第 34 卷 第 5 期 计 算 机 工 程 2008 年 3 月
Vol.34 No.5 Computer Engineering March 2008
·安全技术· 文章编号：1000—3428(2008)05—0163—03 文献标识码：A 中图分类号：TP332

GF(2^m)域上可配置ECC算术模块的设计与实现

戴紫彬，卫学陶，陈 韬
(解放军信息工程大学电子技术学院，郑州 450004)

摘 要：提出一种应用于可配置椭圆曲线密码体制的有限域多项式算术模块结构，乘法器基于已有的 digit-serial 结构乘法器，利用局部并行的 bit-parallel 结构，省去了模约简电路，使乘法器可适用于任意不可约多项式。平方器结构利用 LSB 或 LSD 乘法器以及加法器来计算模平方，通过数据接口控制输入数据的格式，可以满足不同域值有限域点乘运算的需求。

关键词：有限域；二进制有限域；椭圆曲线密码体制

Design and Implementation of Reconfigurable ECC Arithmetic Unit in GF(2^m)

DAI Zi-bin, WEI Xue-tao, CHEN Tao
(Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450004)

【Abstract】A finite field polynomial arithmetic unit architecture is proposed in this paper for reconfigurable ECC. The multiplier based on previous digit-serial multiplier architecture uses bit-parallel architecture of local parallel to eliminate reduction modulo circuit effectively, and the multiplier architecture is the same with arbitrary irreducible polynomials. The squaring architecture computes squares by using an LSB, or an LSD multiplier with an adder. Data format of import is controlled through data interface, which achieves requirements of point multiplication for different finite fields.

【Key words】 Galois Field(GF); GF(2^m); ECC

1 概述

作为一种具有广泛应用前景的密码技术，椭圆曲线密码系统越来越受关注，特别是可配置椭圆曲线密码系统由于具有安全性高、运算灵活和应用范围可扩展等特点，得到了大量的研究[1-3]。高性能可配置椭圆曲线处理器(ECP)由主控制模块(Main Controller, MC)、运算单元控制模块(Arithmetic Unit Controller, AUC)、可配置算术模块(Reconfigurable Arithmetic Unit, RAU)组成[2]。因为MC和AUC实现的复杂性较RAU小得多，所以RAU的复杂性决定了可配置ECP的复杂性。文献[1]提到的有限域正规基算术模块需要以结构重构信号来动态配置AND与XOR阵列，这样会影响算术模块的执行性能。文献[2-3]的有限域多项式基算术模块以域多项式来设计专用的模约简电路。适应不同曲线和有限域值的运算就要求设计不同模约简电路来实现动态配置，因此，模约简成为影响性能的一个关键运算。

本文设计了一种基于多项式基、不需要模约简电路的可配置算术模块，给出了各运算单元的电路设计方案。在省去独立模约简模块之后，无须针对多种二进制有限域和域多项式设计专门的电路，满足了可配置的设计需求，提高了资源利用率。并且在进行多项式乘法时利用基本的 x 乘法电路完成模域多项式的运算，省去了一个大操作数模运算过程，因此，降低了设计难度，大大提高了运算效率。

2 可配置算术模块的结构

二进制有限域上的运算包括模加、模减、模平方、模乘、模除。逆元素可以通过有限域上的模平方和模乘来实现，因此，二进制有限域中不需要单独实现模除运算[2]。整个算术模块结构如图 1 所示，寄存器组存储椭圆曲线参数、预计算值和临时值，主处理器装载曲线参数到寄存器组，并从中提取点乘结果，各运算单元从寄存器组中读取操作数并将操作结果写入其中。

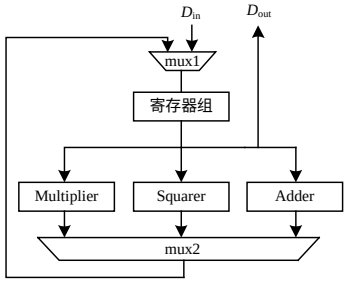

图 1 可配置算术模块的结构

为了支持不同密钥长度ECC的运算需求，达到可配置的目的，本文使用数据接口将操作数变换成高位对齐、低位补零的形式，同时把乘法器、平方器和加法器设计为满足最大域值运算需求的运算单元，如NIST建议使用的有限域为 $GF(2^{163})$, $GF(2^{233})$, $GF(2^{283})$, $GF(2^{409})$和$GF(2^{571})$[4]，给出了它们的域多项式，这样可将运算单元都设计为适应最大域值 571 位的结构，并且能满足不同域多项式的需求。

3 加法器的设计

假设 $GF(2^m)$ 是不可约多项式 $F(x)$ 产生的有限域，

作者简介：戴紫彬(1966－)，男，教授、博士，主研方向：信息安全，军事通信；卫学陶，硕士研究生；陈 韬，助教、硕士
收稿日期：2007-05-10 **E-mail：**ahtoh2000@163.com

$F(x)=x^m+\sum_{i=0}^{m-1}f_i x^i$，域元素为 $A(x)=\sum_{i=0}^{m-1}a_i x^i$， $B(x)=\sum_{i=0}^{m-1}b_i x^i$，运算结果为 $C(x)=\sum_{i=0}^{m-1}c_i x^i$ ，则它们的和可以写成如下形式：

$$C(x)=A(x)+B(x)=\sum_{i=0}^{m-1}(a_i+b_i)x^i \tag{1}$$

二进制有限域上的模加运算非常适合于硬件实现。式(1)中的位加a_i+b_i是模 2 加的，在硬件上，可以通过异或(XOR)操作来实现这种模 2 加，并且不需要进位链。

加法器可在异或门阵列的组合逻辑时延后完成，电路结构总共需要异或门m，关键路径延迟为T_X。如果不加特别说明，本文的T_A表示“与门”延迟，T_X表示“异或门”延迟。

4 乘法器设计

有限域元素的乘法定义如下：

$$c_{m-1}x^{m-1}+\ldots+c_1x+c_0=\begin{pmatrix}(a_{m-1}x^{m-1}+\ldots+a_1x+a_0)\times\\(b_{m-1}x^{m-1}+\ldots+b_1x+b_0)\end{pmatrix}\bmod F(x) \tag{2}$$

本文采取串并混合结构的 Digit-Serial 乘法器，首先将$B(x)$分成 S 段，每段 D 位长，$S=[m/D]+1$。

$$B_i(x)=\begin{cases}\sum_{j=0}^{D-1}b_{iD+j}\cdot x^j & 0\le i\le s-2\\ \sum_{j=0}^{(m\bmod D)-1}b_{iD+j}\cdot x^j & i=s-1\end{cases} \tag{3}$$

则模乘公式可以转化为

$$A(x)B(x)\bmod F(x)=A(x)\begin{pmatrix}x^{(s-1)D}B_{s-1}(x)+\cdots+\\x^DB_1(x)+B_0(x)\end{pmatrix}\bmod F(x) \tag{4}$$

本文根据式(4)设计了一种LSD乘法器，将乘数B从B_0连续送到B_{s-1}。时钟数的减少和资源的增加与D有关。D值越大，周期减少得越多，但是资源的消耗也越多，当D=1 时，乘法器就变成了LSB乘法器。在椭圆曲线密码体制中，为满足安全性要求，用于计算的有限域值通常为质数，如果为合数，即在复合域上椭圆曲线密码体制有安全性问题[5]，则在Digit-Serial结构乘法器中一个时钟送入D位的被乘数$B(x)$，而域值m通常不是D的倍数，因此，B_{s-1}高位须补 0，使被乘数位数成为D的倍数，以便被乘数所有操作数都能输入乘法器。

对于LSD乘法器而言，需要将乘数B由低位补 0 变换为高位补 0；被乘数A、域多项式F仍为低位补 0，以达到最大域值乘法器的操作数要求；输出结果同样是高位对齐。如果乘数B低位补 0，则乘法开始几次送入的B_i为 0，与运算无关，增大了乘法运算所需时钟数。因此，在乘法开始前要根据所进行的有限域运算，控制输入乘数B的分组数，以避免浪费时钟数。

根据式(4)，整个乘法器可以分解为$A(x)$和$B_i(x)$的部分模乘运算、部分积运算结果与恒量x^D的模乘运算以及中间结果的模加运算。

部分模乘运算的定义如下：

$$B_iA\bmod F(x)=(b_0+b_1x+\cdots+b_{D-1}x^{D-1})A\bmod F(x)=\\(b_0A+b_1xA+\cdots+b_{D-1}x^{D-1}A)\bmod F(x) \tag{5}$$

与恒量x^D的模乘运算可以如下定义：

$$\beta=x^D\alpha\bmod F(x) \tag{6}$$

在运算前要将被乘数 $B(x)$分段，同时在被乘数 $B(x)$的高位补上 $D-(m\bmod D)$ 个寄存器，取固定值为 0。由以上分析，模乘结果可以用如下迭代算法实现，令 $K_j=x^{jD}A\bmod F(x)$，$C_j=C_{j-1}+K_jB_j\bmod F(x)$， $j=0,1,\cdots,S-1$且 $C_{-1}=0$，则

$$C=C_{S-1} \tag{7}$$

整个乘法器结构如图 2 所示，乘法器结构由部分积生成器 $M1$、m 位模加法器 $M2$、恒量乘法器 $M3$、移位寄存器、$A(i)$寄存器和 m 位寄存器组成，其中，部分积生成器和恒量乘法器是主要的设计单元。

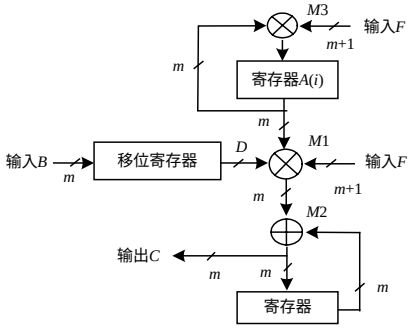

图 2　乘法器结构

4.1 恒量乘法器 *M3* 的设计

恒量乘法器实现式(6)的计算，为得到运算结果，采用x乘法电路实现，对于具有任意不可约多项式的$GF(2^m)$域上的乘法，假设用多项式x乘以 $\alpha(x)$ 有

$$\beta(x)=(\alpha_{m-1}x^{m-1}+\alpha_{m-2}x^{m-2}+\cdots+\alpha_1x+\alpha_0)x\bmod\left(x^m+f_{m-1}x^{m-1}+\cdots+f_1x+f\right) \tag{8}$$

$$\beta(x)=(\alpha_{m-1}f_{m-1}+\alpha_{m-2})x^{m-1}+(\alpha_{m-1}f_{m-2}+\alpha_{m-3})x^{m-2}+\cdots+(\alpha_{m-1}f_1+\alpha_0)x+\alpha_{m-1}f_0 \tag{9}$$

式(9)可以用简单的电路实现。对于 $\beta=x^D\alpha\bmod F(x)$ 可以用D个x乘法电路串联起来实现，电路结构总共需要的与门为mD，异或门为$(m\text{-}1)D$。关键路径延迟为$D(T_A+T_X)$。

4.2 部分积生成器 *M1* 的设计

部分积生成器实现式(5)的计算，部分积生成器可采用 bit-parallel 结构，由恒量乘法器、与门阵列和异或门阵列组成，如图 3 所示，其复杂度由分组长度 D 决定。

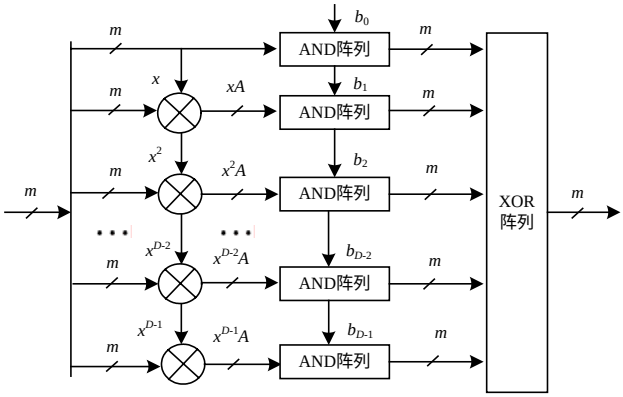

图 3　部分积生成器结构

恒量乘法器实现运算 $x^q\alpha\bmod F(x)$，q=1,2,…, D-1，结构与恒量乘法器 $M3$ 类似，与门阵列和异或门阵列的电路结构都比较简单，电路结构总共需要与门为 $mD(D+1)/2$，异或门为 $(Dm-D+2)(D-1)/2$。关键路径延迟为 $(D-1)(T_A+T_X)+T_A+((D/2)-1)T_X$。

4.3 乘法器的性能分析

乘法器中的 $M1$, $M2$, $M3$ 都可以在一个时钟内完成运算；寄存器的输出是和移位寄存器的分组操作同时得到的，它们可以并行执行。这样在流水状态下，整个乘法器的模乘执行时间由移位寄存器决定，即被乘数 $B(x)$的分组数 S，因此，

一次模乘的执行时间为 S 个时钟周期。电路结构总共需要与门为 $mD(D+3)/2$，异或门为 $(m-1)(D^2-2)/2+D(m+1)/2$，关键路径延迟为 $2DT_A+(5D-2)/2T_X$。

5 平方器的设计

平方器的设计基于$GF(2^m)$上的平方操作可以变形为常量模乘和模加的结论[6]，变形后利用LSB或LSD乘法器以及加法器来计算模平方。

设元素 $A(x)=\sum_{i=0}^{m-1}a_i x^i$，则

$$A^2 \bmod F(x)=\sum_{i=0}^{m-1}a_i x^{2i} \bmod F(x)=\sum_{i=\lceil m/2\rceil}^{m-1}a_i x^{2i} \bmod F(x)+\sum_{i=0}^{\lceil m/2\rceil-1}a_i x^{2i} \tag{10}$$

又因为

$$\sum_{i=\lceil m/2\rceil}^{m-1}a_i x^{2i} \bmod F(x)=\sum_{i=0}^{\lceil m/2\rceil-1}a_{i+\lceil m/2\rceil}x^{2(i+\lceil m/2\rceil)} \bmod F(x)=\sum_{i=0}^{\lceil m/2\rceil-1}a_{i+\lceil m/2\rceil}x^{2i}x^{2\lceil m/2\rceil} \bmod F(x)=A'B' \bmod F(x) \tag{11}$$

其中，

$$A'=\sum_{i=0}^{\lceil m/2\rceil-1}a_{i+\lceil m/2\rceil}x^{2i}$$

$$B'=x^{2\lceil m/2\rceil} \bmod F(x)$$

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

A' 和 C' 的值依赖 A 的值；B' 的值只依赖域不可约多项式。对于不可约多项式 $F(x)=x^m+\sum_{i=0}^{t}f_i\cdot x^i$，$B'$ 可以表示为

$$B'=\begin{cases}\sum_{i=0}^{t}f_i\cdot x^i & m\text{为偶数}\\ \sum_{i=0}^{t}f_i\cdot x^{i+1} & m\text{为奇数且}t<m-1\\ \sum_{i=1}^{t}(f_i+f_{i-1})\cdot x^i+f_0 & m\text{为奇数且}t=m-1\end{cases} \tag{12}$$

研究结论指出ECC的复合域存在安全性问题[5]，只有域值为素数的扩域$GF(2^m)$被推荐用于ECC，当域值为奇素数，式(12)可以转化为

$$A^2 \bmod f(x)=\sum_{i=(m+1)/2}^{m-1}a_i x^{2i} \bmod F(x)+\sum_{i=o}^{(m-1)/2}a_i x^{2i} \tag{13}$$

其中，$A'=\sum_{i=0}^{(m-3)/2}a_{i+(m+1)/2}x^{2i+1}$；$B'=x^m \bmod F(x)=\sum_{i=0}^{t}f_i\cdot x^i$；$C'=\sum_{i=0}^{(m-1)/2}a_i x^{2i}$。

平方器的结构如图 4 所示，其重点在于有效地计算 $A'B' \bmod F(x)$。由于乘法器的一个操作数和域多项式值为 1 的次高位位置有关，针对NIST指定的有限域及不可约多项式[4]，在域值为素数且在 163~571 之间指定的大部分不可约多项式的都有 $m>>t$。因此，对乘积 $A'B' \bmod F(x)$ 使用LSB乘法器同样可以得到较高效率。

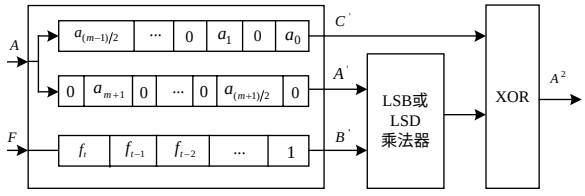

图 4 平方器结构

当使用LSB乘法器计算模平方时，$B'=\sum_{i=0}^{m-1}b_i'x^i$ 可用于乘法器操作数，因为 B' 高位系数为 0，如果使用MSD或MSB乘法器，乘法过程直到高位非 0 系数 b_t' 才能开始，这会浪费大量时钟，所以使用LSB乘法器，平方操作可以在t+2 时钟周期内完成，t+1 个时钟周期用于计算乘积，一个额外的时钟周期用于计算求和。电路结构总共需要与门为 $2m$，异或门为 $3m$。关键路径延迟为T_A+$2T_X$。

当使用 LSD 乘法器计算模平方时，$B'=\sum_{i=0}^{\lceil m/D\rceil-1}B_i'x^{Di}$ 可用于乘法器操作数，B' 的高位非 0 位是 $B'_{\lfloor(t+1)/D\rfloor-1}$，因此，乘积在$[(t+1)/D]$个时钟周期内完成，求和在一个时钟内完成。使用 LSD 乘法器计算平方在$[(t+1)/D]+1$ 时钟内完成。电路结构总共需要与门为 $mD(D+3)/2$，异或门为 $(m-1)(D^2-2)/2+D(m+1)/2+m$。关键路径延迟为 $2DT_A+5D/2T_X$。

6 仿真与验证

本文实现的算术模块最高可以执行 571 位操作数的运算，可以满足所有NIST推荐的有限域及不可约多项式的运算需求。用VHDL语言编写代码，选用在FPGA开发工具QuartusII 5.1 环境下综合编译，并通过了仿真验证，表 1 给出了各运算单元的综合结果。

表 1 各运算单元综合结果

	加法器	平方器	乘法器	
			D=8	D=16
时钟频率/MHz	192.94	186.74	86.32	41.51
占用逻辑单元个数	571	4324	11 468	20 944

7 结束语

本文设计实现了一种适用于可配置 ECC 的有限域算术模块，其优点是不用设计模约简电路，且资源利用率和运算效率较高。可变 $F(x)$结构通过加载不同的域多项式系数以及使用可配置模块输入不同位宽的操作数，使算术模块能够适应基于可配置 ECC 算法的各种点乘运算，满足了 ECC 算法不同安全性的需求。

参考文献

[1] 曾晓洋, 顾震宇, 周晓方, 等. 可重构的椭圆曲线密码系统及其VLSI 设计[J]. 小型微型计算机系统, 2004, 25(7): 1280-1285.

[2] Orlando G, Paar C. A High-performance Reconfigurable Elliptic Curve Processor for GF(2^m)[C]//Proc. of Workshop on Cryptographic Hardware and Embedded Systems. [S. l.]: Springer- Verlag, 2000.

[3] Eberle H, Gura N S, Chang Shan. A Cryptographic Processor for Arbitrary Elliptic Curves over GF(2^m)[C]//Proc. of the IEEE 14th Int'l Conference on Application-specific Systems, Architectures and Processors. Hague, Netherlands: IEEE Press, 2003-06: 444-454.

[4] Department of Commerce, National Institute of Standards and Technology. FIPS 186-2 Digital Signature Standard(DSS)[S]. 2000-01.

[5] Gaudry P, Hess F, Smart N P. Constructive and Destructive Facets of Weil Descent on Elliptic Curves[Z]. (2000-01-17). http://www.hpl.hp.com/techreports/2000/hpl-2000-10.html.

[6] Orlando G, Paar C. An Efficient Squaring Architecture for GF(2^m) and Its Applications in Cryptographic Systems[J]. Electronic Letters, 2000, 36(13): 1116-1117.