

高性能低存储的2维离散小波变换架构

曹鹏 王超

(东南大学国家专用集成电路系统工程技术研究中心, 南京 210096)

摘要 针对 JPEG2000 中的 5/3 小波和 9/7 小波存在的高存储问题, 通过改进离散小波变换 (DWT) 的提升算法, 提出了一种统一的高性能、低存储的 2 维离散小波变换架构。采用该算法实现的 2 维离散小波变换架构不仅省去了行列模块间的转置缓存, 而且减小了片内缓存的大小。对于 $N \times N$ 大小的图像 (N 为图像宽度) 进行 5/3 2 维 DWT 仅需要 $2N$ 大小的片内缓存, 进行 9/7 2 维 DWT 仅需要 $4N$ 大小的片内缓存, 而且通过采用流水线技术还可将关键路径缩短为一个乘法器的延时。和已有的 2 维 DWT 架构相比, 该统一架构具有更低的片内存储器需求和更高的性能。该架构经 Verilog HDL 描述, 并在 ModelSim 中验证正确。在 Altera Stratix II FPGA EP2S60F1024C4 中综合的结果显示, 对于 1024×1024 大小的图像, 需要 1 284 个 ALUT, 片内存储器的大小为 4 K, 最高频率可达 172.56 MHz。

关键词 离散小波变换 提升模式 JPEG2000

中图法分类号: TN919.81 文献标识码: A 文章编号: 1006-8961(2009)11-2198-07

VLSI Architecture for 2-Dimensional Discrete Wavelet Transform with High Performance and Low Memory

CAO Peng, WANG Chao

(National ASIC System Engineering Research Center, Southeast University, Nanjing 210096)

Abstract A unified high-performance and memory-efficient architecture is proposed to perform 5/3 DWT and 9/7 DWT in JPEG2000 with a novel modified lifting algorithm. By applying the proposed lifting scheme, the transposing buffer between the row processor and the column processor is eliminated, resulting in a reduction of the internal memory requirement. For an $N \times N$ image, only $2N$ internal memory is required for 5/3 DWT and $4N$ is required for 9/7 DWT to perform 2D DWT with the critical path limited by one multiplier delay by employing the pipeline technique. Compared with the existing 2D DWT architectures, the proposed 2D DWT architecture has the advantage of regular structure, low memory requirement and high system performance. The proposed architecture was described with Verilog HDL language and verified to be correct in ModelSim. It was also synthesized, placed and routed on an Altera Stratix II FPGA EP2S60F1024C4 using Quartus II version 5.0 toolset. The experimental result shows that 1 284 ALUTs are utilized with memory size of 4 K words for 1024×1024 image and the operating frequency can be up to 172.56 MHz.

Keywords discrete wavelet transform (DWT), lifting, JPEG2000

1 引言

离散小波变换 (DWT) 以其优异的时频局部特

性和良好的去相关能力, 作为多分辨率分析的有效工具, 已被 JPEG2000^[1] 和 MPEG-4 等图像编码标准采用, 在信号处理和视频压缩等领域获得了广泛的应用。而 2 维离散小波变换 (2 维 DWT) 算法的复

基金项目: 国家高技术研究发展计划 (863) 资助项目 (2005AA1Z1214); 国家自然科学基金项目 (60676011)

收稿日期: 2008-04-06; 改回日期: 2008-09-11

第一作者简介: 曹鹏 (1980 ~), 男, 东南大学国家专用集成电路系统工程技术研究中心博士研究生。主要从事多媒体架构设计及 SOC 等研究。E-mail: caopeng@seu.edu.cn

杂性以及较大的片内存储空间,却成为制约 2 维 DWT VLSI 实现的关键问题。

对于 2 维 DWT 而言,存储器的设计和实现决定了整体架构的硬件开销和控制复杂度^[2]。现有的 2 维 DWT 架构按照数据的扫描顺序基本上可以分为以下 3 类^[3]:逐级扫描的 2 维 DWT 架构、基于行扫描的 2 维 DWT 架构和基于块扫描的 2 维 DWT 架构。其中,逐级扫描的 2 维 DWT 架构要先做行变换再做列变换,虽结构简单,但由于需要大小为 $N \times N$ 的转置缓存,以存储行列变换间的中间结果(N 为图像的宽度),因此不利于片内存储器的实现。基于行扫描的 2 维 DWT 架构可以并行完成行变换和列变换,片内缓存由转置缓存和列缓存组成,大小分别为 $K_1 N$ 和 $K_2 N$,其中 K_1 取决于行列模块中数据的扫描顺序,而 K_2 则与采用的 DWT 滤波器相关。基于块扫描的 2 维 DWT 架构可通过分块来处理图像数据,虽然减小了片内存储器大小,但是却导致了额外的片外数据的访问和复杂的块边界处理。

基于行扫描的 2 维 DWT 架构由于具有结构规则、存储量小的优点,已经被很多研究者采用^[2,4-6]。为了进一步地减小 2 维 DWT 中的存储器大小,以便减小 2 维 DWT 架构的硬件面积,本文针对 JPEG2000 中可逆的 5/3 DWT 和不可逆的 9/7 DWT,提出了一种统一的 2 维 DWT 架构,由于其省去了行列 DWT 模块间的转置缓存,从而减少了片内存储器的大小,同时通过采用折叠技术和流水线技术,优化了行列 DWT 模块的实现,并缩短了关键路径,提高了系统性能。

2 改进的离散小波变换算法

2.1 提升小波

Daubechies 和 Sweldens 提出了离散小波变换的提升模式^[7-8],它使得传统的基于卷积的离散小波变换可以由多个提升步骤来完成。5/3 DWT 通过一个提升步骤完成,而 9/7 DWT 通过两个提升步骤和一个伸缩步骤完成,实现过程如式(1)~式(6)所示,即首先将输入数据分为偶数部分 $L^{(0)}[n]$ 和奇数部分 $H^{(0)}[n]$;然后分别通过提升步骤对其进行预测和更新,得到的提升结果表示为 $L^{(m)}[n]$ 和 $H^{(m)}[n]$,其中 m 是提升级数,当进行 5/3 DWT 时, $m=1$,当进行 9/7 DWT 时, $m=1, 2$, $\alpha^{(m)}$ 与 $\beta^{(m)}$ 是

5/3 或 9/7 DWT 的提升系数;最后经伸缩系数 K 和 $1/K$ 即可得到低频小波系数 $L[n]$ 和 高频小波系数 $H[n]$ 。提升系数和伸缩系数的具体值可见文献[1]。

$$L^{(0)}[n] = x[2n] \quad (1)$$

$$H^{(0)}[n] = x[2n+1] \quad (2)$$

$$H^{(m)}[n] = H^{(m-1)}[n] + \alpha^{(m)} \times (L^{(m-1)}[n] + L^{(m-1)}[n+1]) \quad (3)$$

$$L^{(m)}[n] = L^{(m-1)}[n] + \beta^{(m)} \times (H^{(m)}[n] + H^{(m)}[n-1]) \quad (4)$$

$$L[n] = K \times L^{(m)}[n] \quad (5)$$

$$H[n] = \frac{1}{K} \times H^{(m)}[n] \quad (6)$$

2.2 翻转结构

基于提升模式的离散小波变换具有计算量小、结构规则、支持原位计算等优点,但会导致相应的硬件架构的关键路径较长。为了缩短离散小波变换架构中的关键路径,并减少流水线寄存器的个数,Huang 等人通过对传统的提升算法中的乘法器的系数取倒数,提出了 DWT 的翻转结构^[9],用于将提升步骤和伸缩步骤更新为

$$H^{(m)}[n] = a^{(m)} \times H^{(m-1)}[n] + L^{(m-1)}[n] + L^{(m-1)}[n+1] \quad (7)$$

$$L^{(m)}[n] = b^{(m)} \times L^{(m-1)}[n] + H^{(m)}[n] + H^{(m)}[n-1] \quad (8)$$

$$L[n] = k_0 \times L^{(m)}[n] \quad (9)$$

$$H[n] = k_1 \times H^{(m)}[n] \quad (10)$$

其中, $a^{(m)}/b^{(m)}$ 和 k_0/k_1 分别为翻转结构中的提升系数和伸缩系数,可由 2.1 节中所用的系数 $\alpha^{(m)}$, $\beta^{(m)}$ 和 K 计算获得。具体的,在 5/3 DWT 中, $a^{(1)} = 1/\alpha^{(1)}$, $b^{(1)} = 1/\alpha^{(1)}\beta^{(1)}$, $k_0 = \alpha^{(1)}\beta^{(1)}K$, $k_1 = \alpha^{(1)}/K$,在 9/7 DWT 中, $a^{(1)} = 1/\alpha^{(1)}$, $b^{(1)} = 1/\alpha^{(1)}\beta^{(1)}$, $a^{(2)} = 1/\beta^{(1)}\alpha^{(2)}$, $b^{(2)} = 1/a^{(2)}\beta^{(2)}$, $k_0 = \alpha^{(1)}\beta^{(1)}\alpha^{(2)}\beta^{(2)}K$, $k_1 = \alpha^{(1)}\beta^{(1)}\alpha^{(2)}/K$ 。

2.3 改进的提升小波

和传统的提升小波相比,翻转结构的 1 维 DWT 架构能有效地缩短关键路径,并能减小硬件面积,以提高系统性能。但采用这两种算法由于都需要对 3 个源数据进行预测和更新,因此实现 2 维 DWT 架构时,需要先在行列 DWT 模块间加入转置缓存,然后将按逐行扫描的方式将输出的 1 维小波系数重新排

列为按双行逐列扫描的方式输出,从而增加了硬件开销和控制逻辑的复杂度。

本文提出了一种改进的提升小波算法,使得提升步骤的计算只需要 1 个源数据即可进行,对于逐行扫描输出的行变换结果可立即进行列变换,从而省去了行列 DWT 模块间的转置缓存。下面以翻转结构为基础,导出了改进的提升小波的算法。

若将式(7)代入式(8)可得

$$\begin{aligned}
 L^{(m)}[n] &= b^{(m)} \times L^{(m-1)}[n] + H^{(m)}[n] + \\
 &H^{(m)}[n-1] = b^{(m)} \times L^{(m-1)}[n] + \\
 &(a^{(m)} \times H^{(m-1)}[n] + L^{(m-1)}[n] + \\
 &L^{(m-1)}[n+1]) + (a^{(m)} \times H^{(m-1)}[n-1] + \\
 &L^{(m-1)}[n-1] + L^{(m-1)}[n]) \\
 &= L^{(m-1)}[n-1] + a^{(m)} \times H^{(m-1)}[n-1] + \\
 &(b^{(m)} + 2) \times L^{(m-1)}[n] + a^{(m)} \times \\
 &H^{(m-1)}[n] + L^{(m-1)}[n+1] \quad (11)
 \end{aligned}$$

则提升步骤中预测和更新的计算可分解为

$$\begin{aligned}
 L^{(m)}[n] &= X^{(m)}[n] + L^{(m-1)}[n+1] \\
 H^{(m)}[n] &= Y^{(m)}[n] + L^{(m-1)}[n+1] \quad (12) \\
 X^{(m)}[n] &= Y^{(m)}[n] + (b^{(m)} + 2) \times L^{(m-1)}[n] + \\
 &a^{(m)} \times H^{(m-1)}[n] \\
 Y^{(m)}[n] &= L^{(m-1)}[n] + a^{(m)} \times H^{(m-1)}[n] \quad (13)
 \end{aligned}$$

式(13)的计算可进一步分解为

$$\begin{aligned}
 X^{(m)}[n] &= Q^{(m)}[n] + a^{(m)} \times H^{(m-1)}[n] \\
 Y^{(m)}[n] &= P^{(m)}[n] + a^{(m)} \times H^{(m-1)}[n] \quad (14) \\
 P^{(m)}[n] &= L^{(m-1)}[n] \\
 Q^{(m)}[n] &= Y^{(m)}[n] + (b^{(m)} + 2) \times L^{(m-1)}[n] \quad (15)
 \end{aligned}$$

3 2 维 DWT 架构

根据前面叙述的提升小波的改进算法,本文提出了一种适用于 5/3 DWT 和 9/7 DWT 的 2 维 DWT 架构。该架构由 1 维 DWT 行模块、1 维 DWT 列模块和伸缩模块组成(如图 1 所示)。多路选择器选通 0 和 1 时,分别配置为 5/3 2 维 DWT 和 9/7 2 维 DWT。当配置为 9/7 2 维 DWT 时,则行模块由 2 个同构的行提升处理单元(RPE)完成两级提升步骤,而列模块中的两级提升步骤则由 2 个异构的列提升处理单元(CPE1 和 CPE2)完成。当配置为 5/3 2 维 DWT 时,则行模块和列模块分别由 1 个 RPE 和 CPE1 完成。图像数据按逐行扫描的顺序(如图 2 所示)输入 1 维 DWT 行模块,所生成的 1 维小波系数直接输入 1 维 DWT 列模块,列模块的结果经伸缩模块生成子带 LL/LH 和 HL/HH 的数据,并按双行隔列扫描的方式(如图 3 所示)输出。

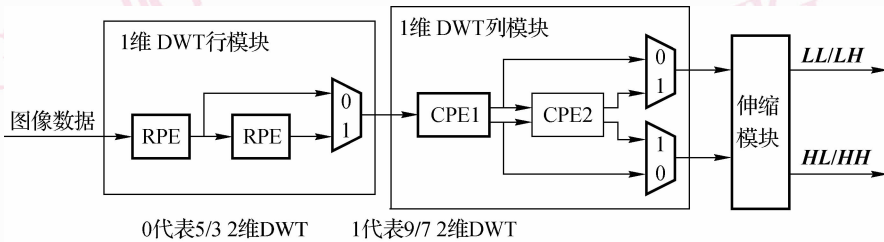


图 1 2 维 DWT 架构

Fig. 1 Architecture of proposed 2D DWT

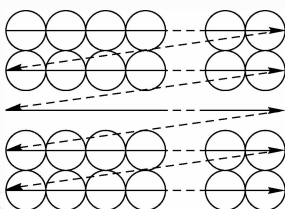


图 2 输入顺序

Fig. 2 Input scan method

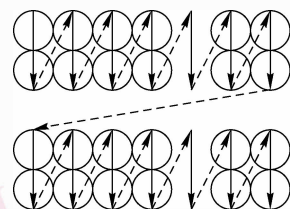


图 3 输出顺序

Fig. 3 Output scan method

3.1 行模块

行模块中的 2 个 RPE 分别完成行模块的两级提升步骤。根据基于翻转结构的提升算法,RPE 的架构如图 4 所示,其由 1 个乘法器、2 个加法器、2 个切换器、4 个多路选择器和 5 个寄存器构成。该架构采用流水线技术将关键路径限制为一个乘法器的

延时,同时通过折叠技术交替来进行预测步骤和提升步骤的计算,从而将乘法器和加法器等计算资源的个数减少了一半。当偶/奇数列的图像数据 $L^{(m-1)}[n]/H^{(m)}[n]$ 输入时,切换器和多路选择器选通 0/1,即完成预测/更新步骤。

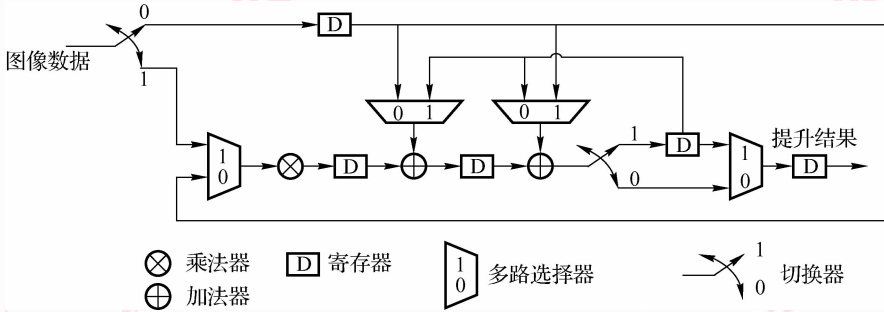


图 4 行变换提升步骤 RPE 的架构

Fig. 4 Architecture of row processor element (RPE)

图 5 为 RPE 的数据流图。由该图可见,奇偶图像数据以每周期一个的速率交替输入 RPE,经过 5 个周期的时延交替即可输出高/低频的提升结果。

步骤。

根据改进的提升小波算法,列方向的一级提升步骤(CPE1)的架构如图 6 所示。该架构由 1 个乘法器、3 个加法器、3 个选择器、4 个流水线寄存器和 2 个行缓存构成。由于采用了流水线技术,因此关键路径被限制为一个乘法器的延时。CPE1 每个周期读入一个行变换的结果,经过 $2N$ 个周期的时延,每当偶数行的 1 维小波系数输入时,CPE1 就生成了一对列方向一级提升的高/低频结果。

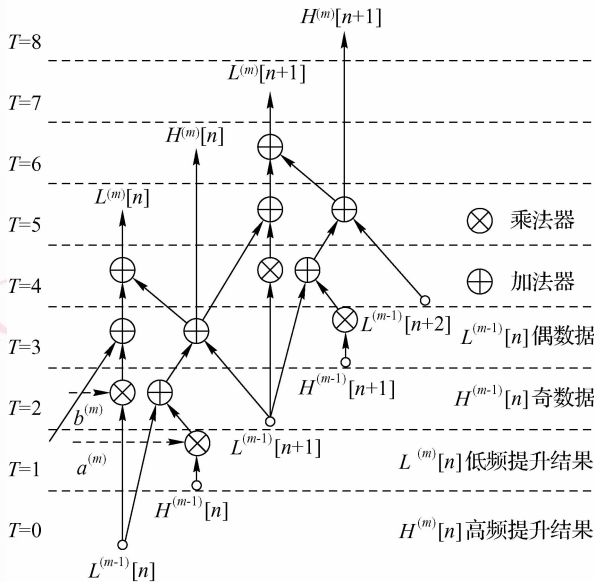


图 5 RPE 的数据流图

Fig. 5 Data flow graph (DFG) of RPE

3.2 列模块

当 1 维小波系数按逐行扫描的顺序输入列模块后,则 CPE1 就完成了列方向的一级提升步骤,并实现了数据的行列转置,然后再按双行逐列扫描的顺序输出一级提升的结果,并由 CPE2 完成二级提升

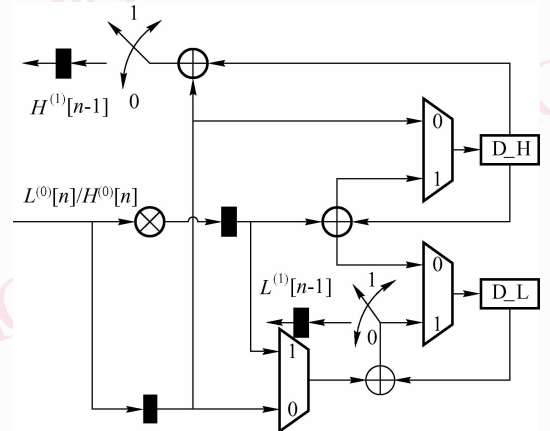


图 6 列变换一级提升步骤(CPE1)的架构

Fig. 6 Architecture of 1st column processor element (CPE1)

当偶数行数据的 1 维小波数据 $L^{(0)}[n]$ 输入时,则多路选择器选通 0,其计算的数据流图如图 7 所

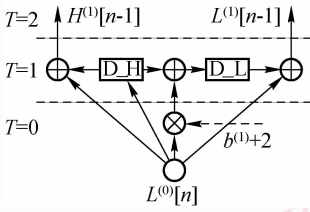


图 7 偶数行数据输入时, CPE1 的数据流图
Fig. 7 DFG of CPE1 when even-row data inputted

示。然后按式(15)将列缓存 D_L 和 D_H 中的当前列的中间结果 D_L 和 D_H 分别更新为

$$D_L \leftarrow D_H + (b^{(1)} + 2) \times L^{(0)}[n] \quad (16)$$

$$D_H \leftarrow L^{(0)}[n]$$

同时,按式(12)生成的高/低频一级提升结果可表示为

$$L^{(1)}[n-1] \leftarrow D_L + L^{(0)}[n] \quad (17)$$

$$H^{(1)}[n-1] \leftarrow D_H + L^{(0)}[n]$$

当奇数行数据的 1 维小波数据 $H^{(0)}[n]$ 输入时,则多路选择器选通 1,其计算的数据流图如图 8 所示。然后按式(14)将列缓存 D_L 和 D_H 中的当前列的中间结果 D_L 和 D_H 分别更新为

$$\begin{aligned} D_L &\leftarrow D_L + a^{(1)} \times H^{(0)}[n] \\ D_H &\leftarrow D_H + a^{(1)} \times H^{(0)}[n] \end{aligned} \quad (18)$$

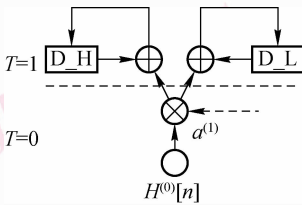


图 8 奇数行数据输入时, CPE1 的数据流图
Fig. 8 DFG of CPE1 when odd-row data inputted

CPE2 用于完成列方向的二级提升步骤,根据改进的提升小波算法,其架构如图 9 所示。该架构由 2 个乘法器、5 个加法器、6 个流水线寄存器和 2

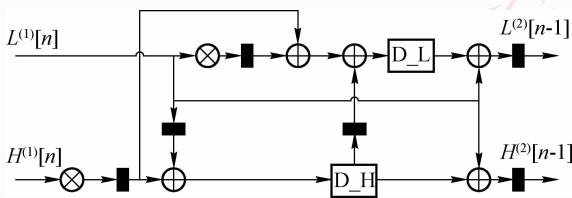


图 9 列变换二级提升步骤 (CPE2) 的架构
Fig. 9 Architecture of 2nd column processor element (CPE2)

个行缓存构成。通过在数据通道中插入流水线寄存器的方式,保证了关键路径的延迟为一个乘法器。CPE2 每隔 N 个周期,以每周期两个的速率从 CPE1 获得两行一级提升的结果,并生成二级提升的高/低频数据。

二级提升步骤的数据流图如图 10 所示。

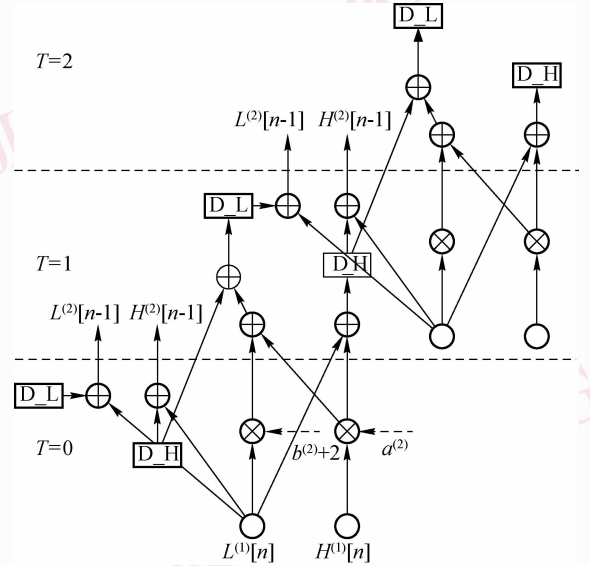


图 10 CPE2 的数据流图
Fig. 10 Data flow graph (DFG) of CPE2

当一级提升的结果 $L^{(1)}[n]$ 和 $H^{(1)}[n]$ 输入时,则按式(12)生成的高/低频二级提升结果可表示为

$$L^{(2)}[n-1] \leftarrow D_L + L^{(1)}[n] \quad (19)$$

$$H^{(2)}[n-1] \leftarrow D_H + L^{(1)}[n]$$

同时,按式(13)将列缓存 D_L 和 D_H 中的当前列的中间结果 D_L 和 D_H 分别更新为

$$\begin{aligned} D_L &\leftarrow D_H + (b^{(2)} + 2) \times L^{(1)}[n] + a^{(2)} \times H^{(1)}[n] \\ D_H &\leftarrow L^{(1)}[n] + a^{(2)} \times H^{(1)}[n] \end{aligned} \quad (20)$$

3.3 伸缩模块

行 DWT 和列 DWT 的伸缩步骤统一由伸缩模块完成。对于 5/3 和 9/7 2 维 DWT,各个子带的伸缩系数如表 1 所示。由于输出的子带 LL/HL 或子带 LH/HH 中只有一个子带需要进行伸缩,因此可以由一个乘法器完成伸缩步骤,伸缩模块的框图如图 11 所示。当子带 LL/HL 输出时,则多路选择器选择 0,对子带 LL 系数进行伸缩;当子带 LH/HH 输出时,则多路选择器选择 1,对子带 HH 系数进行伸缩。

表 1 2 维 DWT 子带伸缩系数

Tab. 1 Scaling coefficients of subbands of 2D DWT

子带	LL	HL	LH	HH
伸缩系数	k_0^2	1	1	k_1^2

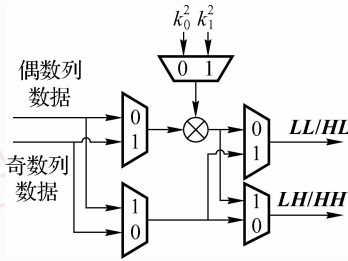


图 11 伸缩模块框图

Fig. 11 Block diagram of scaling module

4 性能分析与比较

在本文提出的 2 维 DWT 架构中,各模块的硬件开销和性能如表 2 所示,其中, N 表示图像的宽度,

t_m 和 t_a 分别表示乘法器和加法器的延时。

表 2 本文提出的 2 维 DWT 架构中各模块的硬件开销

Tab. 2 Hardware resource cost of each module in the proposed architecture of 2D DWT

模块名	乘法器个数	加法器个数	寄存器个数	缓存大小
行模块 RPE	1	2	5	0
列模块 CPE1	1	3	4	$2N$
列模块 CPE2	2	5	6	$2N$
伸缩模块	1	0	0	0

表 3 和表 4 给出了多种 5/3 2 维 DWT 和 9/7 2 维 DWT 架构的性能比较。和已有的架构相比,在面积方面,由于本文通过采用改进的提升算法省去了转置缓存,从而减少了整个架构的存储器面积;在计算资源方面,由于采用了折叠技术,因此所用的乘法器和加法器的个数大大减少,这进一步减少了硬件面积;在系统性能方面,由于采用了流水线技术,整个架构的关键路径延时为 t_m ,从而提高了运行频率。

表 3 2 维 5/3 DWT 架构的性能比较

Tab. 3 Comparison of various architecture of 5/3 2D DWT

架构	乘法器个数	加法器个数	片内存储器大小	关键路径
文献[4]	4	4	$3.5N$	t_m
文献[5]	4	8	$3N$	t_m
文献[6]	4	8	$3N$	t_m
文献[10]	4	8	$5N$	N/A
文献[11]FA	4	8	$3.5N$	$t_m + 2t_a$
文献[11]HA	8	16	$3.5N$	$t_m + 2t_a$
本文	3	5	$2N$	t_m

表 4 2 维 9/7 DWT 架构的性能比较

Tab. 4 Comparison of various architecture of 9/7 2D DWT

架构	乘法器个数	加法器个数	片内存储器大小	关键路径
文献[4]	6	8	$5.5N$	t_m
文献[5]	12	16	$20N$	t_m
文献[6]	12	16	$6N$	t_m
文献[10]	12	16	$7N$	N/A
文献[11]FA	10	16	$5.5N$	$t_m + 2t_a$
文献[11]HA	18	32	$5.5N$	$t_m + 2t_a$
本文	6	12	$4N$	t_m

该架构采用 Verilog HDL 语言描述,中间结果数据采用 16 位表示,5/3 DWT 中的乘法器可通过移

位实现,9/7 DWT 中的乘法器的系数则采用 12 位 CSD 码定点表示,并通过移位加实现。经 ModelSim

进行功能仿真,采用该架构对图像数据进行 2 维 DWT,所生成的 2 维小波分解系数与 JPEG2000 参考软件 jasper^[12] 得到的结果一致,从而验证了该算法及架构的正确性。为了验证该架构的性能,本文采用 Quartus II 5.0 在 Altera FPGA 器件 Stratix II 2S60F1024C4 上进行了综合,综合结果如表 5 所示。

表 5 综合结果

Tab. 5 Experiment results of synthesis

器件	Altera FPGA Stratix II 2S60F1024C4
综合工具	Quartus II 5.0
支持 DWT 滤波器	5/3 DWT 和 9/7 DWT
最大支持图像尺寸	1 024 × 1 024
ALUT(个数)	1 284
存储器(字数)	4 K
关键路径(ns)	5.795
运行频率(MHz)	172.56

5 结 论

基于行的 2 维 DWT 架构虽具有结构规则、存储量小的优点,但由于行列 DWT 处理方向不同,需要由转置缓存来存储部分行的中间结果。本文针对 JPEG2000 中的 5/3 DWT 和 9/7 DWT 的高存储问题,通过改进 DWT 的提升算法,提出了一种统一的高性能、低存储的 2 维 DWT 架构,从而省去了转置缓存,并减小了硬件面积,同时通过采用折叠技术和流水线技术来大大减少所需的存储器个数和计算资源,从而缩短了关键路径,提高了系统性能。该架构经 Verilog HDL 描述,并在 ModelSim 中验证正确。对于 $N \times N$ 大小的图像,进行 5/3 2 维 DWT 仅需要 $2N$ 大小的片内缓存,进行 9/7 2 维 DWT 仅需要 $4N$ 大小的片内缓存。在 Altera FPGA Stratix II 2S60F1024C4 中综合的结果显示,对于 $1\ 024 \times 1\ 024$ 大小的图像,仅需要 1 284 个 ALUT,片内存储器大小为 4 K,最高频率可达 172.56 MHz。

参考文献 (References)

- 1 ISO/IEC 15444-1 JPEG 2000 Part I-Core Coding System [S].
- 2 Huang Chao-tsung, Tseng Po-chih, Chen Liang-gee. Generic RAM-based architectures for two-dimensional discrete wavelet transform with line-based method [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2005, 15 (7): 910-920.
- 3 Zervas Nikos D, Anagnostopoulos Giorgos P, Spiliotopoulos V, et al. Evaluation of design alternatives for the 2D-discrete wavelet transform [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2001, 11 (12): 1246-1262.
- 4 Wu Bing-fei, Lin Chung-fu. A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2005, 15 (12): 1615-1628.
- 5 Seo Young-ho, Kim Dong-wook. VLSI architecture of line-based lifting wavelet transform for motion jpeg2000 [J]. IEEE Journal of Solid-State Circuits, 2007, 42 (2): 431-440.
- 6 Lan Xu-guang, Zheng Nan-ning, Liu Yue-hu. Low-power and high-speed vlsi architecture for lifting-based forward and inverse wavelet transform [J]. IEEE Transactions Consumer Electronics, 2005, 51 (2): 379-385.
- 7 Daubechies I, Sweldens W. Factoring wavelet transforms into lifting steps [J]. Journal of Fourier Analysis and Applications, 1998, 4 (3): 247-269.
- 8 Sweldens W. Lifting scheme: A new philosophy in biorthogonal wavelet constructions [J]. In: Proceedings of SPIE Conference on Wavelet Applications in Signal and Image Processing III [C], San Diego, CA, USA, 1995, 2569: 68-79.
- 9 Huang Chao-tsung, Tseng Po-chih, Chen Liang-gee. Flipping structure: An efficient VLSI architecture for lifting-based discrete wavelet transform [J]. IEEE Transactions on Signal Processing, 2004, 52 (4): 1080-1089.
- 10 Barua S, Carletta J E, Kotteri K A, et al. An efficient architecture for lifting-based two-dimensional discrete wavelet transforms [J]. Integration, the VLSI (Very Large Scale Integrated circuits) Journal, 2005, 38 (3): 341-352.
- 11 Xiong Cheng-yi, Tian Jin-wen, Liu Jian. Efficient architectures for two-dimensional discrete wavelet transform using lifting scheme [J]. IEEE Transactions on Image Processing, 2007, 16 (3): 607-614.
- 12 JasPer [EB/OL]. <http://www.ece.uvic.ca/~mdadams/jasper/>. 2001.