

基于嵌入式 CPU 的 G.722.1 宽带 语音编解码算法优化

谭 伟, 马 琪*

(杭州电子科技大学 微电子 CAD 研究所, 浙江 杭州 310037)

摘要:针对目前语音编码器多采用 DSP 实现,而 DSP 处理器成本相对较高的弊端,提出了一种能在国内自主知识产权 32 位嵌入式 CPU—CK510/520 平台上实现宽带语音实时编、解码的设计方案。首先,研究了基于重叠调制变换技术的宽带语音编解码算法 G.722.1;然后综合该算法原理和 CK-CPU 特性,对算法作了基于 C 语言和汇编语言的各种优化,使得算法程序更适合于硬件实现。实验结果表明,优化后的定点 G.722.1 编码器的算法复杂度得到有效降低,节约了内存空间,并取得了较好的语音重建效果。该研究成果不仅有助于形成基于单 CPU 的比较简单和低廉的语音、编解码应用解决方案,也有利于国产嵌入式 CPU 在语音领域的应用推广。

关键词: G.722.1; 宽带语音编、解码; 32 位嵌入式 CPU; 算法优化

中图分类号: TH73

文献标识码: A

文章编号: 1001-4551(2010)11-0071-05

Optimization for G.722.1 wideband speech codec based on embedded CPU

TAN Wei, MA Qi

(IC CAD Research Center, Hangzhou Dianzi University, Hangzhou 310037, China)

Abstract: Aiming at the drawback that the speech codec was mostly implemented based on DSP, but DSP had a relatively high hardware cost, a 32 bit embedded CPU platform CK510/CK520 with domestic intellectual property was applied to make the real-time implementation of the speech codec. Firstly, the G.722.1 algorithm for wideband speech codec based on modulated lapped transform technology was discussed. Then, combining the principle of the G.722.1 algorithm and the characteristics of CK-CPU, some optimization strategies in C and assembly language were put forward to make the algorithm program more suitable for hardware implementation. The test results indicate that the optimized G.722.1 algorithm can rebuild the speech with high quality, while its time complexity and memory size were both effectively reduced. This research not only helps the formation of a relatively simple and cheap application solution based on single CPU for the speech codec, but also benefits the promotion of domestic CPU.

Key words: G.722.1; wideband speech coding; 32bit embedded CPU; algorithm optimization

0 引言

目前大多数的语音处理系统都是基于窄带语音的,带宽通常被限制在约 200 Hz ~ 3 400 Hz 内,采样率为 8 kHz。但随着数字通讯系统的不断发展,宽带语音越来越受到人们的青睐,其通频带扩展到 50 Hz ~ 7 000 Hz, 16 kHz 采样频率,针对的是不直接与传统数

字 PSTN 进行互操作的应用。与传统窄带语音相比,宽带语音有更好的自然感和可懂度,从而在通信时给人以现场面对面交流的感受^[1-2]。G.722.1 是宽带语音编、解码方案中的一种,尽管宽带语音的传输会占用更高的比特率,但在人们对于高质量音质的需求趋势下它已广泛应用于诸如音视频会议、无线信道、VOIP 以及数字音频广播(DAB)等领域。

一般语音编、解码器多采用数字信号处理运算能

收稿日期: 2010-04-08

作者简介: 谭 伟(1986-),男,湖北天门人,硕士研究生,主要从事集成电路设计及 CAD 方面的研究. E-mail: tanwei19861225@163.com

通信联系人: 马 琪,男,博士,研究员,硕士生导师. E-mail: Maq68@126.com

力强但成本相对较高的 DSP 来实现,而对于语音编、解码的应用,还需要 CPU 来实现网络/通信协议栈和对整个应用系统的主控,这样需要“CPU + DSP”的双核单芯片或双芯片来实施整个应用解决方案。如果能在一款较高性能的嵌入式 CPU 上同时完成语音编/解码、网络/通信协议栈和系统主控,则可有效降低整个应用解决方案的成本。而其技术关键是嵌入式 CPU 如何实时实现语音编、解码。因此在嵌入式 CPU 的平台上深入研究 G. 722.1 语音编解码算法,分析其实时实现的软、硬件要求并提出优化策略,是标准转化到实际应用的关键一步。本研究将就此展开工作。

1 G. 722.1 编解码算法

G. 722.1 是 ITU-T 1999 年制定的标准,是一种提供 50 Hz ~ 7 000 Hz 音频带宽、按 24 Kbit/s 或 32 Kbit/s 比特率工作的数字宽带编码器算法^[3]。编码器输入的数字信号可以是采样频率为 16 kHz 的 14 bit, 15 bit 或 16 bit 二进制补码。

该算法以变换技术为基础,它采用的是被称为“重叠调制变换”(Modulated Lapped Transform),每次处理 20 ms 长度的音频帧,每帧 320 个样点。因为变换的窗长度是 640 个样点,帧与帧之间有 50% (320 个样点) 的重叠,有效预测缓冲区时长为 20 ms。总的 40 ms 算法延迟为帧长度和预测时长之和。所有的其他延迟是由于计算和网络传输引起的,由于计算复杂度的原因,该算法造成的计算延时小于 20 ms,在端对端系统中,总的延时小于 60 ms。

1.1 编码部分

编码系统^[4]框图如图 1 所示。每 20 ms 为重叠调制变换(MLT)提供最新的 640 个时域音频样值,每次变换产生含 320 个 MLT 系数的帧,每一帧的系数将会独立编码,即前一帧中没有状态信息剩余。当工作于 24 Kbit/s 和 32 Kbit/s,每帧分配的比特数分别为 480 和 640。

首先 MLT 变换的系数进入一个幅度包络的模块,计算这些 MLT 系数的幅度包络 $rms(r)$,然后标量量化,组合成矢量后进行哈夫曼编码。该幅度包络近似地表示了 MLT 的频谱图,整个频谱按 20 个 MLT 系数一个区域分成各区域,每区域表示 500 Hz 的带宽。因为信号频域范围为 50 Hz ~ 7 000 Hz,共分为 14 个区域,对于表示 7 000 Hz 以上的 MLT 系数将被忽略。表示幅度包络的代码比特被发送给 MUX(复用设备)以传输到解码器。幅度包络量化和编码余下的比特用于

分类处理过程对 MLT 进行编码。

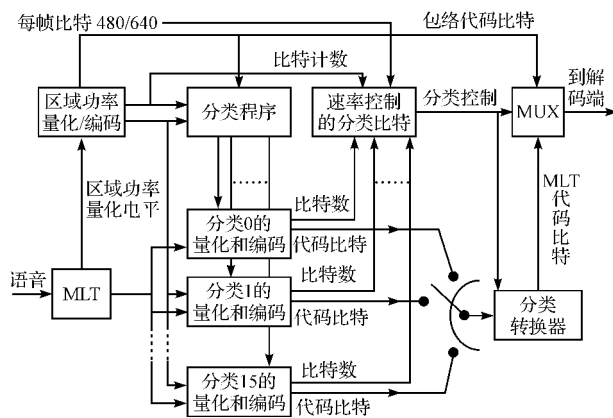


图 1 G722.1 编码器方框图

分类程序使用量化的幅度包络和幅度包络编码(及提供 4 个分类控制比特)之后,基于帧内剩余的比特数目,产生 16 组分类(分类 0 ~ 分类 15)。随后,对 16 种算得分类中的每一种,分别对 MLT 系数进行量化和编码,同时得到每个分类所需要的确切比特数。量化和编码是逐区域进行的,分类方法决定了所有 14 个区域种类分配方式,每个区域 r 的幅度包络 $rms(r)$ 以及分类方式决定了区域中 20 个 MLT 系数所使用的量化和编码参数(其中 $0 \leq r < 14$)。每个区域的 MLT 系数量化后进行哈夫曼编码时所需的比特数是不固定的,出现频度高的矢量指数所需的比特比出现频度低的矢量指数所需的比特少。

在 G. 722.1 中 MLT 的输入是最近的 640 个音频采样值 $x(n)$,其中 $x(0)$ 是最旧的样值,而且 $0 \leq x(n) < 640$,输出的 320 个 MLT 系数值 $mlt(m)$ 由下式得到(其中, $0 \leq m < 320$):

$$mlt(m) = \sum_{n=0}^{639} \sqrt{\frac{2}{320}} \sin\left(\frac{\pi}{640}(n+0.5)\right) \cos\left(\frac{\pi}{320}(n-159.5)(m+0.5)\right) x(n) \quad (1)$$

MLT 变换可以分解为一次窗函数,重叠和加法运算,然后进行 IV 型离散余弦变换(DCT)。其中 G. 722.1 采用正弦窗函数 $w(n)$,其重叠、累加过程为:

$$v(n) = w(159-n)x(159-n) + w(160+n)x(160+n) \quad 0 \leq n \leq 159 \quad (2)$$

$$v(n+160) = w(319-n)x(320+n) - w(n)x(639-n) \quad 0 \leq n < 159 \quad (3)$$

式中, $w(n) = \sin\left(\frac{\pi}{640}(n+0.5)\right)$, $0 \leq n < 320$ 。

联合 $v(n)$ 与 IV 型的 DCT, MLT 结果可表示为:

$$mlt(m) = \sum_{n=0}^{319} \sqrt{\frac{2}{320}} \cos \left(\frac{\pi}{320} (n + 0.5)(m + 0.5) \right) v(n)$$

(4)

利用快速算法可大大降低 DCT 的复杂度。对于码率为 24 Kbit/s 和 32 Kbit/s 的比特流中,每帧的比特数在 480 或 640 之间选择,除了分类控制比特是固定长度(4 位),其余长度全是可变的。帧内比特流的结构如图 2 所示,传输比特自左向右。

首比特		末比特
幅度包络比特 (可变长度)	分类控制比特 (4 比特)	MLT 系数比特 (可变长度)

图 2 比特分配及传输顺序示意图

1.2 解码部分

对于每一帧来讲,首先将收到的表示区域 0 的幅度指数的前 5 个比特进行解码,随后对其余的区域进行哈夫曼解码和重建。然后才能对 4 个分类控制比特进行解码,以确定编码器选择和传输了 16 组可能的分类中的哪一组分类。帧内其余代码比特表示量化的 MLT 系数,这些比特将根据分类情况加以解码。和编码器的情况相似,解码器的分类过程也是同时利用幅度包络和当前帧剩余的比特数来决定分类种类,计算 16 种可能的分类方法。

噪声填充也用于类别 5 和 6,因为这些类别的 MLT 系数有许多是可量化为 0 的。这些按 0 传输的值被置为该区域幅度平均值的极小分数,符号也是随机确定的。

对于标量量化为非零值的那些系数,有一个预先确定的表含有归一化系数的重建值。随后,用 $rms(r)$ 的相应值对重建的值按比例缩放。把表示 7 kHz 以上频率的 40 个 MLT 系数置为 0。重建了 MLT 系数之后,反向重叠调制变换(IMLT)产生 320 个新的时域样值。

除了 IMLT 最后一步重叠和加法运算之外,解码器收到的每一帧中的信息与在上一帧中的信息无关。

2 基于嵌入式 CPU 的 G.722.1 编解码的算法优化

语音压缩算法复杂,运算量大,需要高成本的硬件支持,必须对算法进行优化,才能使得在低速、RAM 空间较小的嵌入式 CPU 上实时实现。算法主要从以下几方面进行优化。

2.1 算法级的优化

从 G.722.1 的算法原理入手,根据原理提出更简

单的实现方法,从而减少运算量。常以损失算法的精度来减少运算的复杂度,这样可能带来编、解码输出和标准的输出不一致,但并不是说明有逻辑错误。优化的原则^[5]是编、解码后得到的语音音质不错,人不能明显听出异声来,这样的 Codec 就能接受。

G.722.1 算法中为了降低离散余弦变换(DCT)的复杂性,采用了一种快速 DCT 变换技术。尽管如此,DCT 变换中大量的乘法和加法运算仍严重影响了变换速度,为了减少运算次数,缩短运算时间,可用查表法实现对快速 DCT 变换方法的优化^[6]。此法是将所有可能的 DCT 系数值事先全部算出放入表中,由于三角函数值的重复性和对称性,在一个 10 × 10 的 DCT 系数表 `dct_core_a[10][10]` 中,其非对角线上的系数值全部关于对角斜线对称。故 10 × 10 的 DCT 系数表中只有对角斜线上的 10 个系数值加上对称的 45 个系数值,总计 55 个 DCT 系数值有用。接着便可将精简后的 DCT 系数表 `array` 重新建立,这样缩小了系数表的规模,更重要的通过建立查表的方法可完全避免乘法,加快了变换速度。

2.2 代码级的优化

算法的关键代码指的就是指程序中最耗时、占总运算量比重较大的函数^[7],对关键代码进行优化可以事半功倍。笔者通过 Microsoft Visual C++ 6.0 中的 `profile` 工具统计各个程序所需时间占算法总耗时的百分比,得到 G.722.1 中最耗时的关键代码为 `Basop32.c` 函数。

`Basop32.c` 函数是整个 G.722.1 算法 C 代码中的基本运算模块,其绝大多数运算操作可以通过 CK510 的汇编指令来完成。如在 `add` 函数中溢出控制的代码可以用一条带溢出控制的 `ADDC` 的指令实现。采用 `Basop32` 函数和单指令相比主要增加两部分代码:第一部分是出栈入栈操作,调用 `Basop32` 函数之前需要将参数压入堆栈,`Basop32` 函数再从堆栈中取出参数;第二部分是 `Basop32` 函数内部执行溢出控制所消耗的代码。根据 CK510 的汇编指令手工汇编关键代码既可有效利用 CK510 的片内资源,又充分发挥了 CK510 潜在的计算能力,降低了编、解码程序对资源的需求,达到低功耗、高效率的要求。

采用汇编语言修改关键部分代码可能是有效的方法之一,但是也可综合运用其他一些代码优化的方法。如编译器的优化编译选项设置^[8],可选择最大优化级别 -O3;又如,可利用 C++ 语言的 `inline` 特性来将函数设置为内联函数,内联函数能将函数内代码直接嵌入到调用处,节省参数压栈和出栈的操作,但缺点是会

使代码量大大提高;此外还可改写算法中的循环语句以减少循环体内总的指令执行周期数^[9],如对于嵌套的多层循环,可将循环次数较少的作为外循环,循环次数较多的放在内部,以提高循环的执行效率;改写循环时要注意到,同样实现循环功能,for 循环的功耗比 while、goto 的要大,goto 语句的功耗明显最低。

2.3 面向硬件资源的优化

不同的嵌入式 CPU 具有不同的体系结构和指令集,如指令集中是否有 SIMD 指令、MAC 指令,cache 是否可配置,可针对硬件特点进行局部的算法优化。

CK510 指令的执行流水线有 7 个阶段,指令同时执行时分别处在流水线的不同阶段。CPU 指令的执行天然是具有反馈性的,也就是前一条指令的执行不可避免地会影响后面指令的执行,即指令之间不可避免相关性。当某条指令进入流水线的某个阶段时,需要使用某个资源,而此时该资源正好被处于另一阶段的某条指令使用时,就产生了冲突。虽然遇到这种情况时,处理器会自动延时以保证程序的正确执行,但却引起流水线的延时,增加了指令执行时间。为了消除这种延时,就需要适当地改变指令执行的顺序来打破这种依赖关系。例如改进数据相关冲突,就是分析前后指令数据使用上的关系,如出现相关,就将前面指令的计算结果尽早取出交给后面的指令使用,这是运行时的数据冲突解决办法^[10]。还有编译时的冲突解决方法,就是将出现冲突的指令序列在不改变逻辑的前提下重新排序,以避免冲突的发生。另外,对于无数据相关性的需要多周期执行的指令,如 lrw、mult、mac 等要尽量使其成对执行,减少流水线的冲突。

此外,本系统实现时可采用 CK-Core 系列中的 CK520 来代替 CK510,与 CK510 相比,CK520 的运算速度更快并且其指令集中多了一个 MAC 指令,这样 MULT 指令(两执行周期)和 ADD 指令(单执行周期)便可合并为 MAC 单周期指令,减少了指令执行周期。

3 算法验证

本研究算法优化的验证在 PC 机 Microsoft Visual C++6.0 环境下进行,利用 ITU-T G. 722.1 提供的测试矢量对优化后的算法作验证测试。

在 VC++ 环境中新建一个名称为 ENCODE_TEST 的工程,加入与编码部分有关的源程序和头文件,编译生成可执行文件 encode.exe;然后按照 ITU 提供的测试矢量,使用编码器所需的命令行格式运行 encode.

exe,如执行:

```
encode.exe 0 g722_1_enc_in.pcm output_test  
24000
```

其中 0 表示采用压缩比特流;g722_1_enc_in.pcm 是 ITU 自带的测试矢量,表示编码器端输入的 PCM 音频文件;24 000 代表 24 Kbit/s 的码率,执行后得到的输出文件是 output_test。通过 output_test 与 ITU 提供的参考矢量 g722_1_enc_out_24000.pak 进行比较完全一致,验证了算法优化源程序的正确性。

笔者还采用一段标准语音文件进行语音测试,首先提取其中的 256 bit-PCM 进行编解码处理,再还原为 wav 文件。通过简单的 MOS 测验和波形观察,发现输入 G. 722.1 编码器的 PCM 语音与重建语音的波形基本相同,在主观听觉上也获得了比较令人满意的语音质量,保持了对原始语音信号的重建性。

4 结束语

本研究介绍了 ITU G. 722.1 语音编解码的基本原理,以及在嵌入式 CPU 平台 CK510/520 上进行该算法的优化、移植和实现。最后通过语音测试表明,经过多种优化策略处理后的该语音压缩算法能够满足语音通信实时性的要求,且能较好地恢复语音,可应用在 VOIP 终端等嵌入式系统产品开发中。目前,已有针对各类业界流行的嵌入式 CPU、DSP 的商用 G. 722.1 算法,但尚未有为国内自主知识产权嵌入式 CPU 专门开发的 G. 722.1 语音编、解码算法,开发该算法,可为国内自主知识产权嵌入式 CPU 以后用于语音产品开发做好准备。

本系统中 G. 722.1 语音编、解码器仍有较大的优化空间,如降低约 40 ms 的算法延迟,提高其语音质量和抗噪声性能。此外,本研究中所完成的 G. 722.1 CK-CPU 算法只能提供一路语音编、解码,若要求实现多路编、解码,则需要给每一路分配一个编码状态的 CoderStat 结构体和解码状态的 DecoderStat 结构体,并在程序上做一些改动。

参考文献(References):

- [1] 陈健,沈雷. 基于 MLT 算法的低码率宽带语音编码研究和实现[D]. 上海:上海交通大学电子工程系,2002.
- [2] 赵晓群. 数字语音编码[M]. 1 版. 北京:机械工业出版社,2007.
- [3] 王炳锡. 语音编码[M]. 1 版. 西安:西安电子科技大学出版社,2002.

(下转第 126 页)

转一圈行程变量减一。最后检测关门极限是否有效,若有效则门已完全关闭,切断电机信号。

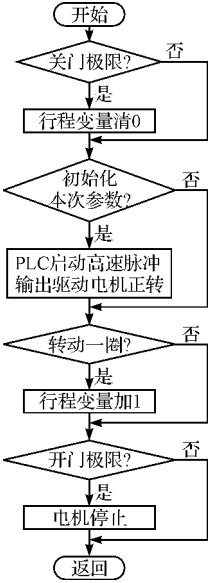


图 3 开门子程序流程图

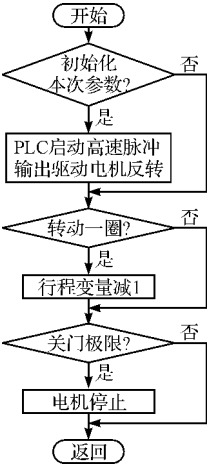


图 4 关门子程序流程图

注意,每次开(关)门的时候,门不一定是完全关闭(打开)的,用行程变量表征电动门所处的位置。高速脉冲输出功能可根据行程变量的大小输出相应的脉冲,使电机按加速、恒速、减速的方式运行。

4 结束语

本研究将 PLC 应用到自动门控制系统中,可以充分发挥 PLC 高可靠性和抗干扰性特点,外部接线简单、灵活,维修方便。采用了双元件型热释电红外传感器检测人体出入情况,同时采用红外线光电感应器防止夹人(或物)的情况出现。传感器灵敏可靠,不受环

境因素的干扰。本系统已通过硬件和软件调试,在实际中获得了应用,具有较高的可靠性、灵活性和经济适用性。

致谢:本论文得到山东省高等学校优秀青年教师国内访问学者项目经费资助。

参考文献 (References) :

[1] 廖常初. S7-200PLC 基础教程[M]. 重庆:重庆大学出版社,2007.

[2] 赵春生. 可编程控制器应用技术[M]. 北京:人民邮电出版社,2008.

[3] 郭艳萍. 电气控制与 PLC 技术[M]. 北京:北京师范大学出版社,2007.

[4] 赵玉安,姚大鹏. 无线热释电红外传感器[J]. 电子制作, 2005,12(2):15-16.

[5] Murata Corporate. Pyroelectric Infrared Sensors[M]. Murata Corporate,2007.

[6] FANG Jian-shuen, HAO Q, BRADY D J, et al. Real-time human identification using a pyroelectric infrared detector array and hidden Markov models[J]. **Optics Express**,2006, 14(15):6643-6658.

[7] 王智明. PLC 基于开关量实现模拟量输出的方法[J]. 机电工程,2009,26(5):105-107.

[8] 李 慧,李海霞,冯显英. 基于 MCU 和 CPLD 的智能移动机器人控制系统[J]. 机电工程,2009,26(8):100-103.

[9] Siemens. SIMATIC S7-200 Programmable Controller System Manual[M]. Siemens,2007.

[10] Siemens. SIMATIC Programming with STEP7 V5.3 Manual [M]. Siemens,2004.

[编辑:柴福莉]

(上接第 74 页)

[4] ITU-T Rec. G.722.1, Low-complexity coding at 24 and 32 Kbit/s for hands-free operation systems with low frame loss [S]. New York: Institute of Electrical and Electronic Engineers, Inc., 2005.

[5] MARTIN R, HEUTE U, ANTWEILER C. Advances in Digital Speech Transmission[M]. Chichester, England: John Wiley,2008.

[6] BESSETTE B. The adaptive multirate wideband speech codec (AMR-WB)[J]. **IEEE Transactions on Speech and Audio Processing**,2002,10(8):620-636.

[7] KIM H K, LEE H S. Interlacing properties of line spectrum

pair frequencies[J]. **IEEE Transactions on Speech and Audio Processing**,1999,7(1):87-90.

[8] 李纯静,沈保锁. ITU-T G.723.1 语音编码算法的分析及优化策略[D]. 天津:天津大学电子信息工程学院, 2007.

[9] 杨树堂,周敬利,余胜生. G.722.1 语音编码器算法的聚类优化策略及其应用[J]. 通信学报,2001,10(2):30-32.

[10] 刘丹青,马 琪,李春强. 基于嵌入式 CPU 的 SBC 音频编解码器的实现和优化[J]. 计算机应用与软件,2008, 15(11):40-42.

[编辑:李 辉]