

MQ 编码算法优化及高速 VLSI 结构设计与实现

邱志雄¹, 史江义¹, 郝 跃¹, 逢 杰¹, 刘 凯², 李云松²

(1 西安电子科技大学宽禁带半导体材料与器件教育部重点实验室, 陕西西安 710071;

2 西安电子科技大学 ISN 重点实验室, 陕西西安 710071)

摘 要: 传统的 JPEG2000MQ 编码器串行编码效率低下, 同时现有的多上下文并行编码的 MQ 编码器占用资源过大. 本文对 MQ 编码算法中的运算流程, 索引值和概率估计值的求解函数, 条件交换和重归一化算法等四个方面进行了优化, 减弱了上下文之间的依赖性, 简化了条件交换和重归一化算法的复杂度. 依据该算法, 本文提出了一种高速的 MQ 编码器 VLSI 结构, 实验结果表明, 本文提出的 MQ 编码器 VLSI 结构能够工作在 532.91MHz, 吞吐率为 532.91 Msymbols/sec, 相比 Dyer 提出的 Brute force with modified 结构, 工作频率提高 1 倍, 吞吐量提高近 27%, 且面积仅为其四分之一.

关键词: MQ 编码器; JPEG2000; 流水

中图分类号: TP402

文献标识码: A

文章编号: 0372-2112 (2012) 11-2158-07

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2012.11.003

Optimization of MQ-Encoder and Implementation of High Speed VLSI Architecture

DI Zhi-xiong¹, SHI Jiang-yi¹, HAO Yue¹, Pang Jie¹, LIU Kai², LI Yun-song²

(1 Key Laboratory of Wide Band-gap Semiconductor Materials and Devices of Ministry of Education, Xidian University, Xi'an, Shaanxi 710071, China;

2 National Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, Shaanxi 710071, China)

Abstract: MQ-encoder, which is a key bottleneck in the JPEG2000 image compression system, presents challenges for realization of a high-speed and low cost VLSI architecture. In this paper, some optimization work has been done to the MQ-encoder arithmetic about the process flow, index and probably estimation function, condition change and renormalization. These optimization schemes are highly effective in simplifying the relation between contexts, reducing the complexity of the condition change and renormalization process, and cutting down the critical path delay of the pipeline. Based on this improved arithmetic, a VLSI architecture for an MQ-encoder with high speed is proposed. Synthesis result shows that the processing speed of the MQ-coder could reach as high as 532.91MHz with a throughput of 532.91 Msymbols/sec. Compared with Dyer's architecture, the architecture with an improved throughput as high as 27% presented in this paper can get a speed two times greater than the former one while its area is only a quarter of the former one.

Key words: MQ-coder; JPEG2000; pipeline

1 引言

MQ 编码算法由于只能串行处理位平面编码模块输出的上下文数据, 编解码效率很低, 已经成为基于 JPEG2000^[1~3] 标准图像压缩系统的速度瓶颈^[4,5]. 文献 [6~9] 均采用了基于流水线的电路结构, 这些结构在工作速度上有了一定的提升, 但是在关键路径上依然存在很多较为复杂的逻辑. 文献 [10] 提出了重归一化规程的一种新的顺序结构, 但是未能解决因两级查找表逻辑复杂导致 VLSI 结构性能较低的问题. 近年来, 对 MQ 编

器的研究主要集中在如何实现并行多上下文编码, 如文献 [11~17] 提出了可以并行处理两个甚至多个上下文的 MQ 编码器. 这些设计虽然上提高了编码器的吞吐量, 但是增大了电路的复杂度和面积, 制约了编码器的性能.

本文从减弱上下文 x 之间的依赖性, 化简逻辑判断分支数, 分离大小概率事件的处理, 降低最长计算路径运算单元的使用率等四个角度, 对基于 JPEG2000 标准的 MQ 编码算法中的运算流程, 函数 \sum_k 与 $\bar{p}(\sum_k)$, 条件交换和重归一化算法四个方面进行了优化. 同时,

依据此算法,本文提出了一种采用四级流水线结构的高速串行 MQ 编码器 VLSI 结构.该结构采用 Altera 公司的 STRATIX 型号 FPGA 完成了综合和实现,结果表明该 VLSI 结构工作频率为 184MHz,约为现有串行 MQ 编码器硬件结构的 1.2 倍,并行 MQ 编码器的 2 倍.此外,本文使用 Synopsys 公司的 Design-Compile 基于 TSMC 0.18 μm CMOS 工艺 ARM 公司单元工艺库进行综合,结果表明本文提出的结构能够工作在 532.91MHz,相比 Dyer 提出的 Brute-force with modified 结构,工作频率提高 59%,吞吐量提高近 27%,且面积仅为其四分之一.

2 MQ 编码器概述

MQ 编码器用来将输入的上下文和二元判决所组成的序列映射成单个压缩的码流 CD^[3],算法如图 1 所示.

```

初始化:  $A \leftarrow 8000h$ ,  $C \leftarrow 0$ ,  $\bar{t} \leftarrow 12$ ,  $\bar{T} \leftarrow 0$ ,  $L \leftarrow -1$ 
设  $\bar{p} = \bar{p}(\sum_k)A = A - \bar{p}$ 
If  $x = s_k$  (对一个 MPS 最可能的记号编码)
    If  $A \geq 2^{15}$ ,  $C \leftarrow C + \bar{p}$  (没有重归一化,从而没有条件交换)
    Else if  $A < \bar{p}$ ,  $A \leftarrow \bar{p}$  (有条件交换)
        Else  $C \leftarrow C + \bar{p}$ 
         $\sum_k \leftarrow \sum_{mps}(\sum_k)$ 
    Do, (执行重归一化平移)
         $A \leftarrow 2A$ ,  $C \leftarrow 2C$ ,  $\bar{t} \leftarrow \bar{t} - 1$ 
        If  $\bar{t} = 0$ , byteout( $\bar{T}$ ,  $C$ ,  $L$ ,  $\bar{t}$ )
    While  $A < 2^{15}$ 
Else if  $A < \bar{p}$ ,  $C \leftarrow C + \bar{p}$  (有条件交换)
Else  $A \leftarrow \bar{p}$ 
 $s_k \leftarrow s_k \oplus X_s(\sum_k)$ ,  $\sum_k \leftarrow \sum_{lps}(\sum_k)$ 
Do, (执行重归一化平移)
     $A \leftarrow 2A$ ,  $C \leftarrow 2C$ ,  $\bar{t} \leftarrow \bar{t} - 1$ 
    If  $\bar{t} = 0$ , byteout( $\bar{T}$ ,  $C$ ,  $L$ ,  $\bar{t}$ )
While  $A < 2^{15}$ 
  
```

图 1 MQ 编码算法

$s_k \in \{0, 1\}$ 为二元判 k 决标识 MPS(最可能编码符号); \sum_k 是一个范围为 0 到 46 之间的 6 比特的索引值. 函数 $\bar{p}(\sum_k)$, $\sum_{mps}(\sum_k)$, $\sum_{lps}(\sum_k)$ 和 $X_s(\sum_k)$ 构成一个查找表,其中,函数 $\bar{p}(\sum_k)$ 为上下文 x_n 所对应的概率估计值;函数 $\sum_{mps}(\sum_k)$ 和 $\sum_{lps}(\sum_k)$ 分别标识大概率编码环境 MPS($k_n = s_n$)和小概率编码环境 LPS($k_n = 1 - s_n$)情况下 \sum_k 的更新值;函数 $X_s(\sum_k)$ 只有 LPS 编码环境下才执行,它表示 MPS 和 LPS 编程环境是否交换,即 s_k 是否应由 $1 - s_k$ 替

换.图 1 所示的重归一化平移步骤负责对编码寄存器 A 和寄存器 C 更新,初始化 $A = 2^{15}$, $C = 0$.此外,设置计数器 $\bar{t} = 12$,在寄存器 A 和 C 左移 1bit 时的同时自减 1,直到 $\bar{t} = 0$ 时,进行码流输出.在编码过程中,编码效率的最大损失出现在 LPS 概率接近 1/2 时,这时为了保证分配给 MPS 的区间大于分配给 LPS 的区间,在 $\bar{p}_n > \frac{1}{2} A_n \in [\frac{1}{4}, \frac{1}{2})$ 时,对 MPS 和 LPS 进行条件交换^[14].

3 MQ 编码算法优化

为了解决 MQ 编码算法实现为硬件结构时的困扰,本文对算法进行了优化,使其更易于被转化为高性能的硬件结构,提升硬件电路的编码效率和吞吐量,降低硬件资源使用率和电路面积.本文从以下角度对 MQ 编码算法进行了优化:

(1)减弱上下文 x 之间的依赖性.根据图 1 所示的编码算法,本文有以下定义:

定义 给定上下文 x ,对上下文 x 编码需已知变量 A 和 \sum ,令变量之间关系为依赖系数 β ,有 $\beta = f(A, \sum)$,其中函数 f 表示编码过程.

由上述定义和图 1 编码算法知,若 $x_n = x_{n-1}$,有 $\beta = f(A_{n-1}, \sum_n)$.故只有得到 x_{n-1} 编码结果 \sum_n 和 A_{n-1} ,才能对 x_n 进行编码.这是硬件 MQ 编码器速度和效率较低的主要原因.若存在 $\beta' = f(A_{n-2}, \sum_{n-1})$,则可减弱上下文 x 之间的依赖性,提升整体性能.

(2)化简逻辑判断分支数.由图 1 知, A 值和 C 值在重归一化平移前需被执行的最大条件判断次数为 $N = 3$,导致 A 值和 C 值赋值方式分支数 $M = 2^N = 8$.故若降低条件判断次数 N ,则可减小赋值分支数 M ,在硬件实现中可降低硬件逻辑复杂度,节约硬件资源使用率,减小 MQ 编码器硬件电路的面积.

(3)分析编码器中各种不同编码环境被执行的概率,对概率较低和概率较高的情况分别单独处理,提升各自的执行效率,减小硬件电路中的传播延时.

(4)降低算法的最长计算路径中如加法器,减法器,移位器等运算单元的使用率,提升电路性能.

依据以上原则,本文将 MQ 编码算法划分为 $\bar{p}(\sum_k)$ 运算、条件交换、重归一化和 \sum_k 更新 4 个阶段,如图 2 所示.

3.1 \sum_k 与 $\bar{p}(\sum_k)$ 优化

编码环境 MPS 总是引起 \sum_k 增加,而 LPS 则引起 \sum_k 减小.其中 LPS 使 \sum_k 向下漂移量为 k ,满足: $p'k$

$= (1 - p') \int_{\frac{1}{2}}^{\frac{1}{2} + p'} 2dx = (1 - p') 2\alpha p'$, 其中 p' 为 LPS 概率, α 为一近似系数. 由此可得: $k = 2\alpha(1 - p') = 1.5(1 - p')$. 故可知, 漂移量 k 与 p' 有关, 并不是一个固定的常数. 同样, MPS 引起 \sum_k 的向上漂移量也是一个变化的量. 因此, 将 MPS 和 LPS 更新映射为一个函数, 然后将其从查找表中分离, 这样的方法在硬件电路中反而会消耗更多的资源, 增大传播延时.

初始化: $A \leftarrow 8000h$, $C \leftarrow 0$, $t \leftarrow 12$, $T \leftarrow 0$, $L \leftarrow 1$

If ($x_k = x_{k-1}$)

If ($k_{k-1} = s_{k-1}$) $\bar{p}'_k \leftarrow \bar{p}(\sum_{mps}(\sum_{k-1}))$

Else $\bar{p}'_k \leftarrow \bar{p}(\sum_{lps}(\sum_{k-1}))$

If ($A_{k-2} - \bar{p}_{k-1} \geq 2^{15}$) $\bar{p}_k \leftarrow \bar{p}_{k-1}$

Else $\bar{p}_k \leftarrow \bar{p}'_k$

Else $\bar{p}_k \leftarrow p(\sum_k)$

If ($(A_{k-1} - \bar{p}_k \geq \bar{p}_k) \cdot (x_k = s_k)$)

$\{A_k = A_{k-1} - \bar{p}_k, C_k = C_{k-1} + \bar{p}_k\}$

Else $\{A_k = \bar{p}_k, C_k = C_{k-1}\}$

Do, 执行重归一化

If ($A_k - \bar{p}_k < 2^{15}$)

If ($k = s_k$) $\sum_k = \sum_{mps}(\sum_k)$

Else

$\{\sum_k = \sum_{lps}(\sum_k)\}$

If ($X_s(\sum_k = 1)$) $s_k = s_k \oplus X_s(\sum_k)$

图2 优化后 MQ 编码算法

\sum_{k-1} 的更新方式有以下三种: $\sum_{mps}(\sum_{k-1})$, $\sum_{lps}(\sum_{k-1})$ 或保持不变. 因此, 可通过 \sum_{k-1} 的更新方式确定 \sum_k 可能的值, 这样不用等待 x_{n-1} 编码完成就可给出 \sum_k 的值, 减弱 x_n 对 x_{n-1} 的依赖. 据此, 本文对 \sum_k 和 $\bar{p}(\sum_k)$ 函数做如下优化:

首先, 根据 k_{k-1} 和 s_{k-1} 的是否相等的关系, 有: 若 $k_{k-1} = s_{k-1}$, 则强制有以下映射关系: $\sum_{k'} = \sum_{mps}(\sum_{k-1})$, $\bar{p}'_k = \bar{p}(\sum_{mps}(\sum_{k-1}))$; 若 $k_{k-1} \neq s_{k-1}$, 则强制满足以下映射关系: $\sum_{k'} = \sum_{lps}(\sum_{k-1})$, $\bar{p}'_k = \bar{p}(\sum_{lps}(\sum_{k-1}))$. 其次, 根据 A_{k-1} 与 \bar{p}_{k-1} 对以上映射关系进行纠正: 若 $A_{k-2} - \bar{p}_{k-1} \geq 2^{15}$, 则说明对 x_{n-1} 的编码并没有引起 MPS 和 LPS 条件交换, 因此对上一步骤得出的结论进行纠正: $\sum_k = \sum_{k-1}$, $\bar{p}(\sum_k) = \bar{p}(\sum_{k-1})$; 若 $A_{k-2} - \bar{p}_{k-1} < 2^{15}$, 则说明 x_{n-1} 引起了 MPS 和 LPS 条件交换, 故上一步骤

得出的结论为: $\sum_k = \sum_{k-1}$, $\bar{p}(\sum_k) = \bar{p}(\sum_{k-1})$. 根据此优化后的算法, 上下文 x 之间的依赖系数为 $\beta' = f(A_{k-2}, \sum_{k-1})$, 且 $f_k = f(x_k, \beta')$ 成立.

3.2 条件交换优化

根据 MQ 编码算法可知, 在 MPS 编码环境下, 若满足 $A - \bar{p} < 2^{15}$ 且 $A - \bar{p} < \bar{p}$, 则进行条件交换; 在 LPS 编码环境下, 若满足 $A - \bar{p} < \bar{p}$, 则进行条件交换. 事实上, 因为 $\bar{p} < 2^{15}$, 所以若 $A - \bar{p} < \bar{p}$, 则一定有 $A - \bar{p} < 2^{15}$. 因此, 综合考虑编码环境的确定条件: x_k 与 s_k 是否相等, 可将条件交换的判定条件优化为: 若满足 $(A_{k-1} - \bar{p}_k \geq \bar{p}_k) \cdot (x_k = s_k) = 1$, 则必有 $A_k = A_{k-1} - \bar{p}_k$, $C_k = C_{k-1} + \bar{p}_k$; 否则, 必有 $A_k = \bar{p}_k$, $C_k = C_{k-1}$.

此优化条件下, 条件判断次数为 $N = 1$, A 值和 C 值赋值方式分支数 $M = 2^N = 2$, 因此减少了因条件交换引起的多级 if-else 逻辑判断, 在硬件实现中可减小逻辑链的长度, 降低传播延时. 此外, 将条件 $A - \bar{p} < 2^{15}$ 优化, 在硬件实现中, 可省去一个 16bit 的减法器和一个 16bit 的比较器, 能够有效地减小关键路径上的延时.

3.3 重归一化优化

MQ 编码算法通过循环检测 A 是否满足 $A < 2^{15}$, 来执行左移操作. 直至 $A \geq 2^{15}$, 停止左移, 完成对寄存器 A 和 C 的重归一化运算. 在硬件实现中, 该 while 语句的循环算法中每一种情况都将翻译成一个单独的电路. 为了减小硬件实现代价, 本文对此进行了如下优化.

如第 1 节所述, 在编码过程中, 为了保证分配给 MPS 的区间大于分配给 LPS 的区间, 在 $\bar{p}'_n > \frac{1}{2} A_n \in [\frac{1}{4}, \frac{1}{2})$ 时, 对 MPS 和 LPS 进行条件交换. 事实上, 根据图 3 所示的 LPS 概率 \bar{p}'_n 分布图可以看出, \bar{p}'_n 接近 $1/2$ 的区域很小, 即大部分情况下编码都处于 MPS 环境下且无条件交换, 即 $A_k = A_{k-1} - \bar{p}_k$, $C_k = C_{k-1} + \bar{p}_k$, 而 $A_k = \bar{p}_k$, $C_k = C_{k-1}$ 则很少会被执行. 这一结论在 MQ 编码仿真中也得到了证实. 因此, 本文对这两种情况进行单独处理, 使最少发生的情况不加载于最常发生的情况, 二者在各自的情况下都能达到最优的处理.

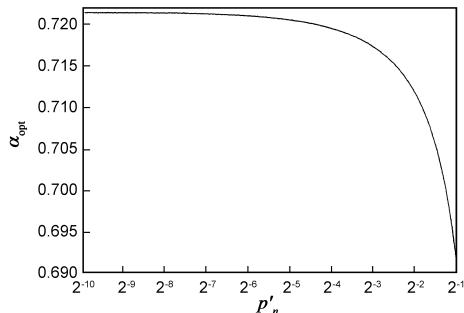


图3 LPS 概率 \bar{p}'_n 概率分布图

在大概率事件下, $\max(\bar{p}(\sum_k)) = 5601_h$, 又 $A \geq 8000_h$, 因此可得 $\min(A - \bar{p}(\sum_k)) = 29FF$. 这意味着在 $A_k = A_{k-1} - \bar{p}$ 被执行的情况下, 前导零个数 $\max(b_n) = 2$, 即 $b_n = \{0, 1, 2\}$. 故可知, $A_k = A_{k-1} - \bar{p}$, $C_k = C_{k-1} + \bar{p}$ 情况下, 前导零个数 b_n 满足以下公式: 若 $A \geq \bar{p} + 8000_h$, 则 $b_n = 0$; 若 $\bar{p} + 8000_h > A \geq \bar{p} + 4000_h$, $b_n = 1$; 其他情况, $b_n = 2$. 而编码过程中出现概率较小的 $A_k = \bar{p}$, $C_k = C_{k-1}$ 情况, 则可预先在查找表中扩展一函数 $b(k)$, 负责统计 \bar{p} 所对应前导零个数. 这样, 在 $A_k = \bar{p}$, $C_k = C_{k-1}$ 被执行的情况下, 就可以直接获得此时的前导零个数 b_n . 获得前导零个数 b_n 后, 可直接对寄存器 A 和 C 左移相应的位数, 同时计数器 i 也直接减去 b_n .

采用该优化后的算法, 不仅避免了 while 循环在软件算法中造成的多次迭代, 更重要的是, 对大小概率事件分别进行单独处理, 降低了硬件电路中各自路径的传播延时. 而且在优化后的算法中, 在最长的计算路径(大概率事件的处理)中, 前导零检测器已经被更为简单的 b_n 所代替. 在硬件结构中, 左移固定 bit 数可用位拼接方法代替, 故移位器、前导零检测器等复杂的逻辑运算单元都被较简单的逻辑电路代替, 节省了硬件资源, 简化了运算逻辑.

4 MQ 编码器 VLSI 结构设计

4.1 四级流水结构

基于以上优化后的 MQ 编码算法, 本文提出了一种四级流水结构 VLSI MQ 编码器, 其中每一级流水都以寄存输出, 如图 4 所示:

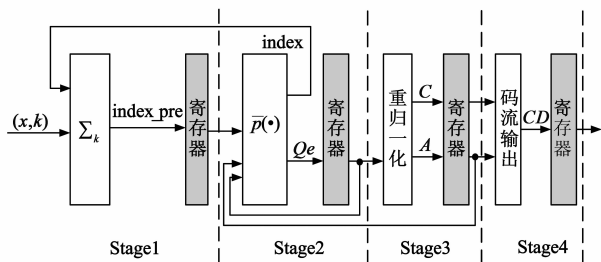
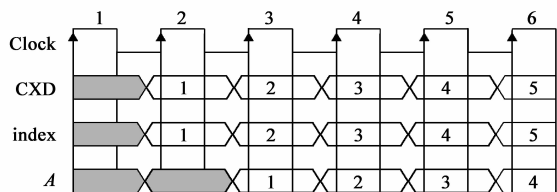


图4 四级流水MQ编码器VLSI设计结构

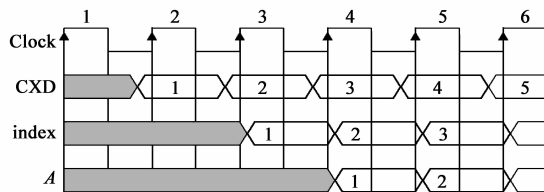
Stage1 完成索引值 \sum_k 查找, 以 2.1 节提出的算法为基础, 获得当前上下文 x_k 所对应的可能的索引值 $\sum_{k'}$, 即硬件线名 index_pre . 同时采用图 2 所示的索引值更新算法对前一上下文 x_{k-1} 的索引值进行更新. Stage2 为概率估计值 $\bar{p}(\sum_k)$ 查找, 该级采用 2.1 节的算法, 获得正确的概率估计值 Qe , 同时将该 x_k 所对应的正确的索引值 index 写回到 stage1 中, 参与索引值更新. 此外, 该流水级采用 2.2 节中的算法, 给出寄存器 A

和 C 归一化的方式, 传递给 stage3. Stage3 为重归一化, 该级以 2.3 节提出的重归一化算法, 完成寄存器 A 和寄存器 C 的重归一化. Stage4 为码流输出, 由于该级的运算不会对关键路径和电路的性能造成影响, 因此该级依然采用基于 JPEG2000 的标准算法实现硬件电路.

采用图 4 所示的流水结构, 本文所提出的 MQ 编码器各关键信号的时序关系如图 5(b) 所示. 由图 5(b) 可知, 编码器对当前待处理的上下文 CXD, 即 x_n 的处理过程需已知两个变量, 分别为其前一个周期的 index , 即 \sum_{n-1} , 和其前两个周期的 A , 即 A_{n-2} . 由图 5(a) 和图 5(b) 对照可知, 在硬件实现中, 本文提出的硬件结构将上下文 x 之间的依赖系数由 $\beta = f(A_{n-1}, \sum_n)$ 改进为 $\beta' = f(A_{n-2}, \sum_{n-1})$, 则可减弱上下文 x 之间的依赖性.



(a) 标准算法中关键信号时序图



(b) 改进后关键信号时序图

图5

4.2 \sum_k 和 $\bar{p}(\sum_k)$ 电路结构

根据 2.1 节提出的 \sum_k 和 $\bar{p}(\sum_k)$ 优化算法, 本文在 stage1 中完成对索引值 \sum_k 和 $\bar{p}(\sum_k)$ 的预测, 即计算 $\sum_{k'}$ 和 $\bar{p}(\sum_{k'})$. 然后在 stage2 中完成二者的纠正, 即得出正确的 \sum_k 和 $\bar{p}(\sum_k)$, 电路结构如图 6 所示.

为了简化电路的运算逻辑, 本文对第一级查找表进行扩充, 将其扩充为 \sum_k , $\sum_{\text{mps}}(\sum_k)$ 和 $\sum_{\text{lps}}(\sum_k)$, 同时将第二级查找表值保留 $\bar{p}(\sum_k)$. 这样仅仅通过一次查找, 就可以获得当前上下文 x_k 所有可能的索引值. 且在硬件电路中, 使两级查找表的扇出得到平衡, 降低了标准算法中第二级查找表由于大扇出带来的延时.

为了使得索引表 \sum_k 能够正确更新, 本文将 stage2 产生的索引值以组合逻辑的形式反馈到 stage1 中. 如果

该值经过寄存后再反馈到 stage1 中,该值将会比 stage1 当前输入的上文晚两个时钟周期.在这种情况下,需

要对上下文的索引值提前两个周期进行预测,该逻辑将非常复杂,不利于硬件实现.

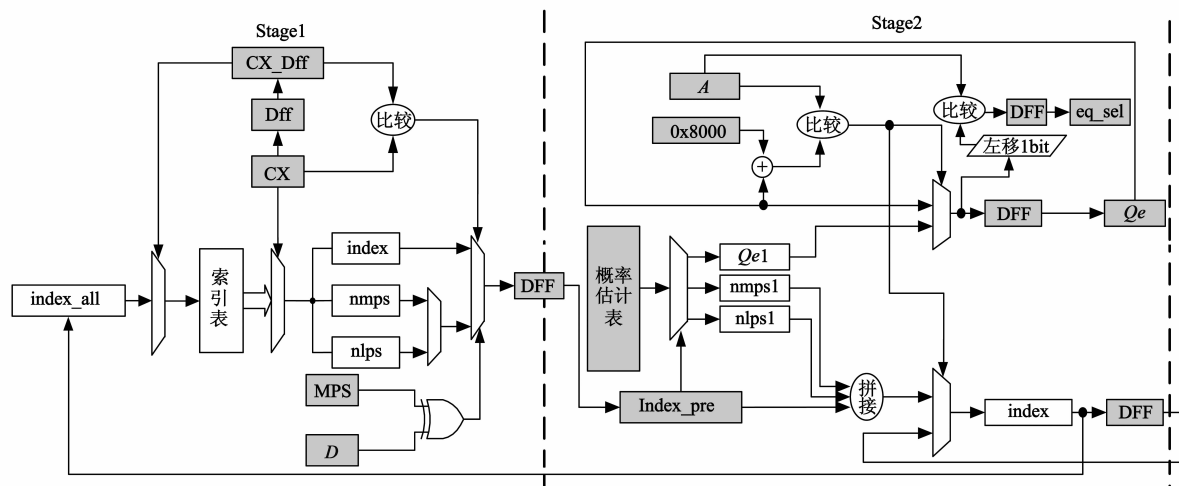


图6 stage1和stage2电路结构

4.3 重归一化电路结构

stage3 硬件电路完成两个功能:(1)编码区间的重整.根据 2.2 节提出的条件优化算法,完成 A 和 C 的计算;(2)依据 3.3 提出的重归一化算法,对重整之后的 A 和 C 完成重归一化运算.电路结构如图 7 所示.

通过以上重归一化移位预判机制,重归一化运算逻辑被大大简化.在传播延时最大的路径中,移位器、前导零检测器等复杂的逻辑运算单元都被较为简单的逻辑电路代替.

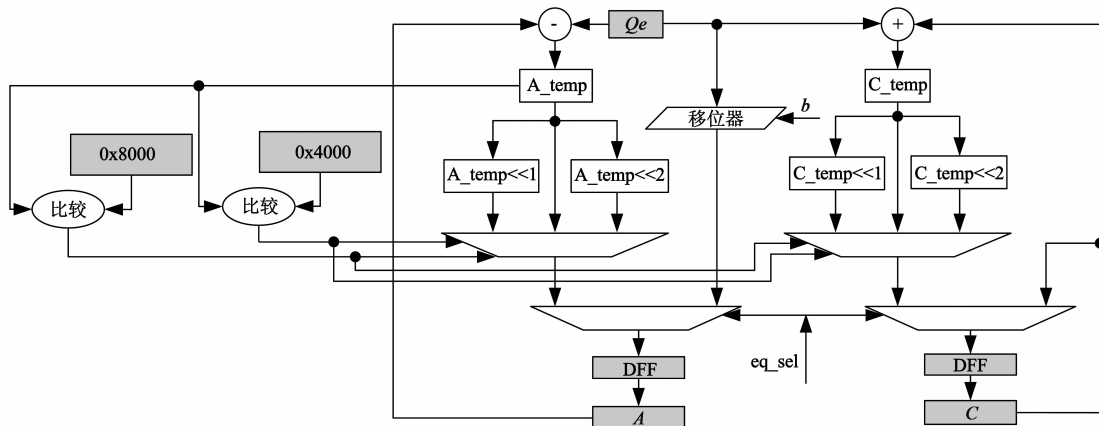


图7 重归一化电路结构

5 实验结果分析

本文使用 Verilog HDL 语言对提出的 MQ 编码器电路结构进行了描述,使用 modelsim6.5b 将仿真生成的码流打印到 .txt 文本中.经过对比,本文提出的算法编码结果同标准 MQ 编码算法结果完全相同.本文采用 Altera 公司的 STRATIX 型号 FPGA 器件完成了综合、实现以及功能测试,并且与其他有代表性的 MQ 编码器结构进行了性能对比,如表 1 所示.

通过表 1 中的数据可知,相比文献[6]串行结构的

MQ 编码器,本文的工作频率提高了 20% 左右,而且面积几乎相等.与文献[12]、文献[15]提出的多上下文并行处理 MQ 编码器相比,本文的吞吐量提高近 2 倍,且资源使用率更小.

本文使用了 Synopsys 公司的 Design-Compiler 工具基于 TSMC 0.18 μ m CMOS 工艺 ARM 公司高密度单元工艺库完成了对电路结构 ASIC 平台的性能评估,如表 2 所示.由表 2 可知,相比文献[7]提出的串行 MQ 编码器,本文的工作频率提高了 23% 左右.相比 Dyer 提出的 Brute force with modified 结构,本文的工作频率提高

59%,吞吐量提高近 27%,且面积仅为其四分之一。

表 1 Altera STRATIX 平台性能对比

Architecture	时钟频率 MHz	频率 归一化	吞吐量 Msymbols/sec	吞吐量 归一化	LEs
文献[6]	153.00	0.83	153.00	0.83	279
文献[9]	145.90	0.79	171.44	0.93	824
Two Symbol ^[12]	48.30	0.26	96.60	0.52	12649
Brute force with modified byteout	48.85	0.26	97.70	0.53	1596
Brute force	36.90	0.20	73.80	0.40	2265
3CXD hypothesis	45.53	0.25	60.78	0.33	1259
3CXD hypothesis with queue	49.50	0.27	71.48	0.39	1744
Bit-plane Coder	61.06	0.33	73.20	0.40	1247
Two Symbol ^[15]	58.56	0.31	117.12	0.63	1488
本文	184.64	1	184.64	1	287

表 2 TSMC0.18um 工艺性能对比

Architecture	时钟频率 MHz	频率 归一化	吞吐量 Msymbols/sec	吞吐量 归一化	面积 μm^2
文献[7]	413.00	0.77	413.00	0.77	8.8K 等效门
Two Symbol ^[12]	220.10	0.41	440.20	0.83	321415.03
Brute force with modified byteout	211.86	0.40	423.72	0.80	381345.14
Brute force	194.17	0.36	388.34	0.73	384817.91
3CXD hypothesis	201.61	0.38	268.14	0.50	248033.02
3CXD hypothesis with queue	205.34	0.39	301.85	0.57	268699.94
Bit-plane Coder	261.78	0.49	314.14	0.59	1255083.98
本文	532.91	1	532.91	1	82478.10

6 结论

本文从硬件实现的角度,对基于 JPEG2000 标准的 MQ 编码算法进行优化,主要优点在于:(1)利用上下文 x 之间的联系,将 MQ 编码算法划分为 $\bar{p}(\sum_k)$ 运算、条件交换、重归一化和 \sum_k 更新 4 个阶段;(2)优化了函数 \sum_k 与 $\bar{p}(\sum_k)$,将上下文依赖性由 $\beta = f(A_{k-1}, \sum_k)$ 改进为 $\beta' = f(A_{k-2}, \sum_{k-1})$,减弱上下文 x 之间的依赖性,提高了电路的并发执行能力;(3)降低了条件判断次数,减小了赋值分支数,充分降低了硬件逻辑复杂度和资源使用率;(4)对大小概率事件分别进行单独处理,降低了硬件电路中各自路径的传播延时。电路综合结果表明,本文设计的 MQ 编码器,不仅比同类的串行编码器具有更高的工作频率,更小的面积,而且与近年来有代表性的并行编码器相比,也具有高吞吐量。

参考文献

[1] M D Adams. The JPEG – 2000 Still Image Compression Stan-

dard[S]. Sep. 2001.

[2] ITUT. JPEG2000 Image Coding System Part1[S]. July 2002.

[3] 刘雷波,王学进,孟鸿鹰,王志华,陈弘毅,夏宇闻. JPEG2000 小波变换器的 VLSI 结构设计[J]. 电子学报, 2002,30(11):1609 – 1612.

Liu Leibo, Wang Xuejin, Meng Hongying, Wang Zhihua, Chen Hongyi, Xia Yuwen. A VLSI architecture of DWT for JPEG2000[J]. Acta Electronica Sinica, 2002, 30(11): 1609 – 1612. (in Chinese)

[4] M D Adams, F Kossentini. Jasper: a software-based JPEG – 2000 codec implementation[A]. Proceedings 2000 International Conference on Image Processing[C]. Canada, 2000. 2: 53 – 56.

[5] 朱悦心,付昀,吴宗泽,郑南宁. 基于多级查询表的 JPEG2000 位平面扫描优化方法[J]. 电子学报, 2004, 32(5):810 – 813.

Zhu Yuexin, Fu Yun, Wu Zongze, Zheng Nanning. A JPEG2000 optimized bit-plane scanning method based on multilevel query table[J]. Acta Electronica Sinica, 2004, 32(5): 810 – 813. (in Chinese)

[6] Sarawadekar. K, Banerjee. S. Low-cost high-peformance VLSI of an MQ coder for JPEG2000[A]. 2010 10th International Conference on Signal Processing Proceedings (ICSP 2010) [C]. China, 2010:397 – 400.

[7] Minsoo Rhu, In-Cheol Park. Optimization of arithmetic coding for JPEG2000[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2010, 20(3):446 – 451.

[8] Mrs. I. Flavia Princess Nesamani, Dr. C. Vasanthanayaki. Implementation of simplified architecture of JPEG2000 MQ coder [A]. 2009 International Conference on Control Automation, Communication and Energy Conservation, INCACEC 2009 [C]. India, 2009:1 – 6.

[9] Kishor Sarawadekar, Swapna Banerjee. VLSI design of memory-efficient, high-speed base line MQ coder for JPEG2000[J]. INTEGRATION, the VLSI Journal, 2012, 45:1 – 8.

[10] 刘文松,朱恩,王健,徐龙涛,林叶. JPEG2000 算术编码器的算法优化和 VLSI 设计[J]. 电子学报, 2011, 39(11): 2486 – 2491.

Liu Wensong, Zhu En, Wang Jian, Xu Longtao, Lin Ye. Optimization algorithm and VLSI design of the arithmetic coder in JPEG2000[J]. Acta Electronica Sinica, 2011, 39(11): 2486 – 2491. (in Chinese)

[11] N Noikaew, O Chitsobhuk. Dual symbol processing for MQ arithmetic coder in JPEG2000 [A]. 2008 International Congress on Image and Signal Processing (CISP) [C]. China, 2008. 1:1521 – 524.

[12] Kai Liu, Yu Zhou, Yun Song Li, Jian Feng Ma. A high performance MQ encoder architecture in JPEG2000[J]. Integration, the VLSI journal, 2010, 43:305 – 317.

- [13] Gupta, A. K., Taubman, D. High speed VLSI architecture for bit plane encoder of JPEG2000[A]. Midwest Symposium on Circuits and Systems[C]. Japan, 2004. 2: II233 – II236.
- [14] M Dyer, D Taubman, S Nooshabadi, A Kumar Gupta. Concurrency techniques for arithmetic coding in JPEG2000[J]. IEEE Transactions on Circuits and Systems—I: Regular Papers, 2006, 53(6): 1203 – 1213.
- [15] PENG Zhou, ZHAO Bao-jun. High-throughout hardware architecture of MQ arithmetic coder[A]. 2010 10th International Conference on Signal Processing Proceedings (ICSP 2010) [C]. China, 2010: 430 – 433.
- [16] Tinku Acharya, Ping-Sing Tsai. JPEG2000 Standard for Image Compression Concepts, Algorithms and VLSI Architectures [M]. New Jersey: John Wiley & Sons INC, Publication, 200: 185 – 196.
- [17] David S. Taubman, Michael W. Marcellin. JPEG2000 Image Compression Fundamentals, Standard and Practices[M]. America: Kluwer Academic Publishers, 2001: 56 – 77, 473 – 483.

作者简介



邱志雄 男, 1984 年 10 月出生于山西忻州. 西安电子科技大学微电子学院博士研究生. 研究方向为 SoC 设计方法学、HDL 源代码质量评估方法学等.

E-mail: zxdli@mail.xidian.edu.cn



史江义 男, 1973 年 9 月出生于陕西韩城, 西安电子科技大学微电子学院副教授. 研究方向为 SoC 设计方法学及物理实现.