm.,* pp. 59-63, May 2000.

[18] A. Reyhani-Masoleh and M.A. Hasan, "Low Complexity Bit Parallel Architectures for Polynomial Basis Multiplication over $GF(2^m)$," Technical Report CORR 2003-19, Dept. of C & O, Univ. of Waterloo, Canada, July 2003.

[19] A. Reyhani-Masoleh and M.A. Hasan, "On Low Complexity Bit Parallel Polynomial Basis Multipliers," *Proc. Cryptographic Hardware and Embedded Systems (CHES 2003),* pp. 189-202, Sept. 2003.

[20] F. Rodriguez-Henriquez and C.K. Koc, "Parallel Multipliers Based on Special Irreducible Pentanomials," *IEEE Trans. Computers,* vol. 52, no. 12, pp. 1535-1542, Dec. 2003.

[21] P.A. Scott, S.J. Simmons, S.E. Tavares, and L.E. Peppard, "Architectures for Exponentiation in $GF(2^m)$," *IEEE J. Selected Areas in Comm.,* vol. 6, no. 3, pp. 578-586, Apr. 1988.

[22] G. Seroussi, "Table of Low-Weight Binary Irreducible Polynomials," HP Labs Tech. Report HPL-98-135, Aug. 1998.

[23] L. Song and K.K. Parhi, "Low Complexity Modified Mastrovito Multipliers over Finite Fields $GF(2^M)$," *Proc. IEEE Int'l Symp. Circuits and Systems (ISCAS-99),* pp. 508-512, 1999.

[24] B. Sunar and C.K. Koc, "Mastrovito Multiplier for All Trinomials," *IEEE Trans. Computers,* vol. 48, no. 5, pp. 522-527, May 1999.

[25] H. Wu, "Bit-Parallel Finite Field Multiplier and Squarer Using Polynomial Basis," *IEEE Trans. Computers,* vol. 51, no. 7, pp. 750-758, July 2002.

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

[27] Y. Wu and M.I. Adham, "Scan-Based BIST Fault Diagnosis," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems,* vol. 18, no. 2, pp. 203-211, Feb. 1999.

[28] T. Zhang and K.K. Parhi, "Systematic Design of Original and Modified Mastrovito Multipliers for General Irreducible Polynomials," *IEEE Trans. Computers,* vol. 50, no. 7, pp. 734-748, July 2001.

Arash Reyhani-Masoleh received the BSc degree from Iran University of Science and Technology in 1989, the MSc degree from the University of Tehran in 1991, both with the first rank in electrical and electronic engineering, and the PhD degree in electrical and computer engineering from the University of Waterloo in 2001. From 1991 to 1997, he was with the Department of Electrical Engineering, Iran University of Science and Technology. Since June 2001, he has been a postdoctoral fellow with the Centre for Applied Cryptographic Research, University of Waterloo. His current research interests include algorithms and VLSI architectures for computations in finite fields, fault-tolerant computing, and error-control coding. He was awarded an NSERC (Natural Sciences and Engineering Research Council of Canada) postdoctoral fellowship in 2002. He is a member of the IEEE and the IEEE Computer Society.

M. Anwar Hasan received the BSc degree in electrical and electronic engineering, the MSc degree in computer engineering, both from the Bangladesh University of Engineering and Technology, in 1986 and 1988, respectively, and the PhD degree in electrical engineering from the University of Victoria in 1992. Since 1993, he has been with the Department of Electrical and Computer Engineering, University of Waterloo, where he is now a professor. At the University of Waterloo, he is also a member of the Centre for Applied Cryptographic Research and the Center for Wireless Communications. His current research interests include algorithms and architectures for computations in Galois fields, data security and reliability, and digital communication networks. From January to December of 1999, he was on sabbatical with Motorola Labs., Schaumburg, Ilinois. He is a recipient of the Raihan Memorial Gold Medal. At the University of Victoria, he was awarded the President's Research Scholarship four times. He has served on the program and executive committees of several conferences and, currently, he is an associate editor of the *IEEE Transactions of Computers*. He is a senior member of the IEEE and a licensed professional engineer of Ontario.