

一种新颖的向量基-8 布斯乘加器的设计

张 喆,张盛兵,黄小平

(西北工业大学 航空微电子中心,陕西 西安 710065)

摘 要: 提出一种新颖的部分冗余向量基-8 布斯编码乘加器,快速地处理了部分积中的正负三倍被乘数项,并以亚字并行的方式支持单指令多数据流操作,可以完成 16×16 的有符号乘法运算以及两个 8×8 的有符号乘法运算.使用了 4:2 压缩器进行部分积的压缩,并使用进位控制逻辑来控制短向量间的进位.

关键词: 向量基-8 布斯;亚字并行;单指令多数据流;4:2 压缩器

中图分类号: TP303

文献标识码: A

文章编号: 1000-7180(2010)11-0020-05

A Novel Design of Vector Radix-8 Multiply-Accumulator

ZHANG Zhe,ZHANG Sheng-bing,HUANG Xiao-pin

(Aviation Microelectronics Center, Northwestern Polytechnical University, Xi'an 710065, China)

Abstract: A novel partially redundant Radix-8 booth-coding Multiply-Accumulator, it support SIMD-operation by the mode of sub-word parallelism, also could complete 16×16 Signed-Multiply or two 8×8 Signed-Multiply. 4:2 compressor is used in Wallace-Reduce Tree and carry-control logic is used to control the carry signal between short vectors.

Key words: Vector Radix-8 booth;sub-word parallelism;SIMD;4:2 compressor

1 引言

数字信号处理已广泛应用于各个领域,随着人们对集成电路设计中 SoC 思想的进一步理解,在通用处理器内部集成数字信号处理能力,使通用处理器具有多媒体处理能力,代替专用的数字信号处理系统已经成为一种趋势.国际上大公司为了使自己的通用处理器芯片具有多媒体处理技术,纷纷对其通用处理器进行多媒体处理能力的扩展^[1].如 Intel 采用 MMX、SSE、SSE2 技术,AMD 采用 3DNow! 技术,Motorola 公司采用 AltiVec 技术等,这些扩展实现了通用处理器对数字信号的快速处理能力.

多媒体技术最重要的一个特色就是基于短向量的并行运算.乘加器是微处理器中最重要的基本运算部件之一,优化乘加器将大大优化整个处理器的性能,所以向量乘加器性能直接影响到多媒体扩展技术的性能.

布斯编码是通过减少部分积的生成来提高速度,现有乘加器一般采用的是编码逻辑和选择逻辑都较简单的基为 4 的布斯编码,基-8 的布斯编码能进一步减少部分积的生成,压缩部分积阵列,减少运算延时,且综合后的电路面积小于基-4 的布斯编码,更适合 SoC 的集成化设计.本设计将向量与布斯编码结合起来,设计基-8 布斯编码向量乘加器.

2 传统的乘加器结构及布斯编码

2.1 乘加器的体系结构

乘加器由部分积生成模块(Partial Product Generation)、Wallace 压缩树模块(Wallace Reduction Tree)和进位传播加法器模块(Carry-propagate adder)三部分组成,见图 1.

2.2 传统的基-8 布斯编码

布斯编码的初衷是加速串行乘法^[2].对于一个用二进制补码表示的数 A , $A = a_{n-1}a_{n-2}a_{n-3}\cdots a_1$

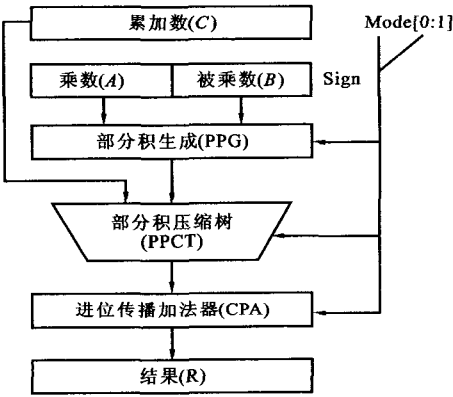


图1 乘法器的体系结构图

a_0 ,基-8的布斯编码如下:

$$\begin{aligned} A &= a_{n-1}a_{n-2}a_{n-3}\cdots a_1a_0 \\ &= -a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \cdots + \\ &\quad a_1 \cdot 2^1 + a_0 \cdot 2^0 \\ &= (-a_{n-1} + a_{n-2}) \cdot 2^{n-1} + (-a_{n-2} + \\ &\quad a_{n-3}) \cdot 2^{n-2} + \cdots + \\ &\quad (-a_1 + a_0) \cdot 2^1 + (-a_0 + a_{-1}) \cdot 2^0 \\ &= \cdots \\ &= (-4a_{n-1} + 2a_{n-2} + a_{n-3} + a_{n-4}) \cdot 2^{n-3} + \\ &\quad (-4a_{n-4} + 2a_{n-5} + a_{n-6} + a_{n-7}) \cdot \\ &\quad 2^{n-6} + \cdots + (-4a_2 + 2a_1 + a_0 + a_{-1}) \cdot 2^0 \end{aligned}$$

它的真值表如表1所示.

表1 基-8布斯真值表

a_{i+2}	a_{i+1}	a_i	a_{i-1}	部分积	M_i
0	0	0	0	0	0
0	0	0	1	Y	0
0	0	1	0	Y	0
0	0	1	1	2Y	0
0	1	0	0	2Y	0
0	1	0	1	3Y	0
0	1	1	0	3Y	0
0	1	1	1	4Y	0
1	0	0	0	-4Y	1
1	0	0	1	-3Y	1
1	0	1	0	-3Y	1
1	0	1	1	-2Y	1
1	1	0	0	-2Y	1
1	1	0	1	-Y	1
1	1	1	0	-Y	1
1	1	1	1	-0	1

2.3 传统基-8布斯编码的缺点

传统的基-8布斯编码在处理 $\pm 3Y$ 时存在一些

问题:首先,如果用CPA简单地完成 $Y + 2Y$,需要很长的进位传播延时,这在单拍完成部分积生成的流水中是不能容忍的;其次,如果采用CLA块进行 $Y + 2Y$ 的求和,见图2.

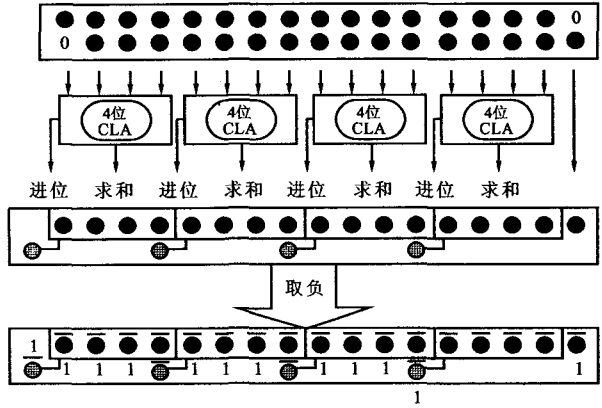


图2 一般的三倍部分积的产生过程

虽然缩短了进位传播延时,但是在求负三倍被乘数的时候,按位取反加1之后使得负三倍被乘数的形式与正三倍被乘数有差异,而且需要再求和一次,所以迫切地需要同一种形式来表示 $+3Y$ 和 $-3Y$;最后,还需要这种形式能够十分简便地完成 $+3Y$ 和 $-3Y$ 之间的相互转化.

3 向量化的基-8布斯编码及部分积的生成

3.1 结合向量处理的布斯编码

首先给出向量的部分积生成框图^[3],见图3.

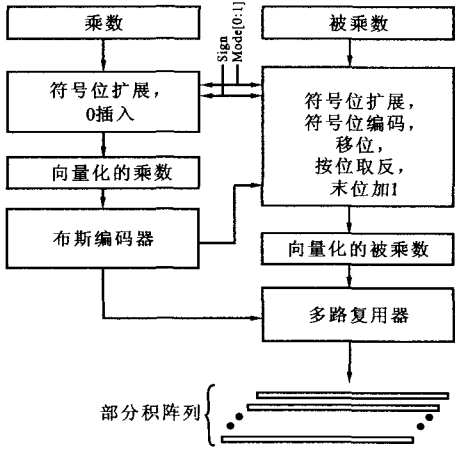


图3 向量部分积生成框图

可以看到,向量的部分积生成首先需要在符号控制信号和模式控制信号的共同作用下对乘数与被乘数进行修改,使之成为向量化的乘数与被乘数.乘数的向量化需要进行符号位扩展和0插入,被乘数的向量化比较复杂,需要进行符号位扩展,符号位编

码,按位取反操作,移位操作,末位加 1 操作.然后对向量化的乘数进行布斯编码,得到不同的选择信号,这些信号再对布斯多路复用器中的向量化被乘数进行选择,最后得到各个部分积.对于向量运算的不同模式来说,每一种模式有相对应的部分积阵列^[4],见图 4.

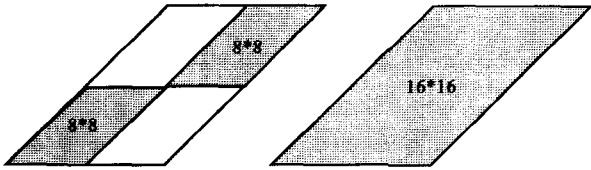


图 4 16 * 16 乘法的 8 位模式和 16 位模式

3.2 对乘数的向量基 -8 布斯编码

向量基 -8 布斯编码需要对被乘数进行符号位扩展,符号位编码,移位,按位取反,末位加 1,每项加 K 以及求和等操作.向量乘加器对乘数的布斯编码需要根据模式的改变来做相应的调整,见图 5.

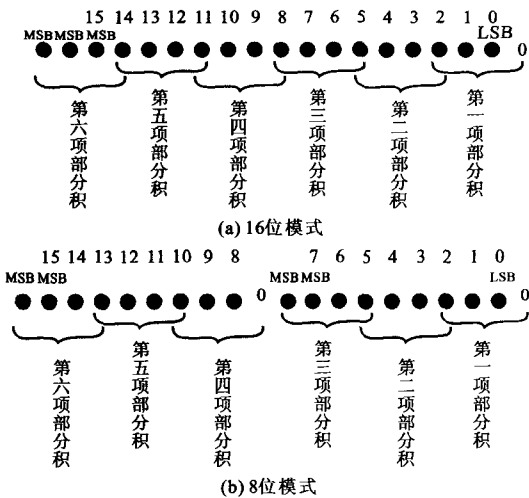


图 5 不同模式下对乘数的基 -8 布斯编码

在标量的部分积生成中,假设乘数为 A ,需要 $A[-1] = 0$.在向量的布斯编码中,不仅仅需要让 $A[-1] = 0$,而且在每个子小组的最低位之后要补 0,如第四个部分积的布斯编码需要公布: $\{A[10], A[9], A[8], 0\}$,而不是标量布斯编码中使用的 $\{A[10], A[9], A[8], A[7]\}$.

为了使无符号数能够正确运算,需要进行符号位的扩展,即每一个子小组的乘数必须向上扩展.为了形成 4 位的子小组(图 5),扩展的位数从 1 位到 3 位不等,这是根据乘数的位宽和最低子小组的起始位置来决定的,见图 6.这个符号位扩展 S 是由符号位控制信号($sign$)和每个子小组的最高有效位(Most Significant Bit)决定的,公式如下: $S =$

$MSB \& sign$.

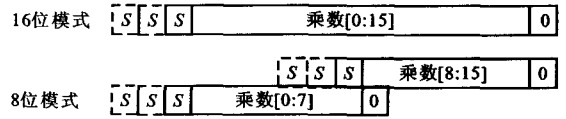


图 6 不同模式下的符号位扩展

3.3 向量基 -8 布斯编码的部分积

与低基的布斯编码不同的是基 -8 的布斯编码需要算出正负三倍的被乘数部分积,所以需要有一个求和的操作,而且为了使正负 3 倍部分积便于处理,又需要作一些修改.对于 16 位乘法, $3Y$ 由 Y 和 $2Y$ 相加得出,用一系列短长度的加法器来完成它们之间的相加,这里同样使用长度为 4 的加法器,如图 2 所示.

但是如果出现 $-3Y$ 的情况下,不仅仅需要按位取反,而且还需要将空隙处的 0 位都转变成 1,如图 2 所示,这就使 $+3Y$ 和 $-3Y$ 的形式变得不相同,求 $-3Y$ 的延时比 $+3Y$ 大很多,而且这样还加大了硬件的复杂性.为了使正数和负数的形式保持一致,这里将每一个部分积都加上一个常数 K ,对于 16 位的乘法来说,这个常数取 $0000_1000_1000_1000_00$,这个 K 在部分积为正或者为负的时候都一样.在修改后的部分积阵列中使用的部分积就是 M ($M = \text{原部分积} + K$),而又存在这样一个常数 Z ,使得 $M + Z = 2K$,见图 7.

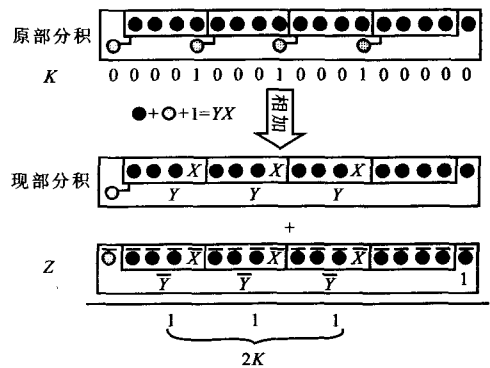


图 7 本文采用的 3 倍部分积的生成过程

所以,可以知道 $Z = 2K - M = K - \text{原部分积}$,并且 M 和 Z 的形式是完全一样的,只是 Z 在最低位加了一个 1,这就完全解决了上面提出来的问题,既缩短了很长的进位传播延时链,又使得 $+3Y$ 和 $-3Y$ 的形式相同并且容易进行相互转化.最后,在部分积阵列加上一个常数,使得所加的这些 K 能够顺利地溢出,从而保持运算结果的正确性.要注意的是对于 8 位和 16 位的乘法,常数 K 是不一样的,如果是做 8 位的乘法, K 取 0000_1000_00 .

而为了使运算结果正确,所有增加的 K 必须溢出,所以需要增加一项额外的部分积 EX(见图 8). 对于一个 16 位的乘法,额外的部分积是 EX = 1111_1101_1001_0000_0000_0100_1110_0000;而如果是同时做两个 8 位的乘法,额外的部分积是 EX = 1111_0110_1110_0000_1111_0110_1110_0000.

对于符号位的扩展,既可能全部为 1,也可能全

部为 0,这部分常量可以通过预先计算它们的和,优化部分积的阵列^[5]. 由图 8 可知,如果部分积为负,需要在最后一位加 1. 有符号的补码乘法与无符号的补码乘法相似,但是符号位的扩展是根据部分积的符号,而不是简单地根据表 1 中的 M_i 来进行的,因为被乘数也可能是负的. 拿两个 8 位的有符号乘法来举例,优化后的部分积阵列如图 8 所示.

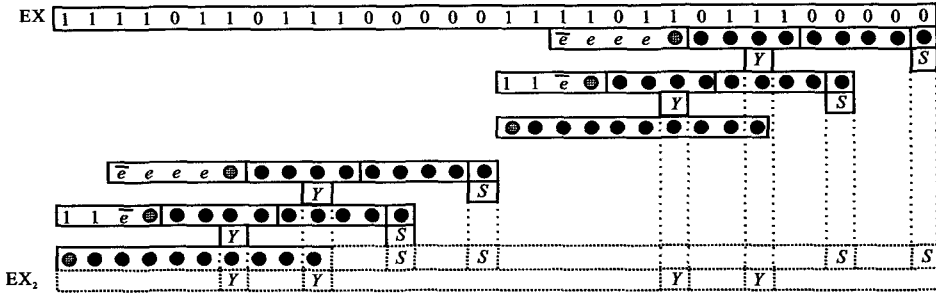


图 8 优化后的部分积阵列

图中, $e = y_i \cdot M_i$, M_i 见表 1, y_i 是被乘数的符号位.

如果每一个部分积都与下面的附加值相加再进入华莱士树的话,需要在进入部分积压缩之前就要进行一次相加,对于 16 位乘法来说,每个部分积增加了一个 16 位的 CPA 延时,对于 8 位乘法来说,每个部分积增加了一个 8 位的 CPA 延时. 在高速的乘加器的设计中,这种延时需要被削减. 本文的方法是再额外增加一项部分积 EX2, 这项部分积由每项部分积的附加值组成(见图 8), 这样就不需要对每项部分积都进行相加处理,只需要在部分积压缩阵列里增加一项 EX2 即可实现. 由图 8 也可以看出,每一项负的部分积需要在最低位进行加 1 处理,由于有的部分积的最低位与上面某些部分积的附加值处于同一列中,因此把最低位处理项也放入 EX2 中不能实现,为了不继续额外的增加部分积,可以把每一项的最低位处理与乘数最高几位编码出的最后一项部分积合并成一项(见图 8). 而最后一项部分积就是所需要累加的数,将它也放入部分积阵列,最后通过部分积压缩单元得出结果.

4 部分积的压缩

部分积压缩单元式乘加器设计的重点,而且也是延时最长的部分. 树型结构通过并行求和方式将求和次数从 $O(N)$ 降为 $O(\log N)$ ^[5]. 而 4:2 压缩器,也就是(5:3)计数器,最大的优点就是它的规则性和它的可布线性,并且它的速度快,每个 4:2 压缩器只有 3 个异或门加 1 级与门的延时^[6]. 所以采用

基于 4:2 压缩器的树型部分积压缩结构.

在不同的模式中,必须考虑短向量单元的边界的进位问题. 传统的办法是分别运算出各种模式下最终的部分积压缩结果,然后给出控制信号用复用器进行选择. 这样的做法,硬件电路器件繁多,面积大. 本文采用的方法是在 4:2 压缩器的短向量单元边界插入一个进位消除信号 nocarry 信号. 这种方法需要在相应的 4:2 压缩器中插入一个 2 输入的与门来控制是否前面的进位被允许进入. 修改后的 4:2 压缩器的表达式如下:

$$S = (a \wedge b) \wedge (c_{in} \cdot \sim \text{nocarry})$$

$$C_{out} = (a \cdot b) + [(a + b) \cdot (c_{in} \cdot \sim \text{nocarry})]$$

很明显,额外增加的 nocarry 信号的 2 输入与门是与其他门逻辑平行的,所以它并不增加关键路径的延时.

5 CPA 模块

最后的 sum 项和 carry 项通过一个特殊的进位传播加法器来完成^[7]. 这个加法器由 4 位的超前进位加法器块组成,并且也需要在短向量的边界加一个 nocarry 信号,用来控制进位传播. 与普通的超前进位加法器稍有不同,它的公式如下:

$$C_{out_0} = g_0 + C_{in} \cdot p_0 \cdot (\sim \text{nocarry})$$

$$C_{out_1} = g_1 + g_0 \cdot p_1 + C_{in} \cdot p_0 \cdot p_1 \cdot (\sim \text{nocarry})$$

$$C_{out_2} = g_2 + g_1 \cdot p_2 + g_0 \cdot p_1 \cdot p_2 + C_{in} \cdot$$

$$p_0 \cdot p_1 \cdot p_2 \cdot (\sim \text{nocarry})$$

因为组间的进位传播信号比组间进位生成信号

生成得早,所以关键路径来自 C_{in} 和组间进位生成 g . 在布局布线中尽量应该将 C_{in} 和 g 信号放到与输出靠近的地方来减少路径延时. 很明显, nocarry 信号并不增加关键路径延时,因此采用这种办法设计最后的加法器.

6 结束语

本文设计的乘加器,采用了一种新颖的基-8 布斯编码,与基-4 的布斯编码相比,生成的部分积个数又有所减少,进一步压缩了部分积阵列. 由表 2 可知,基-8 布斯编码所生成的部分积较基-4 布斯编码少 $\frac{n}{6}$ 个数量级,当 n 越大时,基-8 布斯编码压缩部分积阵列的优势就越大. 所以该向量乘加器十分适合大位宽的多媒体向量运算.

表 2 编码

非布斯编码	基-4 布斯编码	基-8 布斯编码
n	$\frac{n}{6}$	$\frac{n}{3}$

在延时上,该基-8 布斯编码的向量乘加器在处理正负三倍被乘数的部分积时不需要进行传统的长进位链的进位传播加法,只需要一个 4 位的 CPA 延时,以及增加一项 EX2 部分积,较之传统的延时大大减小. 而且还引入进位消除(nocarry)信号,使得关键路径上减少了一个复用器,进一步降低了延时.

在面积上,基-8 布斯编码的向量乘加器比基-4 的更小,更适合苛刻条件下的高性能乘加运算,如航

天领域的处理器.

参考文献:

[1] 黄小平. 基于 Altivec 技术的向量处理单元的结构研究[D]. 西安:西北工业大学,2006:26-33.

[2] Neil H E Weste, David Harris. COMS 超大规模集成电路设计[D]. 3 版. 北京:中国电力出版社,2006:579-590.

[3] Dimitri Tan, Albert Danysh, Michael Liebelt. Multiple-precision fixed-point vector multiply-accumulator using shared segmentation[C]// IEEE Symposium on Computer Arithmetic. Spain, 2003.

[4] Shankar Krithivasan, Michael J Schulte. Multiplier architectures for media processing[C]// Proceeding of the 37th Asilomar Conference on Signals, Systems and Computer. Asilomar, USA. 2003:2193-2197.

[5] Richard P Brent, Paul Zimmermann. Modern research in computer arithmetic[M]. Cambridge University Press, 1997:34-57.

[6] 周旭. 低功耗浮点乘法部件的研究与设计[D]. 北京:中国科学院计算技术研究所,2005:17-42.

[7] 蒋小龙. 定点加法器设计[EB/OL]. [2002-09-14]. <http://www.pld.com.cn/ddvance/MuLD.Pdf>.

作者简介:

张喆 男,(1984-),硕士研究生. 研究方向为计算机系统结构、专用集成电路设计.

张盛兵 男,(1968-),博士生导师. 研究方向为 VLSI 系统设计、计算机系统结构、电子设计自动化.

黄小平 男,(1979-),博士研究生. 研究方向为计算机系统结构、专用集成电路设计.

(上接第 19 页)

作者简介:

张亚歌 男,(1984-),硕士研究生. 研究方向为语音信号处

理、语音识别.

张太镒 男,(1944-),教授,博士生导师. 研究方向为新一代移动通信技术、软件无线电、数字音视频技术.