

H.264/AVC 编码器在 TMS320DM6437 上的 EDMA 优化* ·实用设计·

崔 遥,赵 锴,崔慧娟,唐 昆

(清华信息科学与技术国家实验室,北京 100084)

【摘要】分析了基于 TMS320DM6437 平台的 H.264/AVC 编码器,对 H.264/AVC 编码器结构进行了适当的调整,对代码中大量数据搬移部分进行 EDMA 优化。使编码器的高速缓冲存储器(cache)命中率提高了 20%~30%,编码速率提高了 3~4 f/s(帧/秒)。

【关键词】TMS320DM6437;EDMA;乒乓缓存;编码器;缓冲存储器

【中图分类号】TN919.81

【文献标识码】A

Optimization of H264/AVC Codec Using EDMA on TMS320DM6437 Platform

CUI Yao, ZHAO Kai, CUI Hui-juan, TANG Kun

(Tsinghua National Laboratory for Information Science and Technology, Beijing 100084, China)

【Abstract】In this paper, H.264/AVC codec is analyzed based on TMS320DM6437, the structure of the encoder is adjusted and the part of the codec with large amount of data transfer is optimized. The cache hits rate is improved significantly by 20%~30%, and the whole encoder velocity is enhanced by 3~4 frames per second.

【Key words】TMS320DM6437; EDMA; ping-pong buffer; codec; cache

1 引言

TMS320DM6437(简称 DM6437)数字媒体处理器是一款高性能定点数字信号处理芯片,采用第三代高性能的超长字指令(VLIM)结构,主频可达 400~700 MHz,峰值运算速率高达 3 200~5 600 MI/s(兆指令/秒)。DM6437 芯片所特有的增强型直接内存存取(Enhanced Direct Memory Access,EDMA)数据搬移工具,可以采用乒乓缓存机制,通过配置芯片上的 EDMA 寄存器,实现 CPU 与数据搬移的并行操作,提高 CPU 的运行效率,而 H.264/AVC^[1]编码器中大量的数据搬移,正好可以利用 EDMA 进行优化,提高编码器的高速缓冲存储器(cache)命中率和编码速率。

2 DM6437 EDMA 简介

2.1 EDMA 结构和工作流程

EDMA 在传输过程中完全不占用 CPU 资源,节省大量的时钟周期,支持片内存储区到片外,片内存储区之间和片外存储区之间的数据读写并且传输模式灵活,支持三维传输模式,可以对传输数据进行重排,大大增强代码的灵活性。

EDMA 主要由 EDMA 通道控制器(EDMA Chan-

nel Controller)和 EDMA 传输控制器(EDMA Transfer Controller)组成,如图 1 所示。EDMA 通道控制器主要完成数据传输前的准备工作,如申请传输通道,配置传输参数到参数寄存器(PaRAM)中,发送传输请求以及传输完成检测等工作。EDMA 传输控制器主要是对 EDMA 通道控制器发送过来的传输请求进行处理,然后按照预先的设置,进行数据传输,具体参数配置主要是在 EDMA 通道控制器中完成。

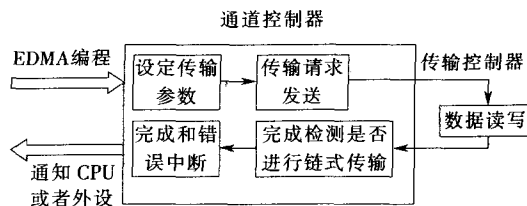


图 1 EDMA 结构和工作流程

2.2 EDMA 传输控制器结构

EDMA 传输控制器主要包括传输事件和中断处理寄存器、传输事件队列、参数寄存器、传输请求发送寄存器和传输完成检测寄存器以及完成中断发送寄存器,结构如图 2 所示。传输事件和中断处理寄存器用于触发传输,传输事件队列会按照优先级对传输事件进行排序,参数寄存器用于存储传输中需要设置的参数,如源地址、目的地址、数据格式等参数,传输完成检测寄存器用

* 国家自然科学基金项目(60572081)

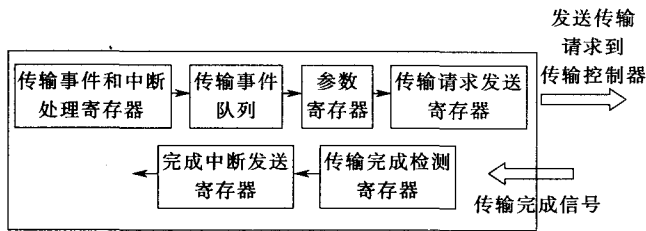


图2 EDMA 传输控制器结构图

于检测传输是否完成,完成后是否进行链式传输,完成中断发送寄存器用于在传输完成后向外设和CPU发送中断信号。

2.3 EDMA 的传输模式

EDMA 传输模式主要有 2 种:A 同步模式和 AB 同步模式。A 同步模式在每次传输被触发后,会传输一个 array, 然后会产生一个中断信号,AB 同步模式每次传输一个 frame, 传输效率高, 本文采用 AB 同步模式。 $ACNT$, $BCNT$ 和 $CCNT$ 分别代表三维传输数据块的尺寸大小,如图 3 所示,每个 array 由 $ACNT$ 个字节组成,每个 frame 由 $BCNT$ 个 array 组成, 每个 block 由 $CCNT$ 个 frame 组成。 $BIDX$ 和 $CIDX$ 分别表示第二维和第三维传输的地址自增偏移量。根据程序需要,对这几个参数进行合理的设置,可以在数据传输过程中实现数据的重排,优化数据存储结构,提高 cache 命中率,从而最终提高代码执行效率。

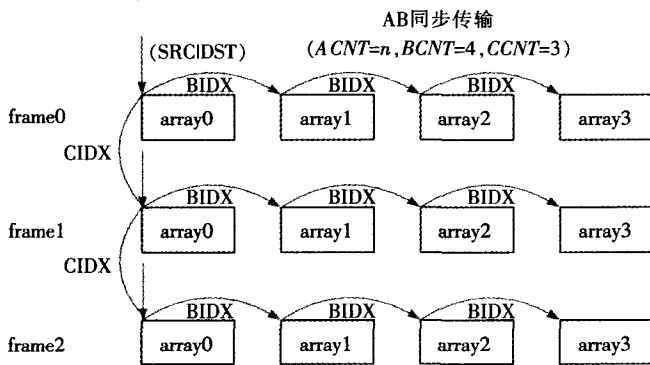


图3 EDMA 三维传输模式

2.4 EDMA 乒乓缓存机制

乒乓缓存机制是 EDMA 最重要的一种传输方式,如图 4 所示。该方式可以实现 EDMA 与 CPU 的并行,本文两处 EDMA 优化均使用该传输方式。其实现主要是将传输目的缓冲区设置为原来目的缓冲区的 2 倍,然后分别用 pingbuffer 和 pongbuffer 作为这 2 块缓冲区的指针,当 EDMA 向 pingbuffer 存储区传输数据时,编码数据指针指向 pongbuffer, 当 pongbuffer 中的数据编码完成后, pingbuffer 存储区数据传输完毕, 然后分别交换

EDMA 传输目的地址和待编码宏块的指针, 即将 pongbuffer 缓冲区赋值给待编码宏块指针, 将 pingbuffer 缓冲区赋值给 EDMA 传输的地址指针。这样 2 个数据通路, 在一路利用 EDMA 传输数据的同时,另一路由 CPU 进行编码的数据运算。如果传输时间小于编码运算的时间,则 EDMA 完全独立于 CPU 在后台运行,不耗费一个时钟周期,因此实现了数据传输和 CPU 并行,提高了代码的运行效率。

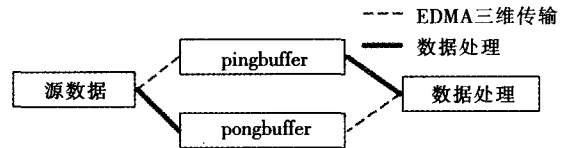


图4 EDMA 乒乓缓存

3 H.264/AVC 编码器 EDMA 优化

3.1 编码器数据存储情况

H.264/AVC 编码器编码步骤如下:1) 编码数据及辅助信息拷贝,主要包括两部分数据,一是待编码宏块数据从片外传输到片内,另一部分是周围重建宏块的数据信息(用于帧内预测)和编码信息(用于 MV 预测和熵编码等)。2)宏块分析,即根据率失真优化得到该宏块的最佳编码模式。3)宏块编码,根据分析出来的最佳编码模式,进行帧内和帧间预测,得到残差信息。4)重建编码宏块,并将当前编码宏块的辅助信息(预测模式、MV 信息等)写入缓冲区(为编码临近宏块使用)。5)宏块熵编码,根据宏块编码模式和残差数据进行进一步的数据压缩。通过以上分析,在帧数据读入后进行边界扩展和编码前宏块拆分阶段,都涉及到了大数据量的搬移,因此对这两部分进行 EDMA 优化。

3.2 帧级 EDMA 优化

编码器在编码前,首先会读入一帧数据到片外存储区 Pic_in,为提高压缩效率,支持越界搜索,对这一帧数据进行边界扩展后存储到片外存储区 Fenc,如图 5 所示。而对帧数据进行边界扩展,需要对整帧数据进行搬移,如果采用数据拷贝指令,占用了 CPU 的资源,费时费力,并且由于只有一块存储区,编码器每次必须等数据传输完成后,才能进行下面的操作,由于数据量大,等待时间较长,因此这里可以使用 EDMA 乒乓缓存机制进行优化。

3.3 宏块级 EDMA 优化

为了更好地发挥 EDMA 在大数据量搬移上的优势,首先对程序进行修改。修改前,编码器从片外整帧数据中读入一个宏块的数据到片内,然后进行编码,但是一个宏块的数据量较小,EDMA 的优势难以得到发挥,因此对程

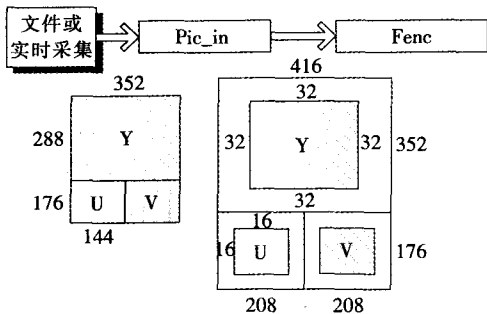


图5 帧级数据传输分析

序进行调整,编码器每次读入若干宏块,如图6所示,宏块的个数通过 *ParamNum* 来控制,增加了每次传输的数据量。因此,进行调整后的编码器,在这里也可以使用 EDMA 进行优化。

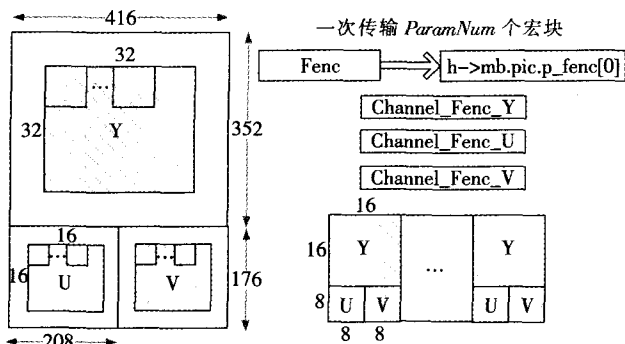


图6 宏块级数据传输分析

帧级和宏块级 EDMA 优化的具体参数配置如表1所示。

表1 EDMA 优化配置参数

参数	整帧数据搬移			N个待编码宏块数据搬移		
	Y	U	V	Y	U	V
ACNT	352	176	176	16	8	8
BCNT	288	144	144	16	8	8
CCNT	1	1	1	N	N	N
SRC_BIDX	352	176	176	416	208	208
DST_BIDX	416	352	352	16	16	16
SRC_CIDX	0	0	0	16	8	8
DST_CIDX	0	0	0	768	768	768

4 试验结果与分析

测试平台使用 DM6437 开发板,采用典型测试序列 Foreman(),对汇编优化^[2]过的 x264^[3]编码器 cache 命中率和编码速度进行测试。根据 DM6437 上片内存储资源,对 cache 和内部 RAM 配置如下:L1P 存储空间全部配置为 cache,大小为 32 kbyte;L1D 存储空间 64 kbyte 配置为内部数据存储存储空间,16 kbyte 配置为数据 cache;L2 存储空间 128 kbyte 全部配置为 cache(程序

cache 或者数据 cache)。经过多次测试,编码器编码前读入的宏块数目为 11 时,cache 命中率得到显著提高。具体的测试结果如表 2 所示。

表2 cache 命中率影响对比

比较项	编码宏块数 N=1	编码宏块数 N=11
编码速度/(f·s ⁻¹)	22.47	23.12
增加的存储量/byte	0	39 800
(Core cycles/Total Cycles)/%	52.75	85.21
(L1PInstruction cache hits)/%	86.70	97.68
(L1DData cache hits)/%	79.22	89.92
(L2 cache data read hits)/%	74.03	67.97
(L2 cache data write hits)/%	99.03	98.28
(L2 cache instruction hits)/%	15.97	97.63

而采用 EDMA 优化后,编码速度也得到了改善,提高了大约 3~4 帧,并且一次读入 11 个宏块与一次读入 1 个宏块相比,前者采用 EDMA 优化的效果更为明显,如表 3 所示。

表3 编码速率影响对比

情况	编码宏块数 N=1	编码宏块数 N=11	f/s
未采用 EDMA	22.47	23.12	
加上宏块编码 EDMA	23.77	25.03	
再加上整帧数据搬移 EDMA	25.76	26.59	

5 小结

通过对 H.264/AVC 编码器进行分析和结构调整,采用 EDMA 乒乓缓存机制,对编码器进行了优化,提高了 cache 命中率和编码速度,证明在处理大数据量的数据传输时,EDMA 有其特有的优势。下一步将分析 H.264 解码器的存储结构和程序流程,思考如何利用 EDMA 进行解码器的优化,以完成实时视频通信的目标。

参考文献:

- [1] WIEGAND T, SULLIVAN G J, BJONTEGAARD G, et al. Overview of the H.264/AVC video coding standard[J]. IEEE Trans. Circuits and Systems for Video Technology, 2003, 13(7): 560-576.
- [2] 彭焯.H.264/AVC 编解码器在 TMS320C6400 上的优化实现[J].电视技术, 2008, 32(9): 21-23.
- [3] X264 Codec[EB/OL].[2010-06-01]. <http://www.videolan.org/developers/x264.html>.

作者简介:

- 崔 遥(1983-),硕士生,主研视频压缩编码与传输;
赵 锴(1985-),博士生,主研视频压缩编码与传输;
崔慧娟,女,教授,从事多媒体通信方向的研究;
唐 昆,教授,从事多媒体通信方向的研究。

责任编辑:丁 雪

收稿日期:2010-06-16