高速整数开方电路的流水线设计

朱维乐[1]，钱贵锁[1]，杨 刚[2]，陈 伟[1]

(1. 电子科技大学电子工程学院 成都 610054; 2. 上海贝尔阿尔卡特有限公司 上海 浦东新区 201206)

【摘要】 对一个位宽为32位整数的开方硬件电路的结构进行设计，介绍了应用流水线技术设计了一个高速求平方根电路，考虑FPGA的内部结构，对采用流水线技术之后占用的硬件资源进行了分析。提出了利用流水线实现开方问题的新算法，在一个时钟周期内对32位整数进行处理，计算出相应的平方根和余数并送出，在算法上具有精度高、速度快、易实现等优点。与传统的算法相比，它完全避免了除法的迭代，从而开方速度提高了一倍左右。

关 键 词 进位保存加法器； 现场可编程门阵列； 流水线结构； 平方根

中图分类号 TN47 **文献标识码** A

A Pipeline Architecture for High Speed Square Root

ZHU Wei-le[1], QIAN Gui-suo[1], YANG Gang[2], and Chen Wei[1]

(1. School of Electronic Engineering, University of Electronic Science and Technology of China Chengdu 610054 ;
2. Alcatel Shanghai Bell Co. Ltd. Pudong Shanghai 201206)

Abstract The technique about how to use pipeline architecture to design high speed square root hardware is illustrated through the process of designing a square root circuit of 32 bits integer. By taking into account of the capacity of FPGA, the resources consumed by the square root hardware is analyzed. The new method to solve the extraction of a root is presented , which can deal with the 32 bits sampled data within a clock period. This method is of high precision, fast speed, easily realization. Compared with the conventional one, the division operations are avoided completely in the new algorithm. Thus, the speed of radication has increased by one time.

Key words carry save adder; FPGA; pipeline architecture; square root

随着大规模集成电路和数字通信的高速发展，越来越多的数字信号处理要求开平方等运算的硬件实现，如较为常用的全周傅氏算法、均方根算法、快速傅里叶算法(FFT)。而目前实现开平方的算法主要有Newton-Raphson算法[1-2]、基2-SRT算法[3-4]以及逐位循环算法[4-6]。

Newton-Raphson算法不仅需要初始化查找表，而且在每次迭代过程中需要乘法器、加法器和减法器。为了加速乘法器，该算法常采用并行Wallace树结构的乘法器和进行传递加法器(CPA)。由于乘法器需要耗费大量的逻辑单元，所以该算法不适合全流水线操作，并且很难恢复余数。基2-SRT算法是一种经典的开方算法，但是每一次迭代都需要相同的硬件资源开销，不能连续输入被开方数，计算的结果还要用CPA硬件电路实现其转化为2的补码，因此限制了它的应用。逐位循环算法每一次循环移位经过相加或相减操作，判断其和或差的符号，得出最终的平方根。

本文介绍了一种基于逐位循环算法的高速整数开方电路的流水线设计，全用加法器实现，无需判断，直接由加法器的进位产生所需要的平方根，加速了硬件电路。

1 整数开方硬件实现的算法介绍

假设32 bits的被开方数 D，其平方根为16 bits；$Q=Q_1Q_2Q_3\cdots Q_{15}Q_{16}$。不恢复余数的逐位循环开方算法[4]运算：

For (k =0; k<16; k = k+1),

If $r_k \geqslant 0$ $r_{k+1}=4r_k-(4q_k+1)$;

Else $r_{k+1}=4r_k+(4q_k+3)$,

If $r_{k+1}\geqslant 0$ $q_{k+1}=2q_k+1$;

Else $q_{k+1}=2q_k$。

收稿日期：2007－04－28；修回日期：2007－10－05

作者简介：朱维乐(1940－)，男，教授，博士生导师，主要从事数字视频、图像处理与HDTV等方面的研究；钱贵锁(1981－)，男，硕士，主要从事数字视频与HDTV方面的研究；杨 刚(1974－)，男，高级工程师，主要从事通信系统研发测试方法方面的研究；陈 伟(1978－)，男，博士生，主要从事数字视频、图像处理与HDTV等方面的研究.

$q_k = Q_1 Q_2 \Lambda Q_k$； $r_0 = D \times 2^{-32}; q_0 = 0$。

因为当 $r_k \geqslant 0$ 时 Q_k=1，所以 $r_k \geqslant 0$ 和 Q_k=1是等价的。另外由于当 Q_k=1时 $q_k = 2q_{k-1} + 1$，当 Q_k=0时，$q_k = 2q_{k-1}$，可得：

If Q_k=1 $r_{k+1} = 4r_k - (4q_{k-1} + 5)$,

Else $r_{k+1} = 4r_k + (4q_{k-1} + 3)$。

再将两个二进制的数相减改成加上一个负数的补码，可得到以下的递推算法：

1) $r_0 = D \times 2^{-32}$, $q_0 = 0$。

2) always $r_1 = r_0 + (-1)$,

If $r_1 \geqslant 0$ $q_1 = Q_1 = 1$;

Else $q_1 = Q_1 = 0$。

3) For (k =0; k<16; $k = k$ +1)。

If $r_k \geqslant 0$ $r_{k+1} = 4r_k + (\overline{8q_{k-1}} + 3)$,

Else $r_{k+1} = 4r_k + (8q_k + 3)$;

If $r_{k+1} \geqslant 0$ $q_{k+1} = 2q_k + 1$,

Else $q_{k+1} = 2q_k$。

4) If $r_{16} < 0$ $r_{16} = r_{16} + (2q_{16} + 1)$。

最后得出平方根是 q_{16} 和余数 r_{16}。

2 整数开方硬件实现的流水线设计

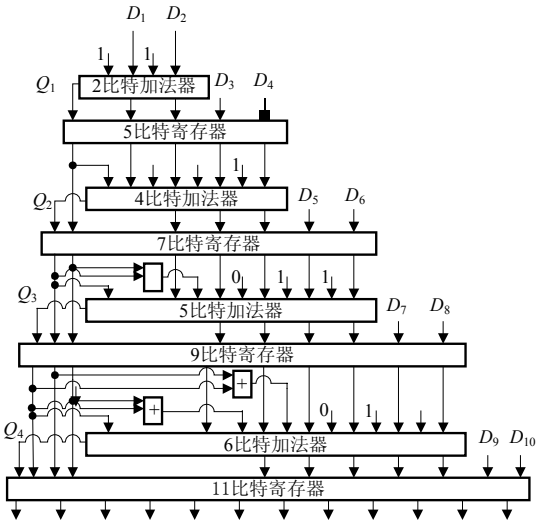

图1 并行开方电路的流水线结构

流水线设计的核心思想是把一个周期内执行的逻辑操作分成几步较小的操作，并在多个高速的时钟内完成。每一次逻辑小操作的结果都存储在寄存器中，被高速时钟同步提供给下一个流水线单元使用。大多数FPGA器件的每个逻辑单元都有寄存器，因此，FPGA适合使用流水线技术。本文设计了位宽为32位整数的开方硬件电路，其流水线结构如图1所示。该流水线结构由全加法器、异或门逻辑、寄存器组成，需要单比特全加器165个，寄存器285个。流水线结构的主径(Critical Path)由16级单比特全加器、寄存器以及异或逻辑门组成，因此流水线结构的工作速度主要取决于单比特全加器、寄存器以及异或逻辑门的工作速度。单比特全加器逻辑结构简单，寄存器和异或逻辑为最基本的时序逻辑器件，三者都能高速地工作。因此，与普通的并行流水线结构相比，这种开平方的硬件电路结构更适合于高速运算，而且硬件的复杂度得到大幅度的缩减。

3 开平方电路进位保存加法器结构

并行开方硬件电路的流水线结构主要耗费大量的加法器，传统的加法器一次只能把两个数相加。在高速开方电路中，需要同时把多个操作数以及前一级的进位相加，为了实现一次能够把多个数相加，需要采用CSA进位保存加法器。

无论是在电路面积还是时序特性上，CSA加法器比传统的Ripple-Carry加法器或是Carry Look-ahead加法器都好很多[6]。当相加的数据越多[7]，CSA加法器的优势越明显。现有的EDA综合工具中已经添加该CSA加法器，如Synopsys公司的新版DC工具。

由CSA加法器实现开方的硬件电路如图2所示。该CSA流水线结构采用了CSA加法器，与采用全加器实现的流水线结构相比，一方面能缩小综合的电路面积，另一方面能提高时钟速度。传统的二操作数加法器构成的全加器，其Critical Path均是在进位的路径上。但CSA加法器则不同，它保留了原有的加法器结构，但是其上一级的进位直接经过移位进入下一级的三操作数相加，从而使Critical Path的延时得到改善，提高了电路的工作速度。

4 仿真结果和性能分析

本文使用ModelSim工具进行功能仿真和时序仿真，得到的时序仿真波形如图3所示。输入为无符号的32位随机整型数，由于采用了流水线设计，首次延时(latency)为16个时钟周期。经过了16时钟输出第一个被开方数的16 bits的平方根和一个17 bits准确的无符号整型余数，随后输入每个被开方数，只需一个时钟周期来获取其平方根和余数。经过Altera公司QuartusII 5.0的硬件资源耗费估计，该设计需要603个逻辑单元，系统的最高工作频率达到了232.72 MHz，数据吞吐量也达到232 MHz。与文献[8]相比，该设计的最高工作频率提高了至少30%，处理的数据位宽增加了一倍，而且能得到准确的余数。

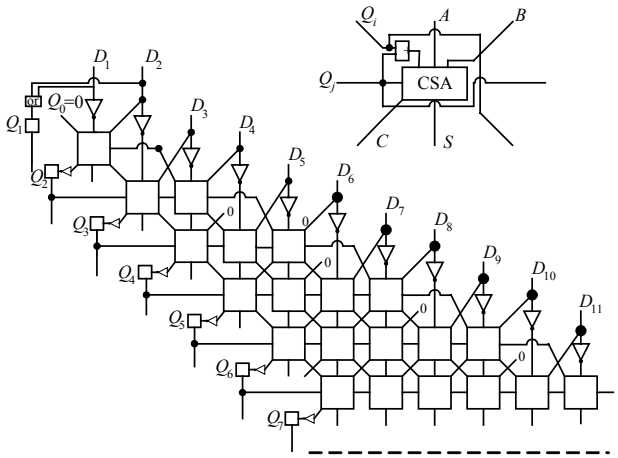

图2　并行开方电路的CSA流水线结构

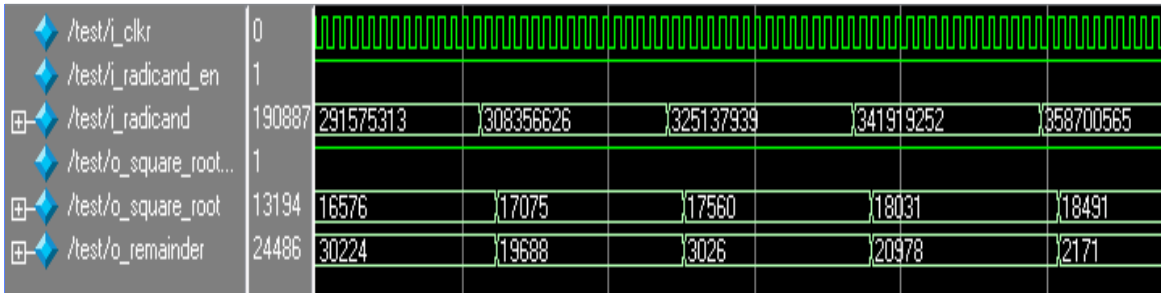

图3　系统仿真波形

5　结　论

本文介绍了一种基于逐位循环递推算法的32位位宽的整数开平方电路的流水线设计，采用了CSA加法器改进了流水线的加法器。其优点是缩小综合的电路面积，并提高整个电路的工作速度。整个流水线设计电路在Altera公司的FPGA器件经过验证，得到了比较好的运算结果。

参　考　文　献

[1] KABUO H, TANIGUCHI T, MIYOSHI A, et al. Accurate rounding scheme for the newton-raphson method using binary representation[J]. IEEE Transaction on Computers, 1994, 3(1): 43-51.

[2] HENNESSY J, PATTERSON D. Computer architecture, a quantitative approach[M]. Second Edition. San Francisco, USA: Morgan Kaufmann Publishers, 1996.

[3] 梁　政, 邵志标, 沈绪榜. 用进位存储加法器快速实现串行乘法器和平方根计算[J]. 西安交通大学学报, 2002, 36(4): 406-409.

[4] LANG T, MONTUSCHI P. Very-high radix square root with prescaling and rounding and a combined division/square root unit[J]. IEEE Trans Computers, 1999, 48(8): 827-841.

[5] WANG Xiao-jun, NELSON B E. Trade offs of designing floating-point division and square root on virtex FPGAS[C]//Proc 11th Ann IEEE Symp FCCM'03. Washington, D. C.: IEEE Computer Society Press, 2003.

[6] ORTIZ I, JIMENEZ M. Scalable pipeline insertion in floating point units for FPGA synthesis[C]//Proceedings of the IASTED International Conference on Circuit, Signals and Systems. Banff, Albert, Canada: IASTED/Acta Press, 2003.

[7] KIM T, JAO W, TJIANG S. Circuit optimization using carry-save-adder cells[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuit and Systems, 1998, CAD-17,10: 974-984.

[8] BANNUR J, VNRMA A. The VLSI implementation of a square root algorithm[C]//Pro IEEE Symposium on Computer Arithmetic. California, USA: IEEE Computer Society Press, 1985: 159-165.

[9] 简弘伦. 精通Verilog HDL: IC设计核心技术实例详解[M]. 北京: 电子工业出版社, 2005.

编　辑　黄　莘