Semi-systolic Architecture for Modular Multiplication over GF(2^m)

Hyun-Sung Kim[1] and Il-Soo Jeon[2]

[1] Kyungil University, School of Computer Engineering,
712-701, Kyungsansi, Kyungpook Province, Korea
[2] Kumho Nat'l Inst. of Tech., School of Electronic Eng.,
730-703, Gumi, Kyungpook Province, Korea

Abstract. This paper proposes a new algorithm and an architecture for it to compute the modular multiplication over GF(2^m). They are based on the standard basis representation and use the property of irreducible all one polynomial as a modulus. The architecture, named SSM(Semi-Systolic Multiplier) has the critical path with 1-D_{AND}+1-D_{XOR} per cell and the latency of m+1. It has a lower latency and a smaller hardware complexity than previous architectures. Since the proposed architecture has regularity, modularity and concurrency, they are suitable for VLSI implementation.

1 Introduction

The arithmetic operations in the finite field have several applications in error-correcting codes, cryptography, digital signal processing, and so on [1]. Information processing in such areas usually requires performing multiplication, inverse/division, and exponentiation. Among these operations, the modular multiplication is known as the basic operation for public key cryptosystems over GF(2^m) [2-3].

Numerous architectures for modular multiplication in GF(2^m) have been proposed in [2-8] over the standard basis. Wang et al. in [5] proposed two systolic architectures with the MSB-first fashion with less control problems as compared to [4]. Jain et al. proposed semi-systolic multipliers [6]. Its latency is smaller than those of the other standard-basis multipliers. Kim in [7] proposed a bit-level systolic array with a simple hardware complexity with the LSB-first modular multiplication. Thus, further research for efficient circuit for cryptographic applications is needed. To reduced the system complexity, Itoh and Tsujii designed two low-complexity multipliers for the class of GF(2^m), based on the irreducible AOP (All One Polynomial) and the irreducible equally spaced polynomial [8]. Later, Kim in [2] proposed various AOP architectures based on LFSR(Linear Feedback Shift Register) architecture.

This paper proposes a new algorithm and a parallel-in parallel-out semi-systolic array architecture to compute the modular multiplication over finite field GF(2^m). They are based on the standard basis representation and use the property of irreducible AOP as a modulus. Let D_{AND} and D_{XOR} be the latency of AND and XOR gate, respectively. The architecture has the critical path with 1-D_{AND}+1-D_{XOR} per cell and the latency of m+1. It could be used to secure cryptosystem application.

V.S. Sunderam et al. (Eds.): ICCS 2005, LNCS 3516, pp. 912 – 915, 2005.

2 Semi-systolic Architecture

GF(2^m) is the finite extension field of finite field GF(2) [2]. An arbitrary element A over GF(2^m) can be represented with $\{1, \alpha, \alpha^2, \cdots, \alpha^{m-1}\}$, which is based on the standard basis, i.e., $A=A_{m-1}\alpha^{m-1}+A_{m-2}\alpha^{m-2}+\cdots+A_1\alpha+A_0$. A polynomial of the form $f(x)= f_m x^m+f_{m-1}x^{m-1}+\cdots + f_1 x+ f_0$ is called an irreducible polynomial if and only if a divisor of $f(x)$ is 1 or $f(x)$. Assume that a polynomial of the form $f(x)= f_m x^m+f_{m-1}x^{m-1}+\cdots + f_1 x+ f_0$ over GF(2) is called an AOP (All One Polynomial) with degree m if $f_i=1$ for $i=0,1, \cdots ,m$. It has been shown that an AOP is irreducible if and only if $m+1$ is the prime and 2 is the primitive modulo $m+1$. Let a set $\{1, \alpha, \alpha^2, \cdots ,\alpha^{m-1}\}$ be generated by α which is a root of AOP $f(x)$ and be the standard basis. In the standard basis, an element A over GF(2^m) is presented by $A=A_{m-1}\alpha^{m-1}+A_{m-2}\alpha^{m-2}+\cdots +A_1\alpha+A_0$. A set with $\{1, \alpha, \alpha^2, \cdots, \alpha^{m-1}, \alpha^m\}$ is called an extended basis of $\{1, \alpha, \alpha^2, \cdots,\alpha^{m-1}\}$. In the extended basis, an element a over GF(2^m) is represented by $a=a_m\alpha^m +a_{m-1}\alpha^{m-1}+\cdots+a_1\alpha+a_0$ with $A_i=a_m+a_i$ ($0\le i \le m-1$). Thus, an element over GF(2^m) has two different representations. Let $F(x)=x^m+x^{m-1}+\cdots +x+1$ be an irreducible AOP of degree m: and let α be a root of $F(x)$. i.e., $F(\alpha)=\alpha^m+\alpha^{m-1}+\cdots +\alpha+1$. Then, we have $\alpha^m=\alpha^{m-1}+\cdots +\alpha+1$, $\alpha^{m+1}=1$.

The multiplication of elements a and b over GF(2^4) in the extended basis can be performed by ab mod p with $p=\alpha^{m+1}+1$ which applied the property of AOP as a modulus. Let the result of this multiplication, ab mod p, be $r=r_m\alpha^m+r_{m-1}\alpha^{m-1}+\cdots+r_1\alpha+r_0$. The recurrence equation for the MSB first algorithm with the property of AOP is as follows: $r = ab \bmod p=\{\cdots[[ab_m]\alpha \bmod p+ab_{m-1}]\alpha \bmod p+\cdots+ab_1\}\alpha \bmod p+ab_0$. From the equation, a new algorithm to compute ab mod p can be derived as following Algorithm 1.

[Algorithm 1] Modular Multiplication

Input : $a=(a_m,a_{m-1},\cdots,a_1,a_0)$, $b=(b_m,b_{m-1},\cdots,b_1,b_0)$

Output : $r=ab \bmod p$

Initial value : $r^{m+1}=(r_m,r_{m-1}, \cdots,r_1,r_0) =(0,0, \cdots,0,0)$

Step 1 for $i=m$ to 0

Step 2 $r^i=Circular\_Left(r^{i+1})+ ab_i$

where *Circular_Left*(x) is the 1-bit-left-circular shift of x and r^i is used to represent the i-th intermediate result for the final result r. In the above algorithm, the modular reduction is performed just by using 1-bit-left-circular-shift operation. Specially, all the operations in the for loop can be performed bit by bit in parallel.

Let a, b, and b^2 be an elements in GF(2^4). Then a and b with an extended basis $\{1, \alpha,\alpha^2,\alpha^3,\alpha^4\}$ can be represented as follows: $a=a_4\alpha^4+a_3\alpha^3+a_2\alpha^2+a_1\alpha+a_0$, $b=b_4\alpha^4+b_3\alpha^3 +b_2\alpha^2+b_1\alpha+b_0$.

When $p=\alpha^5+1$ is used as a modular in the extended basis, we have

$$\begin{aligned} r &= ab \bmod p \\ &= a(b_4\alpha^4+b_3\alpha^3+b_2\alpha^2+b_1\alpha+b_0) \bmod p \\ &= \{\cdots[[ab_4]\alpha \bmod p+ab_3]\alpha \bmod p+\cdots+ab_1\}\alpha \bmod p+ab_0 \\ &= r_4\alpha^4+r_3\alpha^3+r_2\alpha^2+r_1\alpha+r_0. \end{aligned}$$

Fig.1 shows a multiplier named SSM based on Algorithm 1 over GF(2^4). SSM is composed of $(m+1)(m+1)$ basic cells. It is the parallel architecture which a_i and b_i $(0 \leq i \leq m)$ are inputted at the same time. The $(m+1)$-bits of data a are inputted from the top row and transmitted to the adjacent cells following each row. But the data b_i in each row is broadcasted to all cells in the same row at the same time. Let D_{AND} and D_{XOR} be the latency of AND and XOR gate, respectively. SSM has a critical path with 1-D_{AND}+1-D_{XOR} per cell. SSM in Fig. 1 can be generalized for arbitrary m as well as m=4. Fig. 2 shows the basic cells for SSM.

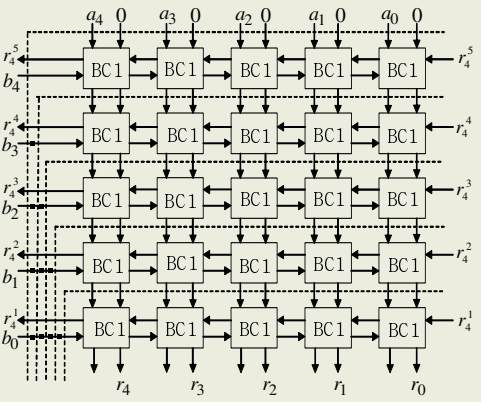

Fig. 1. SSM over GF(2^4)

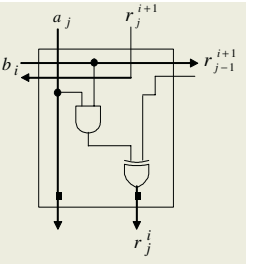

Fig. 2. Basic cell of SSM

3 Comparison and Analysis

Table 1 shows the comparison between the proposed architecture and pervious two architectures.

Table 1. Comparisons

Properties	Jain [6]	Kim [2]	SSM
Basic architecture	Semi-systolic	LFSR	Semi-systolic
Irreducible polynomial	Generalized	AOP	AOP
Number of cell	m^2	m+1	$(m+1)^2$
Cell complexity	2 AND, 2 XOR 3 latches	1 AND, 1 XOR 2 REG, 1 MUX	1 AND, 1 XOR 2 latches
Latency	m	$2m$+1	m+1
Critical path	1-D_{AND} +1-D_{XOR}	1-D_{AND}+ log m(1-D_{XOR})	1-D_{AND} +1-D_{XOR}

It is assumed that AND and XOR represent 2-input AND and XOR gates, respectively, and latch for 1-bit latch. Let D_{AND} and D_{XOR} be the latency of AND and XOR gate, respectively. As a result, the proposed architecture, SSM, has lower latency and smaller complexity than previous architectures in [6] and [2].

4 Conclusions

This paper proposed a new algorithm and a parallel-in parallel-out semi-systolic array architecture to compute the modular multiplication over finite field GF(2^m). The property of irreducible AOP was used as a modulus to get a better hardware and time complexity. Proposed architecture has lower latency and smaller hardware complexity than previous architectures as shown in Table 1. Since SSM has regularity, modularity and concurrency, they are suitable for VLSI implementation.

References

[1] D.E.R.Denning, Cryptography and data security, Addison-Wesley, MA, 1983.
[2] H.S.Kim, Bit-Serial AOP Arithmetic Architecture for Modular Exponentiation, PhD. Thesis, Kyungpook National University, 2002.
[3] S.W.Wei, "VLSI architectures for computing exponentiations, multiplicative inverses, and divisions in GF(2^m)," IEEE Trans. Circuits and Systems, 44, 1997, pp.847-855.
[4] C.S.Yeh, S.Reed, and T.K.Truong, "Systolic multipliers for finite fields GF(2^m)," IEEE Trans. Comput., vol.C-33, Apr. 1984, pp.357-360.
[5] C.L.Wang and J.L.Lin, "Systolic Array Implementation of Multipliers for Finite Fields GF(2^m)," IEEE Trans. Circuits and Systems, vol.38, July 1991, pp796-800.
[6] S. K. Jain and L. Song, "Efficient Semisystolic Architectures for finite field Arithmetic," IEEE Trans. on VLSI Systems, vol. 6, no. 1, Mar. 1998.
[7] H.S.Kim, "Efficient Systolic Architecture for Modular Multiplication over GF(2^m)," will be appeared in PARA 2004 proceeding, 2005.
[8] T. Itoh and S. Tsujii, "Structure of parallel multipliers for a class of finite fields GF(2^m)," *Info. Comp.*, vol. 83, pp.21-40, 1989.
[9] S.Y.Kung, VLSI Array Processors, Prentice-Hall, 1987.