浮点算法转换成硬件定点算法中的问题

■北京航空航天大学　唐清贵　夏宇闻

引　言

DSP 和 FPGA 是信号处理工程设计领域发展最快的两个分支。目前,它们的应用非常普及,但是要开发出占用资源少,运行速度快的高质量硬件体系结构比较困难。在通常情况下,算法的硬件实现都需要采用定点运算并考虑并行结构。

但是在实际的许多应用中,比如图像处理、语音压缩等,需要进行大量复杂的数据运算,而且对数据的精度及动态范围都要求比较高,所以,算法模型大多都以浮点数为基础。另一方面,浮点算法在硬件实现上有相当大的难度,不仅占用的系统资源较多,而且硬件运行的速度也较慢,在很多场合下不能很好地满足系统实时性的要求。因此,电子工程师为了提高系统的性能,一般都会先对浮点算法进行仔细的分析,结合工程的实际要求,综合考虑诸多因素,然后再将其转化为定点算法并通过硬件来实现。因此,如何使转化后的算法在硬件上能正确地运行是设计开发人员特别关心的问题。解决该问题的唯一途径就是使处理后的数据保持系统要求的精度和动态范围,否则就会因为数据处理不当,使系统产生莫名奇妙的故障,从而导致系统的失败。在遇到此类问题时,务必要谨慎对待,以免造成不可估量的损失。

1　浮点与定点运算的比较

在开发过程中,必须要对应用的数据精度及动态范围有清楚的认识,并在开始设计硬件结构前,先进行算法定点化的研究工作,严格按照工程设计的实际需求在数学运算上对算法进行数字定点化处理并计算验证。只有这样,才能保证最终设计的硬件体系结构产生的运算结果符合设计需求。

为了更好的说明如何成功的完成浮点算法向定点算法的转化,首先对浮点数和定点数进行简要回顾。

浮点数由三部分组成:指数部分、尾数部分和符号位,如图 1 所示。图 1 中 m、n 分别为指数和尾数的位宽,浮点数据的数值 v_{f_p} 表示为:

$$v_{f_p} = (-1)^s \cdot v_s \cdot 2^{v_e}$$

其中,s 为符号位的值,v_s 为尾数的数值,v_e 是指数的数值。在 IEEE754 标准中,单精度浮点指数位数为 8,尾数位数为 23,还有 1 位符号位。数据绝对值最小可以是 2.0×10^{-38},最大可以是 $2.0 \times 10^{+38}$,双精度浮点指数位数为 11,尾数位数为 52,同样也有 1 位符号位。数据绝对值最小可以是 2.0×10^{-308},最大可以是 $2.0 \times 10^{+308}$。

定点数的表示方法和浮点数相对应,数据直接用二进制表示,且小数点在数据的位置固定。和浮点数相比,它的运算简单,没有浮点数尾数对齐和归一化问题。因此,在硬件上实现以定点数为基础的算法占用的面积少、性能高,故定点运算是信号处理硬件实现中最常用的一种方式。但是,由于定点数表示的范围远远小于浮点数,所以在实现过程可能会有许多的隐患,如在运算过程中,经常会碰到溢出问题和病态方程等问题。

2　精度处理策略

根据上面的分析,可以看到,浮点算法与定点算法主要的不同之处在于算法中的数据精度和动态范围。因此,只要解决了转化后由于数据的精度和动态范围减小引起的问题,那就等于解决了算法的转化问题。

目前,为了保证转化后的算法能正确地实现系统功能,对数据精度处理的方法主要有舍弃最低位和采用最大字长等。前者以损失精度来保证计算中的无溢出,但是得到的结果往往与实际值有一定的差距,从而使系统在很多情况下不能稳定地运行;后者通过应用分析找出所需数据的最大字长(该字长为运算中精度无损失条件下数据的长度),然后再以最大字长的方式进行算法中所有数据的运算,该方法虽然保证了计算结果的正确和精度,但是它的缺点在于造成了资源的巨大浪费,限制了硬件运行速度的提高,因为实现算法中的某些运算所需的字长远远小于最大字长。最佳的转化应该是使处理后算法中的数据既能

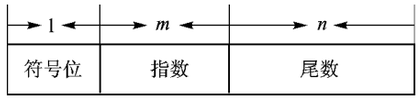

图　1

保证足够的精度,又能使系统达到较高的性能。要达到这一点,必须使硬件资源具有很大的灵活性,能根据不同的需求进行相应的配置。由于 FPGA 硬件资源的利用具有可编程性,因此这种方法的实现是完全有可能的。

对数据精度处理的核心思想就是将数据分段处理,即在精度要求比较高,或者动态范围要求比较大的地方,根据需要采用多种方法来提高数据的精度。在适当的场合选用合适的方法,或将几种方法结合使用,都能起到在有限字长的情况下,使精度得到一定程度的提高。相反,如果在某些区域系统要求的精度不是很高,就可以分配较小的字长来表示该数据的动态范围,以达到硬件实现的数据宽度与系统要求的数据宽度最为接近。通过这种方法对算法进行转化就完全有可能兼顾系统速度与硬件耗费。下面以常见的语音编码为例,对数据分段处理加以说明。

众所周之,人类的听觉能力是有限的,声音能否被听到决定于它的频率和强度,正常人的听觉范围为 20～200 00 Hz,强度范围为 - 5～130 dB 声压级,且在声音频谱中的大部分声音(300 Hz 以下和 10 kHz 以上)只有在 10 dB 的声压强度之上才可以听到。因为从生理学上看,要达到声音听觉阀值,其能量需要达到能在人的耳鼓产生一个驻波,从而使那里的细小毛发产生波动。如果没有这种波动,连接听觉皮层的神经元就不能被触发,因而声音就不能被感知。因此,在语音压缩时,必须对音频信号分段处理,因为对于某些低于人们听觉阀值范围内的语音,在硬件实现上仍按照高精度或者是一定的精度进行压缩是没有实际意义的。另一方面,对于某些频段的语音信号,人耳的灵敏度较高,需要对其采用较高的精度来进行处理,否则就会产生明显的失真现象。

通常情况下,对数字音频信号,往往采用子带压缩编码方法。其原理是将音频信号数字化以后,采用快速傅里叶变换或正交变换,将信号从时间域变换到频率域,然后再将频谱划分成若干个子带,利用听觉掩蔽效应删除下述情况的信号以减少传输信息量:

◇不存在信号频率分量的子带;

◇被噪声掩蔽的信号频率分量的子带;

◇被邻近强信号掩蔽的信号频率分量的子带。

处理完毕冗余信号后,下一步的工作就是对其他的有用信息进行处理。由于系统的传输信息量与频率范围和动态范围有关,而动态范围决定于量化的比特数,因此需要对信号引入比特分配计算。根据不同子带内不同信号表现出的不同性质,使用不同比特数来量化,以达到进一步压缩信息量的作用。

图 2 给出上述压缩编码原理的说明,其中实线表示人耳能听到的最小声压频率特性,即听阈曲线。对于 20 Hz 以下的低频信号和 16 kHz 以上的高频分量,如果声压级低于 60 dB 就听不见,因此可以删除该子带。图 2 中信号 A 在听阈曲线以下听不见,也可以删除。利用强音对弱音的掩蔽,信号 C 由于信号 B 的掩蔽,低于掩蔽效应产生的最小可听值,也可删除。同时,又因在各子带中量化噪声是均匀分布的,所以子带量化所需的比特数可按要求的信噪比来分配,例如,对于图 2 中 60 dB 的信号 B 要求信噪比大于 39 dB,需要以 7 比特进行量化。显然,对于小信号,分配的比特数可以减少。

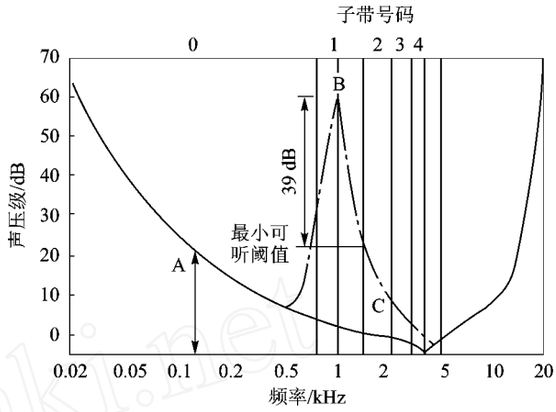

图 2

通过上述方法可以对语音信号进行有损压缩,当然在压缩过程中不可避免的会产生一定误差,但是由于它们不被人耳所察觉,故能保证声音在回放时,人耳不会感到经过处理后的语音在音质效果上与原音有较明显的差别。

3 结 论

综上所述,将浮点算法转化为定点算法后,完全有可能使转化后的算法与原算法一样达到系统要求,同时又使硬件的耗费较小,运行速度提高。但是要使浮点算法高效率地在硬件系统上运行,就必须在硬件实现前,对系统输入/输出数据的处理精度作仔细的计算分析与验证。掌握系统正常运行时各个范围内所要求的最大数据精度和动态范围,然后再结合实际选取最佳的简化方案,最后进行硬件结构的设计,只有这样才能达到最佳的设计效果。

参考文献

1 Randy Allen. Converting floating-point application to fixed-point
2 张铁军. 基于 FPGA 设计的精度管理策略. 微计算机应用,2002(4)
3 江巍,杨军,罗岚,等. MP3 定点解码算法的设计与实现. 计算机工程,2004(3)
4 夏宇闻. Verilog 数字设计教程. 北京:北京航空航天大学出版社, 2003
5 高颉. 定点 DSP 中运算精度的提高. 电子工程师,2001(9)
6 陈丽安,张培铭. 定点 DSP 块浮点算法及其实现技术. 福州大学学报,2004(6)
7 马昌萍,宋丹,马幼鸣. MP3 编码算法分析. 佳木斯大学学报,2005(1)

(收稿日期:2005-08-22)