

# 比特平面并行的 EBCOT 编码及其 VLSI 结构

刘 凯<sup>1),2)</sup> 吴成柯<sup>1)</sup> 李云松<sup>1)</sup> 庄怀宇<sup>1)</sup>

<sup>1)</sup>(西安电子科技大学 ISN 国家重点实验室 西安 710071)

<sup>2)</sup>(西安电子科技大学计算机学院 西安 710071)

**摘 要** 该文提出了比特平面并行处理的 EBCOT 编码算法. 通过分析 JPEG2000 中 EBCOT 编码结构, 指出每一个比特平面的编码信息可以同时获得, 从而给出了比特平面并行处理的块编码方法, 并且详细说明了实现的 VLSI 结构. 与现有的结构相比, 该结构具有并行度高、避免编码位置的时钟浪费等特点. 从实验结果表明, 比特平面并行处理方式所需的时钟周期最少, FPGA 原型系统最高时钟频率可达 52MHz, 图像质量达到了公布的 JPEG2000 标准.

**关键词** JPEG2000; EBCOT 算法; 比特平面并行; 块编码; 上下文模型  
**中图法分类号** TP391

## Bit Plane-Parallel Coder for EBCOT and Its VLSI Architecture

LIU Kai<sup>1),2)</sup> WU Cheng-Ke<sup>1)</sup> LI Yun-Song<sup>1)</sup> ZHUANG Huai-Yu<sup>1)</sup>

<sup>1)</sup>(National Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071)

<sup>2)</sup>(School of Computer, Xidian University, Xi'an 710071)

**Abstract** This paper proposes an efficient architecture composed of bit plane-parallel coder for EBCOT entropy encoder used in JPEG2000. After the detailed analysis of EBCOT architecture in JPEG2000, the coding information of each bit plane can be obtained simultaneously. Therefore, bit plane-parallel coding method is proposed, and its VLSI architecture is shown in detail. Compared with other architectures, it has advantages of high parallelism, and no waste clock cycles for single point. The experimental results show that the proposed architecture reduces the processing time greatly compared with others, and a FPGA prototype chip is designed and works at 52MHz at most. The quality of images reaches the results released by JPEG2000.

**Keywords** JPEG2000; EBCOT; bit plane-parallel; block encoder; context model

## 1 引 言

随着多媒体和网络技术的发展和應用, JPEG 已不能满足当前市场和实际应用的要求. 从 1997 年 5 月开始, 国际上为制定一种静止图像压缩的新标准——JPEG2000 提出各种建议. JTC1/SC29(信息

技术附属委员会)/WG1(静止图像工作组)进行了标准化工作并整理此标准, 于 2000 年 8 月推出了国际标准的最后草案(FDIS), 这个草案从 2000 年 11 月起正式成为国际标准 IS(15444-1). JPEG2000<sup>[1]</sup> 标准支持单分量或多分量的有损和无损编码, 还同时支持 SNR 和分辨率渐进传输、感兴趣区(ROI)编码、码流随机访问, 提供灵活的文件格式, 支持内置

收稿日期: 2003-04-28; 修改稿收到日期: 2004-05-07. 本课题得到国家科技预研项目基金(J20. 12. 1. DZ01)资助. 刘 凯, 男, 1977 年生, 博士研究生, 助教, 研究方向为计算机图形学、图像编码. E-mail: kailiu@mail. xidian. edu. cn. 吴成柯, 男, 1938 年生, 教授, 博士生导师, 研究方向为图像通信、图形学、计算机视觉和模式识别的科研与教学工作. 李云松, 男, 1974 年生, 博士, 讲师, 研究方向为图像编码、计算机视觉. 庄怀宇, 男, 1980 年生, 博士研究生, 研究方向为图像压缩编码、VLSI 结构设计.

用户信息(如版权信息)、图像序列(Motion JPEG), 并具有良好的抗误码特性. 因此, 在图像检索、因特网传输、网络浏览、文本图像、数码相机、医学图像、遥感图像和桌面印刷等多个领域具有巨大的应用价值.

JPEG2000 的编码系统基本结构<sup>[2]</sup>如图 1 所

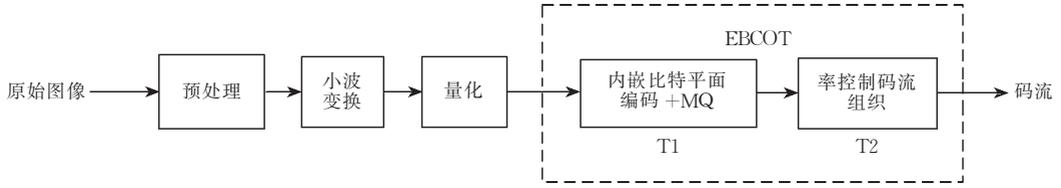


图 1 JPEG2000 的编码系统基本结构框图

可见, JPEG2000 改变了传统 JPEG 标准以 DCT 变换为核心的变换方法, 采用了具有能量特性更为集中的小波变换方法以及率失真优化截取的内嵌码块编码(Embedded Block Coding with Optimized Truncation, EBCOT)算法<sup>[3~5]</sup>. 其中 EBCOT 的硬件实现成为 JPEG2000 硬件化的核心问题, 为此不少学者致力于 EBCOT 硬件结构的研究, 而且提出了一些结构, 如 Andra<sup>[6,7]</sup>提出状态机结构的比特平面编码, Chiang<sup>[8]</sup>提出编码过程并行(coding pass parallel)结构以及 Lian<sup>[9]</sup>提出基于列(column-based)结构等. 通过对这些结构仔细分析后, 我们提出了比特平面并行的上下文模型(bit plane-parallel context modeling)硬件结构, 该结构充分利用了比特平面间的并行度, 编码模块完全由组合逻辑实现, 克服了状态机控制的复杂性, 而且不会在单像素位置产生时钟浪费, 因此编码速度和效率有显著改善. 接下来, 首先给出 EBCOT 码块编码的具体分析, 然后详细介绍比特平面并行处理的 EBCOT 编码结构, 最后给出相应的实验结果.

## 2 EBCOT 码块编码分析

EBCOT 算法分为两部分: T1 和 T2. T1 由内嵌比特平面编码和自适应算术编码器 MQ 组成, 而 T2 部分完成率控制和码流组织. EBCOT 编码时, 各小波子带划分为更小的码块(如  $64 \times 64$ ), 以码块(code-block)为单位独立作 T1 编码. 不同的码块产生的比特流长度是不相同的, 它们对恢复图像质量的贡献也是不同的. 因此对于所有码块产生的比特流, T2 部分采用了率失真优化技术进行后压缩处理(Post Compression Rate Distortion, PCR D), 即对各码块的码流按对恢复图像的质量贡献分层, 完成码流的率控制和组织.

示. 首先对原始图像数据进行离散小波变换, 然后在形成输出码流(比特流)之前, 对变换系数进行量化和熵编码. 压缩图像数据(即码流)通过存储或传输后, 进行熵解码、反量化和逆小波变换, 从而恢复图像数据.

### 2.1 码块比特平面编码

设码块  $B_i$  中的量化系数为  $q_i[m, n]$ , 以符号-幅值的形式表示,  $\Delta_i$  为相应的量化步长,  $M_i$  为  $q_i[m, n]$  的比特平面个数,  $\chi_i[m, n]$  为符号位(0 表示正数, 1 表示负数),  $v_i[m, n]$  为  $M_i$ -比特的幅值. 令  $v_i^p[m, n]$  为  $v_i[m, n]$  第  $p$  层的比特值, 其中  $0 \leq p < M_i$  且  $p=0$  对应于最低比特平面 LSB. 码块中的每个系数有一个二进制状态变量  $\sigma_i[m, n]$  指示该系数是否重要, 初始化为 0, 表示当前系数是不重要的, 在对第一个非零比特  $v_i^p[m, n] \neq 0$  编码时变为 1, 表示当前系数是重要的,  $\sigma_i[m, n]$  称为系数  $s_i[m, n]$  的“重要性状态”. 比特平面编码就是首先从最高比特平面 MSB 开始利用当前系数  $s_i[m, n]$  和相邻样点以前的编码信息(上下文)针对幅值位  $v_i^{M_i-1}[m, n]$  采用不同的编码原语(coding primitives)进行编码, 将编码结果输出给自适应算术编码器 MQ 进行压缩, 然后依次进行下一个比特平面编码, 直至最低比特平面 LSB 完成相应编码. 这里编码原语的选择和上下文的形成是码块编码的关键, 下面对此进行描述.

### 2.2 编码原语及上下文模型

对码块  $B_i$  中的系数  $s_i[m, n]$  在比特平面  $p$  编码时, JPEG2000 使用了 4 种编码原语: “零编码”(ZC)、“符号编码”(SC)、“幅值细化”(MR)和“游程编码”(RLC), 原语的选择是根据系数  $s_i[m, n]$  所处的编码阶段以及和其直接相邻的 8 个系数状态来决定, 见图 2.

$D_0$	$V_0$	$D_1$
$H_0$	$X$	$H_1$
$D_2$	$V_1$	$D_3$

图 2 用来形成上下文的 8 个邻域系数



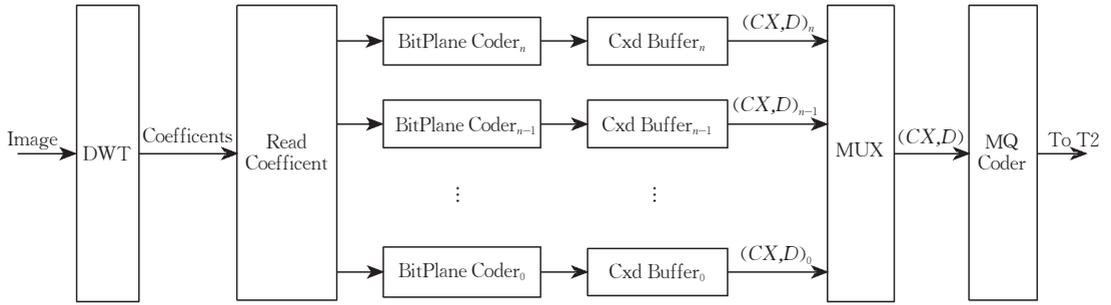


图 5 比特平面并行的编码系统结构

原始图像首先进行离散小波变换 (discrete wavelet transform) 得到频率域的小波系数, 这里我们选择 JPEG2000 建议的整数  $5/3^{[10]}$  小波, 小波系数的量化步长为 1, 小波级数为 3. 接下来, 读系数模块 (read coefficients) 根据分辨率渐进要求, 将小波系数按级数和频带读出送给各个比特平面编码器, 具体为 (LL HL LH HH) $_n$  ( $n=0\sim 2$ ) 其中 0 对应高频子带, 2 对应低频子带. 同时, 根据小波系数在各个比特平面的值, 组织该系数在对应平面的状态变量值, 并将值存储到对应的 RAM 中. 每一级的比特平面编码器接收小波系数和状态变量在该平面的值, 按照算法的编码要求生成对应的上下文和数据信息. 生成的上下文和数据对 (CX, D) 存放到每个平面对应的上下文缓冲 (Cxd Buffer) 中, 各个平面产生的上下文和数据对通过多路选择器 (MUX) 输出到算术编码器 (MQ) 进行压缩编码. MQ 产生的压缩码流提供给 T2 编码模块进行优化截取, 最终形成标准码流.

### 3.2 状态变量组织

小波系数在每一个编码平面都包含 5 个状态变量, 它们分别是重要性状态位 ( $\sigma$ ) 用以表示系数的重要性状态, 符号位 ( $\chi$ ) 用以表示系数的符号状态, 访问位 ( $\eta$ ) 用以表示系数在当前比特平面是否编码过状态 (当该系数在该比特平面某个编码过程中进行了编码后置为 '1', 否则为 '0') 和细化状态位 ( $\sigma'$ ) 用以表示系数是否进行了 MR 编码, 以及 1 位用于表示系数在该比特平面的幅值位 ( $\nu$ ) 状态. 图 6 给出了上下文处理时条带的位置.

从图 6 中可见, 在处理每个条带时, 上下文窗 (context window) 不仅要考虑当前处理的 4 个位置 (图中 7, 8, 9 和 10), 还要考虑左右相邻的条带位置以及上下边界的两个位置 (图中 6 和 11). 由于扫描顺序是以条带为单位从左向右, 从上向下, 所以可以将此 6 个点视为基本处理单元, 表示为一个条带 (strip), 这样可以避免条带内某些位置不产生上下文输出时带来的时钟浪费. 那么, 一个条带的位置个

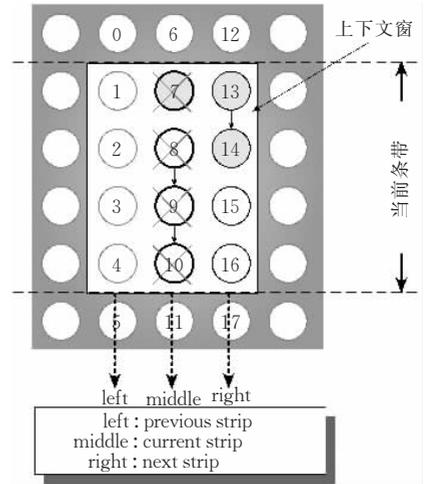


图 6 条带处理位置

数应为  $4+2$  (包含边界), 状态变量的位数为 30 位.

为了达到比特平面并行, 就必须在开始编码时各个比特平面得到关于编码位置的正确状态变量值. 然而, 在 4 个状态变量中只有重要性状态和细化状态是与比特平面位置有关的, 当然各个比特平面的幅值是不相同的, 但幅值位是不会改变的, 所以不用考虑幅值位对并行编码的影响. 对  $p^{\text{th}}$  平面位置  $[m, n]$  处的重要性状态为该平面以上各平面对应位置幅值位的或运算, 而细化位则是  $p+1^{\text{th}}$  平面位置  $[m, n]$  处的重要性状态. 即

$$\sigma_p = \sum_{i=p+1}^{\text{MSB}} \nu_i[m, n], \sigma'_p = \sigma_{p+1}, p \text{ 指平面号}$$

其余的状态位可以直接从小波系数获得. 至此, 每一个比特平面的状态信息可以同时获得, 这样就保证了比特平面并行编码的可行性. 对于每一个条带的状态信息可以组织为一个 30 位宽的寄存器, 分别存放 6 个点的状态信息, 具体可见表 1.

表 1 状态位寄存器

8					7					6				
$\sigma'$	$\sigma$	$\chi$	$\eta$	$\nu$	$\sigma'$	$\sigma$	$\chi$	$\eta$	$\nu$	$\sigma'$	$\sigma$	$\chi$	$\eta$	$\nu$
11					10					9				
$\sigma'$	$\sigma$	$\chi$	$\eta$	$\nu$	$\sigma'$	$\sigma$	$\chi$	$\eta$	$\nu$	$\sigma'$	$\sigma$	$\chi$	$\eta$	$\nu$

表 1 中第 1 行为图 6 中当前处理条带的位置标号,第 2 行为对应位置的状态信息位。

### 3.3 比特平面编码器

根据上下文产生规则和状态位的设置,图 7 给出具体的比特平面编码器的硬件结构。

图 7 中 state ram 存放一个码块所有条带的状态信息,而 left,middle 和 right 分别对应图 6 中的 3 个条带的 30 位状态寄存器,其中 middle 为当前要处理的条带,left-en,mid-en 和 right-en 是这 3 个寄存器的打入使能信号.Context-generate 则是根据 JPEG2000 标准中给出的产生上下文和数据规则利用卡诺图化简后得到的组合逻辑门阵列,是整个编码器的核心模块.Read-context 将 context-generate 生成的上下文和数据读出写入到上下文缓冲中(cxd\_fifo)。Control 模块是整个编码器的控制模

块,主要是产生对 state ram 和 3 个条带寄存器的控制信号.Stage\_cnt 是编码过程的计数模块,产生指示编码过程信号,即当前所处的编码过程(重要性传播、幅值细化和清理更新)。编码器接收离散小波变换(DWT)模块来的启动信号(start),stage\_cnt 指示为重要性传播,并对 control 模块复位开始编码,control 模块将 state ram 中的条带信息依次读出写入到 3 个状态寄存器,同时从组合逻辑阵列产生上下文和数据信息,这时通过读上下文和数据模块将其写入缓冲中完成一个编码过程,然后重复上述过程,直到一个码块的所有条带完成该编码过程。接着是对该码块的下一个编码过程进行编码,stage\_cnt 指示为幅值细化,最后是对码块进行清理更新。至此一个码块编码结束,进行下一个码块编码,直到所有码块编码结束。

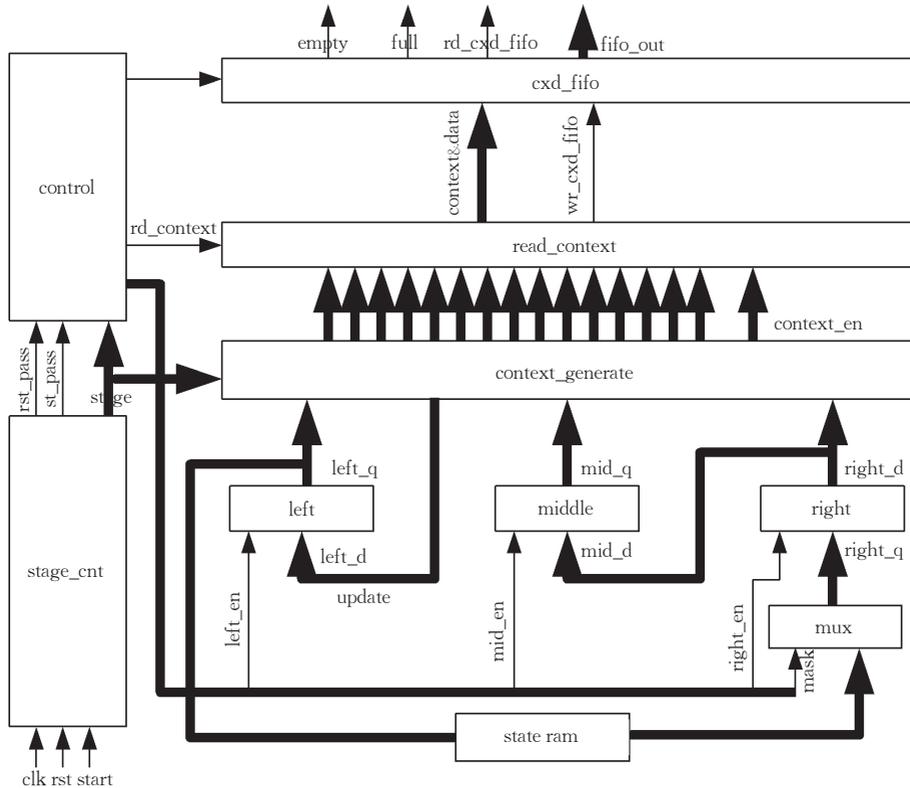


图 7 比特平面编码器结构

### 3.4 条带扫描和上下文生成

JPEG2000 提供了两种可选的条带扫描顺序:Regular 和 Vertical Causal.采用 Regular 方式的条带扫描顺序,如图 8 所示。

设码块大小为  $32 \times 32$ ,共 256 个条带,上下文数据生成是根据 3 个状态寄存器(left, middle 和 right)的值,以组合逻辑形式同时产生 4 个位置的数据。虽然从图 8 中可见一个条带的 4 个位置是依

次处理的,但是每一个位置产生的上下文是完全可以由组合逻辑同时预知的,也就是在一个时钟内该条带应生成的所有上下文数据可以同时获得。对于每一个条带在编码时,最多可以生成的上下文数为 15,它们分别是:位置 7 的 ZC,SC 和 MR;位置 8 的 ZC,SC 和 MR;位置 9 的 ZC,SC 和 MR;位置 10 的 ZC,SC 和 MR 以及 RLC 的 3 个上下文。这些上下文数据可以通过条带的状态值得到,并且可以设

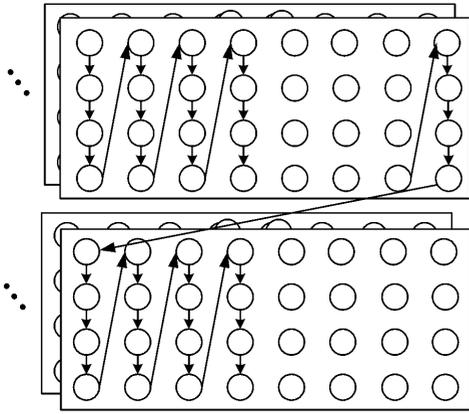
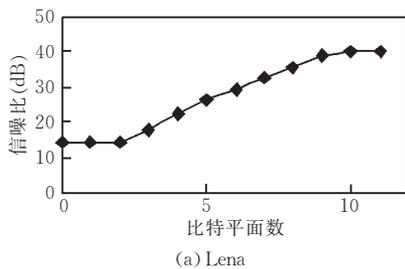
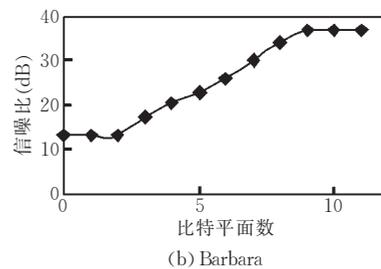


图 8 Regular 方式的条带扫描顺序

定上下文有效信号来指定该上下文是否有效以供输出时使用. 例如, 位置 7 的 ZC 上下文由位置 0, 1, 2, 6, 8 和 11, 12, 13 的状态变量根据标准规定来决定, 而这 8 个相邻位置的状态信息则可在 3 个状态寄存器中得到. 这样就可以使条带内上下文并行输出, 很大程度上节省了时钟. 读上下文模块就是结合这 15 个上下文的有效信号将该条带生成的有效上下文依次输出.



(a) Lena



(b) Barbara

图 9 Lena &amp; Barbara 比特平面数-峰值信噪比(比特率 1.0bpp)

可见曲线两端是平坦的, 中间部分有下降, 这就意味着没有必要设置最大的比特平面数来获取最高的 PSNR. 在硬件资源的约束下, 一般选择中间偏上点作为实际并行编码的平面数, 例如范围为 8~12 之间, 对于比特平面数为 0, 1 和 2 时, PSNR 值非常接近是由于参与编码信息太少, 恢复图像基本上为噪声已经没有意义了.

### 3.7 系统时钟和资源

采用 VHDL 语言, 我们设计并实现了比特平面并行的编码算法. 表 2 给出了具体的实现参数.

表 2 FPGA 芯片说明

目标器件	XILINX XCV600e-6BG432
编程语言	VHDL
综合工具	XILINX ISE5.1
时钟频率	52.247MHz
等效门计数	1,028,182
比特平面数	10

### 3.5 状态变量更新

由于对每一个条带处理时要用到最新的邻近条带状态变量, 这样才能满足 3 个编码过程的要求, 状态变量的及时更新就非常必要. 这种更新具体就是对 state ram 的回写过程, 从图 7 见, Left 寄存器的输出是连接到 state ram 的输入端, 由此来完成对状态变量的更新过程. 那么, 在后续的编码过程中所使用的状态变量就是最新的, 以确保编码的正确.

### 3.6 比特平面数的选择

经过小波变换后, 小波系数的位宽就是决定参加并行编码平面个数的主要因素. 对于基于提升的整数 5/3 小波, 一般小波系数可以用  $12+1$  位源码表示, 其中包括一位符号位. 选择多少个比特平面进行并行编码是图像质量和资源的折衷. 根据 JPEG2000 量化方法, 可以通过设置量化死区 (deadzone) 来调整参与并行编码的比特平面数, 那么必须得到比特平面数与图像质量之间的关系以确定所需的比特平面数目. 图 9 分别给出了用软件得到的 Lena 和 Barbara 的比特平面数与峰值信噪比 (PSNR) 曲线.

## 4 实验结果

通过仿真, 我们得到了几幅测试图像实际所需的编码时钟周期数, 并且和串行方式、编码过程并行方式以及基于列方式进行了比较, 结果见表 3.

表 3 测试图像执行时间

图像 (512×512, gray-scale)	执行时间 (Clock Cycles)			
	串行方式	PASS 并行方式	基于列方式	比特平面 并行方式
Lena	4,164,352	1,312,688	1,743,283	1,116,638
Jet	4,304,128	1,433,483	1,755,450	1,106,172
Pepper	4,550,656	—	1,880,388	1,119,712
Baboon	4,947,712	1,748,956	2,106,820	1,113,828

从表 3 可见, 比特平面并行方式所需的时钟周期数最少. 这一点可以通过如下分析得到: 假设对于

一幅图像,采用串行方式时所需的时钟周期为  $Clock_s$ ,采用编码过程并行(PASS-parallel)方式时所需的时钟周期为  $Clock_p$ ,基于列(column-based)方式时所需的时钟周期为  $Clock_c$ ,比特平面并行(bitplane-parallel)方式时所需的时钟周期为  $Clock_b$ ,编码平面数为  $N$ ,那么理论上  $Clock_b = \frac{1}{N}Clock_s$ ,而由于每次编码最多有 3 个编码过程,那么编码过程并行(PASS-parallel)方式的并行度为 3,即  $Clock_p = \frac{1}{3}Clock_s$ . 然而一般情况下参与编码的比特平面数  $N > 3$ ,所以比特平面并行方式具有更高的并行度. 从上表的实验结果也说明了比特平面并行方式所需的时钟周期最少,编码过程并行(PASS-parallel)方式和基于列(column-based)方式由于采用了其他加速方法如样点跳过(Sample Skipping, SS)、列跳过(Group-of-Column Skipping, GOCS)和多列跳过(Multiple Column Skipping, MCOLS)使得结果非常接近. 但是从并行角度来说,比特平面并行方式的并行度最高.

表 4 给出采用比特平面并行结构对几幅测试图像在不同码率下得到的结果,为了便于比较均采用 9/7 小波滤波器,码块大小为  $64 \times 64$ .

表 4 PSNR(dB)结果

图像	比特率(bpp)	SPIHT[11]	JPEG2000	JPEG2000-BP
Lena (512×512)	1.0	40.41	40.61	40.05
	0.5	37.21	37.43	37.14
	0.25	34.11	34.28	34.10
	0.125	31.10	31.22	31.05
Barbara (512×512)	1.0	36.41	37.37	36.94
	0.5	31.39	32.48	32.28
	0.25	27.58	28.55	28.39
	0.125	24.86	25.55	25.30
Airport (1024×1024)	1.0	—	33.23	33.09
	0.5	—	30.42	30.40
	0.25	—	28.27	28.25
	0.125	—	26.44	26.44
	0.0625	—	24.70	24.62

其中 JPEG2000-BP 指比特平面并行算法, SPIHT 和 JPEG2000 的结果来自文献[3]. 从表中可见,尽管本文的方法在速度上提高了约一个数量级,而在 PSNR 性能上基本和已有的 JPEG2000 以及 SPIHT 相当.

## 5 结 论

传统的 JPEG2000 块编码方法是比特平面串行

的,这种方法速度慢,存在当单点不产生上下文时会有时钟浪费现象. 本文提出的比特平面并行方法大大提高了系统处理速度,而且避免了像素位置上的时钟浪费. 从实验结果看,作为硬件实现方案,效果可以满足大多数应用要求. 而且,整个系统已经通过了 FPGA 验证,最高工作频率可达 52MHz. 通过分析以及实验结果表明,比特平面并行方式在已有的实现方案中并行度上是最高的.

## 参 考 文 献

- 1 JPEG 2000 Part I Final Committee Draft Version 1. 0. ISO/IEC JTC1/SC29/WG1 N1646R, March 2000
- 2 Christopoulos C., Skodras A., Ebrahimi T.. The JPEG2000 still image coding system: An overview. IEEE Transactions on Consumer Electronics, 2000, 46(4):1103~1127
- 3 Taubman D.. High performance scalable image compression with EBCOT. IEEE Transactions on Image Processing, 2000, 9(7): 1158~1170
- 4 Taubman D., Ordentlich E., Weinberger M., Seroussi G.. Embedded block coding in JPEG2000. HP Labs, Palo Alto; Technical Report HPL-2001-35, 2001
- 5 Taubman D., Ordentlich E., Weinberger M., Seroussi G., Ueno I., Ono F.. Embedded block coding in JPEG2000. In: Proceedings of the IEEE International Conference Image Processing, Vancouver, Canada, 2000, 2: 33~36
- 6 Andra K., Acharya T., Chakrabarti C.. Efficient VLSI implementation of bit plane coder of JPEG2000. In: Proceedings of SPIE Applications of Digital Image Processing, San Diego, CA, USA, 2001, 4472:246~257
- 7 Andra K., Chakrabarti C., Acharya T.. A high-performance JPEG2000 architecture. IEEE Transactions on Circuits and System for Video Technology, 2003, 13(3): 209~218
- 8 Chiang J. S., Lin Y. S., Hsieh C. Y.. Efficient pass-parallel architecture for EBCOT in JPEG 2000. In: Proceedings of IEEE International Symposium Circuits and Systems, Scottsdale, Arizona, USA, 2002, 1: 773~776
- 9 Lian Chung-Jr., Chen Kuan-Fu, Chen Hong-Hui, Chen Liang-Gee. Analysis and architecture design of block-coding engine for EBCOT in JPEG2000. IEEE Transactions on Circuits and System for Video Technology, 2003, 13(3): 219~230
- 10 Adams M. D., Kossentini F.. Reversible integer to integer wavelet transform for image compression: Performance evaluation and analysis. IEEE Transactions on Image Processing, 2000, 9(6):1010~1024
- 11 Said A., Pearlman W. A.. A new, fast, and efficient image coder based on set partitioning in hierarching trees. IEEE Transactions on Circuits and System for Video Technology, 1996, 6(3): 243~249



**LIU Kai**, born in 1977, Ph. D. candidate. His major research interests include computer graphics and image coding.

visor. His major research interests include image communication, computer graphics and vision.

**LI Yun-Song**, born in 1974, Ph. D. , lecturer. His major research interests include image coding and computer vision.

**ZHUANG Huai-Yu**, born in 1980, Ph. D. candidate. His major research interests include image compression and VLSI chip design.

**WU Cheng-Ke**, born in 1938, professor, Ph. D. super-

## Background

This work is supported in part by the National Research Foundation under grant numbers J20. 12. 1. DZ01. As EBCOT coding is a bottleneck of JPEG2000 in image coding, this project aims to find out parallelism in EBCOT and design an efficient architecture in VLSI implementation. Since 1999

authors have been working on still image compression, during these years we have designed SPIHT chip for real time image processing, now they begin to design JPEG2000 chip for commercial applications.