

视频信号处理器的并行结构综述及分类

龚大年, 何 芸, 曹志刚

(清华大学微波与数字通信国家重点实验室, 北京 100084)

摘 要: 本文先从并行性的角度提出了视频图像处理器结构的新分类方法. 然后对视频图像处理器的并行结构进行了全面的综述, 指出各类并行结构的特点和以及今后研究和发展的方向.

关键词: 并行结构; 并行性; 视频信号处理器

中图分类号: TN919.81 **文献标识码:** A **文章编号:** 0372-2112 (2000) 07-0096-06

Overview and Classification of Parallel Architectures for Video Signal Processors

GONG Da-nian, HE Yun, CAO Zhi-gang

(Tsinghua University, Beijing 100084, China)

Abstract: A new classification of video signal processors based on parallelism is first presented in this paper. Then an overview of parallel architectures of video signal processors is given and the future development of parallel architectures is pointed out.

Key words: parallel architecture; parallelism; video signal processor

1 引言

视频信号处理算法的计算量很大, 对处理速度要求很高, 传统的处理器无法胜任, 必须研究适合视频信号处理的并行算法和与之相对应的处理器并行结构, 来满足视频信号处理对计算速度的要求. 随着视频压缩编解码算法的成熟, 国际标准的制定和推广以及超大规模集成电路技术的迅速发展, 应用于视频压缩的视频信号处理器已经成为目前国际上众多单位竞相研究开发的对象, 已经有众多不同的视频信号处理器结构问世. 由于视频信号处理器有着重要的理论和应用价值, 因此全面、科学地分析研究已有视频信号处理器的结构并总结各种结构的优缺点, 为进一步设计新的视频信号处理器结构提供借鉴, 具有重要的意义.

许多学者围绕视频信号处理器的结构展开研究并试图对其进行分类. 早在 1990 年, T. Fukushima 在文[1]中就对日本的图像信号处理器芯片作了分类, 共分为完全并行处理器(FPP)、部分并行处理器(PPP)、数字信号处理器(DSP)、功能处理器(FP)和神经网络处理器(NNP). 当时的图像处理器主要还是针对滤波、直方图处理、图像格式转换和 DCT 等处理级别较低的运算. Pirsch 等人根据可编程性将视频编码器分为专用功能处理器和可编程处理器^[2,3]. 随着通用处理器性能的提高、多种媒体的集成、嵌入式设计以及低功耗处理器的出现使得视频编码器的分类更加复杂. 文[4]沿用可编程的角度对多媒体硬件结构分为三类, 即专用结构(Dedicated architecture)、Adapted 结构(Adapted architecture)和可编程结构(Programmable architecture). 专用功能处理器有基于芯片集的实现

方案和单片的实现方案. Adapted 结构处理器分为图形处理器和视频处理器. 可编程结构处理器分为带扩展指令集的通用处理器和媒体处理器. 文[5]把多媒体处理器分为 5 类, 即精简指令集计算机(Reduced Instruction Set Computer, RISC)、复杂指令集计算机(Complex Instruction Set Computer, CISC)、嵌入式微处理器(Embedded Microprocessor)、低功耗数字信号处理器(Low Power DSP)和用于 PC 加速的 DSP, 或称为媒体处理器(Media Processor).

这些工作都从一定的角度出发, 对现有的视频处理器作出分类, 例如文[2,4]是根据可编程性的程度研究视频处理器的, 文[5]是根据应用的角度来研究视频处理器的, 文[1]则是从并行性和应用角度两个方面进行研究. 由于并行性是视频信号处理器的重要特点, 为了满足视频压缩算法运算量大的需要, 视频信号处理器结构都包含了不同类型和层次的并行结构, 因此从并行性的角度出发, 能够比较全面、科学地对已有视频处理器进行研究. 本文将视频信号处理器的并行结构分为四类(七种), 分别总结它们的优缺点, 并以典型的视频信号处理器芯片的并行结构为例加以阐述. 最后指出未来各种视频信号处理器并行结构的发展方向.

2 视频压缩算法的并行性

视频压缩算法的主流是以离散余弦变换和运动估计为框架的混合编码算法, 混合编码算法已经被 H. 261、H. 263、MPEG1、MPEG2 等国际标准所采用. 视频压缩算法中大致有两种类型. 一种是可并行的计算, 它具有数据流确定、运算规则、相关性(数据相关性, 控制相关性和资源相关性)弱的特

收稿日期: 1999-06-08; 修订日期: 2000-01-10

基金项目: 国家自然科学基金(No. 69672028 & No. 69972021) 资助课题

点,例如,离散余弦变换/反离散余弦变换(DCT/IDCT)、运动估计(Motion Estimation, ME)、运动补偿(Motion Compensation, MC)和量化/反量化(Q/IQ)等。另一种是不可并行的计算,这种计算控制和数据相关性强,有较多的转移分支,例如,游程编码、折线形(zigzag)扫描、算术编码、变长解码(Variable Length Decoding, VLD)以及码率控制等。具有并行特点的计算占视频压缩算法计算量的主要部分,特别是DCT和ME,成为视频压缩处理的瓶颈。因此,必须寻求并行算法来实现具有并行特点的计算,从而解决好计算量大的瓶颈。

文[6]指出,软件并行性有两种,控制并行性和数据并行性。控制并行性使得两个或多个操作同时完成,数据并行性使得多个数据元素能在同一时间完成。本文认为视频压缩算法的并行性也有两种不同类型的并行性,即控制并行性(control parallelism)和数据并行性(data parallelism)。并行性根据粒度的大小又可细分为不同的级别,粒度越小,级别越低。本文把粒度定义为最小数据单元大小以及最小任务单元复杂度的度量。最小数据单元是指处理过程中相互之间没有相关性的数据集合。最小任务单元是指相互之间没有相关性的有机组合起来的操作集合。本文把并行性级别大致分为细粒度并行性和粗粒度数据并行性。根据并行性的类型和级别的不同,视频压缩算法的并行性可大致分为细粒度控制并行性、粗粒度控制并行性、细粒度数据并行性和粗粒度并行性。细粒度控制并行性的最小处理任务单元简单,一般为单个算术运算或单条指令,也称为指令级并行性。粗粒度控制并行性的最小处理任务单元较复杂,通常为多个算术逻辑运算的有机结合,每个处理任务完成一定功能,也称为任务级并行性。细粒度数据并行性的最小数据单元一般为像素,不同像素进行各自的运算,互不影响,也称为像素级并行性。粗粒度数据并行性的最小数据单元一般为块、宏块或宏块组,也称为块级并行性或宏块级并行性。

视频算法在不同的级别上都体现出很大的并行度。首先,现有的图像编码国际标准方案大多采用分块的编码方法,例如H.263、MPEG-1和MPEG-2等视频压缩编码标准都采用 8×8 DCT和块大小为 16×16 或 16×8 的ME及MC的算法,不同块在运算过程中相互没有相关性或相关性很小,因此不同块可以同时处理。因此视频算法存在着块级并行性或任务级并行性。其次,图像处理具有规则性、迭代性和局部性的特点^[11]。以H.263、MPEG-1和MPEG-2等视频编码标准中运算量最大的ME和DCT为例说明。ME的基本运算是减法、绝对值、求和以及比较运算;DCT的基本运算是乘累加和转置运算。这些基本运算的特点是运算规则,有大量的循环结构,蕴藏着较大并行性。实验结果也表明,视频计算中存在大量没有相关性的指令,指令级并行度很高^[12]。因此视频算法存在着像素级并行性或指令级并行性。

3 视频处理器并行结构

硬件结构的并行性只有和算法的并行性相匹配才能发挥最大效用。和算法的并行性相对应,本文把硬件并行结构也分为四类,即细粒度控制并行性结构、粗粒度控制并行性结构、

细粒度数据并行结构和粗粒度数据并行结构。本文采用这四类并行结构对已有视频信号处理器结构进行分类。本文将文献上出现的视频处理器结构归纳为七种,即同步流水线结构(Synchronous Pipeline Architecture)、异步流水线结构(Asynchronous Pipeline Architecture)、数据流结构(Data Flow Architecture)、单指令多数据流结构(Single Instruction Multiple Data, SIMD)、分裂ALU(Split ALU)、多指令多数据流结构(Multiple Instruction Multiple Data, MIMD)和超长指令字结构(Very Long Instruction Word, VLW)。本文根据这七种并行结构所开发并行性的类型和级别不同,把它们归类到四类并行结构中,见表1所示。

表1 视频处理器并行结构的种类

并行性类型 级别	并行性类型	
	控制并行性	数据并行性
低	细粒度控制并行性结构(同步流水线结构, VLW, 数据流结构)	细粒度数据并行性(SIMD, Split ALU)
高	粗粒度控制并行性(异步流水线结构)	粗粒度数据并行性(MIMD)

3.1 细粒度控制并行结构

3.1.1 同步流水线结构^[14~18] 同步流水线结构适用于开发细粒度控制并行性。同步流水线结构由若干流水线段组成,流水线段由组合电路组成,每个流水线段之间以时钟锁存器相连,在时钟脉冲到达时,所有数据从锁存器同时将数据输出到下一个流水线段中。同步流水线结构的每个流水线段较简单,时延小,因此同步流水结构可以工作在很高的频率上,从而满足视频压缩处理对计算速度的要求。每个流水线段完成移位、绝对值、加减运算、比较等运算,根据运算的特点采用最简单的电路加以实现,从而达到高速的要求。以VISP^[14]为例,说明高速算术单元(PAU)的流水线结构。PAU包含7个段,第一段是16位移位寄存器段,第二段是带绝对值运算的ALU段,第三、四段是BOOTH乘法器段,第五段是20位累加器,第六段是桶状移位器,第七段是求最小值段。每个段的延迟在2.5纳秒内,时钟频率可达到400兆赫兹。采用同步流水线结构的视频处理器还有^[14]S-VSP^[15]、IDSP^[16]、VDSP^[17]和VSP^[18]。S-VSP工作频率250M赫兹,每段延迟小于4纳秒。IDSP工作频率300M赫兹,每段延迟3.3纳秒。VSP采用双极型CMOS工艺,工作频率高达300兆赫兹。

同步流水线结构的段数决定了并行度(加速比)。在理想情况下,采用N个流水线段的流水线结构的加速比为 $N^{[6]}$ 。由于控制的复杂性,电路实现等问题的制约,流水线段数不能无限地增加^[6],同时由于可能发生流水线中断,加速比往往达不到理想加速比。同步流水线设计是一项很费时间的工作,而且可扩展性不好,即采用新工艺后,流水线要重新设计。采用同步流水线结构的视频处理器工作频率一般较高,必须解决好时钟错位(clock skewing)问题。

3.1.2 超长指令字(VLW)结构^[19~26] VLW结构适用于开发指令级控制并行性。VLW是水平微码和超标量处理两者的结合^[6]。水平微码由多条短指令压缩而成。所谓超标量处理,

是指一个周期内发射多条指令,多个功能单元(Functional Unit, FU)并行地完成指令的执行。VLIW指令是由短指令经过编译器压缩后形成的。一条VLIW指令,包含多个域,共有近百位甚至上百位,如应用于视频处理的HiPar-DSP^[19]采用的VLIW长度为96位。VLIW结构由多个功能部件(FU),指令存储器,译码部件,指令发射部件,交叉互连网络和大型寄存器阵列(RF)组成。FU包括若干个加法器,乘法器以及寻址部件,每个功能部件结构较简单。译码部件将长指令分解成短指令并完成短指令字的译码。指令发射部件控制各FU的执行。大型寄存器阵列一方面作为操作数的暂时存储器,存放中间结果,另一方面用来在不同FU之间传送数据。互连网络完成不同FU之间的数据交换,由于FU之间的通信带宽大,互连网络采用带宽容量较大的交叉互连网络(Cross Bar)。目前已经采用VLIW结构的芯片有^[20]、Phillips TriMedia^[21,22]、Mpaact^[23]和TMS320C6x^[24]。Princeton的Santanu、Wolf和Wolfe等人也正研究VLIW结构^[25,26]。以Phillips的TriMedia TM1000^[21,22]为例,具体说明VLIW的结构。TriMedia的超长指令字有5个域,每个域长度为32位,功能单元有27个,包括整数算术逻辑运算(Integer ALU)单元,浮点算术逻辑运算单元(Float ALU),移位(Shifter)单元,转移(Branch)单元,读取/存储(Load/Store)单元等。寄存器阵列由128个32位寄存器组成。

VLIW和同步流水线结构都开发细粒度控制并行性,但两者有很大的不同。不同点在于:VLIW的功能单元是在超长指令控制下执行,有很强的可编程性;同步流水线结构每个处理单元有固定的执行顺序,不需要指令来控制,适合用于专门的处理任务。

VLIW结构对寄存器阵列和功能单元之间的带宽要求很高(高带宽可以减少存取寄存器时产生的冲突)。高带宽导致互连网络的规模的增加。因此能否在单芯片上实现规模庞大的互连网络是VLIW结构的先决条件^[26]。VLIW的编译时间比较长。另外,兼容性是VLIW固有的缺点,不同VLIW结构没有兼容性,如果功能单元数量变化了,指令集也必须变化^[6]。

3.1.3 数据流结构^[27] 数据流结构(Data flow Architecture)^[6]由若干处理器模块组成,每个处理器能完成一定的视频处理任务,不同处理器之间的同步由数据流机制来实现,可以用来开发细粒度(指令级)的控制并行性。每个处理器除了逻辑运算单元之外,还需要有令牌匹配部件和令牌形成部件。各个处理器的运行由数据的可用性来驱动,不需要有中央控制器,但需要专门的机构来检测数据可用性。数据流结构的优点是能使同步等待产生的损失减到最小,充分开发指令级并行性。采用数据流结构的有Vincent^[27],Vicent5包括BPU、THU和MCU。BPU完成VLD和IQ,THU完成IDCT和MC,MCU完成SDRAM和内部存储器之间的数据传送。VLD对码流处理得到图像头信息和宏块信息,宏块信息加上宏块号标记后存放在控制FIFO(control FIFO)中,控制FIFO和MC和MCU相连。块信息(DCT系数)加上宏块号标记和位置标记后存放在和IDCT模块相连的数据FIFO(data FIFO)中,当FIFO中有足够数据时,IDCT模块工作,IDCT的输出结果传送到MC的输入数据FIFO中,当MC数据FIFO的标记和控制FIFO的标记相同时,

MCU开始从SDRAM取数据,然后MC模块开始工作。整个解码过程无须全局控制器,不同模块之间的同步通过FIFO中的标记(令牌)匹配来实现。

目前用于视频压缩处理的数据流结构较少,目前仅见文[27]中的MPEG2解码器采用数据流结构。

3.2 粗粒度控制并行结构

3.2.1 异步流水线结构^[28~30]

异步流水线结构(Asynchronous Pipeline Architecture)适用于开发粗粒度控制并行性。异步流水线结构由若干处理器组成流水线段,每个流水线段较复杂,可以完成宏块或块级别的运算,处理延迟时间不固定,相邻流水线段通过缓存来连接,并有信号握手(handshaking)协议来控制相邻段的数据流^[6]。信号握手协议可以用中央处理器完成,它严格控制不同流水线段的启动和终止。和同步流水线结构相比,异步流水线结构的每个流水线段的结构以及控制电路较为复杂,工作频率较低。要把视频压缩任务映射到异步流水线结构的方法是进行任务划分。例如,MPEG2的解码过程可以分为VLD、反折线形扫描、IQ、IDCT和MC,把它们合理地映射到流水线段上。当VLD处理的数据集为第I个宏块时,由于第(I-1)个宏块已经完成VLD,已经得到量化DCT系数,因此反折形扫描和IQ可以同时处理第(I-1)个宏块,同样道理,IDCT可以同时处理第(I-2)个经过反量化的宏块,MC可以处理第(I-3)个宏块,这样,在同一个时间里,VLD、IQ、IDCT和MC可以并行完成。不同的任务需要的延迟时间往往不相同,每个流水线段的下一个启动时刻取决于流水线段的最长处理时间。以Nam Ling等人设计的MPEG2解码器^[28]为例,共有4个流水线段,即VLD、反折形扫描/IQ、IDCT和MC,控制策略采用基于中断方式的缓存控制策略,工作频率27兆赫兹,能完成MP@ML的MPEG2实时解码。采用异步流水线控制并行结构的还有文[29,30]。

异步流水线结构的关键在于流水线的控制方法和各个流水线段的设计实现上。由于视频压缩标准的不同子任务所要完成的计算类型和特点有较大的不同,例如MPEG2解码器中,IDCT的计算主要是乘累加,MC主要是半像素插值操作和加法运算,因此如何针对流水线段处理任务的不同计算特点来设计专用电路是流水线结构是否高效的关键之一。由于异步流水线的延迟不象同步流水线延迟那样确定,因此合理划分流水线段,使得流水线段的延迟尽可能相等,从而提高流水线结构的效率,另外,缓冲器大小的确定也是决定异步流水线结构效率高低的关键^[31]。采用异步流水线结构的缺点是,流水线段必须等待所有任务都完成以后才能启动,就是说,先完成的模块必须等待后完成的模块,在等待期间,处理模块处于空闲状态,造成硬件资源的浪费。

3.3 细粒度数据并行结构

3.3.1 SIMD结构^[32~37]

SIMD结构适用于开发细粒度数据并行性,每个处理器单元(Processor Element, PE)完成像素级的处理。SIMD只有一个控制单元(Control Unit, CU),CU一次只发一条指令,这条指令同时发送到各个PE,每个PE对数据执行同样的任务。SIMD结构有两种,即分布存储器SIMD结构和共享本地存储器SIMD结构^[6]。

(1) 分布本地存储 SIMD 结构^[32,33]

采用分布本地存储器 SIMD 结构的视频处理器见文[32, 33]. 文[32]中对分布本地 SIMD 阵列进行了综述, 分布本地存储器 SIMD 结构的每个 PE 的结构简单, 处理器位数少, 一半以上采用了 1 位 ALU. PE 个数众多 (64 - 4096), 组成规模庞大的处理器阵列. 每个 PE 有自己的本地存储器. PE 之间大多采用带宽容量大的静态网络. 分布本地存储 SIMD 结构的缺点有二, 一是 PE 之间的通信瓶颈. PE 结构简单, 要完成视频压缩处理任务需要多个 PE 协同工作, 这就需要 PE 之间互相传送中间数据, 数据通信的瓶颈是影响性能的重要因素^[6]. 二是存储器带宽瓶颈. 由于视频压缩处理对存储器的带宽的要求很高, 众多 PE 和外部存储器的通信将也是影响性能发挥的瓶颈.

(2) 共享存储 SIMD 结构^[34~37]

面向视频压缩编码应用的 SIMD 结构采用共享存储 SIMD 结构为主. 共享存储 SIMD 结构的特点是 PE 个数少, 一般为 4 - 16 个. 每个 PE 较复杂, 有加法器、乘累加器和移位器, 有较强的可编程性, 每个数据通道在一个指令周期里可以完成乘法、累加和移位多种操作. 互连结构采用总线或交叉网络方式, 完成不同处理器之间的数据交换功能. 存储器模块大小与块或宏块大小相当, 个数比分布本地存储器少, 能被不同处理器所共享. 下面以 VCP^[34]为例说明 SIMD 结构的特点. VCP 的视频信号处理器 (VP) 包括两个主要的视频处理加速部件. 一个用于 DCT 和 IDCT 计算, 另一个用于 ME. DCT 加速部件由 4 个 16 位的 PE 组成, 每个 PE 有乘法器、桶形移位器和累加器部件. ME 加速部件由 8 个 16 位 ALU 或 16 个 8 位 ALU 组成. 采用共享存储 SIMD 结构的视频处理器还有 AVPI300E^[35], AxPe640V^[36], NEC 芯片集^[37].

3.3.2 分裂处理器结构^[38,39]

分裂处理器结构 (Split ALU) 可以看成是 SIMD 结构的特例, 它是通用处理器和 SIMD 思想的结合. 通用处理器的位数一般为 32 位或 64 位, 而视频计算中遇到的数据类型为 8 位或 16 位, 造成硬件资源的浪费. 分裂处理器结构在通用处理器结构的基础上, 数据通道引入 SIMD 结构, 将 CPU 分开为若干短位数的处理单元, 一条指令可以完成多个像素的处理. 这种结构的例子是 MMX^[38]. MMX 在 Intel 体系结构的基础上, 改进了逻辑运算单元, 能对 32 位, 16 位或 8 位数据类型进行操作, 还能进行饱和运算. VIS^[39]是采用分裂 CPU 的另一例子.

分裂处理器结构的特点是灵活多样, 可根据精度要求的不同, 设置处理器的位数为 8 位或 16 位, 提高了处理器的利用率和并行度. 同时, 这种结构继承了通用处理器的优点, 操作类型丰富, 除了有加减操作外, 还有饱和运算、压缩、扩展运算、交织运算等. 由于这种结构是通用处理器的改进, 运算单元的特点是高度可编程性, 根据指令类型的不同, 运算单元可以完成加法、减法或移位功能. 但没有专门的加法器、移位器和乘法器, 因此每个操作完成的功能简单, 要完成复杂的视频压缩任务需要多条指令.

3.4 粗粒度数据并行结构

3.4.1 MIMD^[41~45]

MIMD 结构适用于开发粗粒度数据并

行性. MIMD 由多个 PE 组成, 每个 PE 都有控制单元, 算术运算单元的结构比 SIMD 的算术运算单元结构复杂, 处理能力更强. 不同 PE 在各自控制单元的控制下, 执行不同的指令流. 与 SIMD 相比, MIMD 处理任务的粒度更大, SIMD 结构中, PE 处理的数据通常为像素, 而 MIMD 结构中, PE 处理的数据集为块、宏块或宏块组. MIMD 结构使得任务分配和数据分配变得更加灵活, 可以把一幅图像分为若干块, 每块交给 PE 处理, PE 完成宏块所有的运算任务, 包括运动预测、DCT 等操作.

采用 MIMD 结构的一个例子是 TI 的 TMS320C80^[44,45]. TMS320C80 (MVP) 包括四个并行处理器 (Parallel Processor, PP) 和主处理器 (Master Processor, MP). 处理器通过交叉网络与片内存储器相连. 四个并行处理器能完成主处理器交给的视频处理任务. 主处理器是带浮点处理器 (Floating Point Unit, FPU) 的通用 RISC, 主要功能是任务分配和与外部四个并行处理器进行通信. 采用 MIMD 结构的视频处理器还有文[41~43].

MIMD 和 VLIW 有相似之处. 相似之处在于指令都表现为多指令流形式. 不同点在于: VLIW 的多指令流由编译阶段压缩在一条长指令里, 并行性由长指令显示表现出来, 在运行阶段指令流以确定的次序发送到处理单元; 而 MIMD 结构在运行过程中检测并行性, 指令流发送到哪个处理器在运行阶段才确定.

MIMD 的缺点是不同的 PE 之间的通信和控制比 SIMD 复杂, 要采用软件或硬件的方法来实现 PE 之间的同步. MIMD 结构的另一个缺点在于芯片面积大, 在工艺上要求高, 同时必须考虑到制造上的缺陷, 设计时要考虑到冗余度和检测芯片好坏的能力.

4 结论及展望

本文分析了视频处理算法并行性特点, 从并行性的角度对视频处理算法以及视频处理器结构进行分类, 对不同结构的视频处理器进行横向比较并分析它们的特点. 表 2 归纳出不同种类并行结构的特点和例子. 由此可以得到关于各种并行结构的一些结论和展望:

(1) 同步流水线结构应用于设计高速流水运算部件, 将向更高时钟频率的方向发展. VLSI 工艺的迅速发展为高速流水线算术部件的设计提供有利的支持, 另一方面, 视频压缩应用对视频处理计算量的要求越来越高, 需要达到 GOPS 的处理速度, 这也需要开发出高速运算单元满足计算量要求.

(2) VLIW 结构的成功很大程度上不仅取决于硬件的可实现性, 也取决于编译器的效率. 从硬件可实现性考虑, VLIW 结构中有大量的寄存器阵列和互连网络, 能否在一个芯片上实现是其成功与否的关键; 从软件角度考虑, 要设计出功能较强的编译器, 来实现算法和硬件结构的匹配, 提高硬件的利用率. 文[46]详述了 VLIW 结构的关键技术和设计方法. 指出进行 VLIW 结构设计时必须考虑的要素有网络互连结构, 节点数量, 节点 RF 的大小, RF 端口数等等. 如何能在 FU、RF 和互连网络组成的设计空间中找到最佳参数集合是一个有待继续研究的课题.

(3) 采用数据流结构的视频处理器目前仅见于文[27], 它

被应用于 MPEG2 的解码器. 由于数据流结构可以很好地开发指令级并行性, 能最大限度地吸收通信延迟, 使由于同步等待产生的损失最小^[6], 因此在开发视频算法细粒度并行性方面, 数据流结构的优势显著, 不失为一种可行的结构, 有待人们作进一步的研究.

(4) 异步流水线结构目前是解决 MPEG1、MPEG2 等解码器的常见结构, 这种结构设计有两个关键点. 一个关键点是合理对划分流水线的段数和流水线段的延迟时间, 使得每个流水线段的延迟尽可能相等, 提高流水线工作效率, 另一个关键点是合理地确定流水线段之间的缓冲存储器的大小, 取得最佳的性价比.

(5) 分布存储器 SIMD 结构的突出问题是解决好通信瓶颈, 通信瓶颈是限制并行性发挥的重要因素^[6], 解决方法是将逻辑门嵌入 DRAM 或 SRAM 中 (logic in memory)^[32], 从而解决

存储器带宽瓶颈.

(6) 共享存储器 SIMD 结构基于像素的并行处理, 粒度大小适中, 每个处理单元的复杂程度适中, 处理器结构的位数和视频处理应用场合相匹配, 有较好的可编程性和效率. 设计大小合适的片内快速存储器是解决存储器带宽的有效途径.

(7) 分裂 CPU 结构已经成为通用处理器提高多媒体处理能力的重要手段, 事实上通用微处理器只要增加很小的硬件开销就能大大地增强性能. 已经被 Intel 的 MMX^[38]、Sun 的 VIS^[39]、HP 的 MAX2^[40] 所采用. 今后微处理器将提供更丰富的多媒体指令集来增强性能.

(8) MIMD 结构开发粗粒度的数据并行性, 数据划分和任务划分会更灵活, 必须解决好处理器之间的通信和同步问题. 同步问题可以用硬件或软件来解决, 如何减小硬件和软件开销是 MIMD 结构需要解决的问题.

表 2 视频处理器并行结构的特点和举例

类型	特点	举例	
细粒度控制并行结构	同步流水线结构	开发细粒度控制并行性, 有一个或多个高速的流水线算术运算部件. 每个流水段的功能简单, 时延小. 时钟频率很高. 隔代可扩展性不好.	[14 ~ 18]
	VLIW	开发指令级控制并行性, 结构由多个功能部件 (FU), 指令存储器, 指令译码, 发射部件, 交叉互连网络和大型寄存器阵列 (RF) 组成. 每个时钟周期发射一条超长指令, 它的不同域经过译码控制多个功能部件的执行. 大量的寄存器阵和互连网络. 结构是否成功取决于软件编译器.	[19 ~ 26]
	数据流结构	开发细粒度指令级并行性, 采用数据流结构, 以数据驱动机制为基础, 各个模块的运行由数据的可用性来驱动, 不需要有中央控制器来控制各个模块的运行, 但需要专门的机构来检测数据可用性.	[27]
粗粒度控制并行结构	异步流水线结构	开发任务级控制并行性, 采用流水线并行技术, 各个流水段的运算单元较复杂, 处理时间不固定, 不同流水段之间用缓存相连, 具有全局控制器来完成流水段的同步.	[28 ~ 30]
细粒度数据并行结构	分布存储 SIMD	开发像素级数据并行性, 分布本地存储器 SIMD 结构的每个 PE 的结构简单, 处理器位数少, PE 个数多, 组成规模较大的处理器阵列, 互连网络大多采用带容量大的静态网络. 通信瓶颈是影响性能的重要因素.	[32, 33]
	共享存储 SIMD	开发像素级数据并行性, 每个处理器单元结构较复杂, 处理器位数为 8 位或 16 位, PE 个数少, 一条指令能完成对多个像素的处理.	[34 ~ 37]
	分裂 CPU 结构	SIMD 的特例, 是通用处理器结构的增强, 能支持不同精度数据的处理. 每个处理单元高度可编程, 一条指令完成对多个数据的操作.	[38 ~ 40]
粗粒度数据并行结构	MIMD	开发粗粒度数据并行性, 每个处理器单元能完成较复杂运算任务. 不同处理器执行不同的指令或宏操作. 不同处理器处理不同的数据, 数据单元可以是块、宏块或宏块组. 不同处理器之间通信同步的控制复杂, 开销大.	[41 ~ 45]

参考文献

- [1] Fukushima. A survey of image processing LSI's in Japan. IEEE 10th Int. Conf. on Pattern Recog. Atlantic City, NJ: June 1990: 394 ~ 401
- [2] P. Pirsch, Nicolas Demassieux, and Winfried Gehrke. VLSI architectures for video compression: A survey. Proceedings of IEEE, Feb. 1995, 83(2): 220 ~ 246
- [3] P. Pirsch, H. J. Stolberg. VLSI implementations of image and video multimedia processing systems. IEEE Transactions on Circuits and Systems For Video Technology, 1998, 8(7): 878 ~ 891
- [4] Fatemi and S. Panchanathan. Design trends in multimedia hardware architecture. SPIE 3311, San Jose, CA, Jan. 1998, 2 ~ 6
- [5] Kuroda, T. Nishitani. Multimedia processors. Proc. of IEEE June 1998, 86(6)
- [6] Kai Hwang 著. 王鼎兴等译. 高等计算机系统结构. 北京: 清华大学出版社, 1995
- [7] ITU-T Recommendation H. 261, Video Codec for Audio-visual Services at Px64kbit/s, 1993
- [8] ITU-T Recommendation H. 263 Version 2, Video Coding for Low Bit Rate Communication, International Telecommunication Union, February 1998
- [9] ISO CD 11172, Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5Mbit/s, 1991
- [10] ISO/ICE 13818-2, Generic Coding of Moving Pictures and Associated Audio Information: Video, 1994
- [11] S. Y. Kung. VLSI Array Processors, Prentice Hall, Englewood Cliffs, New Jersey, 1988
- [12] Zhao Wu, Wayne Wolf. Parallelism analysis of memory system in single-chip VLIW video signal processors. In: SPIE 3311, San Jose, CA Jan.

- 1998:58~66
- [13] T. Ebrahimi, Murat Kunt. Visual data communication for multimedia applications. Proceedings of the IEEE, June 1998, 86(6)
- [14] K. Kikuchi, et al. A signal-chip 16-bit 25-ns real-time video/image signal processor. IEEE JSSC, Dec. 1989, 24(6)
- [15] J. Goto, et al. 250-mhz BiCMOS super-high-speed video signal processor (S-VSP) ULSI. IEEE JSSC, Dec 1991, 26(12):1876~1884
- [16] T. Minami, R. Kasai, H. Yamauchi, et al. A 300 mops video signal processor with a parallel architecture. IEEE Journal of Solid-State Circuits, Dec. 1991, 26(12):252~253
- [17] T. Araki, M. Toyokura, M. Wakamori, K. Aono. The architecture of a vector digital signal processor for video coding, Proc. IEEE 1992 ICASSP, Mar. 1992:72~73
- [18] T. Inoue, et al. A 300-MHz 16-b BiCMOS video signal processor. IEEE Journal of Solid-State Circuit, Dec. 1993, 28(12)
- [19] K. Ronner, J. Kneip, P. Pirsch. Architecture and applications of the HiPar video signal processor. IEEE Transactions on Circuits and Systems on Video Technology, Feb. 1996, 6(1):56~66
- [20] H. Yamauchi et al. Architecture and implementation of a highly parallel signal chip video DSP. IEEE Trans. Circuits and Systems for Video Technology, June 1992, 2:207~220
- [21] S. Rathnam and G. Slavenburg. An architectural overview of the programmable multimedia processor, TM1, Proc. Compcon, IEEE CS Press, 1996, 319~326
- [22] First trimedia chip boards PCI bus. Microprocessor Rep., 1995
- [23] P. Foley. The Mpack media processor redefines the multimedia PC, Proc. Compcon, IEEE CS Press, 1996:311~318
- [24] T.J. Dillon, Jr. The velociTI architecture of the TMS320C6x, Proc. Int. Conf. Signal Processing and Tech. San Diego, CA, Sept. 1997
- [25] Zhao Wu, Wayne Wolf. Parallelism analysis of memory system in single-chip VLIW video signal processors. In: SPIE 3311, San Jose, CA, Jan. 1998:58~66
- [26] Wayne Wolf. VLIW architectures for video signal processing. In: SPIE 3311, San Jose, CA, Jan. 1998:52~57
- [27] Jeong-Min Kim, Soor-Ik Chae. A cost-effective architecture for HDTV video decoder in ATSC receivers. IEEE Trans. On Consumer Electronics, 1998, 44(3):1353~1359
- [28] Nam Ling, Nier-Tsu Wang, Duan-Juan Ho. An efficient controller scheme for MPEG2 video decoder, IEEE Transactions on Consumer Electronics, May 1998, 44(2):451~458
- [29] M. Yasuda, T. Nishio, M. Toyokura, et al. MPEG2 video decoder and AC-3 audio decoder LSIs for DVD player, IEEE Trans. on Consumer Electronics, Aug. 1997, 43(3):462~468
- [30] Duardo, et al. An HDTV video coder IC for ATV receivers. IEEE Trans. on Consumer Electronics, Aug. 1997, 43(3):628~632
- [31] Nam Ling, Jui-Hua Li. A bus-monitoring model for MPEG decoder design. ICCE Digest of Tech. Paper, Jun. 1997, 180~181
- [32] T. M. Le, et al. SIMD processor arrays for image and video processing: A Review. SPIE 3311, San Jose, CA, Jan. 1998:30~41
- [33] D. G. Elliott, et al. Computational-RAM: A memory-SIMD hybrid and its applications to DSP. IEEE Custom Integrated Circuits Conference, Boston: May 1992, 30(6):1~4
- [34] D. Bailey, M. Cressa, D. Neubauer, et al. Programmable vision processor/controller. IEEE MICRO, Oct. 1992, 12:33~39
- [35] S. K. Rao, et al. A real-time P64/MPEG video encoder chips. Proc. IEEE Int. Solid State Circ. Conf., 1993:32~35
- [36] AXP640V Data Sheet, 1996
- [37] Tamitani et al. An encoder/decoder chip set for the MPEG video standard. IEEE ICASSP, 1992:33~39
- [38] Ales Peleg. MMX technology extension to the Intel architecture. IEEE Micro, August 1996, 16(4):42~50
- [39] M. Tremblay, et al. VIS speed new media processing. IEEE Micro, Aug. 1996:10~20
- [40] Ruby B Lee. Subword parallelism with MAX-2. IEEE Micro, 1996, 16(4):51~59
- [41] K. Gaedke, H. Jeschke, P. Pirsch. A VLSI-based MIMD architecture of a multiprocessor system for real-time video processing applications. J. VLSI Signal Proc. Apr. 1993, 5:159~169
- [42] K. Herrman, Jan Otterstedt, Hartwig Jeschke, et al. A MIMD-based video signal processing architecture suitable for large area integration and a 16.6-cm² monolithic implementation. IEEE Trans. on VLSI Systems, June 1998, 6(2)
- [43] J. Wilberg, M. Schobinger, Peter Pirsch. Hierarchical multiprocessor system for video signal processing. VCIP 1818, Boston, Massachusetts, 1992:1076~1089
- [44] R.J. Gove. Architectures for single-chip image computing. SPIE, Electronic Image. Science and Tech. Conf. on Image Processing and Interchange. San Jose: Feb. 1992
- [45] K. Gattag, R.J. Gove, and J. R. Van Aken. A single chip multiprocessor for multimedia: The MVP. IEEE CG&A, 1992:53~64
- [46] S. Dutta, K.J. O. Connor, W. Wolf, et al. A Design of a 0.25-μm video signal processor. IEEE Trans. Circ. and Syst. for Video Tech., Aug. 1998, 8(4):501~519

龚大年 1975 年出生, 1996 年毕业于浙江大学信息与电子工程系, 1996 年至今在清华大学电子工程系攻读博士学位。主要研究方向为: 视频压缩编码, 视频信号处理器结构及其 VLSI 实现, MPEG2, MPEG4 专用集成电路设计。