

一种适用于 H.264 标准的高度并行双层 流水线结构 CAVLC 编码器

乔 飞, 魏鼎力, 杨华中, 汪 蕙

(1. 清华信息科学与技术国家实验室, 北京 100084; 2. 清华大学电子工程系, 北京 100084)

摘 要: 本文提出一种适用于 H.264 编码器的高度并行、双层流水线的 CAVLC 硬件实现结构. 该结构设计了四路并行扫描统计模块, 克服了以往结构每个时钟周期只能扫描一个系数的处理速率瓶颈; 通过使用 FIFO, 平衡每一级流水线的处理延时, 提高整个流水线工作的效率; 在各个编码模块内部也大量采用流水线结构, 提高数据吞吐率. 基于 $0.18\mu\text{m}$ CMOS 工艺, 新结构在 166.7MHz 工作频率下, 综合等效门数为 20685 门, 数据吞吐率为每秒处理 27M 系数块, 甚至能够实时编码数字影视格式的视频 ($4096 \times 2048 @ 30\text{fp/s}$). 整个设计在数据吞吐率提高到以往结构的 3.46 倍的同时, 硬件资源代价并没有显著的增加.

关键词: H.264; 基于上下文的自适应变长编码; 编码器

中图分类号: TN402 **文献标识码:** A **文章编号:** 0372-2112 (2010) 07-1705-06

Design of a Highly Parallel and Double-Level Pipelined CAVLC Encoder for H.264

QIAO Fei, WEI Ding-li, YANG Hua-zhong, WANG Hui

(1. Tsinghua National Laboratory for Information Science and Technology, Beijing 100084, China;

2. Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

Abstract: This paper presents the design of a CAVLC encoder for H.264 featuring a highly parallel and double-level pipelined architecture. In order to overcome the speed bottleneck of one coefficient per cycle during scanning, the proposed design uses four-channel parallel processing instead of serial scanning. And the delays of all stages in the pipelined architecture are averaged by FIFOs, which achieves a high efficiency for the entire pipeline. The pipelined structure is also widely used in sub-modules for higher throughput. Based on $0.18\mu\text{m}$ CMOS technology, the proposed architecture is synthesized into 20685 logic gates and achieved average 27M blocks/s at 166.7MHz frequency, and even meets the requirements of real-time processing of digital cinema video ($4096 \times 2048 @ 30\text{fp}$). The data throughput of the proposed architecture is 3.46 times of that of the previous reported work with acceptable increase in area.

Key words: H.264; context-adaptive variable-length coding (CAVLC); encoder

1 引言

2002 年基于上下文的自适应变长编码 (CAVLC) 诞生, 并被国际电信联盟 (ITU) 和国际标准化组织 (ISO) 的联合视频专家组用作新一代视频编码标准 H.264^[1] 的熵编码部分. CAVLC 针对残差系数块数据, 复杂度适中, 比特错误时更容易重同步, 其高度自适应特性为 H.264 整体压缩率的提高做出了相当贡献; 但是, CAVLC 的编码过程既用到整个块的统计特性, 又用要自适应地编码, 数据间依赖性很强, 因而其编码过程基本是顺序串行执行的, 很难并行化处理, 硬件实现时数据处理能力不足. 在处理视频信息, 特别是大尺寸实时

视频信息时, 巨大的待处理数据量和有限的处理能力之间的矛盾非常尖锐, 视频编码器的硬件实现日趋必要^[2,3]. 因此尽可能提高 CAVLC 编码器的数据处理能力, 便成为各种 ASIC 设计实现关注的重点.

文献[4]提出的 CAVLC 编码器在设计幅度编码时提出以算术运算代替查表操作, 并仔细设计了扫描器. 文献[5]将大码表划分为小码表, 简化查表操作. 文献[6]则在变换量化后对系数存储时, 每个像素使用一个标志比特表示数据的特征; 扫描时, 根据这些标识比特快速得到各种统计元素的值, 同时也有选择地读入非零系数, 减少存取功耗. 这三种结构没有采用流水设计, 因而工作频率不高, 数据处理率也很低.

文献[7]开始使用 FIFO 寄存统计信息,将扫描环节和查表编码环节分开,以期获得较高的数据吞吐率.文献[8]也采用了相似的块级流水结构;两个 Buffer 以乒乓方式工作,在扫描环节和查表编码环节之间无延时地传递数据.文献[9]针对统计元素中非零系数幅度信息最丰富、编码最耗时的问题,安排了两个幅度编码模块并行工作;在扫描阶段,将非零系数存进串入并出的 FIFO(SIPO);在统计元素编码阶段,用两个幅度编码模块分别处理 SIPO 两个输出端口的数据.文献[10]注意到在使用不同的编码模块得到相应的码字之后,将这些编码码字拼接成输出码流仍要耗费很多的时钟周期,因而用 Buffer 将封装阶段和查表编码阶段分开.由于块级流水结构的引入,以上四种结构数据吞吐率得到了很大提高,能够完成针对 HD1080 @ 30fp(1920 × 1088)格式的视频编码.

尽管采取块级流水结构可以显著提高数据吞吐率,但是由于扫描阶段处理慢(完成一个块的扫描统计至少要 16 个时钟周期),其他阶段的工作即使提前完成也必须等待,整体数据处理率没有太大提高.扫描统计单元成为了 CAVLC 编码器数据处理率提高的瓶颈.针

对此问题,在本文第二部分提出了一种高度并行、双层流水的 CAVLC 编码模块.该模块扫描部分是四路并行处理的;所有统计元素的编码是并行处理的;在幅度编码器内部也是两路并行处理的.扫描、查表编码、码流封装三个环节之间采用块级流水的办法.在各个模块内部也全都采用了编码元素流水处理的办法,割断关键路径,提高工作频率和数据处理速率.

2 高度并行双层流水的 CAVLC 编码器

为了提高数据吞吐率,本文设计了高度并行双层流水的新型 CAVLC 编码器结构,如图 1 所示;其中,标注了非零系数数目(TotalCoeff)、拖尾数目(TrailingOnes)、夹杂在非零系数中零的数目(TotalZeros)、非零系数的幅度(Level)和游程(Run)等.整个编码器分为块扫描级、统计元素并行编码级、码流封装级三个部分.当统计元素并行编码级的各编码模块对当前系数块的统计元素进行查表编码,并把编码得到的码字存入相应的 FIFO 时,扫描统计模块正在统计下一个系数块的统计元素信息;虽然流水结构花在每个块上的编码时间并没有减少,但整体数据吞吐率却可以大幅提高.

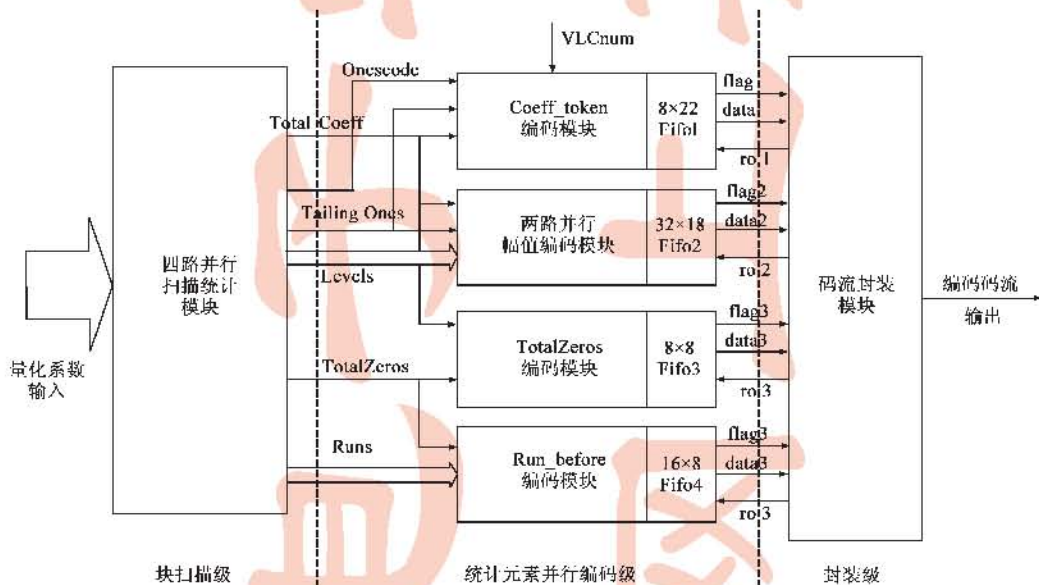


图1 高度并行双层流水CAVLC编码器结构图

2.1 四路并行扫描统计单元

本文设计的四路并行扫描统计模块在每个时钟周期按扫描顺序逆向读入 4 个系数,再得出这个 4 系数序列相应的统计信息.16 个系数扫描完成后,把 4 次得到的统计信息累加起来,并做必要的修正,就得到该系数块全部的统计信息.该扫描统计单元内部仍采用编码系数 4 级流水的结构.图 2 所示是该模块的整体流水结构图.

第一级流水主要是在特定的扫描周期将 4 个合适的量化系数映射到下一级流水线处理.这里的映射网

络实质上是四个 4 选 1 选择器.在映射后,量化系数已经调整为逆扫描顺序了.第二级流水线主要通过组合逻辑得出输入的四个量化系数的统计信息.首先通过四个比较器判断这四个系数是否为 0,得到一个 4 比特的辅助标志 $Aidflag[3:0]$.然后根据 $Aidflag[3:0]$ 由 4 输入的组合逻辑得到三个中间统计信息,中间统计信息 $Smmum$ 表示 $Aidflag[3:0]$ 中 1 的个数; $Smtot$ 表示此四个系数中最后一个非零系数之前的零系数的个数总和; $Smlast$ 表示此四个系数中最后一个非零系数之后的零

个数,同时,跟据 $Aidflag[3:0]$ 将合适的非零系数幅度信息选择传递到寄存器 $Smlev0$ 、 $Smlev1$ 、 $Smlev2$ 、 $Smlev3$ 中。另外,表示游程信息的 $Smrun0$ 、 $Smrun1$ 、 $Smrun2$ 、 $Smrun3$ 也由 4 输入组合逻辑得到。第三级流水则要第二级流水中得到中间统计信息进行汇总。总的非零系数数目等于每次统计的非零系数之和,故将 $Smnum$ 累加到 $TotalCoeff_tmp$ 中。如果已经出现过非零系数,汇总 $TotalZeros_tmp$ 时把 $Smtot$ 和 $Smleft$ 的值都累加进来,同时还应该更新前面统计的最后一个非零系数的游程信息 $Runlast$ 。同时, $Smlev$ 和 $Smrun$ 信息则顺序存入 16 个幅度游程寄存器构成的寄存器组,存入的起始位置由上一个

时钟周期的 $TotalCoeff_tmp$ 决定。事实上,每次存入 $Smlev$ 和 $Smrun$ 信息时,四组数据并不一定全部是有效的。如果此次扫描的 4 个系数中只有两个非零系数,那么实际上也只有两组数据是有效的。虽然每次存入了四组数据,但是 $TotalCoeff_tmp$ 也仅增长了 2,下个周期新的有效的幅值游程是从第三组数据的位置存储的,因而两组无效的幅度游程信息是会被覆盖掉的。第四级流水需要做一些扫描统计的后处理工作。针对 AC 系数块对 $TotalZeros_tmp$ 进行减 1 修正。另外由逆向扫描后存入幅值游程寄存器组的最开始的三个幅度信息生成拖尾的个数 $TrailingOnes$ 和拖尾符号信息 $Onescode$ 。

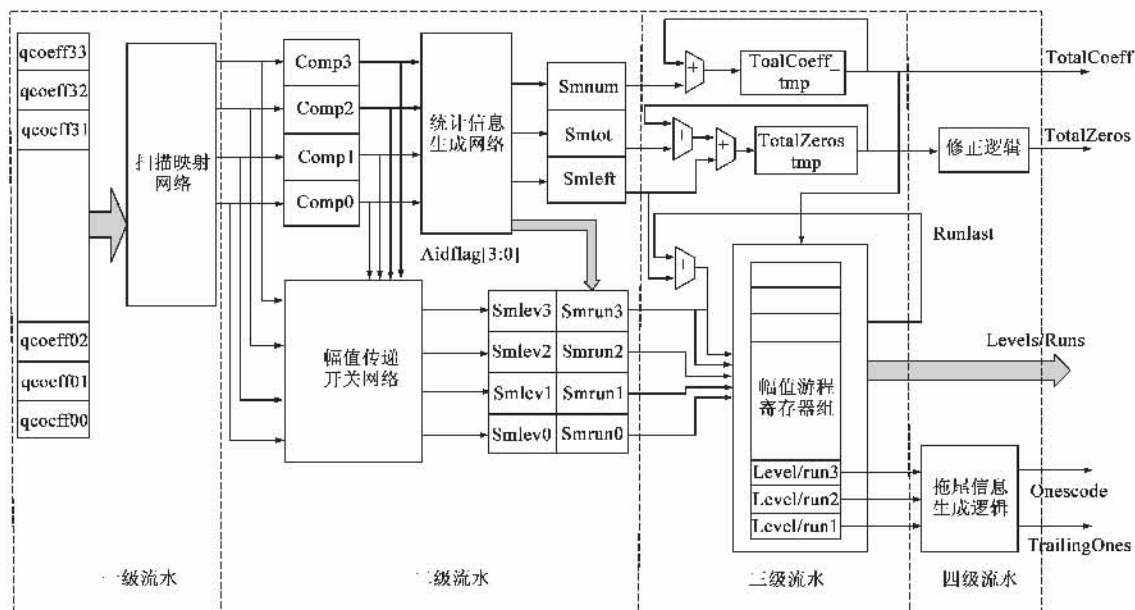


图2 四路并行扫描统计模块流水结构图

整个扫描统计模块的状态转移图如图 3 所示。当开始信号有效后,模块由空闲状态 (idle) 转入扫描状态 (scan1), 如果数据有效信号不为零, 依据输入块的类型分别转到 Scan2 状态或者 Cal1 状态。色度直流块只有 4 个系数, 所以扫描一次就完成了。其余系数块为 4×4 大小, 依次经过 Scan1 到 Scan4 等四个扫描阶段, 进入 Cal1

阶段。由于此模块是流水设计, 前四个系数进入三级流水统计汇总时, 接下来的四个系数正在二级流水生成中间统计信息, 再后边的四个系数正在进行扫描映射。因此, 当 4 个扫描时钟周期结束后, 还需 1 个 Cal1 周期完成最后四个系数的统计信息汇总, 这之后还需一个 Cal2 周期进行数据的修正和拖尾信息的生成。

由于采用了并行扫描技术, 本模块完成一个 4×4 系数块的信息统计只需扫描 4 次, 加上额外的时序开销, 一共耗费 6 个时钟周期。相对于以往每个周期扫描 1 个系数的串行扫描结构, 本模块的数据吞吐率获得了很大提高。

2.2 两路并行幅值编码单元

在统计元素并行编码时, 相对于每个图像块至多只需编码一次的 $Coeff_token$ 、 $TotalZeros$ 信息, 甚至是可以提前结束编码的 $Runs$ 元素, 幅值元素的个数都要多出不少。最坏情况下, 一个图像块需要编码 16 个 $Levels$ 元素。因此, 突破了扫描统计模块的数据处理率瓶颈之

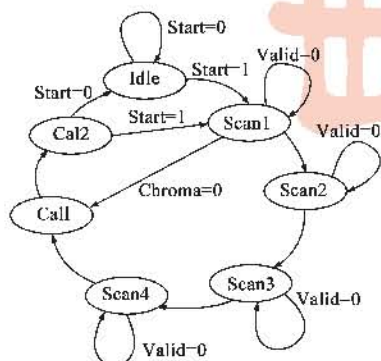


图3 四路并行扫描统计模块状态转移图

后,幅值编码模块的数据处理率就可能成为新的瓶颈.为此本文设计了一个三级流水的两路并行幅值编码

器,如图4所示.

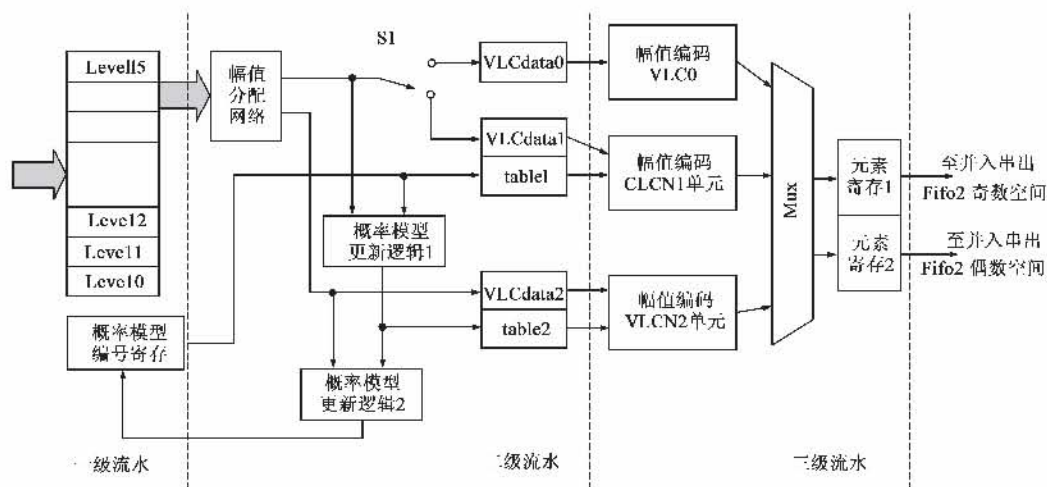


图4 两路并行幅值编码模块

模块的第一级流水主要是寄存好来自扫描单元的幅度信息,并且确定好初始的概率模型.第二级流水主要是准备好两路并行的幅值数据 VLCdata1、VLCdata2 和概率模型编号 table1、table2. CAVLC 的幅值编码一共有七种概率模型,而且要根据当前幅值决定编码下一个幅值时是否更新概率模型.因此,本文采用了两个概率模型更新逻辑,根据标准设定的阈值提前判断是否需要更新概率模型,从而可以并行得到两个幅值编码单元的概率模型号.第三级流水用算术方式对幅度信息进行编码.这里的 VLC0 单元只负责 0 号概率模型的编码,其余的概率模型通过 VLCN1 单元和 VLCN2 单元并行编码.编码时,概率模型编号主要用于确定前缀和后缀的长度分割.根据幅度大小和概率模型编号确定前缀零的数目.幅度的一部分比特编入后缀.前缀、后缀再加上符号信息,形成完整的码字.编码结果存在两个元素寄存器中,并行输出到 Fifo2 中. Fifo2 是一个两端口并行输入串行输出的 FIFO,可以同时写入两个并行数据,输出时却以串行方式输出给码流封装模块.

图5是此两路并行幅值编码模块的状态转移图. Idle 是空闲状态. State1 状态时处理第一个待编码幅值信息.只有第一个幅值信息会用 0 号概率模型,同时当所有非零系数数目大于 3 而拖尾系数数目小于 3 时,第一个编码的幅值信息要首先减 1,再做处理.因而对第一个非零系数采用单独的状态处理.在 State2 状态下,每个时钟周期编码两个幅值,直到所有的幅度信息编码完成,End 信号等于 1 为止.由于采用了流水结构,State2 状态时不仅要准备好当前的 VLCdata1、VLCdata2,还要准备把上一个周期编码得到的码字信息存入 Fifo2 当中.此时特别要注意, Fifo2 可能出现缓存填满, Fifo_full

等于 1 的情况.出现这种情况后,整个幅值编码模块进入 Holdnop 状态,流水线需要全部暂停下来.当 Fifo2 再次空出空间后,整个模块才结束 Holdnop 状态,继续 State2 状态的并行编码.

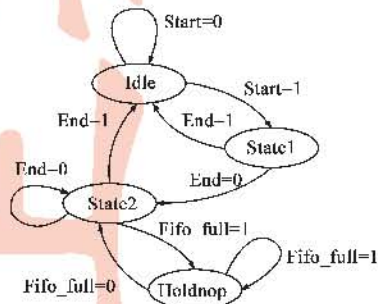


图5 两路并行幅值编码模块状态转移图

2.3 码流封装的多 FIFO 结构

把已经编好的码字合成码流输出便是封装单元的任务.由于实际的系数块不一定有完整的码流结构.依据四个 FIFO 的空标识和各个统计元素编码模块生成并连同码字一起存入 FIFO 的标识,码流封装单元分别从 Fifo1 到 Fifo4 中读取相应的码字,然后将码字封装入输出码流.该封装单元每个周期仅能处理一个码字.

本设计中整个 CAVLC 编码器采用系数块级三级流水线结构.要使流水线达到最佳效率,必须仔细规划每一级流水线的延时,使之比较平均,从而不会出现新的数据吞吐率的瓶颈.本文用运动变化信息最为丰富的 Mobile 序列做了统计,每个系数块生成码字平均为 1.844 个,生成码字数目大于等于 6 的系数块大约占了 10%.因此,从平均效果看,相对于封装模块每个系数块 1.844 个周期的处理时间,本设计 6 个周期的规划流水延时是有很多裕量的.可是,实际上封装模块封装每个

系数块耗费多少时钟周期是由这个块有多少个码字决定的,而不同的块生成的码字数量差别很大.如果一个块生成 10 个码字,那么封装这个块的码字要耗费 10 个周期,而同时对后续的块扫描只耗 6 个周期.于是在流水时,扫描模块就要停下来多等待 4 个周期,从而导致了流水效率的损失.

为了平衡封装模块处理不同块的码字的负担差异,本文设计块级流水结构时,在统计元素并行编码级与码流封装级间添加了 4 个 FIFO.块扫描级与统计元素并行编码级是统一动作的.只有当这两级电路都处理完一个流水系数块之后,才同时开始下一个流水系数块的处理.封装模块则采用不同的动作机制,只要 FIFO 里有未封装的码字,封装模块就不停地工作,直到 FIFO 为空.这样做的好处在于,如果遇到码字特别多的模块,一个流水周期内封装不了的多余码字将暂存在 FIFO 中,以后占用需封装码字少的块的部分流水周期来完成多余码字的封装.

3 仿真综合结果与性能比较

本文设计的 CAVLC 编码器采用了大量并行技术和双层流水的技术,提高数据吞吐率.由于仔细设计了使用 FIFO 的块级流水结构,出现流水线暂停的情况大大

减少了.本文采用了运动剧烈程度不同的 6 种 HDTV1080(1920×1088@30fp)格式的视频进行了仿真验证,结果如表 1 所示.

从表 1 中可以看到,整个 CAVLC 的平均数据块处理时间为每个块耗费 6.1590 个时钟周期.比起不出现流水线暂停时每个块 6 个时钟周期的标准耗费,本设计耗费的时钟周期并没有太多的增加.这证明采用 FIFO 缓冲后,块级流水暂停的情况很少.由于本设计大量采用流水结构,没有太长的延时路径,所以能够工作在较高频率.采用 0.18μm CMOS 工艺综合优化,本设计中 CAVLC 编码器在工作频率 166.7MHz 时,等效门数为 20685 门.

表 1 本设计对各种视频序列的数据块处理周期数

	块处理周期数(cycles/block)
Area	6.1586
Blue_sky	6.1596
Riverbed	6.1650
Rush_hours	6.1570
Stations	6.1565
Sunflowers	6.1581
Average	6.1590

现在表 2 中列出了文献[4~10]的各种 CAVLC 设计的性能,与本设计的性能加以对比.

表 2 各种 CAVLC 编码器 ASIC/FPGA 设计性能对比

	文献[4]	文献[5]	文献[6]	文献[7]	文献[8]	文献[9]	文献[10]	本文
工艺(μm)	0.35	FPGA Virtex-II	0.18	FPGA	0.18	0.18	FPGA Stratix-II	0.18
扫描路数	1	1	1	1	1	1	1	4
是否封装	是	是	是	是	是	否	是	是
是否流水	否	否	否	是	是	是	是	是
幅值编码路数	1	1	1	1	1	2	1	2
综合门数	9171	15363	26598	17207	23584	9724	N.A.	20685
工作频率(MHz)	67	60	27	100	100	125	109.8	166.7
块处理周期数(cycles/block)	1250	43.5	90.9	16	16	16	16	6.1590
数据吞吐率(Mblocks/sec)	0.056	1.388	0.309	6.250	6.250	7.813	6.48	27.066
目标格式	QCIF@10fp	CIF@30f	CIF@30f	HD1080@30fp	HD1080@30fp	HD1080@30fp	HD1080@30fp	4096×2048@30fp

从表 2 中可以看到以往结构都是一路串行扫描,本设计采用了四路并行扫描.以往流水结构编码每个系数块耗费 16 个时钟周期,而本设计只用了 6.1590 个时钟周期.本设计数据吞吐率为 27.066M blocks/sec,提高到以往结构中吞吐率最高的文献[9]的结构的 3.46 倍.以往结构最大仅能处理 HD1080 格式(1920×1088@30fp)的视频,而本设计甚至能处理数字影视格式的视频(4096×2048@30fp).如果以本 CAVLC 编码器来处理 HD1080@30fp 的视频的话,则工作在 50MHz 的频率下就可以满足实时处理的要求,因此 CAVLC 部分的功耗可以大幅度降低.

同时注意到,本设计的四路并行扫描单元资源耗费要大于串行的扫描单元,同时 Fifo1 到 Fifo4 也耗费了

相当的资源.随着集成电路工艺的进步和对于 HDTV 的迫切需求,对处理性能和功耗的需求往往超过了对芯片面积的约束;所以,考虑到本设计数据吞吐率的显著优势和潜在的低功耗特性,其在面积上的增加是可以接受的.

4 结论

本文设计了一种高度并行,双层流水的 CAVLC 编码器.该编码器采用了四路并行扫描结构,突破了数据吞吐率提高的瓶颈;同时规划了系数块级三级流水结构,用 FIFO 缓存不同系数块编码得到的不同数目的码字,使每级流水的延时相差都不大,从而达到很高的流水效率.在 0.18μm CMOS 工艺下,本设计综合电路门数

20685 门,最高工作频率 166.7MHz,数据吞吐率达到每秒处理 27M 系数块,甚至能够处理数字影视格式的视频.整个设计在相对于以往结构取得巨大处理速度优势的同时,硬件资源代价并没有十分显著的增加.

参考文献:

- [1] ITU-T Rec. H. 264 and ISO/IEC 14496-10 AVC, JVT. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video specification[S]. 2005.
- [2] 曹伟,洪琪,侯慧,童家榕,等.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 Jiarong, et al. A high performance reconfigurable multi-transform VLSI architecture for H.264 codec[J]. Acta Electronica Sinica, 2009, 37(4): 673-677. (in Chinese)
- [3] 郑兆青,桑红石,黄卫锋,沈绪榜.用于 H.264/AVC 的 D 级数据重用整数运动估计 VLSI 结构[J].电子学报,2007,35(10):1922-1926.
Zheng Zhaoqing, Sang Hongshi, Huang Weifeng, Shen Xubang. Level D Data reuse integer motion estimation VLSI architecture for H.264/AVC[J]. Acta Electronica Sinica, 2007, 35(10): 1922-1926. (in Chinese)
- [4] Lai Y, Chou C, Chung Y. A simple and cost effective video encoder with memory-reducing CAVLC[A]. Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS) [C]. Kobe: IEEE Press, 2005. 432-435.
- [5] Rahman C A, Badawy W. An area efficient real-time CAVLC IP-block towards the H.264/AVC encoder[A]. Proceedings of IEEE Workshop on Signal Processing Systems Design and Implementation (SIPS) [C]. Canada: IEEE Press, 2006. 368-371.
- [6] Tsai C Y, Chen T C, Chen L G. Low power entropy coding hardware design for H.264/AVC baseline profile encoder[A]. Proceedings of IEEE International Conference on Multimedia and Expo[C]. Toronto: IEEE Press, 2006. 1941-1944.
- [7] Kim D, Jung E, Park H, et al. Implementation of high performance CAVLC for H.264/AVC video codec[A]. Proceedings of the 6th International Workshop on System-on-Chip for Real-Time Applications[C]. Cairo: IEEE Press, 2006. 20-23.
- [8] Chen T C, Huang Y W, Tsai C Y, et al. Architecture design of context-based adaptive variable-length coding for H.264/AVC[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2006, 53(9): 832-836.

- [9] Chien C D, Lu K P, Shih Y H, Guo J I. A high performance CAVLC encoder design for MPEG-4 AVC/H.264 video coding applications[A]. Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS) [C]. Island of Kos: IEEE Press, 2006. 3838-3841.
- [10] Silva T, Vortmann J, Agostini L, et al. FPGA based design of CAVLC and exp-golomb coders for H.264/AVC baseline entropy coding[A]. Proceedings of 3rd Southern Conference on Programmable Logic [C]. Mar del Plata: IEEE Press, 2007. 161-166.

作者简介:



乔飞男,清华大学电子工程系助理研究员,硕士生导师.研究方向为面向视频信号处理的高性能低功耗集成系统体系结构研究,以及低功耗集成电路设计.已经发表学术论文 20 余篇,授权专利 10 余项.

E-mail: qiaofei@tsinghua.edu.cn



魏鼎力男,清华大学电子工程系电路与系统研究所硕士研究生.研究方向为基于 H.264 标准的视频编码器和转码电路设计.



杨华中男,清华大学电子工程系电路与系统研究所教授,博士生导师.获得 2000 年国家杰出青年基金资助.研究方向包括面向通信和多媒体应用的集成数字芯片设计,模拟和射频芯片设计,以及相关电子设计自动化技术的研究;特别是在低功耗集成电路设计领域取得多项成果和专利.已经发表学术论文 100 余篇,专著 6 部.



汪蕙女,清华大学电子工程系电路与系统研究所教授,博士生导师.已经发表学术论文 60 余篇,专著 5 部.