

# 一种支持 H.264 High Profile 的高效可重构反变换 VLSI 结构

陆晓凤<sup>1,2</sup>, 刘 锋<sup>1</sup>, 佟 冬<sup>1</sup>, 王克义<sup>1</sup>

(1. 北京大学微处理器研究开发中心, 北京 100871; 2. 北京大学深圳研究生院, 广东深圳 518055)

**摘 要:** 本文针对 H.264 Fidelity Range Extensions(FRExt, High Profile)解码过程中扩展的所有变换, 采用二维矩阵分解和基于矩阵运算提取公共因子的操作, 利用通用运算单元来设计高效的不可重构 VLSI 结构. 该结构不但节省面积(可重构变换结构只消耗了 4807 门电路), 并且具有高性能(采用 TSMC 0.13 微米工艺, 在 150MHz 的时钟频率下工作, 该电路可以达到 1.2G pixels/s 的吞吐率, 支持分辨率为 4096 × 2048、每秒 60 帧的视频实时解码需求). 根据 DTUA (吞吐率和电路规模的比值)作为效率因子, 本设计在支持 H.264 High Profile 所有变换的设计中具有最高的效率.

**关键词:** 整数变换; High Profile; H. 264

**中图分类号:** TP302 **文献标识码:** A **文章编号:** 0372-2112 (2011) 05-1072-05

## An Efficient Reconfigurable VLSI Architecture for H.264 High Profile Inverse Transform

LU Xiao-feng<sup>1,2</sup>, LIU Feng<sup>1</sup>, TONG Dong<sup>1</sup>, WANG Ke-yi<sup>1</sup>

(1. Microprocessor Research and Development Center, Peking University, Beijing 100871, China;

2. ShenZhen Graduate School, Peking University, Shenzhen, Guangdong 518055, China)

**Abstract:** H.264 High Profile proposes the new 8 × 8 integer transform which achieves average bit-rate reduction of around 10% for HD material. This paper presents an efficient architecture for H.264 multiple block-size inverse transform. 2-D transform decomposition and common matrix operation extraction were applied to design general processing elements to simplify the implementation of transforms. The proposed architecture is both area-efficient (4807 gates) and high performance (can provide real-time decoding of 4096 × 2048@60fps High Profile in 150MHz, 1.2G pixels/s data throughput rate, using TSMC 0.13 μm). This paper supporting all of the H.264 high profile transforms is the most efficient one, compared with the existing designs with the Data Throughput rate per Unit Area (DTUA) adopted as the comparison index.

**Key words:** integer transform; high profile; H.264

### 1 引言

JVT(Joint Video Team)提出的 H.264/AVC 是目前最新的视频编解码标准, 和以前的标准例如 MPEG-2 相比, H.264 的压缩比能够减低超过 50%, 而 H.264 的高压缩率也使得其解码器的复杂度是 MPEG-2 的 4 倍, MPEG-4 visual simple profile (VSP)的 2 倍. 算法的高复杂度导致使用硬件实现其核心模块来加速编解码系统数据速度的必要性.

以往标准中直接采用离散余弦变换(DCT)的定义进行变换, 有如下问题: ①由于变换核都是无理数, 而有限精度的浮点数不可能准确地表示无理数, 再加上浮点数的运算可能会引入舍入误差, 使得编解码出现不匹

配; ②需要进行浮点数操作, 造成计算和硬件实现上的复杂性. 为了克服这些问题, H.264 采用了基于 DCT 的整数变换. 同时由于自适应块大小的概念已经被证明是 H.264 标准中一个有效的编码工具, High Profile 中引入了新的 8 × 8 整数变换(和预测模式). 实验结果显示 High Profile 中引入的 8 × 8 整数变换使得逐行扫描高清视频的平均码率降低近 10%, 这也表明为了支持 High Profile 高清视频的应用, 变换模块的设计要同时灵活地支持 8 × 8 和 4 × 4 变换.

虽然较以往标准采用的 DCT 计算量有所减少, H.264 标准中采用的整数变换仍然是 H.264 编解码系统中计算量较大的关键部分. 例如, 每一个 8 × 8 变换需要 64 个加法和 20 个移位操作, 为了实时处理 30 帧

1080 HD(1920 × 1088)的高清视频,就需要处理 244800 个宏块(MBs),作为 8 × 8 变换就需要 123379200 个加法和移位操作,这个复杂度已经相当于超过 123MIPS,这种情况下用硬件加速来满足实时解码的需求是不可避免的。

有大量研究关注在 H.264 反变换(inverse transform, IT)的有效实现上,由于只有单一类型变换的设计的低效率,最新方法是和本文一样扩展可以支持 High Profile 4 × 4 和 8 × 8 变换的结构.例如,文献[1]用了 1 个一维变换模块和 16 个寄存器的转置阵列,用 3524 个门的代价来实现了单一类型的变换,在 120MHz 的工作频率下能达到 120M pixels/s 的吞吐率.文献[2]和[3]的变换和量化部分都很相似,这个结构每个时钟周期处理 16 个像素,在 107.5MHz 的工作频率下能达到 1720M pixels/s 的吞吐率,但是需要消耗高达 77280 个门,用 DTUA 值来衡量效率非常低.文献[4]中的方法在 1 个时钟周期中就能完成整个 4 × 4 变换,然而,它需要用到矩阵的置换运算来简化实现,在变换前后需要先进行额外的置换操作。

绝大多数已发表的工作都不支持 H.264 High Profile 中新引入的 8 × 8 变换,例如文献[5]和文献[9].另外很多之前的设计没有实现转置寄存器阵列,但是,这并不能显著地减小电路规模,原因是这个阵列同时提供了量化和变换模块之间的临时存储,其他设计必须用相似数量或者更多的寄存器来实现流水线.例如,文献[6]和[7]对比说明了输入输出寄存器的代价某种程度上和转置寄存器阵列是可比的.文献[8]没有使用转置阵列也消耗了和本设计几乎相同的面积,同时本设计能够达到更高的数据处理速度.只有少数几个已发表的文献支持 H.264 High Profile 中最新提出的 8 × 8 变换,然而文献[10]和[11]不能支持 H.264 High Profile,文献[8]和[12]的数据处理速度不如本设计。

和已发表的文献相比,本设计是唯一支持 H.264 High Profile 所有变换并且能达到 1.2G pixels/s 高吞吐率(采用 TSMC 0.13 微米工艺,在 150MHz 的时钟频率下工作)的方案,只消耗了 4807 门电路,能够支持分辨率为 4096 × 2048、每秒 60 帧的 High Profile 视频实时解码需求.根据 DTUA 的概念作为效率因子衡量,本设计在已有的支持 High Profile 的设计中是最有效率的。

## 2 算法分析

H.264 FRExt 解码器采用了四种不同的反变换,式(1)用于亮度 8 × 8 AC 块,式(2)用于亮度和色度 4 × 4 AC 块,(3)和(4)是分别用于色度 4 × 4 和 2 × 2 块的 Hadamard 变换(Hadamard Transform, HT),其中,  $X$  是经过反量化的输入块,  $Y$  是用来重建图像的残差数据。

$$Y = C_{INVERSE8 \times 8}^T X C_{INVERSE8 \times 8} \quad (1)$$

$$Y = C_{INVERSE4 \times 4}^T X C_{INVERSE4 \times 4} \quad (2)$$

$$Y_{DC} = H_{4 \times 4}^T X H_{4 \times 4} \quad (3)$$

$$Y_{DC} = H_{2 \times 2}^T X H_{2 \times 2} \quad (4)$$

$$C_{INVERSE8 \times 8}^T =$$

$$\begin{bmatrix} 1 & 3/2 & 1 & 5/4 & 1 & 3/4 & 1/2 & 3/8 \\ 1 & 5/4 & 1/2 & -3/8 & -1 & -3/2 & -1 & -3/4 \\ 1 & 3/4 & -1/2 & -3/2 & -1 & 3/8 & 1 & 5/4 \\ 1 & 3/8 & -1 & -3/4 & 1 & 5/4 & -1/2 & -3/2 \\ 1 & -3/8 & -1 & 3/4 & 1 & -5/4 & -1/2 & 3/2 \\ 1 & -3/4 & -1/2 & 3/2 & -1 & -3/8 & 1 & -5/4 \\ 1 & -5/4 & 1/2 & 3/8 & -1 & 3/2 & -1 & 3/4 \\ 1 & -3/2 & 1 & -5/4 & 1 & -3/4 & 1/2 & -3/8 \end{bmatrix}$$

$$C_{INVERSE4 \times 4}^T = \begin{bmatrix} 1 & 1 & 1 & 1/2 \\ 1 & 1/2 & -1 & -1 \\ 1 & -1/2 & -1 & 1 \\ 1 & -1 & 1 & -1/2 \end{bmatrix}$$

$$H_{4 \times 4}^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}, \quad H_{2 \times 2}^T = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

采用二维分解和利用矩阵运算提取公共因子的操作设计通用运算单元(PE),简化了逆变换的实现,一维变换公式可以用式(5)和(6)的形式表达便于提取公共部分。

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1/2 \\ 1 & 1/2 & -1 & -1 \\ 1 & -1/2 & -1 & 1 \\ 1 & -1 & 1 & -1/2 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_2 \\ X_4 \\ X_6 \end{bmatrix} + \begin{bmatrix} 3/2 & 5/4 & 3/4 & 3/8 \\ 5/4 & -3/8 & -3/2 & -3/4 \\ 3/4 & -3/2 & 3/8 & 5/4 \\ 3/8 & -3/4 & 5/4 & -3/2 \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_3 \\ X_5 \\ X_7 \end{bmatrix}$$

$$\begin{bmatrix} Y_7 \\ Y_6 \\ Y_5 \\ Y_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1/2 \\ 1 & 1/2 & -1 & -1 \\ 1 & -1/2 & -1 & 1 \\ 1 & -1 & 1 & -1/2 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_2 \\ X_4 \\ X_6 \end{bmatrix} - \begin{bmatrix} 3/2 & 5/4 & 3/4 & 3/8 \\ 5/4 & -3/8 & -3/2 & -3/4 \\ 3/4 & -3/2 & 3/8 & 5/4 \\ 3/8 & -3/4 & 5/4 & -3/2 \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_3 \\ X_5 \\ X_7 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} + \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix}, \quad \begin{bmatrix} Y_7 \\ Y_6 \\ Y_5 \\ Y_4 \end{bmatrix} = \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} - \begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} \quad (6)$$

从式(6)观察得知  $8 \times 8$  的反变换矩阵可以划分成两个  $4 \times 4$  的矩阵,而其中一个恰好和  $4 \times 4$  的反变换矩阵相同.

$$\begin{aligned} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 & 1/2 \\ 1 & 1/2 & -1 & -1 \\ 1 & -1/2 & -1 & 1 \\ 1 & -1 & 1 & -1/2 \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_2 \\ X_4 \\ X_6 \end{bmatrix} \\ &= \begin{bmatrix} X_0 + X_2 + X_4 + 1/2 X_6 \\ X_0 + 1/2 X_2 - X_4 - X_6 \\ X_0 - 1/2 X_2 - X_4 + X_6 \\ X_0 - X_2 + X_4 - 1/2 X_6 \end{bmatrix} \\ &= \begin{bmatrix} (X_0 + X_4) + (X_2 + 1/2 X_6) \\ (X_0 - X_4) + (1/2 X_2 - X_6) \\ (X_0 - X_4) - (1/2 X_2 - X_6) \\ (X_0 + X_4) - (X_2 + 1/2 X_6) \end{bmatrix} \\ &= \begin{bmatrix} \alpha_0 + \alpha_1 \\ \alpha_3 + \alpha_2 \\ \alpha_3 - \alpha_2 \\ \alpha_0 - \alpha_1 \end{bmatrix} \end{aligned} \quad (7)$$

进一步的分析表明  $2 \times 2$  的 Hadamard 变换是一个基本操作,在(6)和(10)等式中都存在,所以设计图 3 中的 PE0 来计算它.

而共有的  $4 \times 4$  反变换矩阵需要另外一种操作:

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} X_0 + X_4 \\ X_2 + 1/2 X_6 \\ 1/2 X_2 - X_6 \\ X_0 - X_4 \end{bmatrix} \quad (8)$$

$$\begin{bmatrix} PE1_{Out0} \\ PE1_{Out1} \\ PE1_{Out2} \\ PE1_{Out3} \end{bmatrix} = \begin{bmatrix} In0 + In5 \\ In1 + In4 \\ In2 - In3 \\ In0 - In5 \end{bmatrix} \begin{pmatrix} In0 = X_0, In1 = X_2, \\ In2 = 1/2 X_2, In3 = X_6, \\ In4 = 1/2 X_6, In5 = X_4 \end{pmatrix} \quad (9)$$

因此设计如式(9)所示的 PE1 来实现式(8),同时, PE1 也可以被复用来做基本的  $2 \times 2$  的 Hadamard 变换,如式(11)所示.因为式(12)只存在于  $8 \times 8$  的 IT 中,再设计 PE2 来计算它.总之,只需要两个 PE0,两个 PE1 和一个 PE2 就可以支持 H.264 High Profile 解码器中的所有变换类型.

$$\begin{pmatrix} \beta_0 & \beta_1 \\ \beta_3 & \beta_2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} \alpha_0 & \alpha_1 \\ \alpha_2 & \alpha_3 \end{pmatrix} (PE0_1) \quad (10)$$

$$\begin{pmatrix} Y_0 & Y_1 \\ Y_7 & Y_6 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} \beta_0 & \beta_1 \\ \gamma_0 & \gamma_1 \end{pmatrix} (PE0_2)$$

$$\begin{pmatrix} Y_2 & Y_3 \\ Y_5 & Y_4 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} \beta_2 & \beta_3 \\ \gamma_2 & \gamma_3 \end{pmatrix} (PE1_2)$$

$$(In0 = \alpha_2, In1 = In2 = \alpha_3, In3 = In4 = \gamma_3, In5 = \gamma_2) \quad (11)$$

$$\begin{bmatrix} \gamma_0 \\ \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} = \begin{bmatrix} 3/2 & 5/4 & 3/4 & 3/8 \\ 5/4 & -3/8 & -3/2 & -3/4 \\ 3/4 & -3/2 & 3/8 & 5/4 \\ -3/8 & -3/4 & 5/4 & -3/2 \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_3 \\ X_5 \\ X_7 \end{bmatrix} \quad (12)$$

### 3 可重构结构设计

本设计的二维反变换结构如图 1 所示,包括两个一维反变换单元和 1 个转置寄存器阵列.为了满足分辨率为  $4096 \times 2048$ ,每秒 60 帧 High Profile 视频的解码需求,IT 模块需要的处理速度为  $754.97 \text{M pixels/s}$ ,而本文的 IT 结构处理能力是  $1.2 \text{G pixels/s}$ ( $150 \text{MHz}$  的工作频率),因此能够满足上述的解码需求.

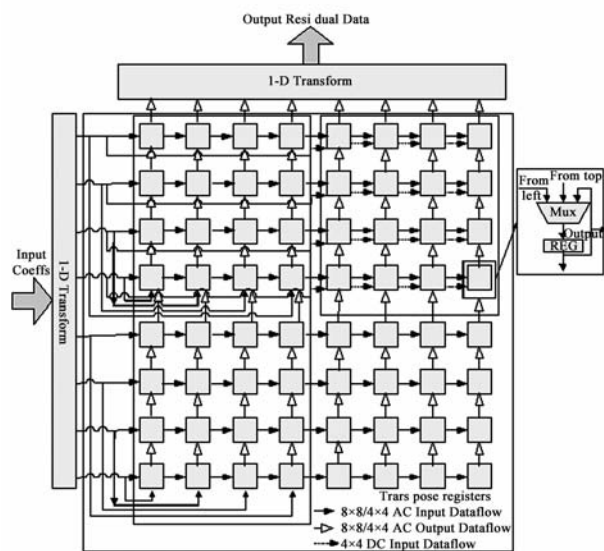


图1 本设计的二维反变换结构

图 1 中的转置阵列包括 64 个基本单元,每个单元包括 1 个多路器和寄存器.多路器的第 1 个输入是寄存器的自反馈,在 NOP 状态使用.第 2 个输入是下方单元

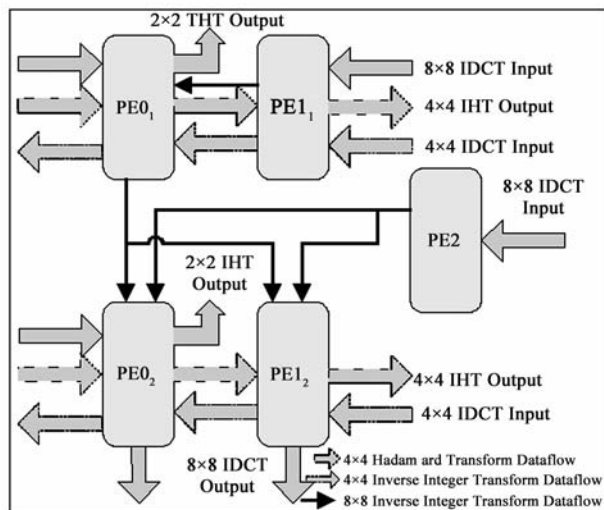


图2 支持H.264 High Profile的一维可重构变换单元

的数据,最后 1 个输入来自左方单元.图 1 中的每种箭头表示了处于输入或者输出模式时相应的数据流以及每种变换类型如何利用转置阵列,方框表明了不同变换类型需要利用的不同区域.

根据本文第 2 部分的算法分析,只需要两个 PE0,两个 PE1 和 1 个 PE2 就能支持 H.264 High Profile 解码器的所有变换类型.图 2 所示为基于图 3 所示 PE 的支持 H.264 High Profile 的一维可重构变换单元,包括参与每种变换的 PE 以及每个 PE 的输入输出数据,并且每种类

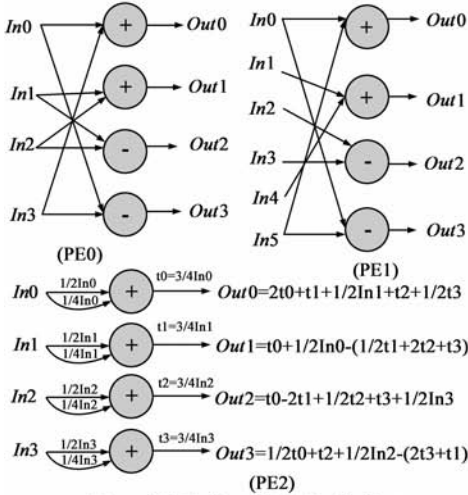


图3 本设计的三种通用运算单元

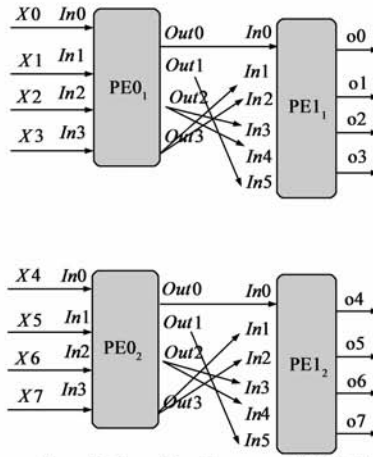


图4 基于PE的一维4×4IHT数据通路

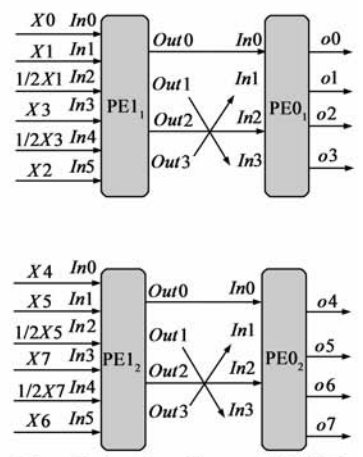


图5 基于PE的一维4×4IT数据通路

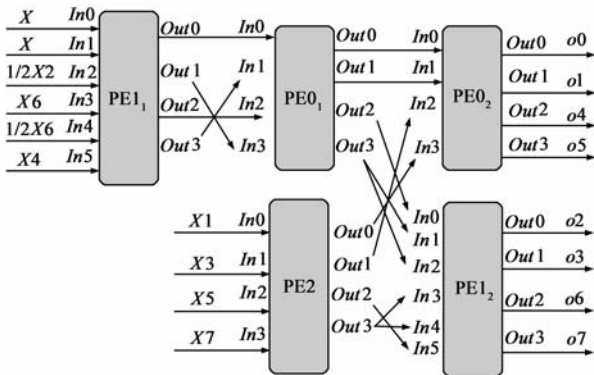


图6 基于PE的一维8×8IT数据通路

### 4 结果分析与比较

整个结构采用 Verilog 硬件描述语言描述并仿真验证通过,使用 Synopsys VCS MX Y-2006.06 模拟器进行硬件模拟,仿真数据利用 H.264 的参考软件 JM13.2 截取.采用 TSMC 0.13 微米 CMOS 标准单元库来实现本文提出的支持 H.264 High Profile 所有变换的可重构结构,综合软件采用 Synopsys 公司的 Design-Compiler.表 1 给出对比电路结构的主要性能指标:电路规模(Area,单位:gate)、最高工作频率(Freq,单位:MHz)、每个时钟周期处理的像素数(DPR,单位: pixels/cycle)、吞吐率(TP,单位: pixels/s).为了衡量电路结构的效率,本文还引入了文献[6]中

型的箭头表示了处于不同变换模式时 PE 之间的相应数据流.对于 4×4 的 IT 和 HT,两组数据能够同时被处理,每组数据用一个箭头来表示.对于 8×8 的 IT,每次处理一组数据,分成两个部分.可重构一维变换结构中的 PE 如图 3 所示,其中的公式表示了每种 PE 的操作.

图 4,5 和 6 利用图 3 中的 PE 分别构建了一维 4×4 HT,4×4 IT 和 8×8IT 的数据通路,每个数据通路的 PE 的输入也在图中相应做了标示.

定义的效率因子 DTUA(单位: pixels/s/gate)来反映电路结构的吞吐率相对于规模的效率,DTUA 值越高,电路的效率越高.采用 DTUA 作为评判参数,我们发现本设计在现有的支持 H.264 High Profile 所有变换的设计中是最有效的,表 1 最后一列是本设计.

表 1 结果对比

	Chen [6]	Wang [7]	Li [8]	Cao [5]	Fan [12]	Ours
High Profile	N	N	N	N	Y	Y
Tech	0.18	0.35	0.18	0.18	0.18	0.13
Area	6428	6538	13651	11038	6537	4807
Freq	100	88	200	200	125	150
DPR	8	4	4	16	8	8
TP	800	352	800	3.2G	1G	1.2G
Trans- pose	N	Y	N	N	Y	Y
Array						
DTUA	124.5	53.8	58.60	289.9	153.0	249.64

### 5 结论

本文提出了一种支持 H.264 High Profile 解码器所有变换类型的结构,包括 4×4 和 8×8 的反变换和 2×2/4×4 的 Hadamard 变换.和已发表的文献相比,本设计是唯

一支持 H.264 High Profile 且能达到 1.2G pixels/s 高吞吐率(采用 TSMC 0.13 微米工艺,在 150MHz 的时钟频率下工作)的方案,可重构变换结构只消耗了 4807 门电路,能够支持分辨率为  $4096 \times 2048$ 、每秒 60 帧的 High Profile 视频实时解码需求.该结构相比于其他支持 H.264 High Profile 的结构,在吞吐率相对于电路规模的效率方面,优势明显.

## 参考文献

- [1] Liu Ling-zhi, Qiu Lin, Rong Meng-tian, Jiang Li. A 2-D forward/inverse integer transform processor of H.264 based on highly-parallel architecture [A]. 4th IEEE International Workshop on System-on-Chip for Real-time Applications [C]. Banff, Alberta, Canada: IEEE Computer Society, 2004. 158 - 161.
- [2] Ihab Amer, Wael Badawy, Graham Jullien. Hardware prototyping for the H.264  $4 \times 4$  transformation [A]. Proc of IEEE International Conference on Acoustics, Speech, and Signal Processing [C]. Montreal, Quebec, Canada: IEEE, 2004. 77 - 80.
- [3] Roman Kordasiewicz, Shahram Shirani. Hardware implementation of the optimized transform and quantization blocks of H.264 [A]. Proc Electrical and Computer Engineering [C]. Niagara Falls, Canada: IEEE, 2004. 943 - 946.
- [4] Woong Hwangbo, Jaemoon Kim, Chong-Min Kyung. A high performance 2-D inverse transform architecture for the H.264/AVC decoder [A]. IEEE International Symposium on Circuits and Systems [C]. LA, USA: IEEE, 2007. 1613 - 1616.
- [5] 曹伟,洪琪,侯慧,童家榕,来金梅,闵昊,荆明娥.一种用于 H.264 编解码的新型高效可重构多变换 VLSI 结构 [J]. 电子学报, 2009, 37(4): 673 - 677.  
Cao Wei, Hong Qi, Hou Hui, Tong Jia-rong, Lai Jin-mei, Min Hao, Jin Ming-e. A high-performance reconfigurable multi-transform VLSI architecture for H.264 CODEC [J]. Acta Electronica Sinica, 2009, 37(4): 673 - 677. (in Chinese)
- [6] Kuan-Hung Chen, Jiun-In Guo, Jinn-Shyan Wang. A high-performance direct 2-D transform coding IP design for MPEG-4 AVC/H.264 [J]. IEEE Transaction on Circuits and Systems for Video Technology, 2006, 16(4): 472 - 483.
- [7] Tu-Chih Wang, Yu-Wen Huang, Hung-Chi Fang, and Liang-Gee Chen. Parallel  $4 \times 4$  2D transform and inverse transform architecture for MPEG-4 AVC/H.264 [A]. IEEE International Symposium on Circuits and Systems [C]. Bangkok, Thailand: IEEE, 2003. 800 - 803.
- [8] Yu Li, Yun He, Shun-liang Mei. A highly parallel joint VLSI architecture for transforms in H.264/AVC [J]. Journal of Signal Processing Systems, 2008, 50(1): 19 - 32.
- [9] Yu-Wen Huang, Bing-Yu Hsieh, Tung-Chien Chen, Liang-Gee Chen. Analysis, fast algorithm, and VLSI architecture design for H.264/AVC intra-frame coder [J]. IEEE Transaction on Circuits Systems for Video Technology, 2005, 15(3): 378 - 401.
- [10] Ihab Amer, Wael Badawy, Graham Jullien. A high-performance hardware implementation of the H.264 simplified  $8 \times 8$  transformation and quantization [A]. International Conference on Acoustics, Speech, and Signal Processing [C]. Philadelphia, PA, USA: IEEE Signal Processing Society, 2005. 1137 - 1140.
- [11] Jeoong Sung Park and Tokunbo Ogunfunmi. A new hardware implementation of the  $8 \times 8$  H.264 transform and quantization [A]. International Conference on Acoustics, Speech, and Signal Processing [C]. Taipei, Taiwan: IEEE Signal Processing Society, 2009. 585 - 588.
- [12] Chih-Peng Fan. Cost-effective hardware sharing architectures of fast  $8 \times 8$  and  $4 \times 4$  integer transforms for H.264-AVC [A]. IEEE Asia Pacific Conference on Circuits and Systems [C]. Singapore: IEEE, 2006. 776 - 779.

## 作者简介



陆晓凤 女, 1984 年生于黑龙江哈尔滨, 北京大学计算机系博士研究生. 主要研究方向为软硬件协同设计、视频编解码和关键算法的 VLSI 实现.

E-mail: luxiaofeng@mprc.pku.edu.cn



刘锋 男, 1977 年出生, 2005 年获西安电子科技大学微电子学与固体电子学博士学位, 现在北京大学从事教学和科研工作. 主要研究方向为 VLSI 设计、视频和图形处理结构、多媒体系统等.

佟冬 男, 1971 年生于吉林长春, 北京大学计算机系副教授. 主要研究方向为高性能微处理器、系统芯片、体系结构等.

王克义 男, 1946 年出生, 北京大学信息科学技术学院教授, 博士生导师, 曾参加和主持多项国家重点科技攻关项目, 主要研究领域为高性能微处理器结构与设计、嵌入式系统等.