

基于 DSP 平台的大分辨率视频实时编码系统^①

王 博, 须 明, 王 凯

(中航工业无线电电子研究所, 上海 200233)

摘 要: 在 TMS320DM642 平台上实现了能够实时处理分辨率为 1024×768 , 60Hz 视频图像的 H.264 视频编码器. 解决了 DM642 平台下大分辨率视频图像的接入问题, 根据 DM642 的片内资源设计了存储器分配方案, 并针对视频采集驱动和 H.264 视频编码器提出了一套优化解决方案. 测试结果表明, 对于 1024×768 这种大分辨率视频图像, H.264 编码器的处理速度能够满足视频编码的实时处理要求.

关键词: 数字信号处理器; 大分辨率; 实时编码; DM642; H.264; 优化

Real-Time Encoder System for Dealing with High Definition Video Based on DSP

WANG Bo, XU Ming, WANG Kai

(Chinese National Aeronautical Radio Electronics Research Institute, Shanghai 200233, China)

Abstract: In this paper, optimization of a H.264 encoder and video capture driver using TMS320DM642 are presented, and some speciality are considered according to the analysis of the video capture driver structure and the architecture of DM642. The problem of dealing with high definition video, such as video frame of 1024×768 resolution, is finally resolved based on DM642. The testing results demonstrate that encoder has sped up remarkably after optimization, which can meet the real-time encoding requirement for the video resolution of 1024×768 .

Key words: DSP; high definition; real-time encoding; DM642; H.264; Optimization

1 引言

H.264 是目前最新一代的视频编码标准, 采用了诸多先进技术, 如帧内预测、多模式运动估计、整数变换及量化、环路滤波和先进的熵编码技术等. 相比以往标准, H.264 在编码性能上有了较大突破^[1]. H.264 标准定义了 4 个档次(profile), 即基本档次(baseline profile), 主要档次(main profile)、扩展档次(extended profile)和高档次(high profile), 每个档次只支持一组特定的编码功能. 考虑到应用环境的需求, 本文仅实现了 H.264 基本档次编码器.

基于 DSP 平台实现 H.264 编码器具有高性能、低成本等特点, 能够满足大部分产品的需求, 是一种理想的解决方案. TMS320DM642 是 Ti 公司 C64X 系列中比较经典的一款 DSP 芯片, 采用两级 Cache 的哈佛结构, 以及

甚长指令字(Very Long Instruction Word, VLIW)和单指令多数据(Single Instruction Multiple Data, SIMD)结构, 具有独立的地址总线 and 数据总线, 其性能超越了传统的多媒体处理器. DM642 在 C6000 公共指令集的基础上扩展了 88 条指令, 可根据数据的不同情况灵活选择最适合的指令, 使 DM642 能够更容易实现图像处理的算法^[2].

本文就 DM642 处理 1024×768 60Hz 视频图像的难度进行了深入的分析, 然后根据视频采集驱动的软件架构, 对采集驱动进行了改造, 在不增加前端外设的前提下实现了主动降帧, 并根据 DM642 芯片的特点, 对 H.264 视频编码器进行了优化实现. 优化后的编码器配合前端视频采集驱动, 使得 DM642 处理 1024×768 这种大分辨率视频图像的速度有了很大程度的提高, 能够满足视频编码的实时处理要求.

① 收稿时间:2012-05-09;收到修改稿时间:2012-06-14

2 DM642实现大分辨率视频图像实时编码的难度分析

2.1 视频采集

DM642 共有 3 个视频采集端口(Video Port), 每个视频采集端口(VPort 口)的频率为 85MHz, 1024×768 60Hz 的 VGA 信号频率达到 65MHz. 由于 DM642 使用 EDMA(增强型 DMA)来进行外设端口和外部存储器之间的数据交换, 所以由 VPort 口采集到的视频图像都是通过 EDMA 搬移到 DM642 的外部存储器中.

DM642 中的 EDMA 一共有 64 个通道, 每个通道都有一个事件与之关联, 由具体的事件触发相应的通道进行数据的传输. 考虑到视频采集过程中, 基本上是每采集到一行视频数据就会触发一次 EDMA 事件, 同时后续编码过程中编码数据在外部存储器和内部 Cache 之间的数据交换也会大量使用到 EDMA, 而视频编码与视频采集基本处于同时工作的状态, 所以如此高频率的使用 EDMA 会导致 EDMA 控制器过于繁忙, 严重影响编码器的编码性能.

2.2 视频编码

图 1 所示的为 H.264 的编码框架. 与 MPEG-4, H.263 等标准类似, H.264 仍然采用的是运动补偿/分块变换的混合编码框架, 采用帧内和帧间两种编码模式, 编码以宏块为单位进行.

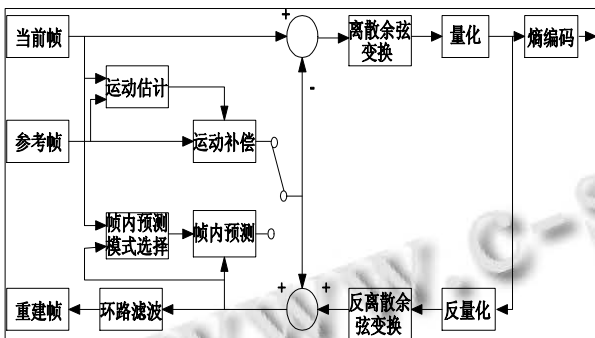


图 1 H.264 编码框架

与以往标准不同, H.264 标准采用多个参考帧进行预测, 经过大量实验发现, H.264 编码算法中 80% 的预测块来自于前一个参考帧, 20% 的预测块来自于其他参考帧^[3]. 因此本文只采用 1 个参考帧进行预测. 当采用 1 个参考帧时, 运动估计占有整个编码 53% 的计算量. 文献[4]分别采用 4 种编码方法对 Foreman 视频序列进行编码, 前 3 种方法分别只采用了 16×16、8×8、4×4 等三种模式, 第 4 种方法采用了所有模式, 只采

用 4×4 编码模式时得到的编码器性能最差, 因此小块的帧间编码模式对编码器性能的贡献较小^[4]. 这一实验结果对本文的编码器优化具有一定的指导意义.

3 基于DM642的视频采集驱动和H.264编码器的优化

目前针对 DSP 的优化方法很多, 主要包括数据传输优化、算法优化和 Cache 优化等. 本文基于 H.264 标准基本档次编码器和 DM642 的特点, 分别从视频采集驱动、优化 Cache 使用方法等方面入手, 对 H.264 编码器进行优化, 配合前端视频采集驱动, 使优化后的 H.264 编码器在编码 1024×768 这种大分辨率视频图像时达到实时性要求.

3.1 视频采集驱动实现主动降帧

DM642 下的视频采集驱动是符合 Ti 公司 IOM mini-driver 规范的一个子集. IOM mini-driver 是构建于 DSP/BIOS 这个可裁剪的实时内核基础上的可分层的驱动程序框架^[5]. 为最大可能的复用代码, DM642 下的视频采集驱动被划分为两个部分: a.通用部分, b.平台相关部分. 这两部分通过统一的内部接口即 EDC(External Device Control)提供对上层的调用, 上层调用通过 Ti 的 FVID/GIO class driver 接口实现. 其层次关系如图 2 所示.

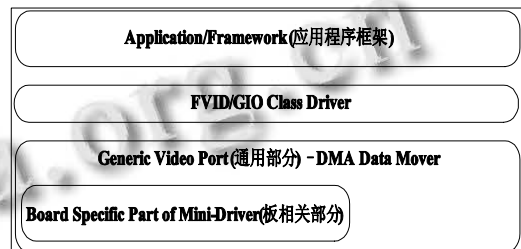


图 2 视频采集驱动层次图

在实际应用中, 原始视频数据通过 VPort 口前端视频采集芯片传输给 VPort, 然后通过 EDMA 控制器将采集到的视频数据搬移到外部缓存中. 其数据流图如图 3 所示.

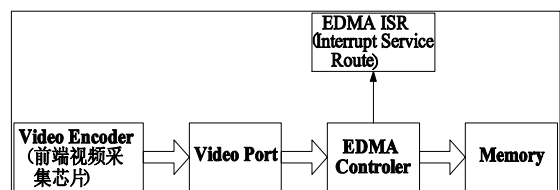


图 3 视频采集数据流图

通过分析视频采集的流程,可以知道视频数据从 VPort 口的 FIFO 到外部缓存都要通过 EDMA 控制器来进行数据的搬移,由于 VPort 的 FIFO 容量较小,而且 EDMA 不支持 Buffer 全空或全满的情况下触发 DMA 传输,所以传输的阈值由用户编程实现. 通常的阈值大小是一整行的数据,这样做有利于后续帧缓存的灵活操作. 当到达一行数据后,会产生相应的 EDMA 事件触发 EDMA 控制器将这一行数据搬移到外部缓存中. 当一帧图像被完整传输到缓存中后,会触发 EDMA 中断,由中断服务程序将完整一帧提交给上层应用. 如果视频数据量较大(如本文中的 1024×768 分辨率视频图像),可以知道 EDMA 的操作是非常频繁的. DM642 下的 EDMA 请求根据优先级不同而分配到不同的优先级队列中,优先级一共划分为 3 级, Q1(Urgent Priority)、Q2(High Priority)和 Q3(Low Priority). 视频采集和后续的视频编码一般都采用 Q2 等级. 当视频采集的 EDMA 操作过于频繁时,会导致处于相同优先级队列中的视频编码的 EDMA 请求长时间得不到相应,从而影响编码的效率.

一般视频的实时性是指 25 帧每秒,根据这个目标,如果能够将前端 60Hz 的视频图像降为 30Hz,在不影响实时性的前提下,就能够大幅降低 EDMA 的操作,从而为后端的视频编码处理带来更大的余量.

这一想法可以通过 VPort 口控制寄存器 VCCTL 来实现, VCCTL 第 30 位 BLKCAP 位可以阻塞视频采集的 DMA 事件,同时刷新采集通道的 FIFO. 通过 DM642 内部定时器,设置一定的延迟时间,在每采集到一帧视频帧后,在 EDMA 中断处理函数中阻塞视频采集的 DMA 事件,同时启动定时器进行延迟,到达延迟时间后触发定时器中断,并在定时器中断处理函数中重新使能视频采集的 DMA 事件,通过这样的操作就能控制 VPort 口的视频采集行为,从而实现主动降帧,流程图如图 4 所示.

其中,延迟时间 T_θ 的计算公式如下:

$$T_\theta = (\text{CAPTURE_FPS} - 1) * T_{\text{frame}} - T_{\text{offset}} \quad (1)$$

CAPTURE_FPS 设为 30,是目标帧率, T_{frame} 是相邻两帧的时间间隔, T_{offset} 是两帧时间间隔的一半,这样做的目的是保证恢复视频采集 DMA 事件的时机恰好是让视频采集在丢弃一帧视频后,距离下一帧一半的时间开始采集,为保证一帧的完整性,需要在距离一

帧视频采集开始前留有一定时间余量. T_{frame} 和 T_{offset} 的计算公式分别如下:

$$T_{\text{frame}} = 1000000\mu\text{s} / 60\text{Hz} \quad (2)$$

$$T_{\text{offset}} = T_{\text{frame}} / 2 \quad (3)$$

经过这样设计,通过一定的延迟,实现了 VPort 口的主动降帧,经过试验,对于 1024×768 60Hz 的视频图像,视频采集帧率可以稳定在 30 帧的目标帧率,使 EDMA 的使用率相比不降帧的情况至少降低了一半,为后续的视频编码争取了更多的余量.

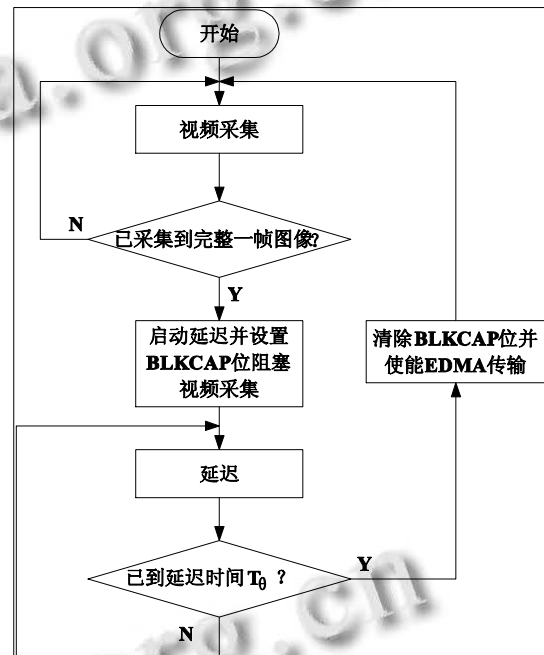


图 4 主动降帧流程图

3.2 基于 Cache 优化的 H.264 视频编码器

DM642 具有高速的两级缓存结构,其中二级 Cache(L2)可以设置为 Cache, SRAM, 或部分 Cache 和部分 SRAM,而设置为 SRAM 的空间可以被映像为 DM642 的寻址空间. L2 Cache 具有速度快的优势,其运行频率等同于 CPU 频率,当其被配置为 SRAM 时,可用于数据缓存. 当缓存在这部分 SRAM 中的数据参与运算时,利用其频率高、速度快的优势,可以大大加快编码的运算速度. 由于同处于 L2 Cache 中,这部分缓存的数据可以很快被替换到 Cache 中,提高了 CPU 对数据的命中率. 但 L2 Cache 分配用作 SRAM 的空间也不能太大,太大会导致用作 Cache 的部分变少,从而增加 Cache miss 的情况,导致系统性能下降. 在实

际应用中,为尽可能减少 Cache miss 的情况,本文所介绍的系统将 L2 Cache 配置为 128KB, 剩余 128KB 配置为 SRAM, 作为数据缓存使用.

根据以上对 L2 Cache 的分配方案,本文对编码器的结构进行了一定的改进.传统的编码器结构都是按照宏块顺序逐块对视频进行处理,这样做的缺陷是会导致视频数据需要在外部存储器和 L2 Cache 之间不停交换,导致大量的 Cache miss,从而影响编码器的效率.而根据文献[2]的研究表明,当采用 1 个参考帧时,编码器中的运动估计部分占有整个编码 53% 的计算量,是最复杂的部分.而运动估计主要的运算过程是一个当前宏块在参考帧运动搜索范围内寻找最匹配宏块得到最佳运动矢量的过程,利用 L2 Cache 中分配作为 SRAM 的部分来当做参考帧的缓存,同时将几个宏块而不是单个宏块一次性搬入 SRAM 中逐个进行处理,当编码 P 帧(前向预测帧)时,将大大加快运动估计的运算速度,同时又提高了 Cache 的命中率,从而提高整个编码效率.

由于 1024×768 视频图像分辨率较大,我们每次按照半行三个宏块行将数据搬入内部存储器暨 SRAM 中,采用乒乓 Buf 的机制,处理当前宏块数据时,将下一个半行三个宏块行的数据也搬入内部存储器中,如图 5 所示.

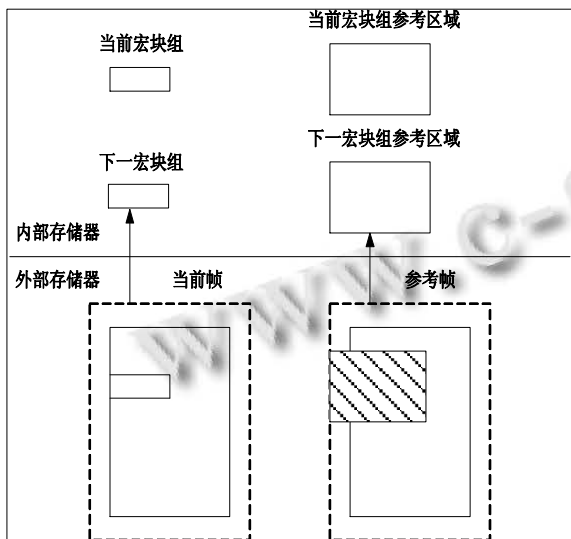


图 5 数据传输图

图中虚线部分是图像加边区域,我们设置运动估计搜索范围为正负 32 像素,为防止运动搜索出界,在图像上下左右各加 32 像素加边区域,阴影部分是当前

宏块组的参考区域,考虑到搜索范围,需要一次搬入三个宏块行的数据. 1024×768 分辨率视频半行数据为 32 个宏块(每个宏块大小 16×16),其处理流程如图 6 所示.

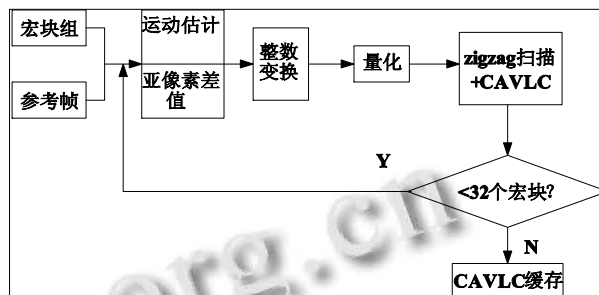


图 6 P 帧编码流程图

图中 CAVLC 缓存处于内部存储器 SRAM 中,在实际应用中,我们将所有码表也都放置于内部存储器中以加快编码速度.

4 实验结果

本编码系统在 600MHz 时钟的 TMS320DM642 评估板上对分辨率为 1024×768 60Hz 的视频图像进行了编码测试,考虑到我们的应用环境中视频图像画面细节较多,运动主要以平移运动为主,我们选取了 Foreman、Football、Mobile 和 Car 四个标准测试序列,并通过软件将这 4 个分辨率为 CIF 大小的视频序列合并成为一个分辨率为 1024×768 大小的视频序列,在计算机上循环播放,并设置计算机分辨率为 1024×768 60Hz,通过 VGA 信号端口接入评估板.

编码采用 IPPP...的编码模式,不包括 B 帧,单帧参考,量化步长固定选择为 25,配合前端视频主动降帧的视频采集驱动,优化前和优化后的实验结果如下表所示.

表 1

性能指标	优化前 QP=25	优化后 QP=25
PSNR (db)	37.58	37.44
码率 (kbps)	3660.56	3762.90
帧率 (fps)	8.60	25.70

通过实验结果,可见本系统在平均 PSNR 损失较小(小于 0.5db)、码率增加较小(小于 5%)的情况下,可将编码速度提高 2.98 倍,已经达到了实时 25 帧每秒编码的要求.

(下转第 128 页)

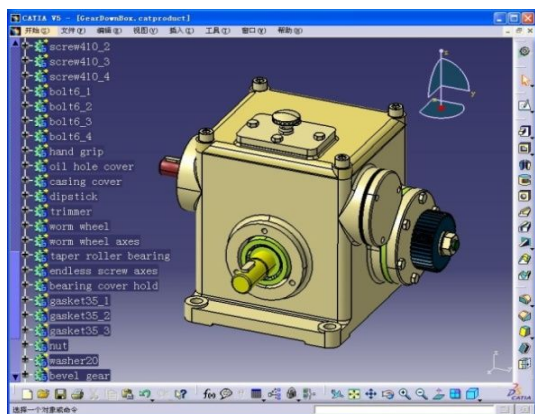


图 4 减速箱的 CAITA 模型

```

- <bomData>
+ <summary suffix="asm,prt,dw" version="0.1" type="CATIA">
+ <errorInfo />
- <part id="0" type="装配件" 名称="GearDownBox">
+ <relation type="文档关联">
+ <模型属性 质量="606.829185 kg" 体积="74404503.650863 mm^3" 平均密度="0.000088 kg/mm^3" 曲面面积="3256368.568299 mm^2" />
- <structure>
+ <childPart id="0" refid="2" 数量="1">
+ <childPart id="1" refid="4" 数量="1">
+ <childPart id="2" refid="6" 数量="1">
+ <childPart id="3" refid="8" 数量="1">
+ <childPart id="4" refid="10" 数量="1">
+ <childPart id="5" refid="12" 数量="1">
+ <childPart id="32" type="零件" 名称="oil hole cover">
+ <document id="33" type="模型文件" 文件名="d:\design\oil hole cover.catpart">
+ <part id="34" type="零件" 名称="casing cover">
+ <document id="35" type="模型文件" 文件名="d:\design\casing cover.catpart">
+ <part id="36" type="零件" 名称="dipstick">
+ <document id="37" type="模型文件" 文件名="d:\design\dipstick.catpart">
+ <part id="38" type="零件" 名称="trimmer">
+ <document id="39" type="模型文件" 文件名="d:\design\trimmer.catpart">
+ <part id="40" type="零件" 名称="worm wheel">
+ <document id="41" type="模型文件" 文件名="d:\design\worm wheel.catpart">

```

图 5 模型结构树信息储存于 XML

5 结语

本文分析了三维 CAD 模型如何向三维装配工艺

系统提供装配信息，重点解决了 CATIA 模型的装配信息提取的若干关键技术及难点，以及如何应用于三维装配工艺系统的集成解决方案。该技术已经在开目三维装配工艺规划软件中得到实际应用，实现了三维 CAPP 与三维 CAD 的紧密集成。

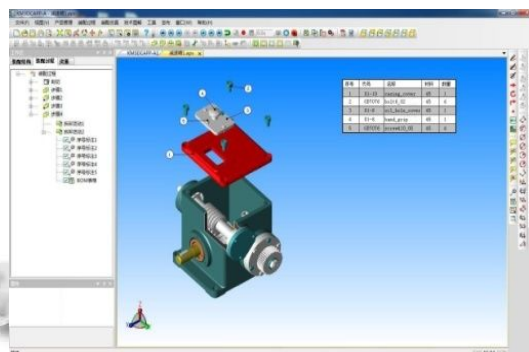


图 6 KM3DCAPP-A 中装配过程规划的一个步骤与 BOM 表格

参考文献

- 1 黄娟,王丽.三维装配 CAPP 的技术和产品浅析.CAD/CAM 与制造业信息化,2011,9:11-14.
- 2 周仙娥,鲁墨武,赵海星.基于 CAA 的 CATIA 二次开发技术的研究.科技信息,2008,36:73-75.
- 3 周桂生,陆文龙.CAITA 二次开发技术研究与应用.机械设计与制造,2010,1:81-83.
- 4 CAA V5 For CATIA Foundations. DASSAULT SYSTEMS, March,2001.

(上接第 32 页)

5 小结

本文在 TMS320DM642 平台上实现了能够实时处理分辨率为 1024×768, 60Hz 视频图像的 H.264 基本档次的视频编码器。文章首先分析了对于 1024×768 这种大分辨率的视频图像, DM642 处理的瓶颈, 并有针对性地设计了一套能够实现主动降帧的视频采集驱动, 大大降低了 EDMA 的使用率。在这个基础上, 充分结合 DM642 的结构特点, 提出了基于 Cache 存储器优化的编码器优化解决方案。实验结果证明, 这些优化方案可以大大提高 H.264 基本档次视频编码器的编码速度, 完全满足实时编码的要求。

参考文献

- 1 沈兰荪,卓立.小波编码与网络视频传输.北京:科学出版社,

- 2005.
- 2 李方慧,王飞,何佩琨.TMS320C6000 系列 DSPs 原理与应用.第 2 版.北京:电子工业出版社,2002.
- 3 Huang YW, Hsieh BY, Wang TC, Chen SY, Ma SY, Shen CF, Chen LG. Analysis and reduction of reference frames for motion estimation in MPEG4 AVC/JVT/H.264.ICME'03, Hong Kong, April 2003,2:809-812.
- 4 Tu YK, Yang JF, Shen YN, Sun MT. Fast variable-size block motion estimation using merging procedure with an adaptive threshold. ICME'03, Hong Kong, July 2003,2:789-792.
- 5 Texas Instruments, TMS320DM642 Video/Imaging Fixed-Point Digital Signal Processor, Texas: Texas Instruments Incorporated, sprs 200J, August 2005.