

# 高效实时的抽取滤波器的FPGA实现

陈俊, 汪学刚

(电子科技大学电子工程学院 成都 610054)

【摘要】介绍了分布式算法实现乘积和运算的原理及工程上的实现方式,提出了适合现场可编程门阵列器件基本查找表结构的改进分布式算法,通过计算机仿真,在具体器件上实现了抽取滤波器。实验结果表明:该分布式算法结构可以充分利用现场可编程门阵列器件的资源,并且只引入固定的流水线延迟,具有很好的高效性和实时性。

关键词 分布式算法; 现场可编程门阵列; 查找表; 抽取

中图分类号 TN957.51 文献标识码 A

## High Efficient and Real-Time Realization of Decimation Filter Based on FPGA

CHEN Jun, WANG Xue-gang

(School of Electronic Engineering, UEST of China Chengdu 610054)

**Abstract** This paper briefly talks about the principle of distributed arithmetic algorithm and its application to the multiply-accumulate before puts forward a improved distributed arithmetic algorithm which suits the look up table structure of filed programmable gate array well. Simulated and tested results on real device shows that high efficient and real-time decimation filter with some invariable pipeline delay can be achieved through this improved algorithm. The fabricated decimation filters are used in real digital receiver.

**Key words** distributed arithmetic algorithm; filed programmable gate array; look up table; decimation filter

抽取滤波器是在数字接收机等诸数字系统中广泛应用的一种滤波器,由于其具有输入和输出数据率满足一定倍数关系的特点,因此在实现这一类滤波器时,设计人员希望找到一种高效、实时的实现方法。

本文针对现场可编程门阵列(Field Programmable Gate Array, FPGA)结构的特点,对分布式算法的结构进行了改进,并利用改进后的结构在FPGA上高效、实时地实现了抽取滤波器。

### 1 分布式算法

DA算法(Distributed Arithmetic algorithm)是一项重要的计算机算法,广泛应用在计算乘积和(Multiply Accumulate, MAC)之中。MAC定义为:

$$y = \sum_{n=0}^{N-1} h[n]x[n] \quad (1)$$

当使用传统的算法实现两个序列的乘积和时,需要用到 $N$ 个MAC循环。如果采用流水线结构,将可以

缩减这一数量,但是也非常有限<sup>[1]</sup>。然而,在数字信号处理(Digital Signal Processing, DSP)应用领域中,系数  $h[n]$  通常一定,乘积项部分就成了简单的“缩放”。有符号数  $x[n]$  可表示为:

$$x[n] = -2^{B-1}x_{B-1}[n] + \sum_{b=0}^{B-2} x_b[n]2^b \quad x_b[n] \in [0,1] \quad (2)$$

式中  $x_b[n]$  为  $x[n]$  的第  $b$  位。由式(1)和式(2),令  $f(h[n], x_b[n]) = h[n]x_b[n]$ , 则有:

$$y = -2^{B-1} \sum_{n=0}^{N-1} f(h[n], x_{B-1}[n]) + \sum_{b=0}^{B-2} 2^b \sum_{n=0}^{N-1} f(h[n], x_b[n]) \quad (3)$$

即只要可以计算出函数  $f(h[n], x_b[n])$ ,  $y$  的计算就可以转化为简单的移位以及带加、减控制的累加。

## 2 查找表 - FPGA的基本逻辑单元<sup>[2]</sup>

目前,全新的FPGA系列芯片越来越受到研究人员的青睐。由于其具有开发时间短、可重复擦写等一系列优点,所以在数字信号处理中得到广泛应用。考虑到工程上对实时性的要求越来越高,因此如何充分利用FPGA的结构特点,进而高效利用FPGA的资源,实时地实现所要求的处理功能,成了研究人员不断追求的目标。典型FPGA的基本逻辑单元由查找表(Look Up Table, LUT)和触发器两个基本部分构成。其中,查找表以4输入1输出最为典型,该结构可以快速查出多达4个输入变量的函数的结果。要提高系统效率,必须充分利用FPGA中的LUT资源。

## 3 基于查找表结构的抽取滤波器

抽取滤波器在信号处理领域的应用十分广泛,关于其原理方面的参考文献非常之多,在此,仅结合具体实现进行讨论。假设抽取滤波器的阶数为  $N$ , 抽取率为  $D$ : 1( $D$ 为整数), 滤波器系数为  $c[n]$ ,  $n=0,1,2,\dots,N-1$ ; 数据为  $x[k]$ ,  $k=0,1,2,\dots$ 。则输出为:

$$y[m] = \sum_{t=0}^{N-1} x[mD+t]c[N-1-t] \quad m=0,1,2,\dots \quad (4)$$

设每个  $x[k]$  均以并行方式流入,且数据率为  $f_s=1/T_s$ 。对于式(4),基本实现方式有两种:(1) 将抽取滤波器看成由无抽取的滤波器和抽取器级联构成,这需要在  $T_s$  时间内完成计算一个滤波结果所需的所有乘法和累加,然后在此基础上抽取有用数据。(2) 由于只需要抽取后的结果,对其余中间数据并不关心,所以只需在  $DT_s$  时间内完成一个输出数据的计算,同时在计算过程中保存好下一个滤波输出所需的输入数据即可。显然,方式(2)结构上比方式(1)略复杂,但可显著节省FPGA资源。构建如下滤波器:阶数为32,抽取因子  $D=8$ ,输入数据  $x[k]$  和滤波器系数  $c[n]$  都是16位、以二进制补码表示的有符号数。以  $y[0]$  的计算为例,设数据和滤波器系数分别为  $x[0], x[1], \dots, x[31]$  和  $c[0], c[1], \dots, c[31]$ 。则:

$$y[0] = -2^{15} \sum_{n=0}^{31} f(c[31-n], x_{15}[n]) + \sum_{b=0}^{14} 2^b \sum_{n=0}^{31} f(c[31-n], x_b[n]) \quad (5)$$

结合FPGA的4输入1输出LUT结构,  $y[0]$  可表示为:

$$\begin{aligned} y[0] = & \sum_{b=0}^{14} 2^b \sum_{q=0}^7 \sum_{p=0}^3 f(c[31-4q-p], x_b[4q+p]) - 2^{15} \sum_{q=0}^7 \sum_{p=0}^3 f(c[31-4q-p], x_{15}[4q+p]) = \\ & 2^8 \sum_{b=0}^6 2^b \sum_{q=0}^7 \sum_{p=0}^3 f(c[31-4q-p], x_{b+8}[4q+p]) + \sum_{b=0}^7 2^b \sum_{q=0}^7 \sum_{p=0}^3 f(c[31-4q-p], x_b[4q+p]) - \\ & 2^{15} \sum_{q=0}^7 \sum_{p=0}^3 f(c[31-4q-p], x_{15}[4q+p]) \end{aligned} \quad (6)$$

为叙述方便,定义“LUT组”: 18个4输入1输出LUT输入端并联构成的4输入18输出查找表结构。LUT组可表示为:

$$\sum_{p=0}^3 f(\text{coef}[31-4q-p], x_b[4q+p]) \quad q=0,1,2,\dots,7; b=0,1,2,\dots,15 \quad (7)$$

由式(6)可知,采用8个LUT组,设为  $LUT_0, LUT_1, LUT_2, LUT_3, LUT_4, LUT_5, LUT_6, LUT_7$ , 并将8个LUT组的输出相加就可实现某一位的滤波。而滤波最后结果  $y[0]$  只是在每一位滤波结果上的移位及累加(最高位对应减)。因数据  $x[n]$  为16位宽度,故可以用串行时钟,将输入数据  $x[0], x[1], \dots, x[31]$  同时、逐位移

入上述一位数据滤波单元, 同时将8个LUT组的输出结果相加并适当左移(实现乘以 $2^b$ )后输入累加器。累加器完成15次累加和1次减法(减去最高位对应的数据), 即可完成32个16位宽度数据的滤波操作。

由于要完成8倍抽取, 所以需将 $x[0], x[1], \dots, x[31]$ 以 $8T_S$ 为周期更新, 滤波过程必须在 $8T_S$ 内完成。而在滤波过程中, 如果数据的每一位以完全串行的逐位移位方式输入到LUT组, 则需要完成16次移位, 显然, 所需移位时钟是数据输入时钟的2倍以上, 这种实现方式以提高时钟为代价来换取较少的资源耗费。此外, 也可以采用和数据输入时钟相同频率的移位时钟, 但需先将一个16位的数据分解成每个数据宽度不大于8的 $l$ 个数据, 但需要多耗费几乎 $(l-1)$ 倍的资源。综合考虑时钟和资源这对矛盾, 可采取如下方式: 把数据拆分为2个8位数据(高8位、低8位)。根据式(6), 易得出滤波器的串行移位寄存器结构原理如下:

$$\begin{aligned} x_7[0], x_6[0], x_5[0], x_4[0], x_3[0], x_2[0], x_1[0], x_0[0] &\rightarrow a[0] \\ x_{15}[0], x_{14}[0], x_{13}[0], x_{12}[0], x_{11}[0], x_{10}[0], x_9[0], x_8[0] &\rightarrow b[0] \\ x_7[1], x_6[1], x_5[1], x_4[1], x_3[1], x_2[1], x_1[1], x_0[1] &\rightarrow a[1] \\ x_{15}[1], x_{14}[1], x_{13}[1], x_{12}[1], x_{11}[1], x_{10}[1], x_9[1], x_8[1] &\rightarrow b[1] \\ x_7[31], x_6[31], x_5[31], x_4[31], x_3[31], x_2[31], x_1[31], x_0[31] &\rightarrow a[31] \\ x_{15}[31], x_{14}[31], x_{13}[31], x_{12}[31], x_{11}[31], x_{10}[31], x_9[31], x_8[31] &\rightarrow b[31] \end{aligned}$$

如上所示的原理中, 每个时钟有效沿移出的64个1位的数据构成查找表的输入, 4位一组。因为最高位是符号位, 所以累加到最高位时要做减法。又因为分为高和低两个部分, 所以需要在高位累加结果与低位累加结果相加时给高位结果乘上一个系数。故得滤波器查找表、求和部分的框图如图2所示。在结构图中, 因为采用流水线结构, 中间输出结果是不需要的, 所以需要严格的时序控制, 以抽取后的数据率输出结果。由于上述实现方法只是具有流水线延迟, 且该流水线延迟只与输入数据精度和抽取率有关, 所以输入数据可以源源不断地进入该滤波结构, 对应的输出结果也将会在流水线延迟之后源源不断地送出。

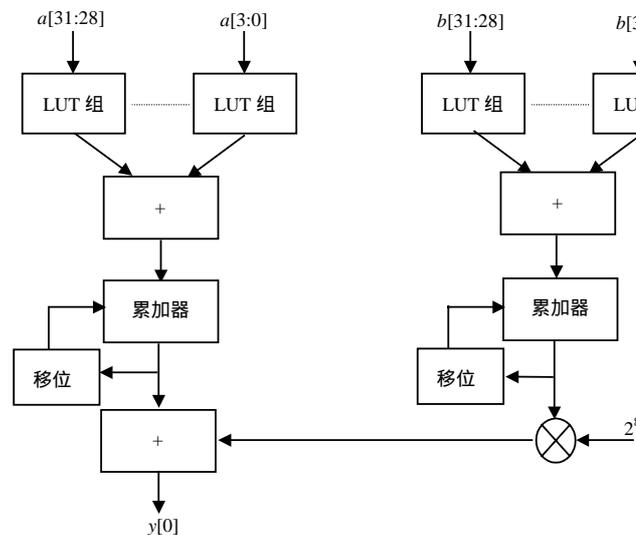


图2 滤波器原理框图(查找表、求和)

## 4 仿真分析

按上述结构, 设计 $D$ 为10、滤波器系数和为“64A92”的20阶抽取滤波器, 采用ALTERA公司提供的开发软件Max+plus II进行时序仿真, 输入数据 $x[n]$ 为14位的二进制数。设 $x[19]=x[18]=\dots=x[0]=5$ , 则输出数据应该为 $64A92 \times 5 = 1F74DA$ 。仿真结果与分析完全吻合, 抽取滤波器仿真结果如图3所示。

将上述提出的算法结构与通常的乘加结构做FPGA资源耗费的对比实验。在ALTERA公司的EP1K30内实现仿真中的抽取滤波器只需要1 580个逻辑单元。而在上述同样器件中以先乘再加的方式实现同样滤波器需要2 450个逻辑单元。所以, 本文提出的算法结构在完成效率上将远远高于传统的乘加结构。

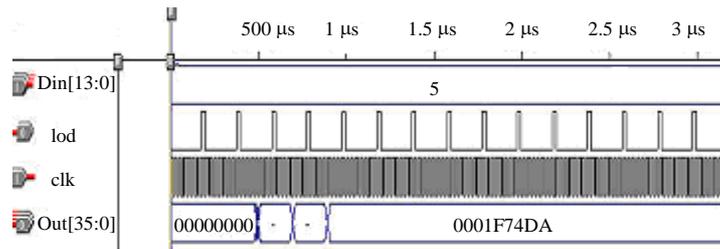


图3 抽取滤波器仿真结果

在实时性方面,由于该抽取滤波结构只用到加法器和移位寄存器,所以处理速度远远快于采用乘法器的结构,并且该结构只是具有长度为15个输入数据时钟周期的流水线延迟,可对连续不断进入系统的数据进行正确处理而不会对系统的正常工作产生任何影响。所以,该结构具有很好的实时处理特性。

## 5 结 论

本文提出了一种充分利用FPGA的基本查找表结构,高效、实时地实现抽取滤波器的改进分布式算法结构。实验证明,该方法完全可行,并且以该方法为核心的数字下变频及处理系统已稳定工作。

### 参 考 文 献

- [1] Uwe M B 著. 数字信号处理的FPGA实现[M]. 刘凌, 胡永生 译. 北京: 清华大学出版社, 2003
- [2] 赵雅兴. FPGA原理、设计与应用[M]. 天津: 天津大学出版社, 1999

编 辑 刘文珍

(上接第738页)

## 4 结 论

多片ADC拼接提高系统采样速率是目前采集技术的研究热点之一,在工程实现中通道不匹配是影响系统性能的关键因素,这也是并行系统中不均衡问题的具体体现。本文讨论了ADC交替采样中出现的时间误差,详细分析了非均匀采样的频谱和误差校正方法,仿真结果表明用FARROW结构的全通滤波器能有效校正时间误差,提高系统的性能。

### 参 考 文 献

- [1] Jenq Y C. Digital spectra of nonuniformly sampled signals : fundamentals and high-speed waveform digitizers[J]. IEEE Transactions on Instrumentation and Measurement, 1988, 37(2): 245-251
- [2] Jenq Y C. Perfect reconstruction of digital spectrum from nonuniformly sampled signals[J]. IEEE Instrumentation and Measurement Technology Conference Ottawa, 1997, 46(3): 649-652
- [3] Black W C, Hodges D A. Time interleaved converter arrays[J]. IEEE J. Solid-State Circuits, 1980, 15(6): 1 022-1 029
- [4] Farrow C W. A continuously variable digital delay element[J]. In: Proc IEEE Int.Symp.Circuits Syst, 1998, 3: 2 641-2 645
- [5] Carson K S, Wu Y C. On the design and efficient implementation of the Farrow structure[J]. IEEE Signal Processing Letters, 2003, 10(7): 189-192

编 辑 徐安玉