JPEG2000 MQ 编码器的设计与实现

雷 磊,罗桂娥
(中南大学,湖南 长沙 410083)

摘 要:提出了一种 JPEG2000 MQ 编码器的硬件设计方案。通过状态更新超前预测、前导零检测、重归一化超前预测等方法以及字节输出的改进处理,使 MQ 编码器的工作速率可达 1CxD/cycle。同时对各流水段中的路径进行优化改进,提高了系统的最高时钟频率。采用 Verilog 语言进行 RTL 级描述,并在 Altera 的 FPGA 上进行了仿真验证。结果表明,在 Altera 的 EP2S60F67214 上,该 MQ 编码器的最高工作时钟频率可达 65.19 MHz。

关键词:图像压缩;JPEG2000;算术编码;流水线
中图分类号:TN919.81 文献标识码:A

Design and implementation of MQ encoder in JPEG2000

LEI Lei, LUO Gui E
(Center South University, Changsha 410083, China)

Abstract: This paper presented a kind of architecture of high-speed MQ encoder in JPEG2000 based on pipeline technology. To make the arithmetic coder can consume 1CxD pair per cycle, we adopted some tricks such as using the probability estimation with next state forwarding and used the method of counting the number of leading zeroes present in arithmetic results and the technology of forwarding renorm signal as well as improving the way of transferring bytes. Further more, we have modefied crucial paths in the system so that it could improve the max frequency of the encoder.The architecture is described with Verilog in RTL and implemented on Altera's FPGA. The result shows that the coder can work up to 65.19 MHz on Altera's EP2S60F67214.

Key words: image compression; JPEG2000; arithmetic coding; pipeline

JPEG2000 是新一代的静态图像压缩标准。与 JPEG 相比,JPEG2000 不仅具有更为优良的压缩性能,而且提供了更多的新特性,例如支持质量、分辨率的可伸缩性和感兴趣域编码等。JPEG2000 编码包括了小波变换、量化、位平面编码和 MQ 编码这 4 个主要的编码流程。其中位平面编码和 MQ 编码是 JPEG2000 中复杂度较高的 2 个模块。这 2 个模块处理的时间花费了整个编码时间的一半以上。而当前的位平面编码处理速度已经远超过 MQ 编码的速度,也就是说 MQ 编码器的编码速率已经成为了 JPEG2000 处理速度快慢的关键所在。

MQ 编码器是一种改进的自适应算术编码器。虽然 MQ 编码器避免了乘法运算,但算法仍然比较复杂,同时采用串行处理方式的 MQ 编码标准算法用硬件实现起来效率低下。而目前国内外对 MQ 编码硬件实现有不少有效的处理方法。本文对面向软件的标准算法进行了改进以提高硬件实现的编码速率:采用 FIFO 进行输入输出的缓存处理,优化了状态更新及 A、C 区间处理过程以提高处理速度,改进了字节输出从而节约了资源面积。该设计方案采用了 4 级流水,能够达到比较高的数据吞吐量。

1 MQ 编码器原理和算法流程

MQ 编码器可以理解为这样一种机器:它将二进制数据判决位 D 和相关的上下文内容 CX 所组成的序列映射成单个的压缩码字,即压缩数据 CD。当数据判决位 D 和其上下文内容 CX 组成的数据对(CX,D)从位平面编码器输出到达 MQ 编码器后,由 MQ 编码器产生压缩数据位 CD。

MQ 编码器通过使用 CX 状态表和概率估值表能够实现自适应的功能。其中 CX 状态表包括 19 个上下文,每个上下文都对应着不同的状态,每个状态包括索引值(index)和大概率符号值(mps)。而概率估值表是一个可以对原始数据快速适应的概率估计模型,包括 47 个索引值。每个索引都对应着不同的状态,这些状态包括下一个状态的索引值 NMPS(6 位)和 NLPS(6 位)、交换位 SWITCH(1 位)和小概率符号概率值 Qe(15 位)共 28 位。这 2 个表的具体内容可以从参考文献[2]中找到。

MQ 编码是基于自适应的算术编码改进而来的。而算术编码的基本操作是递归地划分当前的子区间:当编码器接收到一个新的待压缩码,当前子区间就被划分成 2 个子区间,被划分的边界更新成为新的区间的左边界,也即左区间值,子区间的间隔大小也更新成为新区间的间隔大小。

因此,MQ 编码器采用一个 A 寄存器来存储当前子区间的间隔大小,而用一个 C 寄存器来存储当前子区间的左区间值。当 MQ 编码器接收到输入数据对(CX,D),通过概率估计表和状态表找到相应的 Qe 值,根据当前的情况来决定 A 和 C 如何进行更新,其中包括了 A、C 寄存器值与 Qe 值的加减操作及对 A、C 寄存器的左移重归一化操作,同时伴随着压缩字节输出等过程。

2 MQ 编码器的硬件设计

本文设计的 MQ 编码器采用 4 级流水线,并使用了一些加速技术对关键部分进行了改进,改进后的 MQ 编码器流水线总体架构如图 1 所示。

第 1 阶段:用一个 RAM 对 CX 状态进行存储和更新。把从 FIFO 中输出的(CX,D)数据对做为输入,根据 CX 的值来得到概率估值表的索引和 mps 的值。然后由 D 值与 mps 值比较判断是进行大概率编码(mps)还是进行小概率编码(lps)。要注意的是,要确保 RAM 和 ROM 输出消耗的时间为 1 个时钟,否则就达不到本设计的时序要求。同时,由于目标是 1 个时钟输入 1 对数据对,而在编码过程中如果出现了连续 2 个输入的 CX 具有相同的值并且编码第 1 个数据发生了重归一化时,就会产生时序紊乱。一个解决的办法是对下一个索引值加入一个超前状态分析,这样就可以在编码同一个 CX 的数据时不必等待 RAM 的输出而直接读入由组合逻辑产生的下一个索引值,从而满足了时序的要求。图 2 所示为加入超前状态分析的 CX 表。

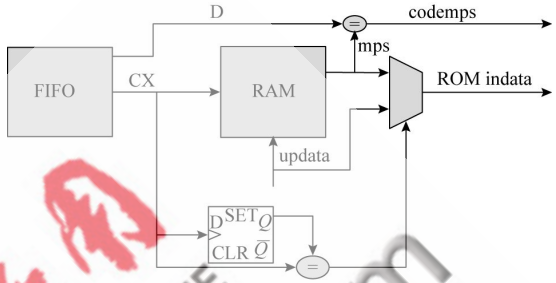

图 2 带有超前状态预测结构的 RAM

第 2 阶段:用一个 ROM 对概率估值表进行存储和读取。把从 RAM 中输出的 index 和 mps 作为输入数据,数据位由高到低排列。根据 index 的值导出相应的 Qe、NMPS、NLPS、SWITCH 等值。本设计由于采用了超前状态预测,没必要把下一次的 lps 和 mps 的概率值加入到估值表中,而仅需添加前导零的个数,使得在第 3 阶段发生重归一化时能够一步完成移位操作,避免了重复和循环过程,大大提高了编码效率。具体的移位思想可以参看参考文献[3]。

第 3 阶段:对 A 寄存器和 C 寄存器低 17 位进行更新处理。把 28 位的 C 寄存器分开处理可以有效缩短关键路径,因此在这个阶段先对 C 寄存器的低 17 位进行处理。另外,由于要对 A 和 2Qe 的大小进行比较从而判断是否需要进行归一化。为了减少路径消耗,可以把A<2Qe 替换成 A[14:0]-Qe[14:0],看是否有借位产生来处理,把它作为是否要进行重归一化的判断条件,同时把 A 和 C 寄存器的加减更新判断逻辑简化成 2 个由于 1 个时钟输入 1 对数据对的关系,A、C 寄存器将在 1 个时钟周期后进行数据更新替换,所以 A、C 的数据处理过程必须要在 1 个时钟周期内完成,因此本设计不能对这段路径进行流水线分割处理。图 3 为对 A 寄存器处理的优化设计结构。C 寄存器的处理结构和 A 类似。

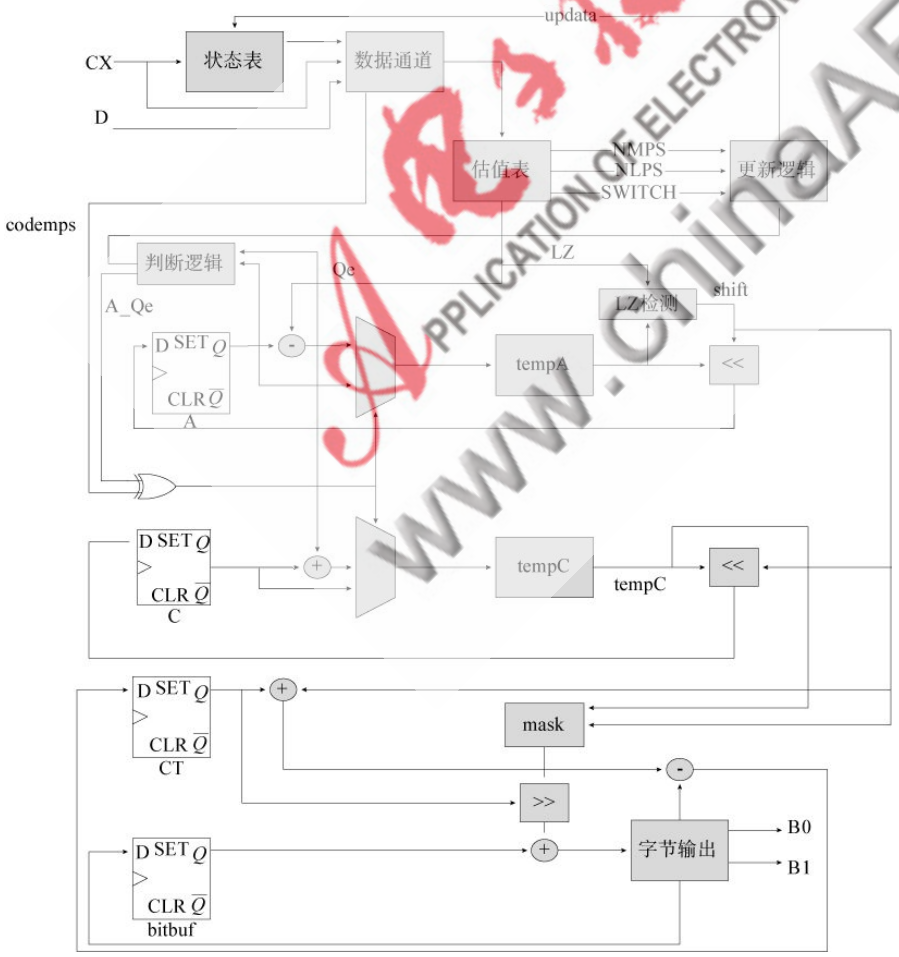

图 1 MQ 编码器总体架构

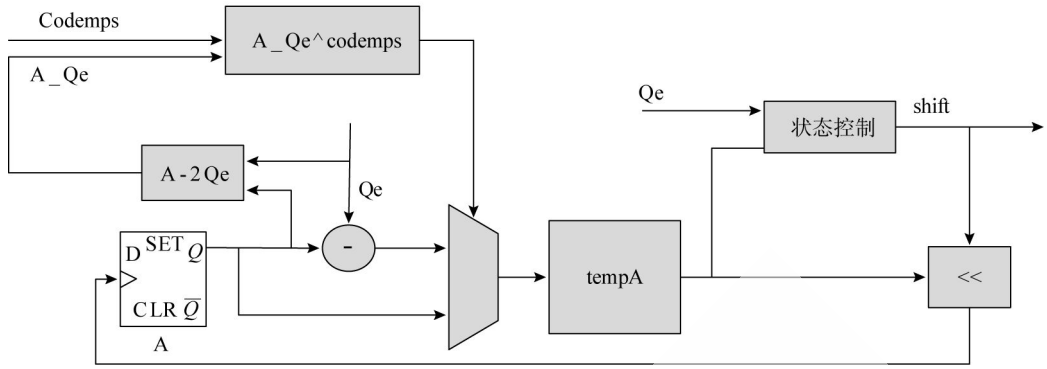

图 3 改进后的 A 寄存器处理结构

第 4 阶段:Chigh 的处理和字节输出。按照标准的字节输出算法流程,需要 1 个缓冲寄存器 B 和 1 个减法计数器来辅助处理:先把高 8 位赋给 B 寄存器,看其值是否是 0xFF 和有没有进位位产生。如果为 0xFF,则进行位填充处理,如果有进位位产生,则对 B 寄存器值进行加 1 处理,再看是否要进行位填充。这种处理方法具有很强的串行性,无法在 1 个时钟周期内完成全部的字节输出过程,因此,有必要对这种方式进行改进。由于原算法加入了 3 位间隔位来限制输出值的范围,为了符合标准,至少要左移 19(即 8+3+8)位才输出 1 个字节,同理,至少左移 27 位才输出 2 个字节,而小于 19 位的则不输出字节,处理好的数据全部放到剩余的数据存储器 bitbuf 中去。而 MQ 编码器的输出方式是增量输出的,因此可以把要左移处理的数据与上一次剩余的数据进行合并,这与标准算法的思想是完全一致的。

首先把 tempC 中的数据进行掩膜处理得到包含左移数据位的 Cmask,同时对 Cmask 进行左移 17 位,使 Cresult 的位数和 augment 的位数相同(augment 为 34 位),再对其进行右移一定位数,使左移数据能够正确地合并到编码数据中去。Cresult 可以表示为:

Cresult={Cmask,17{1'b0}}>>CT-1

其中 CT 为一个 5 位的加法计数器,它对当前剩余的数据进行位数计数。为了减少路径开销,可以把 Cresult 改成:

Cresult={Cmask,18{1'b0}}>>CT

由于剩余的数据位最多为 18 位,因此 bitbuf 采用一个 18 位的寄存器作为处理空间。可以这样把左移数据合并到数据流中:

augment[33:16]=bitbuf+Cresult[33:16]

augment[15:0]=Cresult[15:0]

这是因为仅有高 18 位的 Cresult 需要进行相加,而后 16 位只进行简单的复制即可。这样做就可以不必考虑进位位的值及缓冲值 B 加 1 后是否需要进行位填充这几个因素,可以在一个时钟周期内一步到位地进行字节输出。同时,为了与标准输出一致,把 CT(5 位)的初始值设为-1,即为 11111。图 4 为改进的字节输出的 bitbuf 更新处理部分,图 5 为字节输出的计数更新处理部分。

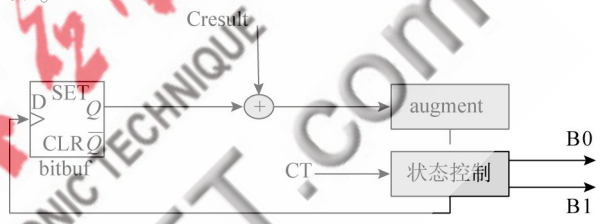

图 4 bitbuf 的更新处理

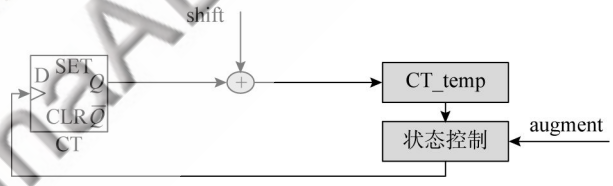

图 5 CT 的更新处理

最后,由于输出的字节数可能为 0、1、2 这 3 种情况,有必要对输出数据进行缓冲,因此需要在最后添加一个 FIFO 对输出的数据进行缓冲。

3 实验结果和性能比较

本文的 MQ 编码器采用 Verilog 语言进行 RTL 级描述,在 Modelsim-Altera 软件下进行仿真,仿真结果和标准算法的计算结果一致,如图 6 所示。在 QuartusII 中选用器件 EP2S60F67214 对代码进行综合、布局布线及时序分析。仿真结果表明,本设计结构最大的时钟频率可达 65.19 MHz,吞吐量可达 65.19 MCxD/s。与参考文献[5]中的方案的比较如表 1 所示。表 2 所示为 MQ 编码器的资源使用情况。

结果显示,本设计占用资源很少的情况下,在最高时钟频率上不及参考文献[5],因为本设计结构为了节省时钟周期在关键路径上没有采用流水线分割,但在整

(下转第 63 页)

（上接 58 页）

图 6 MQ 编码仿真结果

表 1 处理速度比较

方案	时钟频率/MHz	吞吐率/MCxD/s
本设计	65.19	65.19
参考文献[5]	94.585	47.292

表 2 MQ 编码器资源使用情况

Logic utilization	used	utilization
Total logic elements	499	<1%
Total combination fuctions	218	<1%
Total registers	218	<1%
Total memory bits	2661	<1%

体的处理速度上有较大的改进，可以满足硬件高速编码要求。

本文针对 JPEG2000MQ 编码器的硬件实现，提出了一种 4 级流水的设计方案。对 MQ 编码器的原理和算法流程进行了分析，采用超前状态分析避免了时序上的冲突，同时优化了条件判断逻辑以及改进了字节输出的处理结构，减少了路径的开销。最终通过 Aletra 的 FPGA 的验证，处理速度可达 65.19 MCxD/s。

参考文献

[1] TAUBMAN D S, MARCELLIM M W. JPEG2000 图像压缩基础、标准和实践[M]. 魏立江，柏正尧，译. 北京：电子工业出版社，2005.

[2] ISO/IEC. FCD 1544421, final committee draft, JPEG2000 image coding systems[Z].

[3] HSIAO Y T, LIN H D, LEE K B, et al. High-speed memory-saving architecture for the embedded block coding in JPEG2000[C]//IEEE International Symposium on Circuits and Systems, 2002,5:133-136.

[4] DYER M, TAUBMAN D, NOOSHABADI S.Improved throughput arithmetic coder for JPEG2000[J]. Proc.International Conference on Image Processing(ICIP'04),2004.

[5] 周赟. 基于 JPEG2000 的高速 MQ 算术编码器的研究与实现[A].Information technology, 2007(10):49-52.

[6] 王镇道，章兢，曾云，等.一种高速 JPEG2000 MQ 编码器的 VLSI 实现[A].微计算机信息，2006,(9-3):232-233.

（收稿日期：2009-08-17）