

基于 FPGA 的高速双精度浮点乘法器设计

肖 鹏¹, 江先阳^{2,1}, 王高峰², 汪 波², 刘世培²

(1 武汉大学 物理科学与技术学院, 湖北 武汉 430072; 2 武汉大学 微电子与信息技术研究院, 湖北 武汉 430072)

摘 要: 设计了一种基于 FPGA 的高速双精度浮点乘法器。采用了基 4Booth 算法产生部分积, 然后用优化的 Wallace 树阵列结构完成对部分积的累加得到伪和和伪进位, 进而对伪和和伪进位采用了部分和并行相加得到最后尾数结果。采用了优化的 5 级流水线结构的设计在 Cyclone II EP2C35F672C6 器件上经过综合后运行频率可达 123.32MHz。在同等优化下, 相比于 Altera IP 核在调用 DSP 乘法资源情况下运行速度提高大约 11%, 相比于不调用 DSP 乘法资源情况下运行速度提高大约 67%。

关键词: 基 4Booth 编码; 双精度浮点数; 浮点乘法器; 并行结构; 流水线结构; Wallace 树

中图分类号: TP391.9

文献标识码: A

文章编号: 1000-7180(2012)12-0017-05

A High Speed Double Precision Floating Point Multiplier Design Based on FPGA

XIAO Peng¹, JIANG Xian-yang^{2,1}, WANG Gao-feng², WANG Bo², LIU Shi-pei²

(1 School of Physics Science and Technology, Wuhan University, Wuhan 430072, China;

2 Institute of Microelectronics and Information Technology, Wuhan University, Wuhan 430072, China)

Abstract: To considerably satisfy the requirement, a high speed double precision floating point multiplier based on FPGA is presented. The proposed design adopted radix-4 Booth coder to obtain partial products, then an optimal Wallace tree compression architecture is exploited on these partial products to get a pseudo-sum and pseudo-carry, which are partially accumulated in a parallel approach to generate mantissa of the product. The design with 5-stage pipeline architecture can achieve 123.32MHz targeted at a Cyclone II EP2C35F672C6 device. By taking the same optimization efforts, the design gets 11% enhancement on operating frequency over an Altera IP core using DSP multipliers, while about 67% over such IP core without using DSP multipliers.

Key words: radix-4 Booth coder; double precision floating point number; floating point multiplier; parallel architecture; pipeline architecture; Wallace tree

1 引言

浮点乘法在数据计算精确度要求高的应用环境如雷达信号处理^[1]、3D 图像处理等领域被广泛采用, 执行这一算法的电路——浮点乘法器是现代微处理器的最重要部件之一, 人们对提高浮点乘法器的性能进行了大量的研究。IEEE 754 标准定义了单精度和双精度的浮点数, 本文基于 FPGA 从较通用的角度对双精度浮点数乘法器做了性能改进。设计

采用了基 4Booth 算法^[2], 优化的华莱士树阵列压缩结构以及部分和并行相加的处理技术。

IEEE754 标准规定了两种浮点格式: 32 bit 单精度浮点数和 64 bit 双精度浮点数。每个浮点数据包括有符号位 S、阶码 EXP 以及尾数 M。IEEE 754 标准定义的双精度浮点数格式^[3]如图 1 所示。

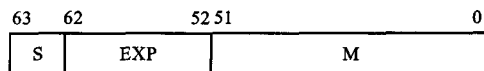


图 1 IEEE754 双精度浮点数据格式

收稿日期: 2012-04-13; 修回日期: 2012-05-11

基金项目: 国家自然科学基金(61072135, 60788402); 武汉市科技攻关计划项目(201110921295)

如果考虑图 1 中没有给出的浮点数隐含位 1, 一个普通的双精度浮点数 P 可以表示为

$$P = (-1)^S * 1.M * 2^{(EXP-BIAS)} \quad (1)$$

式中, $BIAIS$ 表示偏移量, 对于双精度浮点数 $BIAIS$ 值为 1023, $(EXP-BIAIS)$ 为双精度浮点数 P 的指数.

2 双精度浮点乘法算法

一般而言, 双精度浮点数乘法主要包括以下步骤^[4]:

①符号位运算: 被乘数 A 和乘数 B 的符号位异或得到乘积 C 的符号位.

②阶码运算: 两个操作数 A 和 B 的阶码通过式 (2) 得到乘积 C 的阶码.

$$EXP_C = EXP_A + EXP_B - 1023 \quad (2)$$

③尾数运算: 两个操作数 A 和 B 的尾数通过式 (3) 得到乘积 C 的尾数.

$$1.M_C = 1.M_A \times 1.M_B \quad (3)$$

④规格化处理: 使计算结果调整为 IEEE754 标准格式的双精度浮点数, 其中包括对于尾数乘积的舍入处理 (本文采用最近舍入模式), 尾数乘积舍入后的修正处理, 以及阶码的加一修正处理.

⑤最终尾数赋值: 汇总了各种情况的结果. 其中包括处理异常情况: 两个浮点操作数 A 或 B 是特殊数据时的处理, 例如 A 或 B 为 0, 无穷大, 非规格化数以及无效数时的处理; 乘积结果发生了上溢 (OVERFLOW), 下溢 (UNDERFLOW), 结果为 0 (ZERO), 结果为无效数 (INVALID) 时的处理; 结果为正常数的处理.

上述步骤通过图 2 的流程图实现.

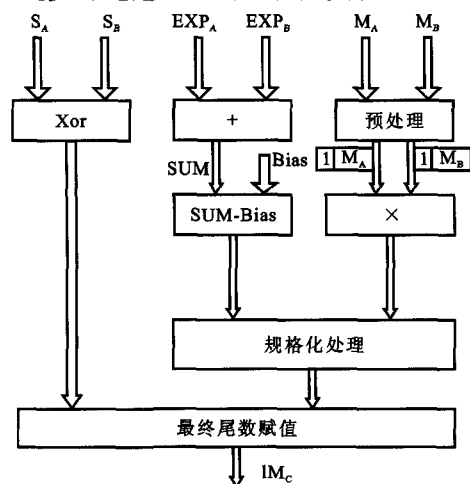


图 2 双精度浮点乘法器流程图

浮点乘法器中尾数位的相乘是最关键且比较复杂

的部分, 以下阐述主要集中于这一部分的设计.

3 尾数相乘的硬件逻辑设计

双精度浮点数的尾数位有 52 位, 再加上一个隐藏位“1”, 有效数共有 53 位, 因此双精度浮点数的尾数相乘实际上是两个 53 位无符号数的相乘, 尾数相乘主要包含以下逻辑步骤:

①Booth 编码: 对于 53 位无符号有效数相乘, 采用基 4Booth 算法, 扩展 53 位无符号定点数据 A 和 B 的最高位使其成为有符号数, 然后对得到的有符号数进行基 4Booth 编码, 根据 Booth 编码输出部分积;

②压缩: 对于 Booth 编码得到的部分积, 采用优化的 Wallace 阵列结构进行压缩, 得到最终的伪和和伪进位;

③部分相加: 对于 Wallace 阵列结构得出的两个 106 位数据伪和和伪进位, 采用部分并行相加逻辑, 得到最后的尾数结果.

3.1 基 4Booth 编码

为了满足基 4Booth 算法要求, 需要定点乘数满足偶数位的要求, 因此对于定点乘数 A 如果其位宽 N 为奇数, 则需要将 A 作符号扩展为 A' , 使其扩展后的位宽 H 为偶数, 这里 H 不小于 N . 如果 N 为偶数, 符号扩展需要增加 2 位变成 A' .

具有偶数位 H 的乘数 A' 可通过式 (4) 表示.

$$A' = \sum_{n=0, A[-1]=0}^{(H/2)-1} ((-2A'_{[2n+1]} + A'_{[2n]} + A'_{[2n-1]}) * 2^{2n}) = \sum_{n=0}^{(H/2)-1} (E[n] * 4^n) \quad (4)$$

式中, $0 \leq n \leq (H/2-1)$, $A'[-1]=0$.

基于式 (4), 定点数据 A 和 B 相乘后所得的乘积 C 可由式 (5) 表示.

$$C = A * B = A' * B = \sum_{n=0}^{(H/2)-1} [(E[n] * 4^n) * B] \quad (5)$$

式 (5) 定义了符号数乘法的基 4Booth 算法. 经过基 4Booth 算法处理后, 产生数目为 $(H/2)$ 的部分积, 与不用基 4Booth 编码方法相比部分积数目降低一半. 基 4Booth 编码表^[5]如表 1 所示.

3.2 压缩

在高速乘法器中通常采用进位保留加法器 (CSA) 对部分积进行相加压缩, 产生伪和和伪进位. 这种结构只在最后一步才采用进位传播加法 (carry propagate addition) 来提高乘法器的速度. 由于基 4Booth 算法产生的部分积数目为 $(H/2)$, 双精度浮

点数乘法压缩阶段由基4Booth算法产生的部分积数目为27,如果使用传统的CSA阵列压缩结构直接进行串行相加,则需要25级CSA,如图3所示.如果使用传统的纯CSA基础上的Wallace树形结构,则需要7级CSA,如图4所示.为了缩短关键路径,本文提出采用由4-2压缩器和CSA(也即3-2压缩器)的混合Wallace树形结构^[6]来执行压缩,如图5所示.

表1 基4Booth编码表

$A'[2n+1]$	$A'[2n]$	$A'[2n-1]$	$E[n]$	对部分积的影响
0	0	0	+0	+0
0	0	1	+1	+B
0	1	0	+1	+B
0	1	1	+2	+2B
1	0	0	-2	-2B
1	0	1	-1	-B
1	1	0	-1	-B
1	1	1	-0	-0

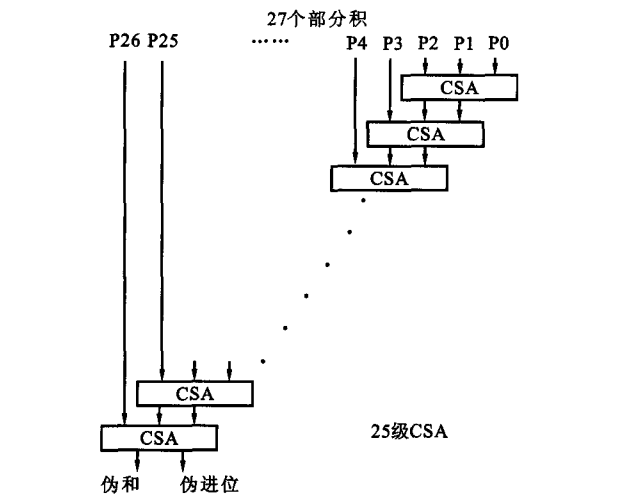


图3 串行CSA阵列

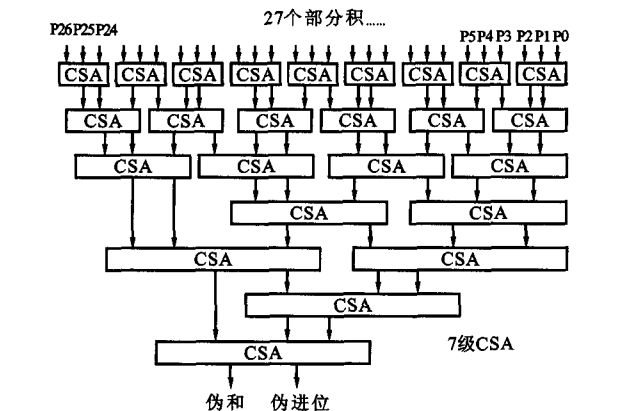


图4 纯CSA阵列Wallace树形结构

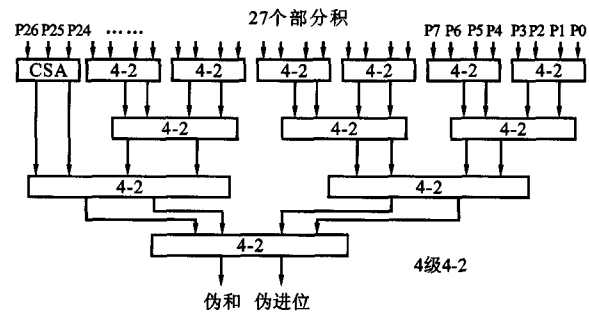


图5 提出的混合Wallace树形压缩结构

CSA也即3-2压缩器由N个一位全加器组成,其延时为一个全加器的延时,并且基本不受位数N的影响.4-2压缩器为改进型4-2压缩器^[7],在同等位数的情况下,改进型4-2压缩器所使用的资源为3-2压缩器的两倍,但延迟仅为3-2压缩器的1.5倍.基于此,提出的混合Wallace树形压缩结构的关键路径延迟为4级4-2压缩器延迟或者说6级CSA延迟,优于纯CSA构成的Wallace阵列压缩结构具有的7级CSA延迟,提高了运算速度.

3.3 伪和和伪进位的部分和相加

Wallace压缩结构输出了2个106 bit的伪和和伪进位,在不损失精度的情况下,采用部分并行相加计算,即用一个55位加法器和一个51位加法器代替一个单一106位加法器^[8],如图6所示,可以有效提高计算速度.新结构中,低51位加法器的进位输出Cout并不作为高55位加法的进位输入信号,而是直接用于尾数的规格化处理,从而同时保证并行计算和计算精度,以提高整个乘法的运算效率.

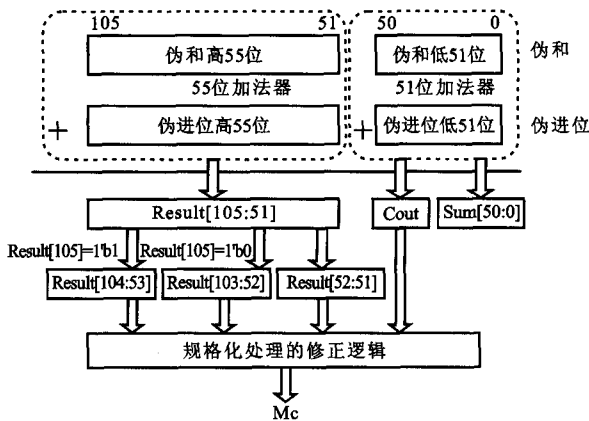


图6 106位数据伪和和伪进位的部分和相加结构

3.4 流水线处理

为了进一步提高乘法器的运算速度,采用了多级流水线结构,作为具体实现,其中5级流水线结构

如图 7 所示。

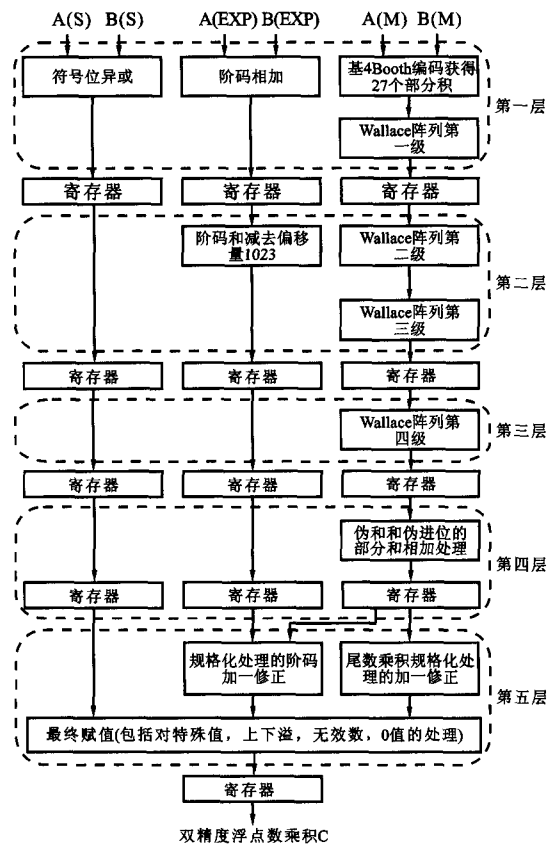


图 7 提出的 5 级流水线乘法器结构

4 实现与验证

4.1 实现

为了实际测试提出的设计的性能,我们基于 Altera 的 Quartus II 在 Cyclone II EP2C35F672C6 器件上对设计进行了实现,表 2 给出了在同等优化和 5 级流水线结构条件下所提出的设计与 Altera 的 IP 核的性能对比。

表 2 提出的设计与 Altera 的 IP 核在 EP2C35F672C6 器件上的性能对比

	LE 数目	寄存器 数目	9-bitDSP 乘法器	运行频率 (MHz)
本文设计	7053	1601	0	123.32
Altera IP (未调用 DSP 资源)	3733	1990	0	73.96
Altera IP (调用 DSP 资源)	938	696	18	110.64

提出的设计完全由逻辑单元实现,没有调用片内高效 DSP 乘法资源。从表 2 的实现结果可以看出,本文设计虽然占用了较多的逻辑资源 LE,但是在速度上有着明显的优势,运行频率达到了 123.

32MHz。在 5 级流水线的情况下,与未调用 DSP 资源的 Altera IP 核相比,运行频率提高了 66.7%;与调用了 DSP 资源(18 个 9 bit 的嵌入乘法单元)的 IP 核相比,运行频率提高了 11.5%。因此表明我们提出的设计具有更高的计算性能。

4.2 验证

设计用 Modelsim 软件进行了功能仿真与时序仿真,在超过 100 万个随机数据测试上,提出的设计与 Altera 的 IP 核具有同样精度的正确结果。图 8 为进行时序仿真的部分对比结果,显示了两者结果的一致性。



图 8 时序仿真的部分对比结果

5 结束语

本文提出了一种基于 FPGA 的高速双精度浮点乘法器。该设计采用基 4Booth 算法和优化的 Wallace 树形压缩结构,减少了一半的部分积数目,缩短了部分积累加的耗时。同时对于尾数乘积运算中 Wallace 树产生的伪和和伪进位采用了部分并行相加的技术,减少了伪和和伪进位的相加时间,进一步采用 5 级流水线结构,提高了乘法器的运算速度。设计通过 Modelsim 仿真与 Altera IP 核进行对比验证了设计的正确性。基于 Cyclone II EP2C35F672C6 器件的实现运行频率可达 123.32MHz,与在同等优化条件下的 Altera IP 核相比,运算速度有较大幅度的提升。该浮点乘法器能够作为 IP 核应用于各种高精度高速数字信号处理系统中。

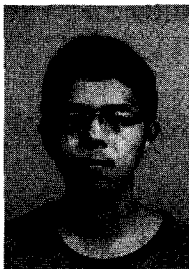
参考文献:

[1] 胡正伟,仲顺安. 10 级流水线双精度浮点乘法器的设计[J]. 北京理工大学学报. 2007,27(4):349-353.
[2] 何晶,韩月秋. 一种双精度浮点乘法器的设计[J]. 微电子学,2003,33(4):331-334.
[3] 张书. 高性能 CPU 中浮点乘法器的设计与实现[D]. 上海:同济大学,2005.
[4] Manolopoulos K, Reisis D, Chouliaras V A. An efficient multiple precision floating-point multiplier. Elec-

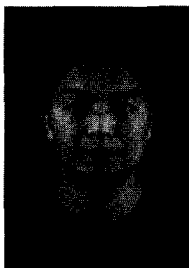
tronics[C]//Circuits and Systems (ICECS). Lebanon, Beirut, 2011.

- [5] 刘鸿瑾,张铁军,侯朝焕. 基于快速舍入的双精度浮点乘法器的设计[J]. 微电子学与计算机, 2006, 23(6): 162-165.
- [6] Gong Renxi, Zhang Shangjun, Zhang Hainan. Hardware implementation of a high speed floating point multiplier based on FPGA[C]//Proceedings of 2009 4th International Conference on Computer Science & Education. China: Wuhan, 2009.
- [7] 旷捷,毛雪莹,彭俊淇,等. 基于 FPGA 的单精度浮点数乘法器设计[J]. 电子技术应用, 2010(5): 17-19.
- [8] Venishetti S K, Akoglu A. A highly parallel FPGA based IEEE-754 compliant double-precision binary floating-point multiplication algorithm[C]//Field-Programmable Technology. Taiwan: Taipei, 2007.

作者简介:



肖 鹏 男,硕士研究生. 研究方向为集成电路设计.



江先阳 男,博士,副教授. 研究方向为 VLSI 设计、计算机体系结构、可重构计算等.

(上接第 16 页)

- [6] Raja T, Mourad, S. Digital logic implementation in memristor-based crossbars [C]//ICCCAS (2009). USA: California, 2009: 939-943.
- [7] Wei Lu, Kuk-Hwan Kim. Two-terminal resistive switches (memristors) for memory and logic applications [C]//ASP-DAC (2011). Japan: Yokohama, 2011: 217-223.
- [8] Q Xia, W Robinett. Memristor-CMOS hybrid integrated circuits for reconfigurable logic[J]. Nano letters, 2009, 9(10): 3640-3645.
- [9] Rose G S. Leveraging memristive systems in the construction of digital logic circuits[J]. Proceedings of

the IEEE, 2011(99): 1-17.

- [10] Zdeněk BIOLEK. SPICE model of memristor with non-linear dopant drift [J]. Radioengineering, 2009, 18 (2): 210-214.
- [11] Yang J J. High switching endurance in TaOx memristive devices[J]. Appl Phys Let 97, 2010, 232102(1-3): 10.

作者简介:

朱平平 男, (1986-), 硕士研究生. 研究方向为忆阻器、嵌入式应用.