An Efficient Squaring Architecture for $GF(2^m)$ and its Applications in Cryptographic Systems *

Gerardo Orlando and Christof Paar†

Electronic Letters, June 2000, vol. 36, no. 13, pp. 1116-1117

Abstract

This contribution describes a squarer architecture for standard basis field representation. This architecture is based on the observation that one can transform a squaring operation in $GF(2^m)$ into an addition and a multiplication of two elements of special form, which computational time depends on the form of the field polynomial.

1 Introduction

This work proposes a new squaring architecture specially targeted for systems requiring the efficient computations of squares and multiplications, such as cryptographic system. For these system, the current architectural alternatives are to use multipliers to compute both multiplications and squares or to use independent multipliers and squarers. As it will be demonstrated

*This research was supported in part through NFS CAREER award CCR-9733246

†ECE Department, Worcester Polytechnic Institute, 100 Institute Rd., Worcester, MA 01609. E-mail: orlandog@WPI.EDU, christof@ece.wpi.edu .

here, it is possible to transform squaring operation into a multiplication by a constant and a sum, and to efficiently compute these operations with least-significant digit-serial multiplier (LSD-multiplier) and what is referred to here as a squaring adapter. The architectures of these two components is very regular, of low complexity, and independent of field polynomials.

When compared against leading squaring architectures such as [1, 2, 3], the most significant features of the proposed architecture are its support for the computation of multiplications and squares; its architecture, as that of [3], is independent of the field polynomials; and its performance and complexity are scalable. The most significant difference from traditional architectures is that its processing time is a function of the form of the field polynomial, and it is often defined by the second most significant coefficient of the field polynomial.

2 Mathematical Background

This work considers arithmetic in extension fields of characteristic two using a standard basis representation. Standard basis representation uses the basis defined by the set of elements $\{1, \alpha, \alpha^2, \ldots, \alpha^{m-1}\}$, for which, α is a root of the irreducible polynomial $F(x)$. For additional information on this subject the reader is referred to [4].

The squaring of a field element $A = \sum_{i=0}^{m-1} a_i \alpha^i$ is expressed by Equation (1). It can be verified that the squaring operation can be expressed as multiplication by a constant and a sum as shown by Equation (2), with A', B', and C' as defined in Equations (3)–(5).

$$A^2 \equiv \sum_{i=0}^{m-1} a_i \alpha^{2i} \bmod F(\alpha) \tag{1}$$

$$\equiv \sum_{i=\lceil m/2 \rceil}^{m-1} a_i \alpha^{2i} \bmod F(\alpha) + \sum_{i=0}^{\lceil m/2 \rceil - 1} a_i \alpha^{2i}$$

$$A^2 \equiv A'B' \bmod F(\alpha) + C' \tag{2}$$

$$A' = \sum_{i=0}^{\lfloor m/2 \rfloor - 1} a_{i+\lceil m/2 \rceil} \alpha^{2i} \tag{3}$$

$$B' \equiv \alpha^{2\lceil m/2 \rceil} \bmod F(\alpha) \tag{4}$$

$$C' = \sum_{i=0}^{\lceil m/2 \rceil - 1} a_i \alpha^{2i} \tag{5}$$

Note that the values of A' and C' depend on the value of A while the value of B' depends exclusively on the field polynomial. For irreducible polynomials $F(x) = x^m + \sum_{i=0}^{k} f_i x^i$, B' can be expressed as shown in Equation (6). For what follows it is crucial to observe that for many practical applications, the degree of B' is relatively low because often $k << m$.

$$B' = \begin{cases} \sum_{i=0}^{k} f_i \alpha^i, & \text{even } m \\ \sum_{i=0}^{k} f_i \alpha^{i+1}, & \text{odd } m \text{ and } k < m-1 \\ \sum_{i=1}^{k} (f_i + f_{i-1}) \alpha^i + f_0, & \text{odd } m \text{ and } k = m-1 \end{cases} \tag{6}$$

3 A New Squaring Architecture

A block diagram of the new squaring architecture is shown in Figure 1. This architecture consists of an LSD-multiplier, such as that introduced in [5], and a squaring adapter. The multiplier is used to compute products and

sums and the squaring adapter is used to generate and propagate operands to the multiplier.

LSD-multipliers implement variations of the multiplication algorithm illustrated by Algorithm 1. The inputs to this algorithm are the field elements $A = \sum_{i=0}^{m} a_i \alpha^i$ and $B = \sum_{i=0}^{m} B_i \alpha^{Di}$, where $B_i = \sum_{j=0}^{D-1} b_{Di+j} \alpha^j$. Note that the field element B is expressed in $\lceil m/D \rceil$ digit, where each digit is represented by D bits. Also, note that the multiplication finishes when the most significant, nonzero digit of B, B_{k_B}, is processed.

Algorithm 1: LSD multiplication

1. For $i = 0$ to k_B do

1.1 $C = B_i * (A * \alpha^{Di} \bmod F(\alpha)) + C$

2. $C = C \bmod F(\alpha)$

¿From Algorithm 1 it is evident that an LSD-multiplier can compute the operation described by Equation (2) by first multiplying A' and B' and then adding to it the product of C' and 1. For the computation of these products, the host system provides the squaring adapter with the operands A and B'. During the computation of the product of A' and B', the squaring adapter generates A' according to Equation (3) and forwards it along with B' to the multiplier. During the accumulation of C', the squaring adapter generates C' according to Equation (5) and forwards it along with the 1 operand to the multiplier.

The computation of a square requires a multiplication and a sum. The computation of the sum requires one clock cycle and the computation of the multiplication requires $\lceil (deg(B') + 1)/D \rceil$ clock cycles. The squaring operation requires $\lceil (k+1)/D \rceil + 1$ clock cycles when m is even, $\lceil (k+2)/D \rceil + 1$

clock cycles when m is odd and $k < m - 1$, and $\leq \lceil (k+1)/D \rceil + 1$ clock cycles when m is odd and $k = m - 1$.

The complexity of the squaring adapter is approximately $3.5m + D$ 2-input gates and its critical path delay is 4 gates. The complexity of an LSD-multiplier depends on its architecture and irreducible polynomial support. As a reference, a realization of an LSD-multiplier documented in [5] that supports field polynomials with order $k \leq m - D$ with h programmable coefficients requires approximately $2Dm + 7m + 4Dh$ gates and $3m + D + h$ registers. (This estimate considers system I/O and accumulator reset, which are not considered in [5].)

4 Squaring Processing Time for Cryptosystems

We conclude by analyzing the suitability of the squarer architecture for cryptographic applications. Table 1 summarizes the squaring-to-multiplication processing time ratio, T_{sq}/T_{mul}, for the field polynomials suggested by the cryptographic standard [6], assuming the use of an LSD-multipliers with $D = 1$. The interpretation of the table is as follows: 32% of all fields in the range considered allow squaring at least 10 times (1/0.1) as fast as multiplication, 23% between 5 times and 10 times (1/0.2)as fast, etc.

References

[1] H. Wu, "Low complexity bit-parallel finite field arithmetic using polynomial basis," in *Workshop on Cryptographic Hardware and Embedded Sys-*

tems (CHES '99) (C. Koc and C. Paar, eds.), vol. LNCS 1717, Springer-Verlag, August 1999.

[2] C. Paar, P. Fleischmann, and P. Soria-Rodriguez, "Fast arithmetic for public-key algorithms in Galois fields with composite exponents," *IEEE Transactions on Computers*, vol. 48, pp. 1025–1034, October 1999.

[3] S. K. Jain, L. Song, and K. K. Parhi, "Efficient semisystolic architectures for finite-fields arithmetic," *IEEE Transactions on Very Large Scale Integration (VLSI) Sytems*, vol. 6, pp. 101–113, March 1998.

[4] R. Lidl and H. Niederreiter, *Finite Fields*, vol. 20 of *Encyclopedia of Mathematics and its Applications*. Reading, Massachusetts: Addison-Wesley, 1983.

[5] L. Song and K. K. Parhi, "Low-energy digit-serial/parallel finite field multipliers," *Journal of VLSI Signal Processing Systems*, vol. 2, no. 22, pp. 1–17, 1997.

[6] A. X9.62-199x, "Public Key Cryptography For The Finantial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)," January 1998. Approved January 7, 1999.

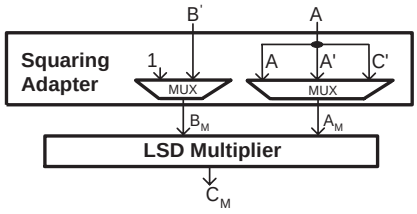

Figure 1: New squaring architecture

Table 1: Distribution of squaring-to-multiplication processing time ratios for fields in range $m = 160 \ldots 1024$

T_{sq}/T_{mul}	Dist. (%)	Cumm. dist. (%)
$0.05 \ldots 0.10$	32	32
$0.10 \ldots 0.20$	23	55
$0.20 \ldots 0.30$	16	71
$0.30 \ldots 0.50$	29	100