

文章编号:1001-5132 (2010) 01-0046-04

AVS 和 H.264 的通用反量化模块设计

陈小垒¹, 彭宗举¹, 黄 晔², 郁 梅¹

(1.宁波大学 信息科学与工程学院, 浙江 宁波 315211; 2.宁波中科集成电路设计中心, 浙江 宁波 315040)

摘要: 为提高 AVS 和 H.264 反量化算法硬件资源的利用率, 设计了一种能够兼容 2 种解码标准的反量化模块. 在分析了 AVS 和 H.264 反量化算法的基础上, 给出了两者之间的异同点. 并根据反量化算法的特点, 采用基于算法和硬件模块的复用技术, 提出了一种资源复用的硬件架构, 不但实现了同块中 4 个像素点的并行运算, 而且实现了 H.264 中直流块和交流块的并行处理. 仿真结果表明: 该模块能满足高清数字视频的实时处理应用.

关键词: 反量化; AVS; H.264; 复用; 并行处理

中图分类号: TN4

文献标识码: A

AVS 是我国第一个具有自主知识产权的数字音视频解码标准, 全称是“信息技术先进音视频编码标准”^[1]; H.264 是国际标准组织 ITU-T Video Coding Experts Group 和 ISO/IEC Moving Picture Experts Group 联合开发的视频压缩标准^[2]. AVS 和 H.264 采用了大量最新的视频编码技术, 能获得较好质量的压缩图像, 但是其压缩效率的提高是以算法复杂度的提高为代价. 它们的解码复杂度是 MPEG-2 的 2~3 倍, 这就造成了对于高清视频的实时解码将更加困难, 需要设计相应的硬件解码电路以实现实时解码. 现有的硬件设计主要都是针对单一标准的反量化设计^[3-5], 笔者在研究了 AVS 和 H.264 视频编码标准和数字视频解码芯片系统结构的基础上, 设计了一种同时支持 AVS 和 H.264 这 2 种标准的高清解码的反量化 IP 核, 能够满足 AVS 和 H.264 Main Profile Level 4.0 的高清视频码流的实时解码要求.

1 反量化算法分析

AVS 标准对亮度块采用 64 级量化, 对色度块采用了 52 级量化; 其量化值 QP 越大, 量化精度越低. AVS 的反量化是将二维量化系数矩阵 $Quant$ - $CoeffMatrix$ 转换为二维变换系数矩阵 $CoeffMatrix$ 的过程. 二维变换系数矩阵 $CoeffMatrix$ 的元素 W_{ij} 可用下式表示为:

$$W_{ij} = (Q_{ij} \times Dequant + 2^{(shift-1)}) \gg shift, \quad (1)$$

其中, 量化系数 Q_{ij} 取值范围为 -211~210; W_{ij} 取值范围是 -213~212; $Dequant$ 为反量化参数; $shift$ 为移位参数, 它们可从标准的规定表中查得.

H.264 标准将 DCT 逆变换中的部分乘法运算归入到逆量化的乘法运算中, 减少了乘法运算的次数, 从而降低运算量. 其反量化过程处理的数据有 3 种类型: 色度块的直流系数、帧内 16×16 预测模式亮度块的直流系数以及除此之外的其他系

数. 如果当前解码的是色度块或者帧内 16×16 预测模式的亮度宏块, H.264 解码器在进行反量化之前, 需要将直流分量进行反 Hadamard 变换, 恢复出整数 4×4 DCT 变换的直流分量^[6].

表 1 分析了 AVS 和 H.264 标准的反量化算法, 给出了两者的异同点.

表 1 2 种反量化算法比较

异同点	AVS 和 H.264
相同点	根据 QP 查找反量化参数, 进行移位操作, 乘法和加法
不同点	其反量化参数的数量不同, AVS 相对较大. H.264 的直流分量反量化前先进行反 Hadamard 变换

2 硬件结构设计和复用

考虑到 2 种标准的兼容性, 反量化模块将从三方面着手设计: (1)提取 2 种标准中相同之处, 设计一种硬件结构的模块, 它能同时实现 2 个标准的功能; (2)提取同标准中算法上相同的部分加以复用; (3)在实现 2 种标准时, 尽量采用同样的架构和接口, 实现接口电路的复用. 结合以上三方面及表 1 的考虑, 笔者将 AVS 的反量化和 H.264 除直流分量之外的其他数据在图 1 中 IQ_main 模块实现, 其余反量化外部 4 个接口共用. 另外, H.264 直流系数块和色度块在量化时, 需要先进行反 Hadamard 变换, 再进行反量化, 因此该部分用图 1 中 IQ_intra16_chroma_dc 模块单独处理, 作为主模块 1 个辅助模块. 本模块的顶层硬件结构如图 1 所示.

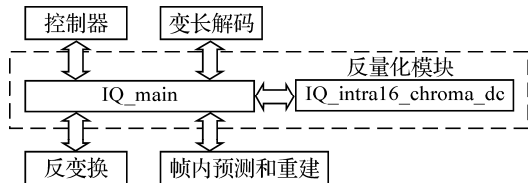


图 1 反量化模块顶层结构

(1) 控制器模块. 通过 32-bit RISC CPU 实行对各个硬件模块的实时控制. 对笔者设计的模块来说, 发送 32 位的命令中包括是序列层还是宏块层、AVS 和 H.264 的判断、CBP(Code Block Pattern)

以及 QP 值的信息等, 具体命令见表 2.

表 2 32 位命令

位数	命令	位数	命令
32~31	1:序列层, 3:宏块层	28~23	QP (色度)
30	IPCM(编码)	22~7	CBP(16bits)*
29	1:AVS, 0:H.264	6~1	QP (亮度)

注: AVS 时低 6 位有效.

(2) 变长解码模块. 解析 2 种标准中的熵编码模式. 表 3 描述其与反量化模块的接口设计, 每次送 16 个点的数据给反量化模块, 即 AVS 中 8×8 块的 2 行或者 H.264 时的 1 个 4×4 块的数据. AVS 时, 块索引的高 2 位表示 8×8 块序号, 低 2 位表示 8×8 块中的行数, 输入 4 次才完成 AVS 的 1 个 8×8 块的数据. H.264 时块索引表示宏块中实际的 4×4 块序号.

表 3 反量化接口设计

位数	命令	位数	命令
198	1:亮度, 0:色度	196~193	块/行索引
197	1:直流, 0:交流	192~1	16 个点的残差数据

反量化模块内部结构设计如图 2 所示, 图中 RAM1 存储的是 AVS 反量化参数数据或者是 H.264 反量化参数数据, RAM2 存放的是 AVS 或者 H.264 反量化时需要进行移位操作的数据, 反量化时只要根据 QP 值查询 RAM1 和 RAM2, 以避免大量的除法运算.

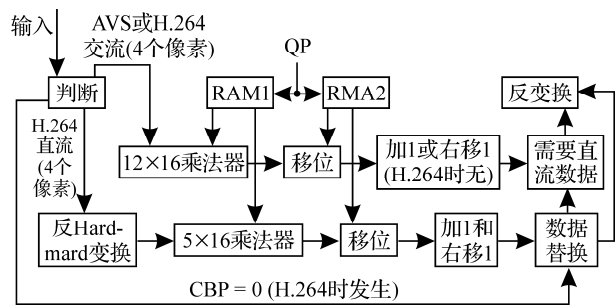


图 2 反量化模块的内部结构设计

反量化模块工作时, 首先根据控制器模块发送来的命令, 判决 AVS/H.264 标志符. 对于 H.264 编解码模式, 如果当前解码是帧内 16×16 预测模式或者色度预测模式的交流分量而且 CBP 为 0, 说明该块没有残差数据. 但直流分量可能不为 0, 虽然省去反量化运算, 但数据替换还要进行操作, 如图

2 中最下面的箭头表示. 除此情况外, AVS 和 H.264 中一旦出现块的 CBP 为 0, 则直接进行下个块的操作; 反之, CBP 不为 0, 考虑到数据处理的效率, 采取 1 次对 4 个像素点的数据进行反量化的并行操作. 如果输入的是 AVS 数据或者 H.264 的交流系数部分数据, 则先通过 FPGA 内部专用 12×16 位的乘法器, 然后对相乘后的数据进行移位操作, 得到反量化的数据. AVS 除了和 H.264 的乘法移位运算复用后, 还需要进行加 1 和右移 1 位操作才完成反量化运算. 宏块开始时, 根据交流直流标志, 直流标志为 1 就表明需要进行数据替换, 否则直接进行输出.

如果输入的是 H.264 的直流数据, 先进行反 Hadamard 变换, 再通过 FPGA 内部专用 5×16 位的乘法器, 然后根据 QP 查询 RAM2 得到反量化参数, 右移反量化参数加 1 位. 之后, 如果解码的是帧内 16×16 预测模式亮度的直流分量时, 还需要加 1 和右移 1 位操作; 色度直流分量时, 直接返回到主控模块进行数据替换和重组.

3 实验结果分析

利用 Verilog HDL 完成所提出硬件架构的 RTL 级代码编写, 根据 H.264 标准的校验模型 JM86 和 AVS 参考模型 RM52j, 编写硬件解码 C 语言模型, 得到硬件仿真所用到的测试向量. 在 Mentor 公司的 Modelsim 6.1b 工具下进行仿真, 仿真结果与运行硬件 C 模型所得到的结果一致, 表明反量化硬件结构的正确性.

本设计不但实现了 AVS 中残差数据的反量化处理, 而且实现了 H.264 中直流残差数据和交流残差数据反量化运算的并行处理. 解码 H.264 时, 第 1 个交流块反量化完成时, 直流块的处理已经完成并返回到主模块 IQ_main, 因此直流系数的替换无需等待. 具体过程如图 3 所示, 宏块开始时读直流残差数据的时间就是 dc_ac_flag 为高电平的处理

时间, 低电平的时间是读取和处理交流数据的周期, 可以看到索引号为 0 的交流块还没有处理完时, 直流数据 i_dc_data 已经返回. 解码 AVS 时, 以每个 8×8 块的 2 行为处理单位, 同样是 16 个像素点. 仿真结果表明, AVS 中宏块亮度和色度分量的处理时间完全相同, 每 2 行的处理周期为 15.

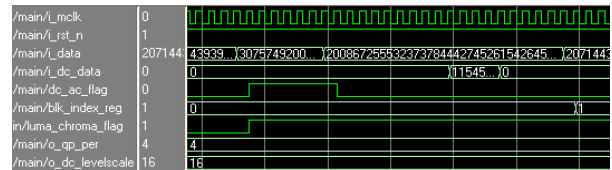


图 3 宏块数据处理

实验中采用了 Smic 0.18 μm CMOS 工艺库, 用 DC (Design Compiler) 工具进行综合. 仿真和综合结果与文献[7]相比见表 4. 从表 4 可见, 在 AVS 标准编解码时, 新方法的宏块处理周期明显低于文献[7], 最大工作频率也相对高一些; 而且其支持 AVS 和 H.264 双标准的反量化, 且 H.264 的最大宏块处理周期为 455, 而文献[7]只支持 AVS 标准. 新方法的处理周期还包括在 1 个新的宏块开始时, 读控制器模块发来的 32 位命令, 然后将宏块 CBP 信息转发给帧内预测和重建模块所需要的 2 个周期.

表 4 综合和仿真结果与参考文献[7]结果对比

	工艺/ μm	最大周期/ 宏块	处理 AVS 8×8 块的时间/ms	最大频 率/MHz
文献[7]	Smic 0.18	396	66	200
新方法	Smic 0.18	455(H.264)	60	220
		362(AVS)		

上述分析是在块的 CBP 标志非零的最差情况下得到的. 一旦出现 CBP 为 0, 部分块的反量化将不用计算, 所以反量化总周期还将有所减少. 当解码器工作频率是 216 MHz 时, 要解析的是 $1920 \times 1080i@30 \text{ fps}$ 的视频流, 只要满足解析每个宏块的周期数小于 882 即可. 根据实验结果分析, 本设计满足高清视频解码的任务.

目前已经完成了系统设计、硬件模块设计、硬件模块 Verilog HDL 语言实现、代码仿真测试以及各个模块的综合等工作, 并取得了阶段性成果. FPGA 平台正在进行搭建, 在 FPGA 平台上进行系

统验证等工作有待下一步开展。

4 结论

在认真分析 H.264 和 AVS 标准反量化算法的基础上, 针对两者的相同之处, 采用了同一个硬件资源实现。由于采用了四个点的并行解码, CBP 的引入加速了对数据的快速处理, 节约了不必要的宏块处理时间, 大大提高了宏块的解码速度。仿真结果表明, 本文所提出的兼容 AVS 和 H.264 标准的通用反量化模块, 在硬件资源共享的基础上, 一定程度上提高了解码效率。用 Smic 0.18 μm CMOS 作了综合, 评估了电路面积和性能, 能够满足高清视频图像解码的要求。

参考文献:

[1] Audio Video Coding Standard Workgroup of China (AVS).

Advanced coding of audio and video-part 2: Video[S]. 2004.

- [2] Wiegand T, Sullivan G. Draft ITU-T recommendation and final draft international standard of joint video specification[C]. ITU-T Rec. H.264/ISO/IEC 14 496-10 AVC, in joint video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, 2003.
- [3] 赵爽, 王勇, 周晓方, 等. H.264 反变换反量化器的硬件设计[J]. 微电子学与计算机, 2008, 25(1):144-148.
- [4] 刘艳, 陈永恩. AVS 数字视频解码中反量化/反变换的 FPGA 实现[J]. 现代商贸工业, 2008, 20(3):293-294.
- [5] 杜国泽, 张春, 杨昆, 等. H.264 中反变换反量化模块的硬件实现[J]. 电视技术, 2007, 31(8):23-25.
- [6] 毕厚杰. 新一代视频压缩编码标准—H.264/AVC[M]. 北京: 人民邮电出版社, 2005.
- [7] Bin Sheng, Wen Gao, Di Wu. An implemented VLSI architecture of inverse quantizer for AVS HDTV video decoder. IEEE International Conference on ASIC[C]. China: Shanghai, 2005:306-309.

Universal Inverse Quantization Module Design for AVS and H.264

CHEN Xiao-lei¹, PENG Zong-ju¹, HUANG Chao², YU Mei¹

(1.Faculty of Information Science and Technology, Ningbo University, Ningbo 315211, China;

2. Ningbo CAS IC Design Center, Ningbo 315040, China)

Abstract: In order to improve the efficiency of hardware resources of AVS and H.264 inverse quantization algorithm, a new inverse quantization module is designed, which is compatible with two video decoder standards. Analysis of AVS and H.264 inverse quantization algorithm is first conducted by presenting the similarities and differences between these two standards. Based on multiplying technology in the algorithm and hardware module, a hardware architecture scheme for resources multiplication is proposed. The presented design has not only realized parallel computing for four pixels in a block, but also implemented the parallel processing of the AC block and DC block in H.264. The simulations show that the design can meet the real-time requirements for HDTV applications.

Key words: inverse quantization; AVS; H.264; multiplexing; parallel processing

CLC number: TN4

Document code: A

(责任编辑 章践立)