

MELP 解码器系统的 FPGA 实现

韩琼磊, 郭立, 杨帆, 高路

HAN Qiong-lei, GUO Li, YANG Fan, GAO Lu

中国科学技术大学 电子科学与技术系, 合肥 230027

Department of Electronic Science and Technology, USTC, Hefei 230027, China

E-mail: hql007@mail.ustc.edu.cn

HAN Qiong-lei, GUO Li, YANG Fan, et al. Implementation of MELP decoder on FPGA. Computer Engineering and Applications, 2009, 45(9): 74-76.

Abstract: MELP algorithm with low bit-rate is complexity, which makes implementation of MELP coder/decoder in real time very difficult. Based on the algorithm of MELP decoder, presents a SOC scheme for implementing a MELP decoder system on a FPGA chip, and the system has been verified on FPGA. The important parts of the system are NiosII processor and custom IP cores. The custom IP cores remedies the weak capability of NiosII processor. Experiments shows that MELP decoder is achievable in real time.

Key words: Mixed Excited Linear Predication (MELP); real time; pipeline; IP core

摘要: 低比特率混合激励线性预测 (MELP) 算法的复杂性使得 MELP 声码器系统的实时实现比较困难。根据 MELP 声码器的算法, 提出了一种新的基于现场可编程门阵列 (FPGA) 实现整个解码器系统的单片方案, 并在 FPGA 平台上完成了对整个系统的验证。该系统主要包括 NiosII 微处理器和自定义 IP 模块, 通过自定义 IP 弥补了 NiosII 处理器运算能力的不足。实验结果表明, 实现了 MELP 解码系统的实时处理。

关键词: 混合激励线性预测; 实时; 流水线; IP 模块

DOI: 10.3778/j.issn.1002-8331.2009.09.021 文章编号: 1002-8331(2009)09-0074-03 文献标识码: A 中图分类号: TP311

1 引言

近年来, 随着通信的迅速普及和发展, 移动通信信道资源变得越来越紧张, 人们迫切需要在较低速率上高质量的传输数字语音, 在这种背景下, 低比特率语音编解码技术取得了突飞猛进的发展。混合激励线性预测编解码算法是美国国防部语音信号数字处理协会选定的 2.4 kb/s 标准^[1], 其在很低的比特率下仍能模仿较丰富的语音特征集合, 但同时其编解码运算复杂, 较难在嵌入式处理器上实时实现。

目前, MELP 解码器的实时实现可以通过 DSP 器件实现^[2-3], 但需要外围存储器支持, 难于单片实现, 这会大大提高解码器的成本。国外有线性预测类声码器 (LPC) 单片实现的研究, 而完整的 MELP 解码器单片实现的研究尚不能检索到相关文献。国内文献^[4]中对 MELP 解码器中某些模块及单片实现方法进行了少量研究, 但并未形成有效的实时系统。

本文采用的 MELP 解码器硬件实现结构不同于 DSP 器件实现方案及文献^[4]中的指令设计方案, 而是采用嵌入式 CPU 辅加硬件 IP 的方案。该方案不需要额外的存储器, 降低了成本; 同时也不需要设计专门的复杂指令系统, 只需要修改软件代码就能够重新配置解码器, 增加了设计的灵活性。

2 MELP 解码器算法

MELP 声码器在传统的二元激励 LPC 模型基础上作了改进, 采用了混合激励、非周期脉冲、自适应谱增强、脉冲整形滤波和傅氏级数幅度值五项新技术, 使得合成语音更好地拟合自然语音。图 1 中所示是 MELP 解码器的原理框图^[6]。

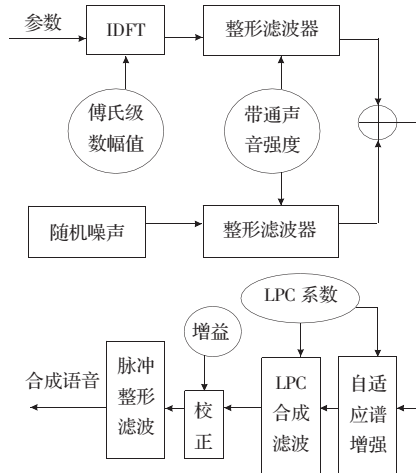


图 1 MELP 解码器语音合成原理框图

基金项目: 国家自然科学基金 (the National Natural Science Foundation of China under Grant No.60772032)。

作者简介: 韩琼磊 (1983-), 男, 硕士, 主要研究方向为语音压缩及实现技术; 郭立 (1946-), 男, 教授, 博士生导师, 主要研究方向为集成电路与系统, 语音与音视频压缩, 传输理论与实现技术等; 杨帆 (1983-), 男, 硕士, 主要研究方向为语音压缩及实现技术; 高路 (1984-), 男, 硕士, 主要研究方向为语音压缩及实现技术。

收稿日期: 2008-09-17 修回日期: 2008-11-17

MELP 声码器的标准采样率为 8 kHz, 每 180 个样点为一帧, 帧长为 22.5 ms, 每帧量化比特为 54, 总的速率为 2.4 kb/s^[5]。在我们的设计中, 对标准算法进行了一些改动, 取 200 个样点为一帧, 帧长 25 ms, 即时要求为 50 ms/2 帧。

3 MELP 解码器系统实现

3.1 系统整体架构

NiosII 是 Altera 公司用于片上系统的一个高效的通用 CPU 核, 选取该核作为解码器系统的处理器。在 100 MHz 的时钟频率下, MELP 解码算法在 NiosII 处理器上解码 2 帧的执行时间如表 1, 显然, 距实时处理要求差别仍然很大(14 倍)。

表 1 解码器各部分在 NiosII 上的执行时间

模块名	运算时间/ms	百分比/(%)
IDFT	(max)404.00	58.13
滤波器	247.55	35.62
LPC 系数转换	(max)14.00	2.04
其他	29.77	4.28
总时间	(max)695.32	100.00

因此, 考虑把其中运算量集中的模块(IDFT、滤波器、LPC 系数转换)提取出来单独用硬件电路实现, 以弥补 NiosII 处理器大规模浮点运算能力的不足, 从而达到实时处理要求。经过综合分析, 最终确定的解码器系统的片上架构如图 2。

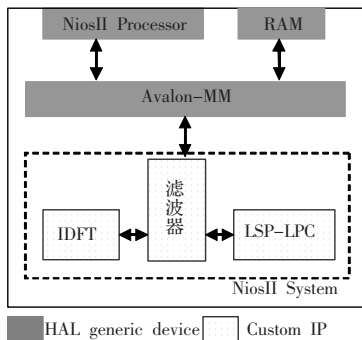


图 2 解码器 NiosII 硬件系统实现方案

图 2 中, 整个解码器系统在一片 Altera 公司的 FPGA 上实现。我们利用了 Altera 提供的 NiosII 微处理器及与之相匹配的 Avalon-MM 总线, 然后把自己的 IP 模块挂到 Avalon-MM 总线上, 从而实现了整个硬件系统。

三个自定义 IP 模块都采用 Verilog-HDL 语言进行 RTL 级设计。各个模块的设计验证都是在 Altera 的 FPGA 平台上进行的。为了减少总线的读写操作, 同时考虑到三个自定义 IP 模块之间的关系, 最终将三个 IP 模块整合到一起, 向 Avalon-MM 总线提供统一的通信接口。下面分别介绍三个自定义 IP 模块的设计。

3.2 自定义 IP 模块设计

由于所有的运算都是浮点型运算, 所以各个模块最基本的单元是浮点乘法器和浮点加法器。在设计中, 浮点乘/加法器都采用三级流水线的方式实现。为了达到最快的运算速度, 设计中使浮点乘/加法器尽量以流水线方式工作。

3.2.1 IDFT 模块

该模块用于产生脉冲激励, 作为整形滤波器的源数据。每解码一帧要进行 200 次 IDFT 运算。其基本公式如下式:

$$e(n) = \sum_{k=1}^{T/2-1} M(k) \cdot \cos[k \cdot w(n)], n=0, \dots, 199$$

其中

$$M(k) = \begin{cases} \text{傅氏级数幅值}, & k=1, \dots, 10 \\ 1, & k=11, \dots, (T/2-1) \end{cases}, w(n) = \sum_{x=-\infty}^n \frac{2\pi}{T_x}$$

上式中, 由于基音周期 $T(10 < T < 148)$ 为一变量, 因此不能利用 IFFT 进行快速计算, 只能按基本公式进行累加。

为了节约运算时间, 充分地使用了流水线——余弦模块流水和乘累加模块流水, 并且两个模块并行运算。图 3 是其内部结构简图。这样最后处理的总时间就主要取决于余弦模块的时间。在最差情况下, 余弦模块每 2 个时钟周期计算出一个结果, 解码过程中 IDFT 运算所需的时钟周期最大值为:

$$2 \times (148/2 - 1) \times 200 = 29\ 200/\text{帧} = 58\ 400/2 \text{ 帧}$$

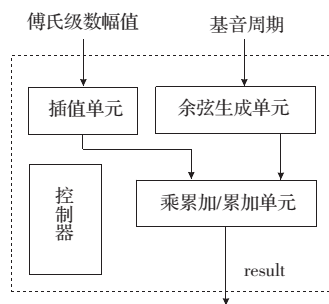


图 3 IDFT 内部结构

在 100 MHz 的时钟频率下, 时间为 0.584 ms/2 帧, 再加上一些预处理及累加后处理时间, 总的最长处理时间为 0.76 ms/2 帧, 比软件处理速度提高 530 倍。

3.2.2 滤波器模块

该模块主要是用于混合激励生成时的整形滤波及后滤波, 解码过程中用到 3 种滤波类型: 30/10 阶 FIR、10 阶 IIR, 其差分实现表达式如下:

$$\text{FIR}: y(n) = \sum_{i=0}^{M-1} b(i)x(n-i), M=30, 10$$

$$\text{IIR}: y(n) = \sum_{i=1}^N a(i)y(n-i), N=10$$

从上面表达式可以看出, 在满足时间要求的情况下, 为了尽可能地减少面积, 一个乘累加单元加上一些控制逻辑就能够实现 3 种滤波器。但在解码算法中, 30 阶滤波器的系数是不固定的, 因此必须有一个系数的生成单元。图 4 是其内部结构简图。

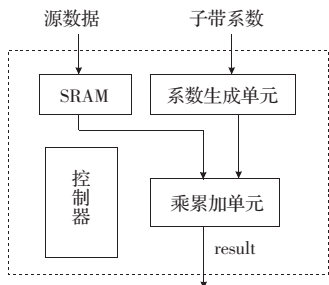


图 4 滤波器内部结构

由于系数生成的复杂性,使运算时间基本都集中到系数的生成上,求解一个系数需要 5 次乘累加操作(5+9 个时钟周期),这样解码过程中 2 个 30 阶整形滤波器仅系数生成需要的时钟周期为:

$$2 \times (5+9) \times 31 \times 200 = 173\,600 / \text{帧} = 347\,200 / 2 \text{ 帧}$$

在 100 MHz 的频率下,时间为 3.472 ms/2 帧,另外再加上 10 阶滤波器时间(主要是乘累加),总的滤波时间为 4.895 96 ms/2 帧,比软件处理速度提高 60 倍。

3.2.3 LPC 系数转换模块

该模块用于产生 LPC 系数,作为 10 阶滤波器的系数。

与前两个模块不同的是,该模块并没有像累加这种集中、重复的运算,因此必须选择一种新的实现结构。为了更好地处理该模块中复杂的逻辑和分散的运算,在设计中采用微程序控制方法实现。

微程序控制的基本思想^[7]是把操作控制信号编成微指令,存放在只读存储器中,当机器运行时逐条地读出这些微指令,从而产生各种操作控制,使相应的部件执行所规定的操作。这样在硬件结构相对简单的前提下就可以处理复杂的逻辑。图 5 是其内部结构简图。

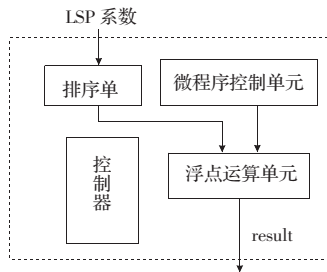


图 5 LPC 系数转换内部结构

其时间消耗即为执行存放在只读存储器中微指令的时间,经实际测试为 2.06 ms/2 帧,比软件运算提高 15 倍。

3.2.4 自定义 IP 模块时间性能分析

以上三个自定义硬件模块的设计、验证及与 NiosII 处理器的集成工作已经完成。在 100 MHz 的频率下,测得各个模块在解码出 2 帧语音过程中的运算时间如表 2。

对比表 1,用硬件模块处理的速度较软件处理大大提高。

表 2 自定义硬件模块时间性能

模块名	运算时间/ms
IDFT	(max)0.760
滤波器	4.896
LPC 系数转换	(max)2.060
总时间	7.716

除去这三部分的运算时间,剩下由 NiosII 处理器处理的软件部分时间为 29.77 ms,再加上总线读写时间,总的解码运算时间为 39.95 ms/2 帧,满足实时要求 50 ms/2 帧。

4 结论

针对 MELP 语音解码算法,提出了一种基于 FPGA 的软硬件混合设计的实现结构。该结构采用简单灵活的嵌入式通用处理器代替高成本的 DSP 处理器,配合自定义 IP 实时地实现了解码器系统,为 MELP 声码器的实时实现指出了一种新的方法,也为 MELP 解码器的最终 ASIC 实现提供了一个过渡环节。

参考文献:

- [1] Supplee L M, McCree A V. MELP: the new federal standard at 2400 bps[J]. Acoustics, Speech, and Signal Processing, 1997, ICASSP-97: 1591-1594.
- [2] Xiao Lin, Choon Boon Lim, Soo Peng Hoh. Real time implementation of low bit rate speech encoder MELP on TMS320C54x DSP[C]// 5th International Conference on Signal Processing Proceedings, Aug 2000: 651-654.
- [3] Chen Liang, Zhang Xiongwei. A new 1.2 kb/s speech coding algorithm and its real-time implementation on TMS320LC548 [C]// WCC-ICCT 2000 International Conference on Communication Technology Proceedings, Aug 2000: 388-391.
- [4] 丁涛. MELP 算法声码器的功能模块与指令系统的设计与实现[D]. 西安: 西安交通大学, 2003.
- [5] McCree A, Truong K. An enhanced 2.4 kbit/s MELP coder[J]. IEEE. Speech Coding for Telecommunications, 101-102, Sept. 20-22, 1995.
- [6] 王炳锡. 音编码[M]. 西安: 西安电子科技大学出版社, 2002.
- [7] Stallings W. Computer organization and architecture[M]. 张昆藏, 译. 北京: 电子工业出版社, 2002.
- [4] Kai -Tai F, Xuan L, Winker P. Lower bounds for centered and wrap-around L2-discrepancies and construction of uniform designs by threshold accepting[J]. Journal of Complexity, 2003, 19(6): 692-711.
- [5] 均匀试验设计表 [EB/OL]. [2008-01-01]. http://www.math.hkbu.edu.hk/UniformDesign/.
- [6] M Yu-Seung, K Yong-Rae, Offutt J. Inter-class mutation operators for Java[C]// Software Reliability Engineering, 13th International Symposium on, 12-15 Nov. 2002.

(上接 73 页)

参考文献:

- [1] W3C. W3C 词汇表[EB/OL]. (2003). [2007-09-25]. Http://www.w3c.org/tr/ws-gloss.
- [2] Zhu Hong. A framework for service-oriented testing of WSs[C]// 3rd International Workshop on Quality Assurance and Testing Web-based Applications, Sept. 2006, Chicago, USA, 2006: 679-691.
- [3] 本书编写组. 正交试验设计法[M]. 上海: 上海科学技术出版社, 1979.