文章编号: 2095-4980(2019)03-0524-07

适用于 FPGA 的浮点型 DSP 硬核结构设计

赵 赫^{1,3},黄志洪¹,余 乐*2,杨海钢*1,3,许仕龙⁴,郝亚男⁴

(1.中国科学院 电子学研究所,北京 100190; 2.北京工商大学 食品安全大数据技术北京市重点实验室,北京 100048; 3.中国科学院大学 微电子学院,北京 100049; 4.中国电子科技集团公司 第五十四研究所,河北 石家庄 050081)

摘 要:提出一种浮点型数字信号处理器(DSP)硬核结构,在兼容定点数运算的同时,也为浮 点数运算提供较好支持。目前各大现场可编程门阵列(FPGA)主流厂商在实现浮点数运算功能时均 采用软核实现方式,即将浮点数运算算法映射到芯片上,通过逻辑资源和 DSP 模块实现。相比于 传统方法,提出的硬核结构在不占用 FPGA 中其他逻辑资源情况下,仅利用 DSP 模块便能完成浮 点数运算。设计中,充分考虑负载和时延影响,插入多级流水线,显著提高浮点数的计算效率。 采用中芯国际(MCI)28 nm 工艺设计并完成所提出的浮点型 DSP 硬核结构。仿真结果表明,所提出 的硬核结构的单个浮点数加法和乘法效率为 0.4 Gflops。

关键词:现场可编程门阵列;数字信号处理器;硬核结构;浮点数运算 中图分类号:TN47 **文献标志码:**A **doi:** 10.11805/TKYDA201903.0524

A suitable design of floating-point DSP hard core structure in FPGA

ZHAO He^{1,3}, HUANG Zhihong¹, YU Le^{*2}, YANG Haigang^{*1,3}, XU Shilong⁴, HAO Yanan⁴

(1.Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China;

2.Beijing Key Laboratory of Big Data Technology for Food Safety, Beijing Technology and Business University, Beijing 100048, China;
 3. School of Microelectronics, University of Chinese Academy of Sciences, Beijing 100049, China;

4.The 54th Research Institute, China Electronics Technology Group Corporation, Shijiazhuang Hebei 050081, China)

Abstract: A floating-point Digital Signal Processor(DSP) hardcore structure is proposed. The hardcore structure not only is compatible with fixed-point operation, but also provides better support for floating-point operation. At present, the major manufacturers of Field-Programmable Gate Array(FPGA) use soft core to implement floating-point arithmetic, which maps floating-point arithmetic to the chip and realizes it through logical resources and DSP module. Compared with traditional methods, the proposed hard-core architecture can complete floating-point arithmetic only by using DSP module without occupying other logic resources in the FPGA. At the same time, in the design process of the DSP, the influence of load and time delay is fully considered, and the multi-level pipeline is reasonably inserted, which greatly improves the calculation efficiency of floating-point number. The floating-point DSP hard-core structure is designed and completed by the Semiconductor Manufacturing International(MCI) 28 nm technology. The simulation results of the proposed hard-core structure show that the single floating-point efficiency of addition and multiplication is 0.4 Gflops.

Keywords : Field-Programmable Gate Array(FPGA); Digital Signal Processor(DSP); hardcore structure; floating point

目前工业界主流的 FPGA 产品中基本都集成了可编程数字信号处理(DSP)模块。例如, Xilinx 公司 Virtex-7 中含有 3 600 个 DSP48E1 单元,支持乘加/乘减/乘累加等操作^[1]; Altera 公司 Stratix-V 含有 532 个 DSP 单元。

收稿日期: 2017-11-27; 修回日期: 2018-01-24

·通信作者:余乐 email:ladd_u@163.com;杨海钢 email:yanghg@mail.ie.ac.cn

基金项目:国家自然科学基金资助项目(61876172,61704173);北京工商大学食品安全大数据技术北京市重点实验室开放课题基金资助项目 (BKBD-2017KF05);北京工商大学食品安全大数据技术北京市重点实验室开放课题基金资助项目(BKBD-2017KF05);北京市科技 重大专项课题(Z171100000117019)

单个 DSP 的 IP 核可根据应用需求进行拆分,以最小的资源实现最多的功能,支持乘加、乘减、乘累加等操 作,但不支持加法、累加运算^[2]。FPGA 在进行数字信号处理的过程中需调用众多的 DSP 模块对信号进行各种 数学运算,商用 FPGA 产品基本采用定点型硬核 DSP 结构,定点数的计算更加高效、功耗低^[3-6],但随着数据 量的不断增加,所需要处理的信号也由原来的定点数表示变为数值范围更大的浮点数表示,如雷达信号、导航 等大多采用浮点数的表示方式,这些信号在进行处理时需调用大量的浮点数运算单元参与信号运算。

面向该应用需求,Altera 和 Xilinx 公司的 FPGA 产品都提供了浮点数运算的相关 IP 软核。以 Xilinx 公司为例,通过 Vivado 中的 IP Catalog 功能进行 IP 调用,能够支持包括多种浮点数运算,除基本运算之外,还提供指数、对数、开方等运算^[7]。目前的 FPGA 产品中的 DSP 模块都是采用定点型的 DSP 结构,在 Altera 的 EDA 软件 Quartus II 和 Xilinx 的 EDA 软件 Vivado 中,将浮点数运算中的逻辑控制部分映射到 FPGA 的 LUT 表等逻辑资源中,将浮点数运算的乘法、加法等操作映射到 DSP 的定点乘法器、加法器中。这种方法虽然方便,但占用的资源过多,且 IP 软核的运算效率不高。由于对浮点数高速运算的应用需求,Intel 公司在其最新的产品中嵌入了浮点型硬核 DSP 模块来提高 FPGA 对浮点数计算的支持,但目前为止未有芯片提供。因此,本文针对浮点型 DSP 的设计展开研究,提出一种硬核的浮点型 DSP 结构。

1 浮点型硬核 DSP 结构

设计的 DSP 架构框图如图 1 所示,主要由 5 部分组成:输入模块、乘法器单元、选择器组模块、算术逻辑 单元(Arithmetic Logic Unit, ALU)、输出模块。输入模块主要为乘法器单元提供操作数,为保证 DSP 的工作频 率,在本模块中可以引入多级流水线,对关键路径进行分割,从而缩短关键路径。同时,设计 DSP 级联模式, 通过级联的方式高效完成更复杂的操作,提高 DSP 的灵活性。



乘法器单元包括一个二输入的乘法器单元(A×B),通过 booth 编码的方式对操作数进行乘法操作,并压缩 部分积的个数,同时结合树形的加法器进一步对部分积进行压缩,以减少面积开销,提高计算速度。该单元可 同时支持定点数和浮点数乘法运算。结合前导 0 电路对所得的结果进一步修正,并同样引入流水线结构,提高 工作频率。选择器组模块由多个选择器组成,由相应的选通信号对选择器进行选择,从而完成不同功能之间的 切换,也可改变输入到下一级加法器中的数据来源。ALU 的主要单元是为浮点数和定点数提供加减以及乘法运 算所使用的加法器,同时也为定点数提供逻辑运算。其中,为了使 ALU 单元能够支持浮点数的运算,除了本身 的基本运算结构以外,还加入了前导 0 预测(Leading Zero Anticipation, LZA)电路、一位误差调整电路以及舍入 模块,从而在完成浮点数运算的同时,将输出结果规格化,产生符合标准的浮点数输出。

输出模块引入流水线结构,同时其寄存器为前级加法器单元提供了寄存单元,计算后的结果可以被寄存到 加法器中,这样可以将结果用于累加的操作中。同时,输出模块预留了级联的端口。 执行浮点数乘法时,需要对符号位进行异或操作,以确定最终结果的符号位。将指数位相加得到新的指数位,尾数位相乘得到新的尾数部分,再根据浮 点数格式对得到的指数位和尾数位进行调整,从而得 到浮点数乘法结果,操作如图2所示。

为了实现上述计算过程,在输入模块中加入前加 器模块,算法的 DSP 结构实现框图如图 3 所示,前加 器作为一个专用的指数加法器会对输入数据的 32 bit 数据中的指数部分,即 24 位至 31 位的指数部分进行 前加操作,得到的结果作为浮点数乘法的指数部分, 但在这个过程中存在如下问题:当输入数据是一个非 规格化浮点数数据,即指数位为全 0 或全 1,这种情 况下无法进行浮点数的运算。IEEE754 协议也规定了 在这种情况下的输出结果:当输入的指数位全为 1 时,此时的输出结果为原符号位、8 bit 全 1 的指数位 以及 23 bit 全 0 的尾数位;当输入指数位全为 0 时, 此时的输出结果为原符号位、8 bit 全 0 的指数位以及 23 bit 全 0 的尾数位^[8]。因此在前加器的设计中加入逻 辑来探测非规格化的浮点数,并对其进行标记。

根据 IEEE-754 协议,任何规范浮点数的尾数部分为 23 bit,在这 23 bit之前还有一个数值为 1 的隐藏位不在浮点数中表示出来,但在参与浮点数运算时,这一位隐藏位需补出来参与运算,因此将乘法器位宽设计为 25×25 bit,数据格式如图 4 所示。对尾数进行符号位补 0 操作,并将尾数视为正数进行运算,同时为得到真正的符号位,将输入的 2 个浮点乘数的符号取出,单独进行逻辑异或运算,从而得到结果真正的符号位。浮点数的尾数位宽为 25 bit,按照常规乘法实现方式会产生 25 个 25 bit 位宽的部分积,这个过程会产生很大的延时,严重影响 DSP 模块的工作频率。为了减少部分积的数量,采用 Booth 算法生成部分积。

Booth 算法的核心思想是每次用多于一位的乘数 和被乘数做逻辑运算,通过这种方法压缩部分积的数 目^[9]。根据操作数的位宽进行分析可知,如果采用乘 数相邻 2 位来决定对被乘数操作的基为 2 的 Booth 算





图 4 尾数乘法的补位操作

法,部分积的数量并不会被压缩。如果采用乘数中 4 位以上相邻位数,虽然可以产生更少的部分积,但硬件开 销太大^[10]。因此,本文采用基为 4 的 Booth 算法,即根据乘数中的相邻三位来决定对被乘数的操作,将部分积 的个数压缩为 13 个。

此外,为进一步提高乘法器运算速度,乘法器中还使用了 Wallace 树压缩结构^[11]。为得到乘法最终结果, 传统方法是通过按顺序累加部分积的方式,但是累加操作会制约乘法器运算速度。Wallace 树结构会将部分积进 行分组,各组进行累加操作,通过这种方式压缩部分积的数量,然后再进行分组累加的过程,直至得到结果, 该结构可大大提高部分积的运算速度。

在得到指数部分和尾数部分的值之后,需要对指数部分和尾数部分 进行调整,对相关的浮点数进行规格化处理。在进行规格化处理过程中 需要用到前导 0 探测(Leading Zero Detection, LZD)模块来探测尾数部分 结果中 0 的个数,从而对尾数部分进行移位操作^[12]。LZD 模块中,第一

| 表12 bit LZD 模块真值表 | | | | | | |
|--|----------|-------|--|--|--|--|
| Table1 Truth value table of 2 bit LZD module | | | | | | |
| series | position | valid | | | | |
| 1X | 0 | 1 | | | | |
| 01 | 1 | 1 | | | | |
| 00 | Х | 0 | | | | |

个 1 之前有多少个 0,这个数值由位置位进行表述, 此外还有另外一个有效位来表示所输入的数列是否有 效。以 2 bit 的数列为例, 2 bit 的 LZD 模块真值表如 表 1 所示,当数列为 "00"时,1 bit 的位置表示位已 经无法表示 0 的个数,因此位置位为 "X",无效。通 过该表可以看出,位置位的作用是表示 0 的个数,有 效位是表示所输入的数列是否有效。

由 2 bit 的 LZD 模块推广到 4 bit 的 LZD 模块, 其真值表如表 2 所示,结构如图 5 所示。在 4 bit 的 LZD 模块中用 V0 和 P0 表示数列的高两位,用 V1 和 P1 表示数列的低两位,利用 2 bit 的 LZD 模块的真值 表可以得到 V0,V1,P0,P1 的值。4 bit 电路有一个深度 为 2 的逻辑层:输出的有效位的结果由前一层的 2 个 有效位通过或运算得到,只有当前级的所有有效位都 为 "0"时,最终的输出有效位才为 "0"。同时,前一 层的高两位的有效值 V0 选通多路选择器的输出,决 定了最终输出的位置位的值。将上述的基本模块组成 树形结构,可以将 LZD 模块推广到多 bit 的情况,用 于对浮点数尾数部分乘积结果的前导 0 探测。

在获得了前导 0 的个数值之后,对尾数结果进行 相应的移位操作,同时将指数的值减掉前导 0 的个数 值,再将最终的符号位、8 bit 指数部分和 23 bit 的尾 数部分组合起来,可得到最终浮点数乘法的结果。

1.2 浮点数加法的实现

第3期

执行浮点数加法时,先通过比较器选出 2 个输入 值 A 和 B 中绝对值比较大的数,然后比较 2 个数的指 数部分,对小的数进行对阶操作,使其和数值大的数 的指数相同,同时对小的数的尾数部分进行相应的移 位操作,然后将两数的尾数部分进行加减操作,对结 果进行调整,从而得到浮点数加法的最终结果,其算 法流程图如图 6 所示。所采用电路结构如图 7 所示,2 个浮点数 A 和 B 输入到 ALU 单元中,会分别送入 2 条路径,一路信号利用加法器对 2 个浮点数进行加减 操作,得到的结果通过 LZD 模块进行一个初步的移位 和指数的调整;另一路信号被编码后送入探测树结构 中,产生指示信号,决定是否需要对初步移位的信号 进行进一步调整,以得到浮点数加减法运算的结果。

实际运算中,同符号数加法不需要通过编码和探测树模块,因为这种运算条件下结果不会产生误差, 但异号的加法或同号的减法需要进行结果调整。在进 行异号加法或同号减法的过程中存在1bit误差^[13]。以 同号减法为例,可以得到如下的等式:

$$W = A - B \tag{1}$$

式中 A 和 B 的每一位都在进行减法的操作,即

$$w_i = a_i - b_i, \ w_i \in \{-1, 0, 1\}$$
(2)

将-1 表示为 ī。数列 W 决定了前导 0 的个数,即 决定了浮点数规格化时的指数部分值以及尾数部分的



(3)

移位。现在考虑 W>0,W<0 和 W=0 的情况,使用 x 表示值为任意的子数列, 0^k , 1^k , $\overline{1}^k$ 分别表示数列中有 k 个连续的 0,1, $\overline{1}$, $k \ge 0$ 。

当 *W*>0 时, *w_i*中第一个不为 0 的数一定是 1。因此 *w* 数列为 0^{*k*}1(*x*)。在子数列中的第一个数可能为 0 或为 1。通过下面两种情况进行说明:

1) 当第一个1后面是1时,即

$$W = 0^k 11(x)$$

很显然第一个 1 的位置是在 *k*+1, 由于 *k*+2 位置也为 1, 因此即使子数列 *x* 中出现借位的情况, 也不会影响 *k*+1 位上的数值。

2) 当第一个1之后的数为0时,即

$$W = 0^k 10(x) \tag{4}$$

现在有两种可能的情况:如果(x)是正值(或者 0),即

$$W = 0 \cdots 010000 \cdots 1 \cdots \tag{5}$$

因为没有来自(*x*)的借位,所以第一个1的位置在*k*+1。 如果(*x*)是负值,由于(*x*)产生了借位,因此第一个1的位置为*k*+2,即

 $W = 0 \cdots 010000 \cdots \bar{1} \cdots = 0 \cdots 001111 \cdots 1 \cdots$ (6)

因为这个位置是依靠 $\bar{1}$ 之前的 0 的个数来确定,因此,第一个 1 的位置就在 k+1 或 k+2。利用相同的方法,也可以发现当 W<0 时,也存在着由于借位使第一个 1 的位置发生一位右移的问题。所以假设第一个 1 的位置在 k+1,之后再对结果进行修正。将 W 数列进行编码^[14],首先将数列 W 中的每一位 w_i 的值用 $e_i(w_i > 0)$ 、 $p_i(w_i > 1)$ 和 $m_i(w_i > -1)$ 来表示,其次根据新得到的数列 W,可得到数列 U,V,S,T,Z,其中每个数列的每一位通过式(7)表述。

$$\begin{cases}
u_{i} = (e_{i+1}p_{i} + \overline{e_{i+1}}m_{i})e_{i-1} \\
v_{i} = (e_{i+1}m_{i} + \overline{e_{i+1}}p_{i})e_{i-1} \\
s_{i} = (e_{i+1}p_{i} + \overline{e_{i+1}}m_{i})p_{i-1} \\
t_{i} = (e_{i+1}m_{i} + \overline{e_{i+1}}p_{i})m_{i-1} \\
z_{i} = 1 - (u_{i} + v_{i} + s_{i} + t_{i})
\end{cases}$$
(7)

主? 会》 描述主

通过式(7)可知,在*i*相同情况下,XXXXXX 只有一个为1,可得到探测树真值表。在得到以上5个数列之后,即可得到数列*R*,数列*R*的每一位用*u*,*v*,*s*,*t*,*z*表示。根据探测树真值表,对数列*R*进行化简,即可得到最终的值,如果最终的结果是*x*或*y*,则需要对LZD模块中的输出结果进行一位修正,否则,不需要进行修正。

1.3 浮点数的舍入操作

本文舍入模块支持 4 种舍入模式:向最近的偶数舍入、向 0 舍入、向正无穷舍入、向负无穷舍入^[15]。在进 行浮点数运算的过程中,由于尾数乘法和尾数加法的移位对阶操作,尾数部分的位宽变为 48 bit,但实际只需 要 24 bit 的数据,因此在数据的截取过程中需要对数据进行舍入操作。在舍入模块中将数据的低 22 位进行或运 算,结果作为粘滞位(S 位),第 23 位做为舍入位(R 位),第 24 位做为保护位(G 位),这三位拼接在高 24 bit 后 面,如图 8 所示。依据模式、(LSB,G,R,S)、以及符号位确定是否进位,具体的舍入方法见表 3。

| | | | | | | | | 仪 | 5 百八侠以衣 | | |
|--|-------|------|-------------------------------|-------------|-------------------------------|-----------|---------|--------------------|-------------|------|-------|
| | | | | | | | | Table3 Round table | | | |
| | [24] | [23] | [22] | [21] | [20] | [19] | [0] | round | (LSB,G,R,S) | sign | carry |
| | | | I | | | | | rounding to zero | XXXX | х | 0 |
| | | | | | x000 | х | 0 | | | | |
| | OR OR | | | | rounding to position infinity | othomyjag | + | 1 | | | |
| LOD | | | |] | | | | | outerwise | - | 0 |
| LSB | G | R | S | • | | 1 | | | x000 | х | 0 |
| Fig.8 Guard, round, and sticky logic 图 8 保护、舍入和粘滞逻辑 | | | rounding to negative infinity | otherwise - | + | 0 | | | | | |
| | | | | | - | 1 | | | | | |
| | | | rounding to nearest even | ≤0100 | | 0 | | | | | |
| | | | | | >0100 | х | 1 | | | | |

2 实现与比较

本文提出的浮点型硬核 DSP 结构具有浮点数加法、乘法、以及累加的功能,为验证本结构的功能性是否正确,在测试文件中调用 DC 标准单元库中的浮点数加法、乘法模块,并搭建了浮点数的累加模块作为测试 基准,与本文所提出的结构输出结果进行对比,以确定结构功能的正确性。在输入相同的情况下,对比本文结构和基准模块,浮点数乘法、加减法、累加的结果都相等。

采用标准中芯国际 28 nm CMOS 工艺库,电压为 1.05 V,温度 25 ℃,利用 DC 工具完成浮点型硬核 DSP 的电路实现,通过布局布线得到的整体面积为 11 091 μm²。

商用 FPGA 采用的都是定点型硬核 DSP 结构,因此本文利用浮点数软核作为比较对象。由于 Xilinx 器件给出的浮点数性能指标与 FPGA 片上的 DSP 模块数量直接相关,单个 DSP 模块的浮点数性能乘以 DSP 模块的数量即为手册中浮点数性能指标的数值。因此本文只比较单个浮点数软核与硬核之间的参数指标。

对比同样工艺节点的 Xilinx 7 系列 xc7v585tffg1157-3 FPGA 芯片,在 Vivado 中设置相同 latency 的情况下 (本文采用 6 级流水线结构),调用浮点数加减法软核模块,选择速度优先的结构优化,该模块被映射到 1 个 DSP48E1 和逻辑资源上,分别设置性能和资源作为优化目标。性能指标通过 Vivado 的综合报告给出,最高工作 频率通过给模块施加的时许约束获得,将上述指标和本文结构进行比较。同时调用浮点数乘法软核模块进行比较,结果如表 4 所示。

表 4 综合结果对比表

| Table4 Comparison of simulation results | | | | | | | |
|---|-----------------------------|--------------|-------------------------|--|--|--|--|
| | | structure | Xilinx 7 implementation | | | | |
| | maximum operation frequency | 400 MHz | 220 MHz | | | | |
| float-point addition | power | 1.12 mW | 267 mW | | | | |
| | performance | 0.4 Gflops | 0.22 Gflops | | | | |
| | maximum operation frequency | 416 MHz | 290 MHz | | | | |
| float-point multiplication | power | 1.65 mW | 284 mW | | | | |
| | performance | 0.416 Gflops | 0.29 Gflops | | | | |

可以看出,在执行浮点数加法时,本文所提出的浮点型硬核 DSP 结构的浮点数加法运算性能是性能优先条件下 IP 软核的 1.8 倍;浮点数乘法运算性能是性能优先条件下 IP 软核的 1.43 倍。

本文所提出的浮点型硬核 DSP 结构的浮点数运算效率明显优于 Xilinx 7 系列用软核的方式实现浮点数运算。这是由于 IP 软核实现的过程中需要片上逻辑资源的参与,数据由片上逻辑资源处理后送入 DSP 运算和 DSP 运算后的结果再送出到片上逻辑资源再处理这两个过程需要花费较长的时间,而本文所提出的浮点型硬核 DSP 结构对数据的处理和运算都在该结构内部完成,而且利用 FPGA 中的逻辑资源映射浮点数调整等电路结构 不如专用的电路设计高效。

3 结论

本文提出了一种 FPGA 中的嵌入式浮点型硬核 DSP 结构,该结构能够通过配置实现加减法、累加、乘法等 功能,在兼容传统商用 FPGA 的 DSP 结构所能实现的定点计算的同时,能有效实现应用越来越广泛的浮点运 算。相比于原有商用 FPGA 通过定点 DSP 模块结合逻辑模块实现浮点数运算的方法,该结构减少了浮点数计算 过程中对 FPGA 中逻辑资源的消耗,相比于 Xilinx 公司的浮点数运算 IP 软核,该结构具有更高的浮点数运算性 能和较低的功耗。

参考文献:

- [1] XILINX. Virtex-5 FPGA XtremeDSP design consideration user guide[R/OL]. Xilinx Inc, 2017. (2017-07-27)[2018-01-24]. http://www.Xilinx.com/support/documentation/user_guides/ug193.pdf.
- [2] ALTERA Corporation. Stratix V device handbook[R/OL]. Altera Corporation, 2015. (2015-01-23)[2018-01-24]. https:// www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/stratix-v/stratix5_handbook.pdf.
- [3] 王楠,黄志洪,杨海钢,等. 一种支持高效加法的 FPGA 嵌入式 DSP IP 设计[J]. 太赫兹科学与电子信息学报, 2017, 15(5):867-873. (WANG Nan,HUANG Zhihong,YANG Haigang,et al. A design of FPGA embedded DSP IP core supporting efficient addition[J]. Journal of Terahertz Science and Electronic Information Technology, 2017,15(5):867-873.)

- [4] 余洪敏,陈陵都,刘忠立. FPGA 中专用可重构乘法器的设计[J]. 半导体学报, 2008,29(11):2218-2225. (YU Hongmin, CHEN Lingdu,LIU Zhongli. Design of a dedicated reconfigurable multiplier in an FPGA[J]. Journal of Semiconductors, 2008,29(11):2218-2225.)
- [5] PFAENDER Oliver A, PFLEIDERER Hans-Joerg. EMMA-a suggestion for an embedded multi-precision multiplier array for FPGAs[C]// International Conference on Field Programmable and Logic Applications. Heidelberg, Germany: IEEE, 2008:434-437.
- [6] CHONG Yee Jern, PARAMESWARAN Sri. Configurable multimode embedded floating-point units for FPGAs[J]. IEEE Transactions on Very Large Scale Integration(VLSI) Systems, 2011,19(11):2033-2044.
- [7] 高亚军. 基于 FPGA 的数字信号处理[M]. 2版. 北京:电子工业出版社, 2015. (GAO Yajun. The digital signal processing based on FPGA[M]. 2nd ed. Beijing:Publishing House of Electronics Industry, 2015.)
- [8] IEEE Computer Society. IEEE standard for floating-point arithmetic: IEEE 754-2008[S]. New York, USA: IEEE, 2008.
- [9] 王源,贾嵩,崔小欣,等. 超大规模集成电路分析与设计[M]. 北京:北京大学出版社, 2014. (WANG Yuan, JIA Song, CUI Xiaoxin, et al. Analysis and design of VLSI circuits [M]. Beijing: Peking University Press, 2014.)
- [10] 李楠. 快速乘法器的设计[D]. 哈尔滨:哈尔滨工业大学, 2007. (LI Nan. Design of high speed multiplier[D]. Harbin, China:Harbin Institute of Technology, 2007.)
- [11] 石敏,王耿,易清明. 基于改进的 Booth 编码和 Wallace 树的乘法器优化设计[J]. 计算机应用与软件, 2016,33(5):13-16. (SHI Min,WANG Geng,YI Qingming. An optimized design of multiplier based on improved Booth encoding and Wallace tree[J]. Computer Applications and Software, 2016,33(5):13-16.)
- [12] OKLOBDZIJA Vojin G. An algorithmic and novel design of a leading zero detector circuit: comparison with logic synthesis[J]. IEEE Transactions on Very Large Scale Integration Systems, 1994,2(1):124-128.
- [13] BRUGUERA Javier D, LANG Tomas . Leading-one prediction with concurrent position correction[J]. IEEE Transactions on Very Large Scale Integration Systems, 1999,48(10):1083-1097.
- [14] KORNERUP Peter. Correcting the normalization shift of redundant binary representations[J]. IEEE Transactions on Computers, 2009,58(10):1435-1439.
- [15] SOHN Jongwook, SWARTZLANDER Earl E. Improved architectures for a fused floating point add-subtract unit[J]. IEEE Transactions on Computers, 2012,59(10):2285-2291.

作者简介:



赵 赫(1993-),男,山东省青岛市人,在 读硕士研究生,主要研究方向为超大规模集成 电路设计.email:zhaohe93@126.com.

杨海钢(1960-),男,上海市人,研究员/教授,博士生导师,主要研究方向为超大规模集成电路设计、数模混合信号集成电路设计等.

黄志洪(1984-),男,福建省莆田市人, 博士,助理研究员,主要研究方向为可编程芯 片设计、人工智能芯片设计.

余 乐(1983-),男,湖北省荆州市人,博士,讲师,主要研究方向为智能硬件设计.

许仕龙(1978-),男,河北省青龙市人, 硕士,高级工程师,主要研究方向为超大规模 集成电路设计与射频电路设计.

郝亚男(1983-),男,石家庄市人,博 士,工程师,主要研究方向为大规模数字电路 设计.