

# 基于FPGA的AVS环路滤波器设计与实现

黄中华, 支 琤

(上海交通大学芯片与系统研究中心, 上海 200240)

**摘要:** 根据 AVS 音视频编码标准中提出的环路滤波算法设计了一个高效的环路滤波器。通过适当地增加片上数据存储空间, 使得整个环路滤波的过程占用的带宽资源最小。该滤波器经过仿真验证, 对一个完整的 4:2:0 格式的宏块数据进行环路滤波仅需 293 个时钟周期。可以被用于 1920×1080 高清图像的 AVS 解码芯片中。

**关键词:** AVS; 环路滤波; 去块效应

## Design and Implementation of AVS Loop Filter Based on FPGA

HUANG Zhonghua, ZHI Cheng

(IC and System Research Center, Shanghai Jiaotong University, Shanghai 200240)

**【Abstract】** In this paper, according to the arithmetic of loop filter in AVS audio video coding standard, a high effective loop filter circuit is designed. The band width is reduced lowest by increasing data memory on the chip acceptably. The circuit is verified through stimulation and processes a complete chroma format 4:2:0 macroblock data in 293 clock cycle. As a result, the circuit can be used in 1920×1080 high definite image AVS decoding IC.

**【Key words】** Audio video coding standard(AVS); Loop filter; Deblock

音视频编码标准是我国具有知识产权的第二代信源编码标准。在压缩效率上是 MPEG-2 的 2.4 倍, 跟 H.264 相当。为了改善图像质量, AVS 采用了环路滤波, 来消除用于帧间预测的参考帧中由于量化、运动补偿以及帧内预测而产生的图像块噪声。环路滤波的应用在不影响压缩效率的前提下使图像质量有了很大的改善。但与此同时, 解码端的计算复杂度也增大了很多, 基于宏块的滤波处理过程, 消耗了大量的时钟周期。为了在有限的总线带宽资源下, 保证解码的实时性, 文中引入了 Shih-Chien Chang 等提出的一个基于 H.264 标准的环路滤波电路结构, 该结构把环路滤波器作为一个总线设备, 环路滤波过程中所用数据及参数都通过总线从外部存储器得到。Shih-Chien Chang 通过预先计算边界强度(bs)来决定是否需通过总线下载当前宏块数据。为了使环路滤波能够应用于具有实时解码的 AVS 芯片中, 本文对 Shih-Chien Chang 的电路结构进行了改进, 依靠适当增加片上的存储, 以及增大与系统内前级模块的交互, 使环路滤波高效运算的同时把占用的系统带宽减到最小。

### 1 AVS 环路滤波算法

除了图像边界及条带的边界之外, 宏块的所有边界都应进行滤波。此处宏块边界定义为宏块内部各个 8×8 块的边界, 以及当前宏块与相邻宏块的上边界和左边界, 如图 1 所示(以 4:2:0 为例)。

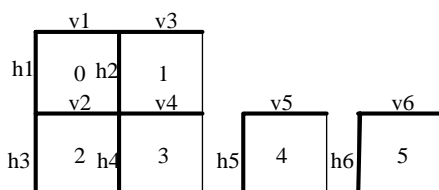


图 1 滤波边界

在滤波过程中, 首先计算宏块中每个 8×8 块的上边界和左边界的边界强度(bs), 色度块的边界强度用对应位置亮度块边界的 bs 代替。然后根据边界强度的数值选择相应的滤波器对边界进行滤波处理。滤波时的顺序是先垂直边(从上到下), 后水平边(从左到右)。当前宏块的上边或者左边像素值可能在以前的环路滤波过程中被修改, 当前宏块的滤波处理的输入为这些可能被修改的像素值, 并且当前宏块环路滤波可能进一步修改这些像素值。

#### 1.1 边界强度(bs)的计算

AVS 中定义了 3 种环路滤波强度, bs 的取值为 0、1 或 2。其中 bs 等于 0 的边界不作处理, bs 等于 1 和 2 的边界都要进行滤波, 2 的滤波强度要大于 1。强度大的滤波器处理的数据范围和修改的数据幅度都要大于强度小的滤波器。

对 bs 值的计算如下:

- (1) 如果边界两边的两个 8×8 块有一个或两个都属于帧内预测宏块, 那么 bs 等于 2;
- (2) 否则, 如果满足以下两个条件中的任一个, bs 等于 1;
  - 1) 两个块的参考图像不同;
  - 2) 两个块的参考图像相同, 但是两个运动向量分量中任一个分量的差值大于一个整像素。
- (3) 否则, bs 等于 0。

#### 1.2 块边界门限的计算

图 2 表示块 p 和块 q 在水平或垂直边界(边界用黑色粗线表示)两侧的 6 个像素点(实线框内)。

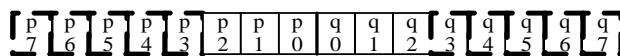


图 2 滤波的像素

**作者简介:** 黄中华(1982-), 男, 硕士生, 主研方向: 高清电视, 数字视频处理, 视频编解码芯片设计; 支 琤, 高工

**收稿日期:** 2006-03-20 **E-mail:** hzhonghua@sjtu.edu.cn

如果下式为真，对边界样本滤波：

$$bs \neq 0 \ \&\& \ Abs(p_0 - q_0) < \alpha \ \&\& \ Abs(p_1 - p_0) < \beta \ \&\& \ Abs(q_1 - q_0) < \beta \quad (1)$$

其中  $\alpha$  和  $\beta$  为块边界门限值，可以根据两个块的 QPav 和图像头中的 alpha\_c\_offset、beta\_offset 计算查表索引 IndexA 和 IndexB，两个块的 QP 平均值 QPav 为

$$QP_{av} = (QP_p + QP_q + 1) \gg 1 \quad (2)$$

索引 IndexA 和 IndexB 为

$$IndexA = Clip3(0, 63, QP_{av} + AlphaOffset) \quad (3)$$

$$IndexB = Clip3(0, 63, QP_{av} + BetaOffset)$$

根据以上两个查表索引可从 AVS 标准给定的一个表中查出两个门限值。

### 1.3 滤波计算

当式(1)为真时，根据 bs 的值来选择不同的滤波处理方法，对图 2 中边界两边的像素点进行滤波处理。

## 2 实现架构

### 2.1 实现框图

根据对 AVS 中环路滤波的算法描述，如何减少与片外存储器的数据交换以及快速有效地调整待滤波的数据成为提高滤波速率实现实时性解码的关键。对于整个 AVS 系统来说，实际需要带宽等于从片外存储请求的数据占用带宽与向片外存储输出数据占用的带宽之和。由于总线本身的利用效率因素，在一定的时钟频率下，系统实际可用的带宽是很有限的。运动补偿已经占用了大部分的带宽资源从片外存储请求数据。所以在环路滤波模块中，我们适当地增加了片上存储空间，并设计了存储策略，使本模块在能够完成环路滤波处理的同时占用最小的带宽(只向总线输出数据，不请求数据)。采用的环路滤波结构如图 3 所示。

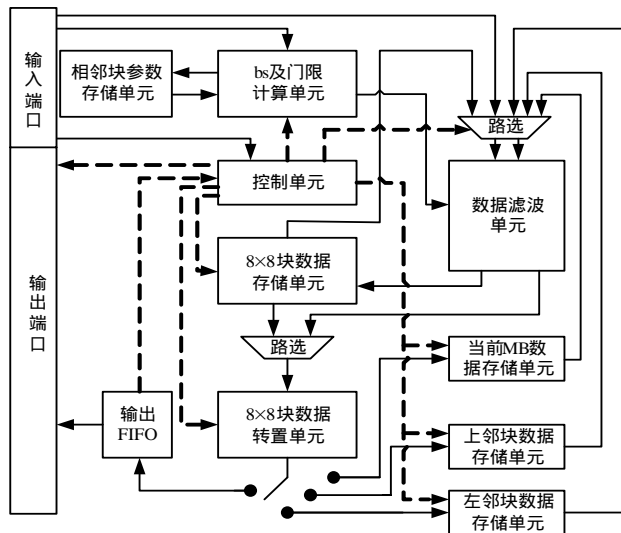


图 3 整体框图

### 2.2 模块描述

#### (1)控制单元

采用了一个大的有限状态机来严格控制滤波的流程。在不影响滤波结果的前提下，适当调整滤波的顺序(如图 1 所示)，先处理 h1h2h3h4h5h6，再处理 v1v2v3v4v5v6。这样处理顺序可与宏块的扫描顺序一致，降低了复杂度。

#### (2)相邻参数存储单元

此模块中存储着计算边界 bs 以及门限的相邻宏块的参数。例如宏块类型，运动矢量等。

#### (3)bs 及门限计算单元

根据参数存储单元提供的参数计算出当前宏块环路滤波所需的 bs 及门限值。

#### (4)数据滤波单元

根据上面模块提供的参数及控制信息对输入的图像像素数据进行滤波计算。每个端口都是 8 像素 64 比特数据。参与计算的是图 2 中边界两边实线框内 6 个点的像素值。

#### (5)8x8 块数据存储单元

此单元用来对滤波的中间结果进行缓存，缩短随后边界滤波中数据的准备时间。此模块很大程度上加快了滤波速度。

#### (6)8x8 块数据转置单元

按照宏块的扫描顺序，可以使垂直边界的水平滤波顺利进行，但为了对水平边界进行垂直滤波，就需要对已经完成水平滤波的 8x8 块转置。此模块内包含一个 8x8x8 比特的寄存器堆，根据块的具体情况选择正确的输出方式(转置或不转置)和输出路径。由于数据处理的特殊性，这里采用“列进行出”和“行进行出”的方式。描述如图 4 所示。

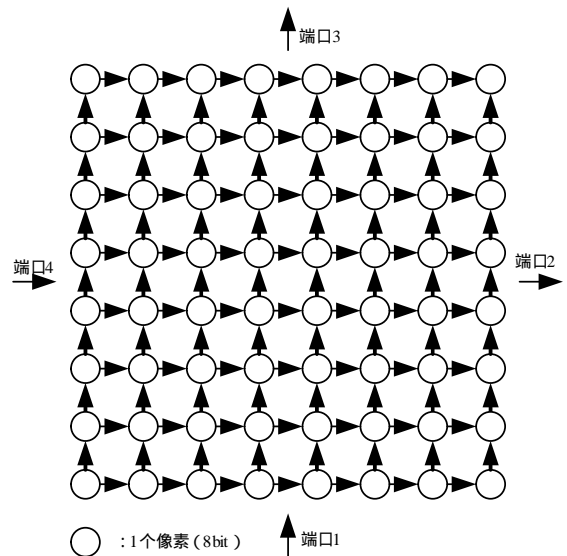


图 4 8x8x8 比特的寄存器堆

如果块的每行数据(8 像素)从端口 1 输入，经过 8 个时钟，8x8 块的数据全部进入矩阵，然后用 8 个时钟从端口 2 把数据取出，这样数据就完成了转置。当端口 2 输出一行数据后，下个块的数据就可以从端口 4 输入了，全部充满矩阵后由端口 3 输出。同样，从端口 3 输出数据的同时，端口 1 可以输入数据。这种输入输出的方式也很大程度上缩短了数据处理时间。需要注意的是，在不同的滤波阶段，寄存器中的块数据会改变输出方式和路径，需要及时地调整输入输出端口，保证数据的正确。

宏块中所有的 8x8 块数据在处理过程中都会被进行 2 次转置操作，只是由于滤波边界的位置而导致转置的时机不同。正确地对相应的 8x8 块数据进行转置操作，是整个环路滤波处理过程高效运行的关键。

#### (7)当前 MB 数据存储单元

此模块用于存放已经完成垂直边滤波并经过 1 次转置的宏块数据。存储器容量为 48x64 比特(按照 4:2:0 格式)。如果采用亮度色度复用，此处容量可减小到 32x64 比特。

#### (8)上邻块数据存储单元与左邻块数据存储单元

在这两个存储单元里存放了当前宏块环路滤波将会涉及到的像素数据。其中上邻块数据存储单元所占的片内存储容

量最大,共需要  $3\ 840 \times 24$  比特的片上 RAM(可用于  $1\ 920 \times 1\ 080$  的 HDTV)。左邻块需要空间为  $32 \times 64$  比特。总共需要 11.5KB 存储空间。

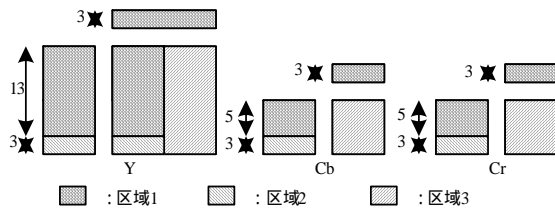


图 5 宏块数据的不同后处理区域

当一个宏块的环路滤波结束后,图 5 中不同区域的数据要分别进行不同的后处理。区域 1 中的数据按照扫描顺序直接由  $8 \times 8$  块数据转置单元输出到输出缓存,区域 2 中的数据被转置后送到上邻块数据存储单元存储,区域 3 中的数据按照扫描顺序送到左邻块数据存储单元存储。

#### (9)输出缓存

此模块为输出的数据计算输出地址,并包含一个 FIFO。为了更好地与片外存储器进行数据交互,这里把滤波完的数据先存放在 FIFO 中,FIFO 的大小为  $16 \times 64$  比特。输出缓存中有数据后,就向外部存储器发出写请求,待响应后开始把 FIFO 中的数据输出。当 FIFO 满的时候,FIFO 发出信号给控制单元,控制单元就中断整个滤波过程。但是为了保证每个块的数据在处理过程中不会因为随机性的中断而不连续,这里对 FIFO 采用一个“预满”机制的满信号。为 FIFO 设置一个门限,当 FIFO 中数据数量超过门限值,FIFO 就发出满信号给控制单元,控制单元根据当前的滤波状态开始准备中断,等到一个完整的块进入滤波器模块后,中断才真正开始。中断时所有已经处于处理过程中的数据仍可以进入 FIFO,这样保证了块数据的连续性。

### 2.3 工作流程

当前一级把当前宏块数据(亮度和色度)以及相应参数准备好时,控制单元便发出  $bs$  及门限计算单元的开始信号。待  $bs$  及滤波门限生成后,控制单元进入对图 1 中垂直边的滤波控制状态。滤波器的路选器在控制单元的控制下选通正确的两条数据输入路径,从数据滤波单元输出的两路数据,一路数据(图 2 中  $q$  数据)将被输入到  $8 \times 8$  数据存储单元,该数据在下一条边界滤波开始时,经过滤波器的路选进入滤波单元。在垂直边界滤波时 1、3、4 和 5 块数据以及水平边界滤波时的 2、3、4 和 5 块数据将直接由  $8 \times 8$  数据存储单元输出到  $8 \times 8$  数据转置单元;另一路数据(图 2 中  $p$  数据)将被输入到  $8 \times 8$  数据转置单元,再根据当前滤波状态为数据选择正确的

输出路径。当垂直边  $h6$  滤波结束,控制单元会向前级模块发出结束信号。此时的结束并不是真正的结束,只是说明不再需要前级提供数据。而后环路滤波进入水平边界滤波阶段,当最后一条边  $v6$  滤波结束,表明当前宏块环路滤波的结束。控制单元中状态回到开始的等待状态。

### 3 仿真结果

仿真环境见表 1。

表 1 仿真环境

编程	Verilog 语言 RTL 级描述
仿真软件	Modelsim6.0
测试码流	AVS rm52c 软件中提取的 QCIF 测试数据

经过对仿真波形的观察,完成一个宏块只需要 293 个时钟。此时钟是不计由于总线对请求的判决而需要等待的时钟数以及由于前级模块未准备好数据而需要等待的时钟数的。假设 AVS 系统频率为 108MHz,要达到对  $1\ 920 \times 1\ 080$  高清图像的实时性解码,AVS 系统对每个宏块(包含色度)的处理时钟数上限设置为 441 个时钟。可看出本结构满足了时钟周期上可实时性的要求。仿真结果对比如图 6 所示。



图 6 仿真结果对比

### 4 结论

本文设计了一种高效的环路滤波处理结构,此结构可被用在实时高清解码的 AVS 芯片中。设计时充分考虑了影响芯片实时解码的两个主要因素:总线带宽和数据处理速度。并从本模块角度最大程度改善了以上两个因素对解码的影响。

#### 参考文献

- 1 数字视音频编解码技术标准化工作组. 信息技术先进音视频编码第 2 部分: 视频(草案)[S]. 2004. <http://www.avs.org.cn>.
- 2 Chang S C, Peng W H, Wang S H, et al. A Platform Based Bus-interleaved Architecture for Deblocking Filter in H.264/MPEG-4 AVC[C]//Proc. of IEEE Intl. Conf. on Consumer Electronics. 2005.
- 3 ITU-T Rec. H.264/ISO/IEC 14496-10 AVC. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification[S]. 2003-03.
- 4 中国科学院计算技术研究所. AVS 参考软件 rm52c[EB/OL]. 2003. <http://www.avs.org.cn/>.

(上接第 202 页)

#### 参考文献

- 1 Qi Yaohu. The Theory Study of  $C^3$  Countermeasure Decision-making Supporting System[C]//Proceedings of CIE International Conference of Rada. 1996.
- 2 梁百川.  $C^3$ CM 决策支持系统理论研究[J]. 系统工程理论与实践, 1996, 16(8): 21-25.

- 3 程 岩. 数据库中挖掘决策偏好信息的粗糙集方法研究[J]. 计算机工程, 2003, 29(6): 14-16.
- 4 Shinichi O, Hideyuki I. Evaluation for a Stability of Fuzzy Decision Making Using a Sensitivity Analysis[C]//Proc. of NAFIPS Conference of the North American. 1998.
- 5 曲长文, 何 友, 马 强. 应用多属性决策的威胁估计方法[J]. 系统工程与电子技术, 2000, 22(5).