

基于 $GF(2^m)$ 上椭圆曲线点乘的实现

陈光化, 张晓莉, 朱景明, 刘 名
(上海大学 微电子研究与开发中心, 上海 200072)

摘 要: 给出了椭圆曲线加密算法的点乘实现. 在实现模乘运算时, 把相乘过程和模约多项式过程结合起来, 以改善运算效率. 片外双口 RAM 的使用, 加快了数据存取速度, 同时通过预留 RAM 空间, 增强了系统的可扩展性. 本设计用 VerilogHDL 语言作为设计工具, 在 synopsys DC Z-2007 03 solaris9 工作平台上, 基于 chartered 0.35 CMOS 的综合库, 50MHz 约束下综合出结果约为 18657 门.

关键词: 椭圆曲线; 有限域; 点乘; 双口 RAM

中图分类号: TN4

文献标识码: A

文章编号: 1000-7180(2010)01-0090-04

Implementation of Scalar Multiplication Based on $GF(2^m)$

CHEN Guang-hua, ZHANG Xiao-li, ZHU Jing-ming, LIU Ming

(Microelectronic R & D Center, Shanghai University, Shanghai 200072, China)

Abstract: The implementation of scalar multiplication is described in the paper. The efficiency of module multiplication is improved by the combination of the multiplication process and the reduction process. Double interface RAM is used to accelerate the speed of data access. The capability of expansion is enhanced by the reserved RAM. The scalar multiplication module is synthesized with a library of synopsys DC Z-2007 03 solaris9 0.35 CMOS-technology and the result is about 18657 gates.

Key words: ECC; finite field; scalar multiplication; double interface RAM

1 引言

由于椭圆曲线密码系统(Elliptic Curve Cryptography, ECC)^[1-2]的密钥较短, 运算耗费的资源少, 目前广泛应用于无线连接设备中^[3].

在 ECC 常见的实现方法中, 滑动窗口编码法^[4]引进了预计算, 存放预计算结果需要占用存储空间, 并且编程复杂, 不利于硬件实现; 蒙哥马利(Montgomery)^[5]算法是在射影坐标下完成, 虽然避免了点加倍点运算中的模逆运算, 但完成点乘运算难免存在一次模逆运算; 一个大数经过冗余(NAF)^[6]编码法编码后可能会比其二进制数长. 无论采用哪种方法, 其点乘运算中的倍点运算的次数是不变的, 不同表示方法的实质就是尽量减少表示式中非零元的个数, 以减少点乘运算中点加运算的次数, 进而提高点乘运算的速度.

文中采用二进制编码法, 不需要编码转化和预计算, 编码过程与 ECC 参数无关, 从而得到灵活的实现结构. 本设计完成模乘运算过程中根据中间乘积的第 234 位的值是 1 还是 0 进行判断是否进行约减运算, 实现在做乘法的同时完成约减运算.

2 ECC 数学理论

假设 $F2^m$ 表示一个有限域, $E(F2^m)$ 是定义在 $F2^m$ 上的非超奇异椭圆曲线: $y_2 + xy = x_3 + ax_2 + b$, 其中 $a, b \in F2^m, b \neq 0$. $E(F2^m)$ 是由满足上述方程式的点 $P = (x, y)$ 以及无穷远点 O 共同构成. 如果 $P = (x_1, y_1)$ 和 $Q = (x_2, y_2)$ 是 $E(F2^m)$ 上的两点, 则 $E(F2^m)$ 上的加法运算可定义如下:

$$\textcircled{1} P + O = O + O = P;$$

$$\textcircled{2} \text{点 } P = (x_1, y_1) \text{ 的加法逆元为 } P' = -P = (x_1, y_1 + x_1), P + P' = O;$$

③ $P \neq \pm Q$ 时, $P + Q = (x_3, y_3)$, x_3, y_3 由下式确定:

$$\begin{cases} x_3 = \lambda^2 + \lambda + x_1 + x_2 + a \\ y_3 = \lambda(x_1 + x_3) + x_3 + y_1 \end{cases}$$

其中 $\lambda = (y_1 + y_2)/(x_1 + x_2)$.

④ $P = Q$ 时, $P + Q = 2P = (x_3, y_3)$, 此时 $x_3 = \lambda^2 + \lambda + a$, $y_3 = x_2 + (\lambda + 1)x_3$, 其中 $\lambda = x_1 + y_1/x_1$.

如果将椭圆曲线上的点 P 自相 K 加次, 把计算结果记作 KP , 则称根据 P 计算 KP 的运算为点乘.

根据以上数学理论分析, ECC 的实现从层次上可分为 4 层: 最底层为有限域上的模加、模减、模乘和模逆运算; 第二层为椭圆曲线上的点运算, 包括点加运算、倍点运算; 第三层为椭圆曲线点乘运算; 最上层为 ECC 机制协议实现层, 运算层次如图 1 所示.

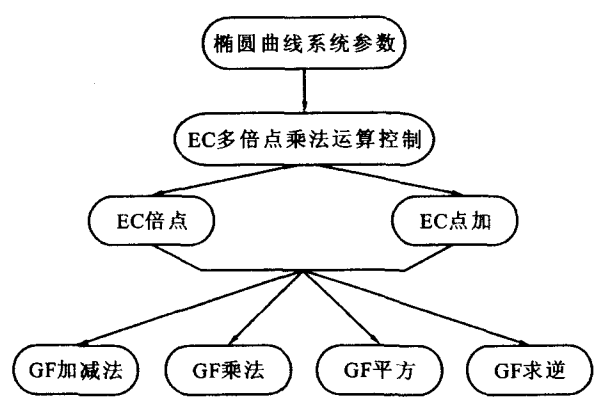


图 1 ECC 的运算层次

其中点加、倍点和点乘运算都是由模加、模乘和模逆这三个基本的域运算模块组成.

3 椭圆曲线加密算法的实现

3.1 二元有限域上乘法运算的实现

在 $GF(2^m)$ 的二进制域中, 通常多项式基表示下的元素乘法先进行多项式相乘, 然后再进行模约减多项式的算法. 硬件实现时需要引入 2 倍于乘数的存储单元存放多项式相乘的中间结果, 考虑到硬件实现该算法时的效率, 可对上述算法进行改进, 把元素的相乘过程和模约多项式过程结合起来, 改进后的算法描述如下:

输入: $a, b \in GF(2^m)$, 既约多项式 $f(x)$

输出: $c = a \cdot b \bmod f(x)$

步骤:

(1) $c = 0$;

(2) if($c[232] = 0$)

$c = 2c + a \cdot b[i]$;

else $c = 2c + a \cdot b[i] + f$;

模乘实现时, 每个时钟周期读入的 32 bit 数据, 根据当前 b 位与 c 位的组合是 00、01、10、11 进行相应的运算如图 2 所示, 在计算 $C[i]$ 的同时将 $C[i + 1]$ 的值写回到 $ramC[i + 1]$.

3.2 二元有限域上模逆运算的实现

椭圆曲线密码体制中, 计算 $GF(2^m)$ 域中某个非零元素的逆可以采用多种算法^[1]. 对于在 $GF(2^m)$ 中求逆来说, MAIA 算法比其它算法更为直接, 因此在目前各种求逆算法的硬件实现中, 一般都

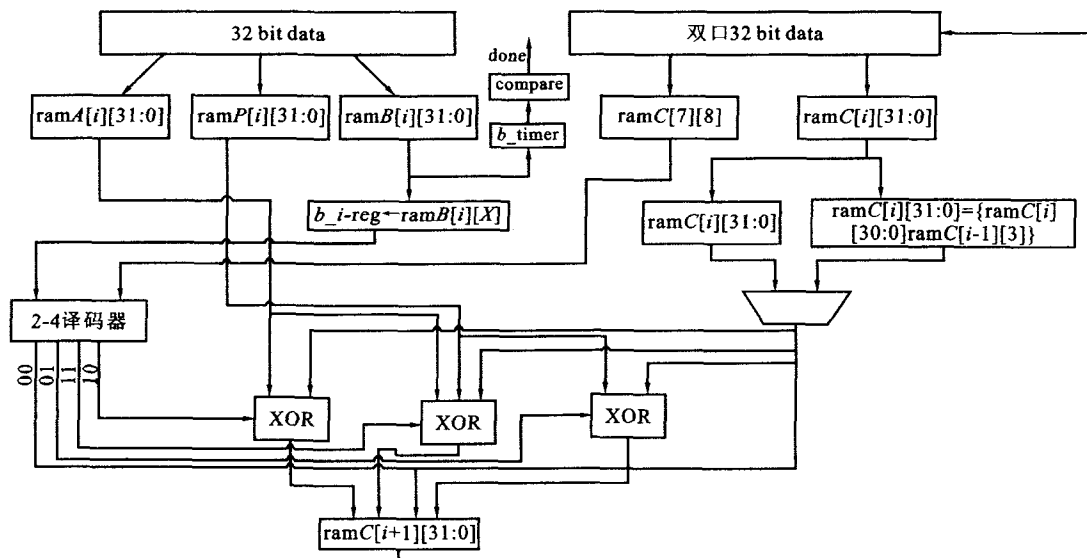


图 2 模乘的实现框图

采用 MAIA 算法^[2]. MAIA 的实现框图如图 3.

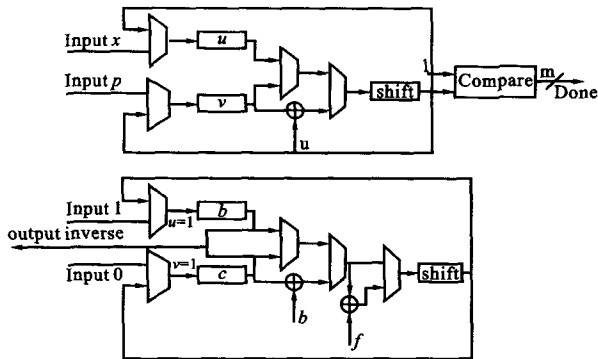


图 3 MAIA 的实现框图

输入: $a \in \text{GF}(2^m) \setminus \{0\}$, 既约多项式 $f(x)$

输出: $a^{-1} \bmod f(x)$

步骤:

(1) $b \leftarrow 1, c \leftarrow 0, u \leftarrow a, v \leftarrow f$;

(2) $u \neq 1$ 且 $v \neq 1$;

① If $x \mid u$ do

$u \leftarrow u/x$;

if $x \mid b$ then $b \leftarrow b/x$;

else $b \leftarrow (b + f)/x$;

② If $x \mid v$ do

$v \leftarrow v/x$;

if $x \mid c$ then $c \leftarrow c/x$;

else $c \leftarrow (c + f)/x$;

③ If $\deg(u) > \deg(v)$

$u = u + v, b = b + c$;

else $v = u + v, c = b + c$;

(3) If $u = 1$, inverse = b ;

else inverse = c .

MAIA 算法的实现过程中,首先判断 $u[0]$ 是否为 0,若为 0 则进行右移操作,接下来判断 $b[0]$ 是否为 0,若为 0 则进行右移操作,若为 1 则先与 p 相异或后再进行右移操作,这样的操作直至 $u[0]$ 为 1 结束;接着判断 $v[0]$ 是否为 0,若为 0 则进行右移操作,接下来判断 $c[0]$ 是否为 0,若为 0 则进行右移操作,若为 1 则先与 p 相异或后再进行右移操作,这样的操作直至 $v[0]$ 为 1 结束;然后比较 u 与 v 大小,当 $\deg(u) > \deg(v)$ 时 $u = u + v, b = b + c$,否则 $v = u + v, c = b + c$;至此操作回到对 $u[0]$ 的判断,顺序执行下面的操作,直到 u 为 1 时 inverse 的返回值为 b 的当前值或是 inverse 的返回值为 c 的当前值.

3.3 点加和倍点的实现

设计中,点加、倍点是在状态机的控制下,由控制

信号协调基域上的运算模块实现^[6],为节省硬件开销,平方单元用模乘模块完成.其实现框图如图 4 所示.

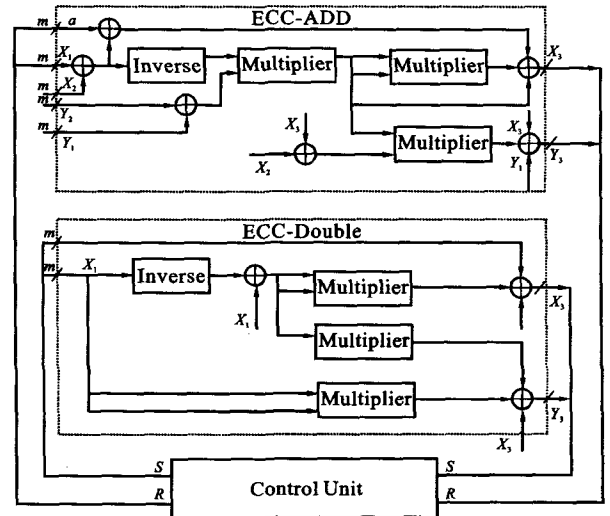


图 4 点加、倍点实现框图

点加实现是在控制单元(Control Unit)的调度下, x_1 和 x_2 异或的结果作为模逆模块(Inverse)的输入进行运算,其结果与 y_1 和 y_2 的异或结果作为模乘模块(Multiplier)的输入,得到 $\lambda = (y_1 + y_2) / (x_1 + x_2)$. 一般在进行 λ^2 运算时采用专门的平方运算模块,本设计为了节省硬件资源用 Multiplier 模块完成平方部分的运算,即用两个 λ 作为 Multiplier 的输入,得到的结果再与前面 x_1 和 x_2 异或的结果. 椭圆曲线参数 a, λ 相异或得到 x_3 的点加结果. 利用 x_3 及计算 x_3 过程中得到的 λ 值很容易得到 y_3 , 即把 x_3 与 x_2 的异或结果与 λ 作为 Multiplier 的输入,结果与 x_3 相异或得到 y_3 的点加结果. 倍点的过程与此类似^[7].

3.4 点乘运算的实现

椭圆曲线中点乘算法描述:

输入: 基域 $\text{GF}(2^n)$ 及其上的椭圆曲线 $E: y^2 + xy = x^3 + ax^2 + b$, E 上的点 P , n 比特私钥 K .

输出: $Q = K \cdot P$

步骤:

$Q = P$

for $i = n - 2$ to 0

$Q = 2 \cdot Q$

if $K[i] = 1$ then

$Q = Q + P$

endif

endfor

return Q

按照椭圆曲线的分层运算机制,首先进行椭圆曲线点乘运算,此时控制寄存私钥的移位寄存器 K 从高往低移一位, $K[n-1]$ 为当前私钥比特位,并将其值作为状态机的控制反馈;按照椭圆曲线的分层运算机制,接下来进行第二层的椭圆曲线上点的运算,包括点加运算、点倍运算,即调用倍点运算模块 Double(图4)进行倍点运算,然后根据当前私钥比特位 $K[n-1]$ 的值判断是否调用加法模块,当 $K[n-1]$ 的值为1时,需要调用加法模块 Add(图4)进行点加运算,当 $K[n-1]$ 值为0时则直接进入私钥的下一位扫描处理;最后通过调用最底层的有限域上的模加、模减、模乘和模逆的运算结果完成第二层的椭圆曲线上点的运算进而完成点乘的运算.点乘工作原理如图5所示.

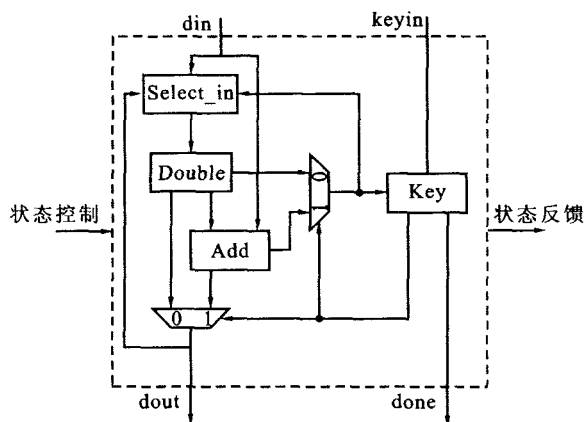


图5 KP的工作原理

4 双口RAM在ECC中的应用

在椭圆曲线加密算法的底层运算中,对233位的大数运算,目前硬件实现模乘单元直接使用233位寄存器进行数据交换与运算,因此占用了大量的片内资源,为减少模乘单元的面积,文中使用了数据调度机制与双口RAM相结合的方式,其系统框图如图6所示.

计算过程使用的233位数据以32位为一个字,利用8行存储空间存储于片外的RAM中,片内数据计算、存储则使用32位寄存器.在RAM控制器的调度下,实现每个时钟周期读入相应RAM中的一行32位数据于片内寄存器进行数据计算,同时将上次的运算结果按RAM控制器的调度顺序写入存放结果的RAM中.设计中通过利用双口RAM同时读写性能节省了时间,通过片内32位寄存器的使用节省了面积,同时通过预留RAM空间,增加了系统

的可扩充性.RAM的配置如图7所示.

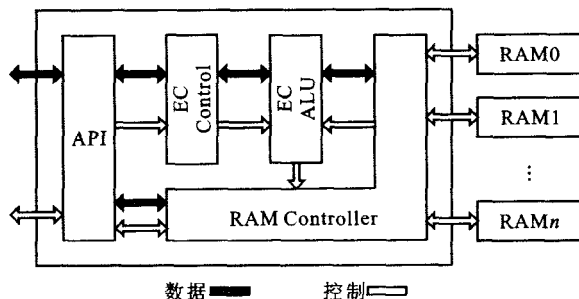


图6 ECC芯片的系统框图

0_00	Control_reg
0_01	Status_reg
0_02	Command_reg
0_03	Reserved_reg
1_00	...
c_00	Y_i $(KP_y,$ $(W+Q)_y,$ $(W+W)_y)$
c_20	Reserved_reg

图7 RAM单元设置框图

5 结束语

文中给出了椭圆曲线加密算法的点乘实现.点乘运算采用二进制编码法,它是最简单的编码方法,不存在任何编码转化和预计算,编码过程与ECC参数无关,从而得到灵活的实现结构.域运算中的模乘运算通过把相乘过程和模约多项式过程结合起来,改善了运算效率.片外双口RAM的使用,加快了数据存取速度,同时通过预留RAM空间,增强了系统的可扩充性.本设计用VerilogHDL语言作为设计工具,在synopsys DC Z-2007 03 solaris9工作平台上,基于chartered 0.35 CMOS的综合库,50MHz约束下的综合结果约为18657门.

参考文献:

- [1] Miller V. Use of elliptic curves in cryptography[C]// Lecture notes in computer sciences, 219 on Advances in cryptography - CRYPTO 85. New York: Springer - Verlag, 1986:417-426.
- [2] Koblitz N, Vastone S, Menezes A. The state of elliptic curve cryptography[J]. Designs, Codes and Cryptography, 2000(19):173-193.
- [3] Lauter K. The advantages of elliptic curve cryptography for wireless security [J]. IEEE Wireless Communications, 2004,11(1):62-67.

(下转第98页)

6 结束语

文中提出一种从用户服务请求到产生结构化服务组合的方法.虽然方法需要一些人工输入,但是基本上是完全自动的.算法的关键部分是给定的服务集和它们之间的依赖关系,然后完全自动的构建结构化组合来满足给定的依赖.该方法的特点是构建结构化组合,并且利用了基本工作流程模型.与文献[5]相比,不仅考虑单个服务间的依赖关系,还考虑了多个服务输出联合起来为一个服务提供输入的情况.与文献[2]相比,不仅考虑了输入/输出参数之间的依赖,更考虑了服务的行为约束条件.整个方法中并没有涉及到循环结构的处理,下一步工作将对算法进行扩展以处理循环结构^[6-7].

参考文献:

- [1] Foster I, Kesselman C, Nick J, et al. An open grid services architecture for distributed systems integration[C]// Physiology of the Grid. Chicago: University of Chicago, 2002.
- [2] Lang Qha, Su Syw. AND/OR graph and search algorithm for discovering composite Web services[J]. Journal of Web Services Research. 2005, 2(4): 46 - 64.
- [3] Rik Eshuis, Paul Grefen, Sven Till. Structured service composition[C]//BPM 2006. The Netherlands, LNCS 4102, 2006: 97 - 112.
- [4] Seyyed Vahid Hashemian, Farhad Mavaddat. AGraph - based approach to web services composition[C]// Proc of the Symp on Application and the Internet. Canada: Waterloo University, 2005: 183 - 189.
- [5] 付燕宁, 刘磊, 金成植. 基于服务链的 Web 服务组合方法[J]. 通信学报, 2007, 28(7): 92 - 97.
- [6] 李刚, 马修军, 韩燕波, 等. 动态网络环境下的透明服务组合[J]. 计算机学报, 2007, 30(4): 579 - 587.
- [7] 许兰, 朱巧明, 贡正仙, 等. 一种优化服务组合执行的设计方案[J]. 微电子学与计算机, 2007, 24(12): 144 - 146.

作者简介:

黄 炜 男, (1985 -), 硕士研究生. 研究方向为网格计算、中文信息处理.
朱巧明 男, (1963 -), 教授. 研究方向为中文信息处理、Web 信息处理、嵌入式系统.
李培峰 男, (1971 -), 副教授. 研究方向为中文信息处理和多文种信息处理、分布式信息系统、网格计算.
刘梅梅 女, (1981 -), 硕士研究生. 研究方向为中文信息处理、网格计算.

(上接第 93 页)

- [4] Jerome A Solinas. National security agency: an improved algorithm for arithmetieona family of elliptic curves [R]. USA: Ft. Meade, MD20755.
- [5] King B. An improved implementation of elliptic curve cryptography over GF(2m) when using projective point arithmetic[M]. Berlin Heidelberg: Springer, 2001: 134 - 150.
- [6] Menezes A, Hankerson D, Hernandez J. Software implementation of elliptic curve cryptography over binary field [J]. Journal of Electronic Science and Technology of China, 2003, 1(1): 1 - 24.
- [7] Luis Enrique Erro. GF(2m) arithmetic modules for elliptic curve cryptography electronics[J]. Computer Science Department, 2006(9): 689 - 696.

作者简介:

陈光化 男, 副教授. 研究方向为数字集成电路设计.
张晓莉 女, 硕士研究生. 研究方向为数字集成电路设计.