文章编号:1674-2974(2022)02-0101-05 DOI:10.16339/j.cnki.hdxbzkb.2022264

一种改进的模逆算法与硬件实现

胡锦†,李勇彬
(湖南大学 物理与微电子科学学院,湖南 长沙 410082)

摘 要:在公钥密码体系中,无论是RSA密码还是椭圆曲线密码,模逆运算都是非常关键的运算.模逆运算的前提是两数的最大公约数为1,否则结果是没有意义的.基于现有的二进制模逆算法的基础上提出了一种可以同时求最大公约数和进行模逆运算的算法,并且对算法进行优化,用VERILOG HDL语言进行硬件实现.通过功能仿真和FPGA验证,结果表明该设计可以正确进行32~1 024 bit的大数模逆运算.该设计应用于一款汽车安全芯片的PKI模块,采用UMC 55 nm工艺进行流片,芯片面积为10 mm^2,工作电压3.3 V,钟频率为200 MHz时,功耗约为30.2 mW.

关键词:RSA密码;椭圆曲线密码;公钥密码;模逆;最大公约数
中图分类号:TN492 **文献标志码**:A

An Improved Modular Inversion Algorithm and Hardware Implementation

HU Jin†,LI Yongbin
(School of Physics and Electronics,Hunan University,Changsha 410082,China)

Abstract: In public-key cryptosystems (PKI), whether it is RSA cryptography or elliptic curve cryptography (ECC), modular inversion operations are very critical operations. The premise of modular inversion operations is that the greatest common divisor (GCD) of the two numbers is 1, otherwise the result is meaningless. Based on the existing binary modular inversion algorithms, an algorithm that can simultaneously find the GCD and perform modular inversion operations is proposed, and the algorithm is optimized and implemented in hardware using VERILOG HDL language. Through functional simulation and FPGA verification, the results show that this design can correctly perform the modular inversion operation of large number from 32 to 1 024 bits. The design is applied to a PKI module of an automotive security chip, using UMC 55 nm process for tape-out, and the chip area is 10 mm^2. When the working voltage is 3.3 V and the clock frequency is 200 MHz, the power consumption is about 30.2 mW.

Key words: RSA cryptography; elliptic curve cryptography (ECC); public-key cryptosystems (PKI); modular inversion; greatest common divisor (GCD)

RSA密码和椭圆曲线密码是目前使用最广泛的公钥密码算法.随着物联网的发展,用户信息安全越

* **收稿日期**:2021-01-18
基金项目:国家自然科学基金资助项目(61674055),National Natural Science Foundation of China(61674055)
作者简介:胡锦(1964—),男,湖南长沙人,湖南大学教授
† 通信联系人,E-mail:hujin@hnu.edu.cn

来越重要,例如现今高速发展的智能汽车,安全性和实时性要求极高,软件实现加密算法的方式已经无法满足需求,采用硬件方式实现算法是发展的趋势.模逆算法是公钥密码体系中的核心算法[1-2],尤其在椭圆曲线密码体系中[3],也是最复杂[4]和最消耗时间的算法之一[5]. 在RSA密码算法中,生成密钥对时,需要计算$d=e^{-1}\bmod \phi$,其中e表示公钥,满足$1<e<\phi$且$\gcd(e,\phi)=1$,ϕ表示欧拉函数,d表示私钥. 为了安全,ϕ至少为1 024位,用软件实现模逆运算,运算量大,运算时间长,效率低[6]. 在椭圆曲线密码算法中,进行点加[7]、点乘和倍点运算时,用雅可比坐标进行坐标变换[8]也只能减少模逆运算使用的次数而不能完全避免. 本文在现有的二进制模逆算法基础上进行了改进,提出了一种在求逆过程中同时可以求取最大公约数的算法. 此外,出于对实现算法的硬件资源考虑,对算法做了优化,最后通过VERILOG语言进行硬件实现.

1 算法介绍

1.1 二进制模逆算法

二进制模逆算法原理是根据贝祖等式gcd(a,b)=gcd(b−a,a)=ax+by推导得出. 目前常用的模逆算法还有扩展欧几里得算法[9-10],但是由于算法中每步都要用到除法操作,开销很大[11],尤其在进行大素数模逆运算时缺点更突出. 算法1只用到了移位操作和减法运算,比扩展欧几里得算法更加高效.

算法1 二进制模逆算法

输入:素数p和$a\in[1,p]$

输出:$a^{-1}\bmod p$

1,$u\leftarrow a$,$v\leftarrow p$.

2,$x_1\leftarrow 1$,$x_2\leftarrow 0$.

3,当$u\neq 1$和$v\neq 1$时,重复执行

3.1 当u是偶数时,重复执行

$u\leftarrow u/2$.

若x_1是偶数,则$x_1\leftarrow x_1/2$,否则,

$x_1\leftarrow (x_1+p)/2$.

3.2 当v是偶数时,重复执行

$v\leftarrow v/2$.

若x_2是偶数,则$x_2\leftarrow x_2/2$,否则,

$x_2\leftarrow (x_2+p)/2$.

3.3 若$u\geq v$,则$u\leftarrow u-v$,$x_2\leftarrow x_1-x_2$,否则

$v\leftarrow v-u$,$x_2\leftarrow x_2-x_1$.

4,若$u=1$,返回$(x_1 \bmod p)$,否则,返回

$(x_2 \bmod p)$.

1.2 改进的模逆算法

算法1有一个缺陷是无法判定输入的两数是否互质,如果$\gcd(a,p)$不等于1时,再用算法1计算就会得到错误的结果,或者说算出的结果是没有意义的. 结合stein算法[11],可以将上述算法改写为算法2,算法2在模逆运算的同时可以求解最大公约数$\gcd(a,p)$,从而保证模逆运算的结果是在$\gcd(a,p)=1$的情况下得到的正确结果. 在算法1中可以看出,循环计算时u和v的计算基本上是一致的,为了节省资源,可以进行进一步优化. 由于算法2中的步骤3和步骤5.2保证u和v每次循环最多只有一个为偶数,利用这个特点可以去掉u的循环.

算法2 改进的模逆算法

输入:正整数a和p

输出:$a^{-1} \bmod p$和$\gcd(a,p)$

1,$x\leftarrow a$,$y\leftarrow p$.

2,$g\leftarrow 1$.

3,当x和y都为偶数,重复执行:

$x\leftarrow x/2$,$y\leftarrow y/2$,$g\leftarrow 2g$.

4,当x为偶数时,执行:

$u\leftarrow y$,$v\leftarrow x$,$A\leftarrow 0$,$C\leftarrow 1$,$B\leftarrow 1$,$D\leftarrow 0$.

否则:

$u\leftarrow x$,$v\leftarrow y$,$A\leftarrow 1$,$C\leftarrow 0$,$B\leftarrow 0$,$D\leftarrow 1$.

5,当$v\neq 0$,重复执行:

5.1 当v是偶数时,重复执行

$v\leftarrow v/2$.

如果C和D都是偶数,执行

$C\leftarrow C/2$,$D\leftarrow D/2$.

否则:

$C\leftarrow (C+y)/2$,$D\leftarrow (D-x)/2$.

5.2 当$u\geq v$时,执行

$u\leftarrow v$,$v\leftarrow u-v$;

$A\leftarrow C$,$C\leftarrow A-C$,$B\leftarrow D$,$D\leftarrow B-D$.

否则:

$v\leftarrow v-u$,$C\leftarrow C-$A,$D\leftarrow D-B$.

6,计算$\gcd(a,p)=u*g$.

7,返回$(A \bmod y)$和$\gcd(a,p)$.

2 算法硬件结构框图

模逆算法主要通过状态机来控制算法流程,通过ram存储大量的中间数据,图1是模逆算法硬件结

构框图,主要分为两个模块,INV_FSM模块是状态机模块,INV_CAL是模逆算法的运算模块,受状态机模块的调度. ram为伪双端口ram,主要用来存储在运算过程中产生的中间数据和运算结果.

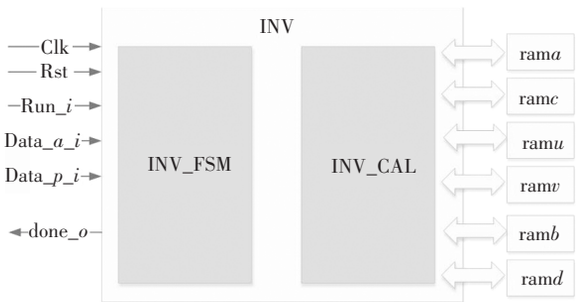

图1　模逆算法硬件结构框图

Fig.1　Modular inversion algorithm hardware structure block diagram

2.1　算法状态机

图2是模逆算法的状态转移图,严格控制模逆算法的运算流程. 该状态机共有9个状态,IDLE是空闲状态,开始运算时,进入SHIFT状态,SHIFT状态完成x和y同时向右移位(即除以2),g向左移位,直到x和y不同时为偶数. LOAD状态主要完成u、v、A、B、C、D数据的装载. INV_CAL状态完成模逆算法的主体运算直到v等于0时才跳出该状态. 在GCD_CAL状态中完成gcd (a,p)计算,在GCD_CAL状态计算完成后判断A的极性进行状态跳转. 若A为负数,则跳转到A_NEG状态进行$A+y$运算,直到A为正数,则跳转到DONE状态;若A为正数,则跳转到A_POS1状态进一步判断A是否大于y,若A小于y,则跳转到DONE状态,若A大于y则跳转到A_POS2状态进行$A-y$运算,直到A小于y,然后跳转到DONE状态,模逆运算完成,最终回到IDLE状态.

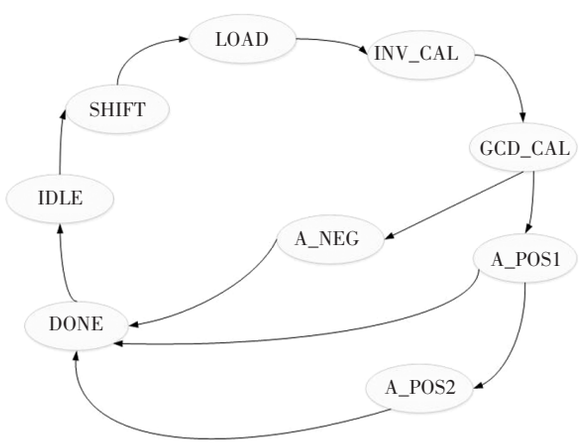

图2　模逆算法状态转移图

Fig.2　State transition diagram of modular inversion algorithm

2.2　算法运算模块

算法运算模块主要功能是在状态机的控制下进行数据的运算. 从算法2中可以看出,求逆的过程需要用到大数加法和大数减法,还需要进行两数大小判断和移位. 大数加法和大数减法可以通过算法3和算法4实现,两数的大小比较可以通过使用大数减法的借位信号来判断. 图3是运算模块主要运算电路,data_v和data_u通过mux来进行选择,mux的选择信号通过比较data_u和data_v的大小得到,data_c和data_a、data_b和data_d类似. 从图3中可以看出:改进后的算法使用mux可以进行u和v的选择,从而实现资源的复用,减少了资源的消耗. 运算的中间结果和运算结果保存在ram中.

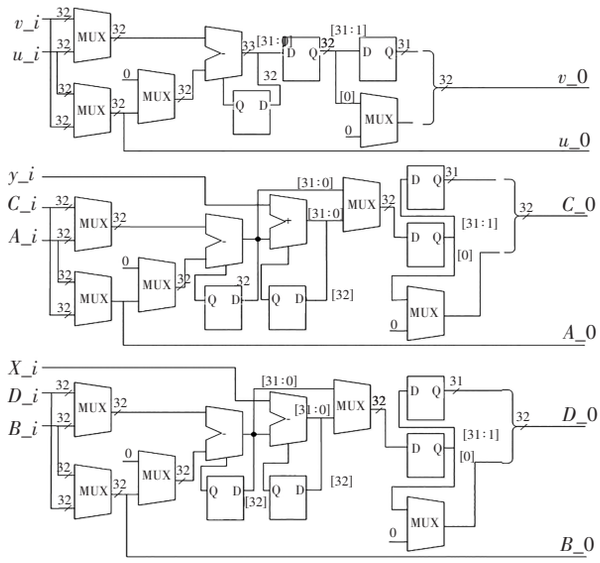

图3　算法模块电路图

Fig.3　Algorithm module circuit diagram

该电路中输入数据都是以32位为最小输入单元,如果进行1 024位的模逆运算,此电路需要计算32次,可以看出此电路其实还可以实现2 048位甚至更大的模逆运算. 需要注意的是. 计算的位宽越大,需要保存的中间数据也越多,ram的容量就需要越大.

算法3　大数加法算法

输入:整数$A=(a_{k-1},a_{k-2},\cdots,a_0)_{2^{32}}$

整数$B=(b_{k-1},b_{k-2},\cdots,b_0)_{2^{32}}$

输出:$\{c,s\}=A+B$.

1,$\{c,s_0\}=a_0+b_0$.

2,对于i从1到$k-1$,重复执行

2.1　$\{c,s_i\}=a_i+b_i+c$.

3,返回(c,s).

算法4 大数减法算法

输入:整数$A=(a_{k-1},a_{k-2},\cdots,a_0)_{2^{32}}$

整数$B=(b_{k-1},b_{k-2},\cdots,b_0)_{2^{32}}$

输出:$\{c,s\}=A-B$.

1,$\{c,s_0\}=a_0-b_0$.

2,对于i从1到$k-1$,重复执行

2.1 $\{c,s_i\}=a_i-b_i-c$.

3,返回(c,s).

3 结果与分析

通过VCS对该电路进行功能仿真,如图4所示.模逆算法电路能正确计算出最大公约数和模逆运算的结果,将计算的结果保存在ram中.仿真结果表明:该设计可以正确进行32~1 024 bit的模逆运算.

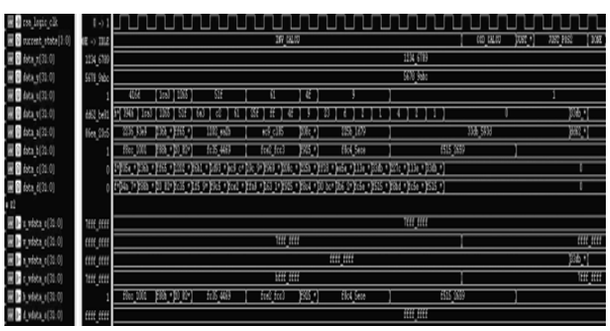

图4 算法功能仿真图

Fig.4 Algorithm function simulation diagram

使用Xilinx virtex-II FPGA开发板进行了板级验证,验证了该设计的正确性.表1列出了该设计与同类设计的资源消耗和性能比较.表格中的时间为计算256 bit模逆运算使用的时间.

表1 FPGA结果对比

Tab.1 FPGA result comparison

设计	最大频率/MHz	面积/slice	时间/μs
文献[12]	146.38	1 480	2.32
文献[13]	68.17	2 085	11.60
本设计	185	1 347	1.48

该设计应用于一款汽车安全芯片的PKI模块,用于实现RSA和SM2算法.图5为该安全芯片的版图,采用UMC 55 nm工艺进行流片,该芯片总面积为10 mm^2,在工作电压3.3 V,时钟频率为200 MHz时,功耗仅为30.2 mW,流片测试结果表明,芯片达到设计指标要求.

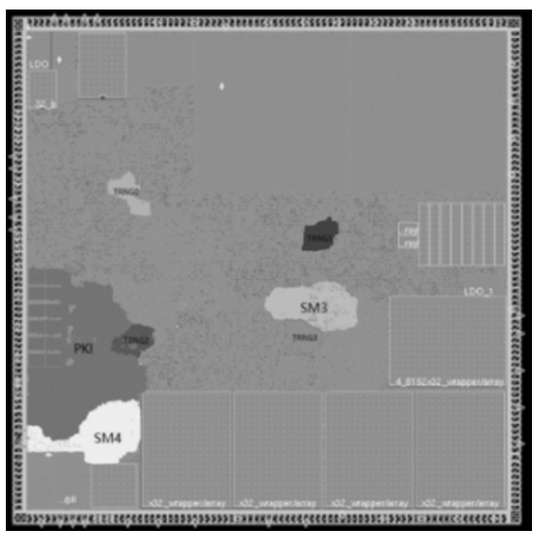

图5 芯片版图

Fig.5 Chip layout

4 结 语

本文提出了一种在现有二进制模逆算法的基础上进行改进的算法.该算法不仅可以进行模逆运算,还同时可以计算最大公约数,并且在算法上进行了优化,减少了算法实现的资源消耗,最后通过VERILOG语言实现了该算法.该设计电路结构简单,可扩展性强.最后该设计采用UMC 55nm工艺进行流片,流片测试结果表明,芯片达到设计指标要求.

参考文献

[1] CHEN C P, QIN Z P. Fast algorithm and hardware architecture for modular inversion in GF(p)[C]//2009 Second International Conference on Intelligent Networks and Intelligent Systems. Tianjian, China: IEEE, 2009: 43-45.

[2] WANG J, JIANG A P. An area-efficient design for modular inversion in GF(2m)[C]//APCCAS 2006 - 2006 IEEE Asia Pacific Conference on Circuits and Systems. Singapore: IEEE, 2006: 1496-1499.

[3] XU S, GU H H, WANG L Y, *et al*. Efficient and constant time modular inversions over prime fields[C]//2017 13th International Conference on Computational Intelligence and Security (CIS). Hong Kong, China: IEEE, 2017: 524-528.

[4] LIN S Y, HE S, GUO X, *et al*. An efficient algorithm for computing modular division over GF(2m) in elliptic curve cryptography [C]//2017 11th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID). Xiamen, China:IEEE,2017:179-182.

[5] TIAN C L, YU J, ZHANG H L, *et al*. Novel secure outsourcing of modular inversion for arbitrary and variable modulus [J]. IEEE Transactions on Services Computing, 2019:1-1.

[6] CHOI P, KONG J T, KIM D K. Analysis of hardware modular inversion modules for elliptic curve cryptography [C]//2015 International SoC Design Conference (ISOCC). Gyeongju, Korea (South):IEEE,2015:313-314.

[7] MRABET A, EL-MRABET N, BOUALLEGUE B, *et al*. An efficient and scalable modular inversion/division for public key cryptosystems [C]//2017 International Conference on Engineering & MIS (ICEMIS). Monastir, Tunisia:IEEE,2017:1-6.

[8] RAHMAN M S, HOSSAIN M S, RAHAT E H, *et al*. Efficient hardware implementation of 256-bit ECC processor over prime field [C]//2019 International Conference on Electrical, Computer and Communication Engineering (ECCE). Cox′sBazar, Bangladesh:IEEE,2019:1-6.

[9] FU B W. A rapid algorithm and its implementation for modular inversion [C]//2009 Fifth International Conference on Information Assurance and Security. Xi′an, China:IEEE,2009:697-700.

[10] PHIAMPHU D, SAHA P. Redesigned the architecture of extended-euclidean algorithm for modular multiplicative inverse and jacobi symbol [C]//2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI). Tirunelveli, India:IEEE,2018:1345-1349.

[11] 谭丽娟,陈运. 模逆算法的分析、改进及测试[J]. 电子科技大学学报,2004,33(4):383-386.
TAN L J, CHEN Y. Analysis and improvement of modular inverse algorithm [J]. Journal of University of Electronic Science and Technology of China, 2004, 33(4):383-386. (In Chinese)

[12] HOSSAIN M S, KONG Y N. High-performance FPGA implementation of modular inversion over F_{256} for elliptic curve cryptography [C]//2015 IEEE International Conference on Data Science and Data Intensive Systems. Sydney, NSW, Australia: IEEE, 2015: 169-174.

[13] VLIEGEN J, MENTENS N, GENOE J, *et al*. A compact FPGA-based architecture for elliptic curve cryptography over prime fields [C]//ASAP 2010 - 21st IEEE International Conference on Application-specific Systems, Architectures and Processors. Rennes, France:IEEE,2010:313-316.