

高速双域乘法器设计及其应用

郑朝霞, 资义纯, 田 园, 吴 浩

(华中科技大学 光学与电子信息学院, 湖北 武汉 430074)

摘 要: 双域乘法器在椭圆曲线密码学中具有重要意义, 是构成双域模乘器的重要组件. 考虑到双域乘法器的关键路径主要由 $GF(p)$ 域决定; 因此, 在传统的基 4 Booth 编码乘法器的基础上进行优化设计, 改进部分积产生电路以及 Wallace 压缩电路, 使其能够同时支持 $GF(p)$ 域和 $GF(2m)$ 域. 设计的双域乘法器在 FPGA 实现结果表明, 双域乘法器比单独实现两个域面积减小 16.9%; 延时比单独的 $GF(p)$ 域增加 1.188 ns. 将设计的双域乘法器应用到模乘器, 结果表明, 该设计完成一次 256 bit 的模乘操作比已有的在时间上节约了 7.35%.

关键词: 双域乘法器; 双域模乘; Booth 编码; 双域 4-2 压缩器

中图分类号: TN47

文献标识码: A

文章编号: 1000-7180(2016)05-0001-05

Design and Application of High Speed Dual-Field Multiplier

ZHENG Zhao-xia, ZI Yi-chun, TIAN Yuan, WU Hao

(School of Optical and Electronic Information, Huazhong University of Science and Technology, Wuhan 430074, China)

Abstract: Dual-Field multiplier is the important component of Dual-Field Montgomery multiplier and has important meaning for elliptic curve cryptography. Considering that the critical path of Dual-Field multiplier depends on the $GF(p)$ field, Partial product generating circuit and Wallace compressor circuit was optimized based on traditional radix-4 Booth encoder multiplier($GF(p)$) to support both $GF(p)$ and $GF(2m)$ field. The designed Dual-Field multiplier implemented on FPGA shows that the area is decreased by 16.9% compare to the separating implement. The delay is increased by 1.188 ns than the traditional field $GF(p)$. The designed Dual-Field multiplier is applied to Montgomery multiplier. The time for 256 bit Montgomery multiplier of our design is reduced by 7.35% than paper proposed.

Key words: dual-field multiplier; dual-field montgomery multiplier; booth encoder; dual-field 4-2 compressor

1 引言

近年来, 信息安全得到了越来越多的重视, 国内外都有相关的组织进行此方面的标准化工作, 如美国国家标准局 NIST 以及中国国家密码管理局. 在公钥加密方案上, ECC 以其存储空间小、带宽要求低等特性正逐步取代 RSA 在金融智能卡等领域应用^[1]. Satoh^[2]等人提出了一种可伸缩的基于字的 montgomery 模乘算法, 证明了 $GF(p)$ 域和 $GF(2m)$ 域具有相同的数据通路, 基于字的算法可以保证模乘的操作数位长

可变, 因此能够同时支持 RSA 和双域的 ECC 算法, 但没有对双域乘法器进行优化设计. $GF(p)$ 和 $GF(2m)$ 双有限域乘法运算是双域模乘器的关键构成, 双域乘法器电路结构的设计对整个系统的面积/速度等性能有着直接的影响. 如果分别实现两个域的乘法器会占用更多的面积资源. 基于以上考虑, 本文在传统(素数域 $GF(p)$)基 4 Booth 乘法器的结构基础上, 设计了一种可复用的高速双域乘法器, 有效减小电路的面积, 提高电路的性能.

收稿日期: 2015-06-30; 修回日期: 2015-08-21

基金项目: 湖北省重大项目(2015ACA063); 中央高校基本科研业务费资助, HUST(2014TS041); 深圳市技术创新计划(CYZZ20140829104843693)

2 传统高速乘法器

高速乘法器在高性能处理器、加密协处理器、DSP、FFT 等领域具有重要作用;如何高速、小面积、低功耗地实现乘法器具有重要意义.同时,很多乘法器实现技术被提出,例如:Booth 编码器、Wallace 树加法器、Dadda 树加法器、进位保留加法器、阵列乘法器、改进 Booth 编码器等^[3].

一般乘法器的实现可分为图 1 所示的 3 个步骤:部分积产生、部分积求和、最后和的产生.

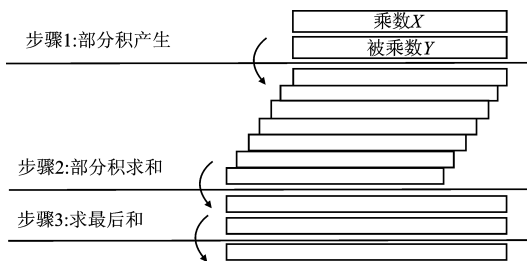


图 1 乘法器实现的 3 个基本步骤

高速并行乘法器在设计上主要针对步骤 1 和步骤 2 进行优化,其中基于 Booth 编码的技术主要是优化步骤 1 的部分积产生,通过 Booth 编码减少部分积乘积项.而树形压缩结构则针对步骤 2 的部分积求和过程进行优化,通过树形结构设计减小加法器级数从而减小总的传播延时.通常,高速并行乘法器设计会同时采用 Booth 编码和树形压缩来减小总延时,提高乘法器速度.

3 高速双域乘法器设计

3.1 双域乘法器设计基础

椭圆曲线等加密算法通常定义在两个有限域上,素数域 $GF(p)$ 和 2 元扩域 $GF(2^m)$.其中 $GF(2^m)$ 域中的元素可以用多项式表示,其加减法为 2 进制异或,同时没有进位信号产生;而对比 $GF(p)$ 域可以发现,素数域加减法也是异或电路,只不过根据情况会产生进借位.只要在传统乘法器上稍做改进即可实现 $GF(2^m)$ 域乘法器从而形成双域乘法器.

$GF(2^m)$ 域因为只有异或项,没有进位传播,因此路径延时小,双域乘法器的关键路径主要由素数域乘法器决定.素数域乘法器的结构会对整个双域乘法器的性能起到关键性作用,因此,本文主要基于传统高速素数域乘法器进行改进使其能够支持双域乘法操作.

3.2 改进 Booth 编码器

基于 Booth 编码的并行乘法器能够减小延时,

提高乘法器速度;但同时会增加面积.Booth 编码主要是通过减小部分积的项数从而间接减小后续求和电路的级数,减小传播延时.主要有基-4、基 8、基-16、基 32 等 Booth 编码.然而高于基-4 的 Booth 电路需要额外的 2 进制补码转换电路,会增加路径延时;因此,实际设计中最常用的是基-4 Booth 编码架构.

本文采用的是改进的基-4 Booth 编码 MBE-III 编码方式^[4].当计算两个 32 位无符号数相乘,如 $P=A \times B$,可以式(1)表示其乘积结果;对于 n 等于 0 时的 b_{-1} 和 n 等于 16 时的 b_{32} 、 b_{33} 都等于 0,对于有符号数的乘法运算,需要令 b_{32} 、 b_{33} 等于 b_{31} ;由式(1)可以推导出部分积 PP 的表达式,如式(2)所示.

$$P = A \times \left(\sum_{i=0}^{31} b_i 2^i \right) = A \times \left[\sum_{n=0}^{16} (b_{2n-1} + b_{2n} - 2b_{2n+1}) 4^n \right]$$

$$= \sum_{n=0}^{16} A(b_{2n-1} + b_{2n} - 2b_{2n+1}) 2^{2n} \quad (1)$$

$$PP_{ni} = \left(\sum_{i=0}^{31} a_i 2^i \right) \times \left(\sum_{n=0}^{16} (b_{2n-1} + b_{2n} - 2b_{2n+1}) 2^{2n} \right) \quad (2)$$

根据式(2)可以得到其 MBE-III 编码方式如表 1 所示.

表 1 素数域乘法的基 4 Booth 编码

b_{2n+1}	b_{2n}	b_{2n-1}	PP_n	NEG	B_1	B_2	Z
0	0	0	$+0 \times A2^{2n}$	0	1	0	1
0	0	1	$+1 \times A2^{2n}$	0	1	1	1
0	1	0	$+1 \times A2^{2n}$	0	0	1	0
0	1	1	$+2 \times A2^{2n}$	0	1	0	0
1	0	0	$-2 \times A2^{2n}$	1	1	0	0
1	0	1	$-1 \times A2^{2n}$	1	0	1	0
1	1	0	$-1 \times A2^{2n}$	1	0	1	1
1	1	1	$+0 \times A2^{2n}$	1	1	0	1

根据表 1 可以推导出 NEG、 B_1 、 B_2 、 Z 等 Booth 控制信号的逻辑表达式.根据 Booth 控制信号逻辑可得部分积逻辑.其电路结构分别,如图 2、图 3 所示.

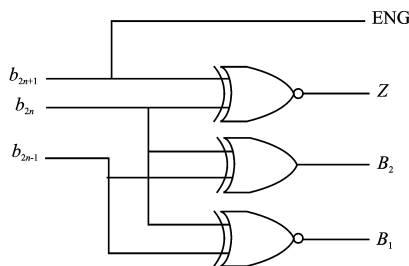


图 2 控制信号逻辑电路

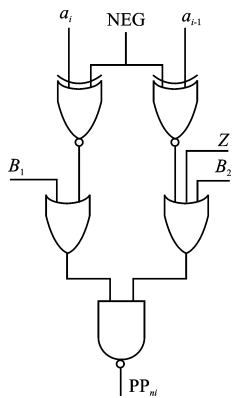


图 3 部分积产生逻辑

3.3 双域 Booth 编码设计

由于 GF(2 m)域运算的特殊性,域的加法运算是按位异或操作,所以最后产生的部分积的求和操作是通过异或运算完成的,另外,GF(2 m)域的乘法不涉及符号位的运算,所以编码的时候不需要考虑符号位的情况.

根据 GF(2 m)域的运算特点,对部分积的产生电路进行相应的调整,使其能够同时支持 GF(p)和 GF(2 m)域.可以采用统一的基 4 Booth 编码方式来实现双域的 32 bit 乘法运算,这样在提高电路灵活性的同时减少了电路面积,其具体的编码方式如表 2 所示.其中 field 信号为 1 表示 GF(p)域,为 0 表示 GF(2 n)域.

表 2 双有限域乘法的基 4 Booth 编码方式

Field	b_{2n+1}	b_{2n}	b_{2n-1}	PP_{ai}	NEG	B_1	B_2	Z
1	0	0	0	$+0 \times A2^{2n}$	0	1	0	1
1	0	0	1	$+1 \times A2^{2n}$	0	0	1	1
1	0	1	0	$+1 \times A2^{2n}$	0	0	1	0
1	0	1	1	$+2 \times A2^{2n}$	0	1	0	0
1	1	0	0	$-2 \times A2^{2n}$	1	1	0	0
1	1	0	1	$-1 \times A2^{2n}$	1	0	1	0
1	1	1	0	$-1 \times A2^{2n}$	1	0	1	1
1	1	1	1	$+0 \times A2^{2n}$	1	1	0	1
0	0	0	0	$0 \times A2^{2n}$	0	1	1	1
0	0	0	1	$0 \times A2^{2n}$	0	1	1	1
0	0	1	0	$1 \times A2^{2n}$	0	0	1	0
0	0	1	1	$1 \times A2^{2n}$	0	0	1	0
0	1	0	0	$2 \times A2^{2n}$	0	1	0	0
0	1	0	1	$2 \times A2^{2n}$	0	1	0	0
0	1	1	0	$(A \cdot 2 A) \times A2^{2n}$	1	0	0	0
0	1	1	1	$(A \cdot 2 A) \times A2^{2n}$	1	0	0	0

从表中可以得到 NEG, B₁, B₂ 和 Z 的表达式如下:

$$\text{Neg} = b_{2n+1} \cdot (b_{2n} + \text{field});$$

$$B_2 = ((b_{2n} \wedge b_{2n-1}) \cdot \text{field}) + ((\sim b_{2n+1}) \cdot$$

$$(\sim \text{field}));$$

$$B_1 = \sim((\text{field} \cdot b_{2n-1}) \wedge b_{2n});$$

$$Z = ((\sim b_{2n+1}) \cdot (\sim b_{2n})) + (b_{2n+1} \cdot b_{2n} \cdot \text{field});$$

从表达式可以得到双域乘法器的控制信号逻辑图和部分积产生逻辑电路图分别如图 4 和图 5 所示.

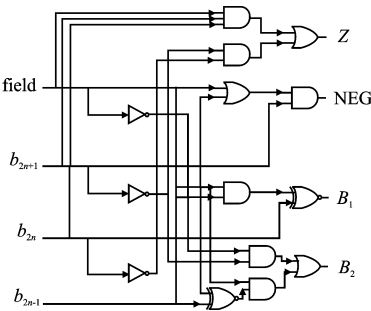


图 4 双域乘法器控制信号逻辑电路

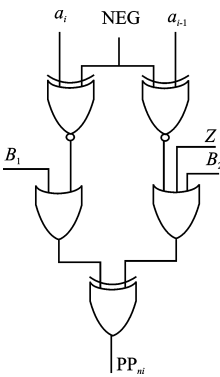


图 5 双域乘法器部分积产生逻辑电路

3.4 双域部分积产生电路进一步改进

对于 GF(p)域的 $-1 \times A2^{2n}$ 和 $-2 \times A2^{2n}$, 产生的部分积是取反后的值,得到的部分积还需要加 1 才能得到补码,如果在取反之后直接加 1,会增加电路的延时,所以把加 1 的操作给到下一个部分积后面两位上.产生的 17 个部分积均为 34 位,如果考虑到取反的情况,那么除了生成的部分积以外,还需要在高位补上对应位数的 1,这样必定会造成资源浪费.为了进行不必扩展,先假设所有的部分积都是负的,那么对于 32 位的乘法可以计算出所有 1 相加的部分积 PP₁₇为

$$PP_{17} = \sum_{i=0}^{16} (2^{33} - 2^{2i}) \bmod 2^{33} = -(1010 \cdots 10101) = (1010 \cdots 1011).$$

32 位乘法器的 PP₁₇为 32'b10101010_10101010_10101010_10101011.综合上面可以得到进行扩展后的部分积的格式如下:

GF(*p*)域:

符号位	34 位部分积	2 位扩展数据位
-----	---------	----------

GF(2^{*m*})域:

1'b0	1'b0, 33 位部分积	2'b00
------	---------------	-------

GF(*p*)域数据的符号位由部分积电路决定,由于计算 PP₁₇时已经假设所有数据为负,所以部分积为负的符号位为 0,部分积为正的符号位为 1. 2 位拓展数是根据上一部分积确定,上一部分积为负,则数据为 2'b01,否则为 2'b00. 对于 GF(2^{*m*})域,相加没有进位,所以对应的数据如上所示.

3.5 双域压缩器设计

3.5.1 Wallace 压缩器选择

Booth 编码减少了乘法器的部分积项数,树形压缩器则减少级联的加法器级数;在乘法器部分积项数一定的情况下,乘法器的延时主要由压缩器级联产生,一个全加器可以看做是一个 3 : 2 压缩器,对于 *r* bit 乘法器而言,级联级数为 log_{3/2}*r*,延时约为 log_{3/2}*r* × Td(fulladder). 高速大型乘法器中通常会采用高压缩比的压缩器^[5]. 对于 *n* : 2 的压缩器,级联级数约为 log_(*n*/2)*r*,图 6 示意了不同压缩比的压缩级数. 显然,对于 32 bit 乘法器而言,4 : 2 压缩器能有效地减少级联级数.

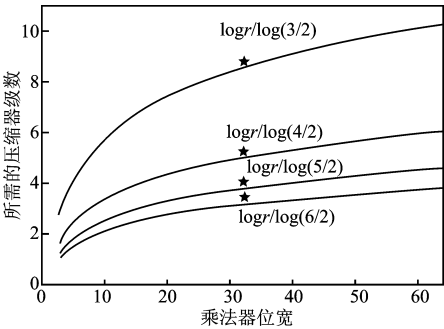


图 6 不同压缩比的压缩性能曲线

3.5.2 双域 4 : 2 压缩器设计

4 : 2 压缩器的作用是将 4 个输入压缩为两个输出,从而加快运算速度. 其逻辑表达式:

$$S=a\hat{b}\hat{c}d;$$
$$Carry=(a\hat{b}\hat{c})\cdot d+(a\hat{b}\hat{c})\cdot Cin+d\cdot Cin;$$
$$Count=a\cdot b+a\cdot c+b\cdot c;$$

通过计算 {Count, sum} + {Carry, 1'b0} 可以得到最后计算结果. 对于双域 4-2 压缩器,由于 GF(2^{*m*})域的压缩没有 Carry 和 Count 位都为 0,可以得到结果如图 7 所示.

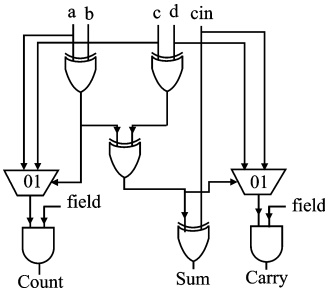


图 7 双域 4-2 压缩器

基于双域 4-2 压缩器的双域 32 位乘法器结构图 8 所示;对于 GF(*P*)域是产生 18 个部分积进行相加,而对于 GF(2^{*m*})域产生的是 16 个部分积相加. 对产生的部分积采用 Wallace 树结构进行压缩求和可得乘法结构. 图 8 中 4-2 压缩器为双域 4-2 压缩器.

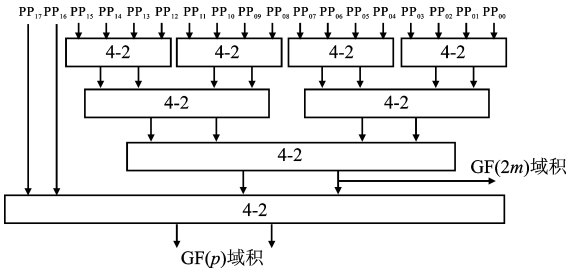


图 8 双域乘法器电路结构

4 双域乘法器电路实现结果及其应用

4.1 双域乘法器实现结果

使用 xilinx 的 KC705 的 FPGA 平台进行双域乘法器的实现占用 LUT 为 1714,比起单独实现两个域的乘法器的面积减少了 16. 9%;采用双有限域 Booth 编码之后,电路的延时将主要受 GF(*p*)域的影响,双有限域最大路径延时为 5. 629 ns,从实现结果表来看,路径延时比单独的 GF(*p*)域乘法器增大 1. 188 ns,且主要延时由 route 布线延时引起. 结果表明本文实现的双域乘法器速度较快,面积较两个域的乘法器单独实现面积省,如表 3 所示.

表 3 32-bit 乘法电路的 FPGA 性能评估

乘法电路	Slice LUT	MaxDelay/ns
GF(<i>p</i>)域	1 575	4. 441
GF(2 ^{<i>m</i>})域	429	1. 619
Dual-Field	1 714	5. 629

(1. 760 1 logic+3. 869route)

4.2 双域乘法器在椭圆曲线加密中的应用

Satoh^[2]等人推导证明了 $GF(p)$ 域和 $GF(2m)$ 域的模乘算法具有相同的数据通路,在底层实现一个双域乘法器则可以实现统一的双域模乘电路;进而实现能同时支持双域的椭圆曲线加密算法。

本文采用自底向上的方法,先优化设计了底层的双域乘法器;然后采用 Satoh^[2]的双域模乘统一架构实现双域模乘器;结合本文设计的 32 bit 双域乘法器,设计了基于字的双域模乘器. 设计的双域模乘器完成一次 256 bit 的模乘操作需要 56 个周期. 对设计的模乘器使用 Xilinx KC705 FPGA 进行实现;并将结果与其他文献进行比较,结果如表 4 所示. 从表 4 看出,与文献[6-9]设计的双域模乘器相比,本文完成一次 256 bit 的模乘操作时间较短,只需要 0.315 μs ,比文献[6]中的模乘器所需时间少 7.35%. 文献[9]完成 256 bit 模乘操作只需 0.28 μs ,比本文快,但其只支持 $GF(p)$ 域.

表 4 双域模乘器实现结果对比

论文	时间/ $\mu s@256\text{ bit}$	频率/MHz	周期数	域
本文	0.315	177.6	56	双域
文献[6]	0.34	143	49	双域
文献[7]	6.60	80	528	双域
文献[8]	0.81	384	312	双域
文献[9]	0.28	250	70	$GF(p)$

5 结束语

本文基于传统的基 4 Booth 编码乘法器,对其部分积产生电路及压缩电路进行改进,使其能够同时支持 $GF(p)$ 域和 $GF(2m)$ 域. 结果表明双域乘法器能有效改善电路面积,速度上较单素数域 $GF(p)$ 慢. 将本文设计双域乘法器应用到双域模乘器中,能有效提高模乘器运算速度.

参考文献:

[1] 全国金融标准化技术委员会. 中国金融集成电路(IC)卡规

范(PBOC3.0)[EB/OL]. (2013-05-22). <http://www.cfstc.org/publish/main/19/60/20130522094255168899577/index.html>

[2] Satoh A, Takano K. A scalable dual-field elliptic curve cryptographic processor[J]. Computers, IEEE Transactions on, 2003, 52(4): 449-460.

[3] Swee K L S, Hiung L H. Performance comparison review of 32-bit multiplier designs[C]// Intelligent and Advanced Systems (ICIAS), 2012 4th International Conference on. Kuala Lumpur, IEEE, 2012 (2): 836-841.

[4] Yeh W C, Jen C W. High-speed booth encoded parallel multiplier design[J]. Computers, IEEE Transactions on, 2000, 49(7): 692-701.

[5] Ma W, Li S. A new high compression compressor for large multiplier[C]// Solid-State and Integrated-Circuit Technology, ICSICT 2008. 9th International Conference on. Beijing, IEEE, 2008: 1877-1880.

[6] 郭晓, 蒋安平, 宗宇. SM2 高速双域 Montgomery 模乘的硬件设计[J]. 微电子学与计算机, 2013, 30(9): 17-21.

[7] Tenca A F, Savas E, Koc C K. A design framework for scalable and unified multipliers in $GF(p)$ and $GF(2m)$ [J]. International Journal of Computer Research, 2004, 13(1): 68-83.

[8] 史焱, 吴行军. 高速双有限域加密协处理器设计[J]. 微电子学与计算机, 2005, 22(5): 8-12.

[9] 刘建国, 张军, 杨晓辉, 等. 有限域模乘专用指令设计[J]. 计算机工程, 2011, 37(21): 105-107.

作者简介:

郑朝霞 女,(1975-),博士,副教授. 研究方向为超大规模集成电路与 SOC 系统设计、嵌入式及信息安全.
E-mail:zxzheng@hust.edu.cn.

资义纯 男,(1991-),硕士. 研究方向为超大规模集成电路与 SOC 系统设计、嵌入式及信息安全.

田园 女,(1991-),硕士. 研究方向为超大规模集成电路与 SOC 系统设计、嵌入式及信息安全.

吴浩 男,(1993-),硕士. 研究方向为超大规模集成电路与 SOC 系统设计、嵌入式及信息安全.