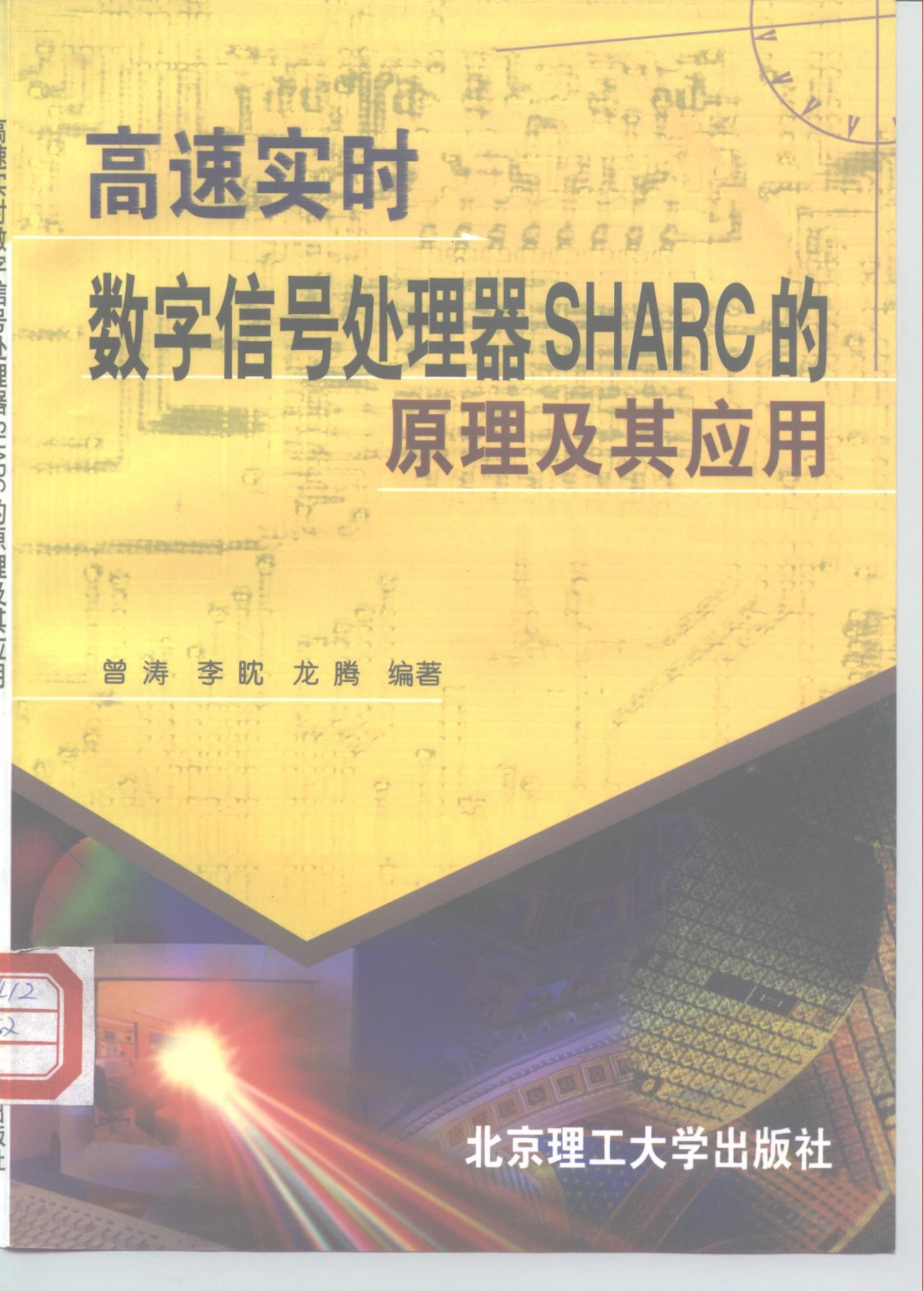




高速实时

数字信号处理器 SHARC 的 原理及其应用

曾涛 李旼 龙腾 编著

北京理工大学出版社

73.412
652

高速实时数字信号处理器 SHARC的原理及其应用

曾涛 李眈 龙腾 编著



北京理工大学出版社

2001100

内 容 简 介

JS115/06

北京理工大学雷达技术研究所的几位教授、青年教师和博士生,根据他们长期研究、开发、设计 SHARC DSP 芯片的经验和心得体会,写作了这本书,本书从对 DSP 技术的介绍开始,预测了今后 DSP 技术的发展趋势。详细论述了 SHARC 芯片的构造、特点、原理、编程方法等内容,并给出了在工程实现中常用的实例。还结合我们实际工作经验,对在 SHARC 系统设计中出现的问题进行了总结,并给出了相应的解决方案。同时,还讲述了高速电路设计中常见的问题,以及解决办法。最后介绍了 SHARC 家族的其他成员,包括 SHARC 二代、Tiger SHARC 等产品的基本情况。相信本书的出版会为 SHARC 技术的应用和推广提供方便并起到一定的促进作用。本书可供 DSP 专业设计人员、SHARC 系列产品硬件、软件开发人员阅读,亦可作为相关专业研究生或高年级本科生的教学用书。

图书在版编目(CIP)数据

高速实时数字信号处理器 SHARC 的原理及其应用/曾涛,李眈,龙腾编著. —北京:北京理工大学出版社,2000.6

ISBN 7-81045-677-6

I. 高… II. ①曾… ②李… ③龙… III. 数字信号发生器 IV. TN911.72

中国版本图书馆 CIP 数据核字 (2000) 第 08567 号

责任印制:王 军 责任校对:陈玉梅

北京理工大学出版社出版发行

(北京市海淀区白石桥路 7 号)

邮政编码 100081 电话(010)68912824

各地新华书店经售

北京国马印刷厂印刷

*

787 毫米×1092 毫米 16 开本 12.25 印张 277 千字

2000 年 6 月第 1 版 2000 年 6 月第 1 次印刷

印数:1—4000 册 定价:18.50 元

※ 图书印装有误,可随时与我社退换 ※

0011005

序

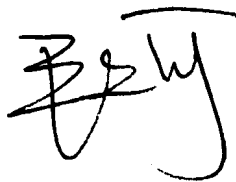
数字信号处理(DSP)技术的发展已经有 20 年的历史。目前公认的第一个成功的商业 DSP 芯片,是 1982 年美国 TI 公司推出的 TMS32010 定点 DSP 芯片。它采用了数据总线和程序总线分离的 Harvard 体系结构,并具有内部硬件乘法器,完成一次乘法只要一个指令周期。Harvard 体系结构和硬件乘法器,构成了第一代 DSP 区别于通用计算机的主要特点,为此, TI 公司曾获得当年的 EDN 技术创新奖。

经过将近 20 年的发展, DSP 已经发展为一个庞大的家族,其体系结构也从早期简单的 Harvard 体系结构,发展到现在的 SHARC、VLIW 等复杂的体系结构;其运算速度从早期的 200ns 指令周期、5MIPs,发展到今天的 5ns 指令周期、1600MIPs;此外, DSP 的寻址能力、片内存储器容量、周边控制电路的灵活性、以及与上一位计算机的接口等能力都得到极大的扩充。

一般认为, DSP 区别于一般通用 CPU 的主要特点是 Harvard 体系结构和硬件乘法器。DSP 的另一个重要特点是处理和控制的实时性,即信号处理的时间是严格可预测的。但是随着 DSP 结构的日益复杂,以及 CPU 中也开始逐渐采用硬件乘法器, DSP 的这些早期特点正在逐渐模糊。但是无论如何, DSP 在运算密集、实时性要求高的场合,还是具有不可替代的优越性。为此, DSP 市场近年来获得了长足的发展,在通信、雷达、航空航天、汽车电子、消费电器等各个领域都获得了广泛的应用,预计到 2002 年其市场将达到上百亿美元。

北京理工大学电子工程系雷达技术研究所从 1985 年即开始跟踪国际 DSP 的最新动态,从那时开始,基本上全系列地开发了国际主要的 DSP 芯片。在 DSP 的研究、开发过程中,发现美国 AD 公司的 SHARC 系列 DSP 芯片具有运算能力强、并行扩展性好、使用方便等众多优点,在浮点信号的大规模实时处理方面具有很大的优越性。为此,研究所的几位青年教师和博士生根据他们开发 SHARC DSP 芯片的经验,写作了这本书,其目的是为国内 SHARC 技术的应用和推广提供方便。本书介绍了 SHARC 芯片的主要特点,以及他们在 SHARC 系统设计中的心得体会,相信本书的出版能对国内 DSP 技术的应用和推广起到一定的促进作用。

中国工程院院士
北京理工大学电子工程系教授



2000年3月

前 言

当前信息产业的发展越来越趋向于数字化、小型化、集成化,数字信号处理(DSP)技术已经成为必不可少的手段。无论在民用还是军用,DSP技术已经深入到通信、雷达、航空、航天、家用电器等各个领域,成为电子系统的核心。1998年在芯片总体市场呈下降趋势的环境下,DSP市场却达到了20%的年增长率,销售额达39亿美元。预计到2002年,DSP市场规模将达到134亿美元,平均年增长率达33%。

目前在国际上,有TI、朗讯、AD、Motorola等几大DSP生产厂商,这4家的产品占据了DSP市场90%以上的销售份额。其中,AD公司推出的SHARC系列浮点DSP芯片,以其强大的功能、精巧的设计、优良的实用性被业界公认为“最好的浮点DSP芯片”。AD公司还将于近期推出Tiger SHARC(虎鲨系列)浮点DSP芯片,主频将达到250MHz,将成为“最快的浮点DSP芯片”。从DSP概念、实现、应用等各方面来看,AD公司的SHARC系列芯片,都是该公司最成功的代表作品。

在国际上,SHARC早已受到广泛的第三方支持,各个国际知名的公司竞相推出自己的SHARC板级产品,并被广泛运用到DSP技术涉及的各个领域。我国于近年开始接触SHARC系统设计与集成,SHARC系列DSP芯片已经受到国内业界的广泛承认。但是,由于SHARC本身是一种高性能的复杂DSP芯片,而目前国内还没有完整的讲述SHARC系统设计的资料与书籍,这给国内的设计者带来很多不便。有鉴于此,我们撰写了本书,旨在促进SHARC在国内的推广。

本书面向的对象是:DSP专业设计人员、SHARC系列产品硬件、软件开发人员,并可以作为研究生或大学高年级学生的教学参考资料。

在撰写本书的过程中,得到了毛大可院士、何佩琨教授的悉心指导,在此表示衷心的感谢。王宏伟、田黎育、莫力博士、杨凡、朱仕银硕士认真细致地校对了初稿,并提出了很多好的建议,在此一并感谢。

由于作者水平有限,书中难免出现一些缺点和错误,恳请广大读者批评指正。

编 者

目 录

第 1 章 概述	(1)
1.1 DSP 的基本概念	(1)
1.1.1 DSP 的结构特点	(1)
1.1.2 DSP 的发展	(4)
1.2 SHARC 简介	(7)
1.2.1 SHARC 的基本结构	(7)
1.2.2 SHARC 指令的特点	(9)
1.2.3 程序开发过程	(10)
第 2 章 SHARC 的基本结构与概念	(11)
2.1 SHARC 运算单元	(11)
2.1.1 SHARC 的数据格式	(11)
2.1.2 算术逻辑单元(ALU)	(12)
2.1.3 乘法器(MPU)	(13)
2.1.4 移位器(SHIFTER)	(15)
2.1.5 数据寄存器	(15)
2.1.6 小结	(16)
2.2 SHARC 的程序流控制	(16)
2.2.1 流水线机制	(16)
2.2.2 指令 Cache	(17)
2.2.3 循环指令的程序流控制	(18)
2.2.4 分支指令的程序流控制	(19)
2.2.5 小结	(20)
2.3 SHARC 的中断	(20)
2.3.1 SHARC 中断的基本原理	(20)
2.3.2 中断向量表及相关的控制寄存器	(20)
2.3.3 SHARC 中断使用	(23)
2.3.4 中断现场保护	(24)
2.3.5 中断重入	(25)
2.3.6 小结	(25)
2.4 地址空间和寻址	(26)
2.4.1 地址空间	(26)
2.4.2 寻址方式	(29)

2.4.3 各种字长数据的传输	(32)
2.4.4 小结	(34)
第3章 DMA 传输与串口、Link 口通信	(35)
3.1 DMA 传输	(35)
3.1.1 DMA 的基本概念	(35)
3.1.2 与 DMA 相关的寄存器	(36)
3.1.3 外口 DMA 传输	(41)
3.1.4 DMA 中断	(42)
3.1.5 DMA 的优先级	(43)
3.1.6 链式 DMA	(43)
3.1.7 小结	(47)
3.2 串口通信	(47)
3.2.1 SHARC 串口简介	(47)
3.2.2 重要寄存器及其使用	(49)
3.2.3 串口中断	(54)
3.2.4 串口 DMA 传输	(54)
3.2.5 应用实例	(55)
3.2.6 小结	(58)
3.3 Link 口	(58)
3.3.1 Link 口的构成	(58)
3.3.2 Link 口的传输协议	(59)
3.3.3 Link 口的控制	(59)
3.3.4 Link 口缓存	(62)
3.3.5 Link 口的 DMA	(63)
3.3.6 Link 口中断	(64)
3.3.7 DMA 片内数据块搬移	(67)
3.3.8 小结	(72)
第4章 SHARC 系统设计	(73)
4.1 SHARC 的多处理器系统——簇结构	(73)
4.1.1 簇结构的基本形式	(73)
4.1.2 总线仲裁	(73)
4.1.3 多处理数据传递	(75)
4.1.4 向量中断(VIRPT)	(76)
4.1.5 相关的状态寄存器	(77)
4.1.6 多 SHARC 互连	(77)
4.1.7 小结	(77)
4.2 MeshSP(网格)结构	(78)

4.2.1 MeshSP(网格)结构的基本形式	(79)
4.2.2 小结	(80)
4.3 通过 RTOS 实现并行处理	(81)
4.3.1 RTOS 简介	(81)
4.3.2 在 RTOS 环境下开发并行系统	(82)
4.3.3 APEX PRO	(83)
4.3.4 并行程序的开发与调试	(83)
4.3.5 小结	(83)
4.4 关键信号、JTAG 接口与加载模式	(83)
4.4.1 关键信号的定义与连接	(83)
4.4.2 JTAG 接口	(84)
4.4.3 程序自加载	(86)
4.4.4 小结	(89)
4.5 高速数字电路设计简介	(89)
4.5.1 非理想因素	(89)
4.5.2 端接匹配	(92)
4.5.3 串扰性质及其消除	(94)
4.5.4 传输延时一致性	(95)
4.5.5 EDA 设计	(95)
4.5.6 SHARC 系统设计中应注意的问题	(98)
4.5.7 小结	(100)
4.6 SHARC 与 FLASH MEMORY 的接口	(100)
第 5 章 SHARC 的指令系统	(106)
5.1 SHARC 指令系统的组成与分类	(106)
5.2 数学运算指令汇编	(110)
5.2.1 ALU 的数学运算指令	(110)
5.2.2 MPU 的数学运算指令	(123)
5.2.3 Shifter 的数学运算指令	(125)
5.3 程序流控制指令后缀	(128)
5.4 常用算法	(129)
5.4.1 开平方	(129)
5.4.2 除法	(130)
5.4.3 正弦、余弦函数	(130)
5.5 SHARC C 语言	(132)
5.5.1 SHARC C 程序的开发过程	(132)
5.5.2 混合编程	(135)
5.5.3 语法扩展	(141)
5.5.4 小结	(143)

第 6 章 SHARC 家族其他成员	(144)
6.1 ADSP21061、ADSP21062、ADSP21065 简介	(144)
6.2 四 SHARC 的联合模块——AD14060	(145)
6.2.1 AD14060 的特点	(145)
6.2.2 AD14060 的内部结构	(145)
6.3 高性能的 SHARC 二代——ADSP21160	(146)
6.3.1 ADSP21160 的特点	(146)
6.3.2 单指令多数据流机制(SIMD)	(147)
6.3.3 多样化的内存访问方式	(148)
6.3.4 时钟	(149)
6.4 SHARC 家族的飞跃——TigerSHARC	(150)
6.4.1 TigerSHARC 的主要特点	(150)
6.4.2 TigerSHARC 的内核	(151)
6.5 网上信息	(153)
附录 A 引脚功能描述与芯片封装形式	(154)
附录 B 时序	(160)
附录 C 寄存器	(168)
参考文献	(183)

第 1 章 概 述

1.1 DSP 的基本概念

DSP 是 digital signal processor^{1*} 的简称,是一种专门用来实现信号处理算法的微处理器芯片。根据使用方法的不同,DSP 可以分为专用 DSP 和可编程 DSP。专用 DSP 只能用来实现某种特定的数字信号处理功能,如数字滤波、FFT 等。专用 DSP 不需编程,使用方便,处理速度快,但是灵活性差。可编程 DSP 则像 GPP(General Purpose Processor,如 Pentium)一样有完整的指令系统,通过软件实现各种功能。若无特别说明,在本书中 DSP 均指可编程 DSP。

DSP 的产生主要是为了满足通信、雷达、数字电视等领域对实时数字信号处理的需要。典型的数字信号处理算法包括数字滤波、FFT 等。这些算法的共同特点是要进行密集的数学计算,因此 DSP 在体系结构上采取了一系列措施,使之在数学计算方面具有特别突出的性能;而在其它方面,例如文字处理、数据库管理等则不如 GPP。除了密集的数学计算之外,DSP 应用的另一个突出特点是实时性。在许多应用领域,如通信中的调制、解调、雷达中信号检测等等,数据是以帧为单位更新的,每帧的长度一般为微秒到毫秒量级,DSP 必须在这段时间之内完成处理并输出结果,这就是所谓实时处理。显然,实时处理要求处理器具有极高的处理速度,能够对外部事件迅速做出反应(能够及时迅速地响应中断),并且具有强大的 IO 吞吐能力。

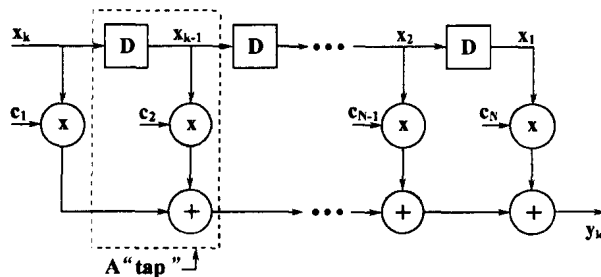
1.1.1 DSP 的结构特点

为了适应实时信号处理的需要,DSP 采用了一些特殊的结构,使 DSP 从 GPP 中分离出来,成为一类独立的微处理器。

1.1.1.1 MAC(乘累加)指令

至少在 DSP 发展初期,MAC 指令是 DSP 最突出的特点。让我们以一个最典型的数字信号处理算法,即 FIR 滤波器为例来说明 MAC 操作对 DSP 的重要意义。

如图 1-1 所示,FIR 滤波器由若干阶组成,每一阶包含一个延时器(在图中用字母 D 表示)、一个乘法器和一个加法器。要高效率地实现 FIR 滤波器,就



$$y_k = x_k \times c_1 + x_{k-1} \times c_2 + \dots + x_2 \times c_{N-1} + x_1 \times c_N$$

图 1-1 FIR 滤波器

* 在许多场合,DSP 指的是 digital signal processing,也就是数字信号处理。这是一个相当广义的概念,几乎涵盖了数字技术的各个方面;但是当硬件工程师和芯片制造商提到 DSP 时,它们所指的是专门用来实现信号处理算法的微处理器芯片,如 TMS320C50,ADSP21060 等。

必须能够快速实现乘法运算。大家知道,早期的 GPP 是用软件来实现乘法运算的,即使是现在,依然有许多 GPP 需要多个指令周期才能完成一次乘法运算。而 DSP 都配备了硬件乘法器,一般能够在一个时钟周期内完成乘法。另外,由于每产生一个 FIR 滤波器的输出 y_k ,都需要对多次乘法结果进行累加,所以 DSP 一般都设有专用累加器;为了防止溢出,累加器的字长要超过普通寄存器。在这些硬件的支持下,DSP 设置了 MAC 指令,可以在一个指令周期内同时完成乘法和累加操作,从而最大限度地提高效率。由于大多数数字信号处理算法都像 FIR 滤波器一样,主要由乘加运算组成,所以 MAC 操作对 DSP 有着特殊的意义,以至于在一些场合 MACs/sec(每秒钟实现 MAC 操作的次数)可以作为一种衡量 DSP 处理能力的指标。

1.1.1.2 存储器结构

DSP 和 GPP 的另一个重要区别是存储器结构。仍然以 FIR 滤波器为例,实现每一阶 FIR 滤波器一般需要进行三次内存访问*:取 MAC 指令、取输入数据、取系数 c_i 。

GPP 一般采用 Von Neumann 结构,如图 1-2 所示,系统只有一套总线(包括数据总线和地址总线)和单一存储器,无论是数据还是指令都要经过同样的数据通道进入处理器内核。所以当用 GPP 实现 FIR 滤波器时,存储器就自然成了系统瓶颈,处理器内核会经常因为等待数据而闲置(假设 MAC 操作能够在一个周期内实现)。

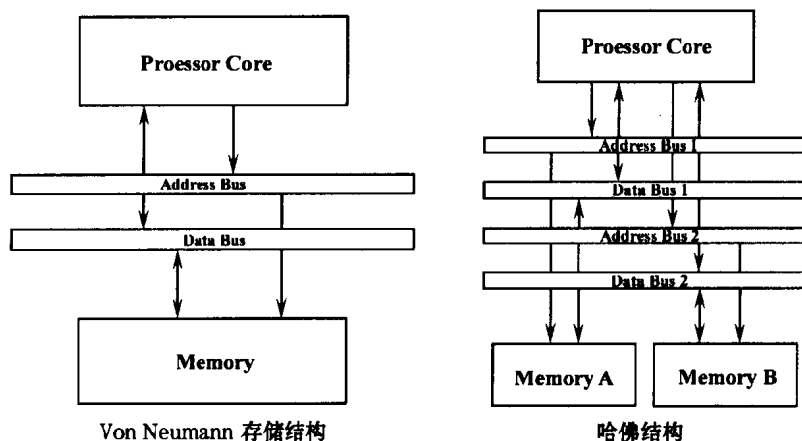


图 1-2 两种存储结构的比较

要避免存储器瓶颈,力争在尽量少的指令周期甚至单周期内实现一阶 FIR 滤波器,就必须能够在一个周期内访问多次存储器,因此,DSP 一般采用了如图所示的哈佛结构。

哈佛结构将指令存储空间和数据存储空间分离开,各自拥有独立的总线,这就使取指令和访问数据可以同时进行,从而缓解了存储器的瓶颈效应。AD 公司的 ADSP-21xx 系列 DSP 和 Lucent Technologies 的 DSP16xx 系列 DSP 都采用了这种结构。

许多 DSP,如 TI 公司的 TMS320C50,使用了所谓改进的哈佛结构。这种结构依然保持了哈佛结构双存储区双总线的基本特征,但是每个存储区都既可以存放数据也可以存放指令。这样处理器内核在一个周期里仍旧可以访问两次存储器,但比起标准哈佛结构来显得更加灵活。

对哈佛结构的另一个改进是进一步增加存储空间和总线的数量。这种做法的主要问题是

* 事实上这是一个比较保守的估计,由于延时器的作用,缓冲区内的数据需要不断平移,也就是被复制到相邻的存储器单元;此外,如果访问数据要用指令显式实现,则要取的指令也将不止 MAC 一条。

成本高,不过随着微电子工业的发展,这一问题有可能逐步得到解决。目前 TI 公司的最新产品 TMS320C6000 系列和 AD 公司将要推出的 TigerSHARC 就都拥有三个存储器空间。

在采用哈佛结构的同时,一些 DSP 还使用了其它方法增加总线带宽。比较成功的方法是增加数据总线宽度。例如 TMS320C6201 的程序存储空间的数据总线的宽度是 256 位,所以在—个周期内可以传送 8 条 32 位指令。

这里有两点需要说明,第一,由于管脚数目的限制,大多数 DSP 芯片在访问片外存储器时仍然只能通过单一总线,所以如果处理器内核要从片外存储器取数据和指令,处理速度就会大大降低。第二,高性能的 GPP 在片内设有数据 CACHE 和指令 CACHE,并且分别由总线与内核相连。从物理上看,这种 CACHE 结构与哈佛结构十分相似,但二者在逻辑上是截然不同的。DSP 的片内存储器是整个存储器地址空间的一部分,完全由软件来管理。而 CACHE 没有自己的地址,它的内容是某一块存储器的副本,由硬件完成管理,程序员一般不能干预哪些数据将进驻 CACHE。多数 DSP 设置了片内存储器而没有设置 CACHE,这固然有成本方面的考虑,但同时也是由 DSP 的用途决定的。CACHE 结构是建立在数据局部性原理之上的,但在实时信号处理过程中,DSP 处理的数据是不断更新的,数据 CACHE 很难发挥作用,因此 DSP 极少使用数据 CACHE。由于 DSP 的程序一般是循环执行的,所以有些 DSP,如 ADSP21060 在内核中设置少量的指令 CACHE,以便腾出总线访问更多的数据。DSP 很少使用 CACHE 的另一个原因是便于预测指令的执行时间,对此后面还会加以介绍。

1.1.1.3 寻址方式

在进行数字信号处理时,处理器往往要同时维护多个数据缓冲区,而且每个缓冲区的指针都要频繁移动。例如在复数 FIR 滤波器中,输入数据 x_i 和系数 c_i 各需要两个缓冲区,每完成—阶 FIR 之后,各缓冲区的指针都要移向下一个位置。GPP 虽然有基址和变址寄存器,但是这些寄存器不能自动更新;而且数量也有限,每个寄存器往往要管理多个缓冲区。DSP 用特殊的硬件来寻址数据存储器,有大量寄存器可以用作数据指针(如 ADSP21060 有 16 个),指针的更新可以和其他操作并行执行,所以不占用处理时间。DSP 还支持一些特殊的寻址方式,如用来实现环形缓冲区的环形寻址,实现 FFT 变换所必需的逆序寻址等。

1.1.1.4 零耗循环(zero overhead loop)

GPP 每执行—次循环都要用软件判断循环结束条件是否满足,更新循环计数器,还要进行条件转移。这些例行操作要消耗几个周期的时间,这种消耗对于短循环是相当可观的。与 GPP 不同,DSP 可以用硬件实现更新计数器等例行操作,不用额外消耗任何时间,所以是一种零耗循环。由于数字信号处理程序百分之九十的执行时间是在循环中渡过,所以零耗循环对提高程序效率是非常重要的。

1.1.1.5 程序执行时间可预测

实时处理不仅要求处理器必须具有极高的计算速度,而且还要求程序的执行时间要容易预测,否则开发人员无法判断自己的系统是否满足实时要求。高性能 GPP 普遍采用了 CACHE 和动态分支预测技术,这些动态特性虽然能够从统计角度提高处理速度,但也使处理时间很难精确预测,因为当前指令的执行时间要受到程序运行的历史过程的影响。尽管从理论上说,程序员可以推测出最坏情况下的执行时间,但是由于各种动态特性的相互影响,最坏执行时间可能远远超过程序的典型执行时间,这将导致系统设计过于保守,严重浪费资源。与 GPP 不同,DSP 的动态特性较少,而且还通过设置 MAX(求最大值)、MIN(求最小值)、CLIP

(限幅)等指令来减少条件判断对执行时间的影响。这些措施使 DSP 的执行时间容易预测,同时,DSP 生产商还提供了能够精确模拟每一条指令执行状态的软件仿真器 Simulator,使设计人员在硬件系统完成之前就能够调试程序并验证处理时间。值得注意的是,TI 的最新产品 TMS320C6011 设置了可选择的两级 CACHE,而 AD 将要推出的 TigerSHARC 采用了动态分支预测技术。这是否意味着 DSP 正在丧失程序执行时间可预测的特点,或者正在准备采取其他措施(如提供工具软件)来弥补因芯片结构日趋复杂对预测时间造成的不利影响,我们将拭目以待。

1.1.1.6 外围设备

GPP 硬件系统(如 PC 机)的开发一般由专业公司承担,用户只从事软件开发。而 DSP 工程师往往要自己设计硬件平台,而且许多 DSP 应用系统特别是嵌入式系统对体积、功耗有严格的限制,所以 DSP 必须具备开发简便的特点。多数 DSP 支持 IEEE1149.1 标准,用户可以通过 JTAG 端口对 DSP 进行在线实时仿真。另外 DSP 体现了片上系统(system on chip)的设计思想,在片上集成了 DMA、中断控制、串行通信口、上位机接口、定时器等外围设备,有的 DSP 还包含 AD 和 DA 转换器。所以用户通常只需要外加很少的器件就可以构成自己的 DSP 系统。

1.1.2 DSP 的发展

1.1.2.1 DSP 的发展历程

尽管 70 年代末就出现了一些具有 DSP 性质的处理器,如 NEC 的 μ PD7720 等,但是公认的第一种商业上成功的 DSP 是 1982 年 TI 公司推出的 TMS320C10。从那时算起,DSP 大致经历了 4 个发展阶段。

(1) 1982 年 TI 公司推出的 16bit 定点 DSP TMS320C10 是第一代 DSP 的代表,它的哈佛结构和 MAC 指令标志着 DSP 的诞生,不过当时 TMS320C10 完成 MAC 操作还需要 390ns。

(2) DSP56001 是 Motorola 公司于 1987 年推出的,它的数据和指令的长度都是 24bit,可以支持单多指令的硬件循环,有循环寻址能力,完成 MAC 操作需要 75ns。同时代的其它产品还有 AT&T 公司的 DSP16A、AD 公司的 ADSP-2100, TI 公司的 TMS320C50 等。这些构成了第二代 DSP 产品。

(3) 第三代 DSP 出现在 1995 年,代表产品有 Motorola 公司的 DSP56301、TI 公司的 TMS320C541 和 AD 公司的 ADSP21060 等。这些产品开始对传统的 DSP 结构进行改进,在数据通路上增加了专用功能单元或可并行的功能单元。有的产品,如 TI 的 TMS320C80 和 Motorola 的 MC68356 还在片内集成多处理器,所有这些措施都是希望通过并行工作来提高运算能力。这时的 DSP 还普遍增加了片内存储器的规模,并开始支持 3V 或 3.3V 电压。为了适应 ASIC 的发展需要,各大 DSP 厂家除了提供成品 DSP 之外,还同时提供 DSP 内核,用户可以把购买的 DSP 内核与自己设计的外围设备集成在一个芯片上,获得最佳的系统集成效果。第三代 DSP 完成 MAC 操作大约需要 20ns。

(4) 第四代 DSP 出现在 1997 到 1998 左右,代表产品是 TI 公司的 TMS320C6201。它通过片内锁相倍频和复杂的流水线结构使指令周期达到了 5ns。采用了甚长指令字(VLIW: Very Long Instruction Word)结构,极大地提高了片内并行能力。这些措施使 TMS320C6201 平均在 3ns 左右就可以完成一次 MAC 操作。

由于各 DSP 厂商都有自己的产品系列和发展战略,所以并非所有 DSP 产品都能按上述方法划归为某一代 DSP,而同属一代的 DSP 产品的特点也不尽相同,但是上面的分类方法还是基本反映了 DSP 的整体发展过程。从这一发展过程中我们可以发现 DSP 正在经历一个重要的转折时期,表现在以下几个方面:

(1) DSP 的市场竞争空前激烈,产品换代的速度不断加快,从第二代 DSP 发展到第三代用了将近 8 年,而第三代和第四代产品相隔不到三年。

(2) 时钟频率突然大幅度提高。以 TI 公司为例,TMS320C50 的指令周期是 35ns,TMS320C80 的指令周期是 25ns,提高并不显著,这说明在这一时期提高时钟频率并不是 DSP 发展的重点。但是到了 TMS320C6201,时钟频率猛增到 200MHz,而且 TI 已经宣布要推出 300MHz 的产品。

(3) DSP 的结构出现了重要变化。在 DSP 发展的最初 10 年中,DSP 的结构并没有重大改变,但是近几年 DSP 厂家纷纷推出新的 DSP 体系结构。其目的是在提高时钟频率的同时,通过增加并行性进一步增强 DSP 的处理能力。具体的方法有两个:一是在一条指令里面完成更多的功能,二是在一个周期内完成多条指令。单指令流多数据流(SIMD)是第一种方式的代表,这种结构的 DSP 设置有多个相同的功能单元,各单元同时执行相同的指令(但处理的数据不同)。SIMD 方式的 DSP 一般称为增强型的传统结构,因为它在指令上和过去产品有较好的兼容性,结构上和传统 DSP 结构也有较强的继承性。SIMD 典型的产品有 ADSP21160。由于 SIMD 型 DSP 与它的前辈产品在一定程度上具有兼容性,所以比较容易开发,但是 SIMD 方式对 DSP 性能的改善是有限的,一些公司只是把它作为一种过渡性的解决方案。相比之下,第二种方法,即在一个周期内完成多条指令更有发展前途。TMS320C6201 就是这方面的成功代表。TMS320C6201 片内集成了 8 个功能单元,利用 VLIW 方式,在一个周期内完成 8 个操作。超标量结构是另一种在一个周期内完成多条指令的方法,例如 AD 公司的 TigerSHARC 有两条流水线,每秒钟可以执行 20 亿次 MAC 操作。由于新一代 DSP 的处理能力急剧膨胀,对数据带宽的要求迅速增加,如果总线的传输速度跟不上内核的处理速度,内核就会经常处于停工待料的状态。所以新一代 DSP 普遍采用了三个独立的片内存储器区,每个存储器区拥有自己的 64 位或 128 位总线,内核与片内存储器的传输速度达到了惊人的水平,如 TMS320C6201 达到 8G 字节每秒,而 TigerSHARC 是 12G 字节每秒。相比之下,DSP 对片外存储器的访问能力提高相对缓慢,因为片外存储器只能通过单一总线,访问周期一般也比较长。因此,目前外存储器瓶颈是影响新一代 DSP 实际性能的一个突出问题。最近出现的另一个趋势是 DSP 的 RISC(精简指令集)特征日益明显。传统的 DSP 多采用复合指令,试图在一条指令中实现多个功能,例如 ADSP21060 最多可以在一个指令内实现加、减、乘法各一次,同时还可以访问两次内存并且修改两个地址指针,而现在最新的 DSP 的指令功能却变得相对简单,这是与时钟频率的提高和多指令并行相联系的,新型 DSP 主要通过更短的时钟周期内完成多条简单指令来提高处理能力。

(4) 随着新一代 DSP 逐渐成为主流产品,DSP 的开发方法也正在发生变革。首先在电路设计上,DSP 的频率已经提高到 100MHz 以上,有的芯片的管脚间距只有 20mil*,手工布线已经不能保证系统可靠工作,必须借助先进的 EDA 工具完成布局和布线,并对电路板的电特性

* 1mil 等于千分之一英寸。

进行充分仿真。新一代 DSP 的另一个问题是编程难度增加。为了使 DSP 达到最佳处理能力,程序员必须尽量避免流水线的断流和功能模块的闲置,但是由于新一代 DSP 复杂的流水线结构(这是时钟频率增加的一个代价)和大量采用并行功能模块,做到这点越来越困难。再加上 DSP 软件的规模在不断扩大,这都使得 DSP 软件开发的成本迅速增加,所以过去那种完全依靠程序员用汇编语言手工编程,用原始的命令行形式的编译器、连接器产生代码的开发方式越来越不能适应 DSP 的发展。今后,尽管高级语言的应用范围依然会受到编译器效率的影响,内层循环也仍需要手工优化,但是使用高级语言开发的 DSP 软件必将逐渐增加。同时,集编辑、编译、连接、调试、评估于一体的集成开发环境也会逐渐取代原始的代码开发工具。

1.1.2.2 DSP 的发展预测

随着微电子技术向深亚微米发展,DSP 一定会变得速度更快,片内集成的功能单元和存储器更多,同时为了降低功耗,供电电压也会进一步降低。除此之外,我们还可以从另一个角度,也就是 DSP 与 GPP 和 μC (微控制器,也就是通常所说的单片机)的互动关系来考察 DSP 的发展趋势。

前面讲过,哈佛结构和 MAC 指令等特征使 DSP 从普通微处理器中独立出来,并且在数字信号处理方面超过了 GPP,例如图 1-3 对 TMS320C50 和同档次的 GPP ARM7TDMI 做了比较,显然 C50 的速度优于 ARM7TDMI。图中的 BDTI 指标是 Berkeley Design Technology Incorporation 开发的一种评估处理器速度的指标,它用一系列数字信号处理的核心算法,如 FFT、FIR 等作为测试程序,从而避免了 MIPS 或 MFLOPS 等指标的片面性。

但是近年来 GPP 的发展速度比 DSP 更快。GPP 的主频要高出 DSP 许多,片内的并行程度也更高,而且由于 DSP 应用市场的迅速发展,Intel 等著名的 GPP 生产商也开始涉足 DSP 市场。一个典型的例子就是 Pentium MMX,Intel 公司在这种处理器中增加了一个指令子集,专门用于支持数字信号处理。有人认为 MMX 实际上是 GPP 和 DSP 的混血儿,并称之为 GP/DSP。其它一些 GPP 厂商也相继推出了自己的混合型处理器。这种发展的结果是,目前 GPP 和 GP/DSP 在处理速度上已经超过了 DSP,即使是在数字信号处理方面也是如此。如图 1-4 所示,MMX 和 PowerPC 的处理能力远远超过同时期的浮点 DSP,仅略逊于 TMS320C67,需要注意的是 167MHz 的 TMS320C67 现在还没有正式上市,而 MMX 等 GPP 则早已被新产品所取代,现在的奔腾 III 处理器的主频已经达到了 550MHz。图 1-4 给出了一些 DSP 与 GPP 的速度比较:

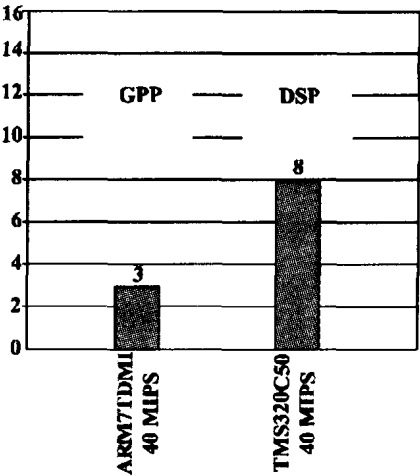


图 1-3 GPP 和 DSP 的速度比较
(单位:BDTI 指标)

可见,DSP 与 GPP 在速度方面的差距在加大。虽然由于功耗、封装、执行时间的可预测程度和调试工具等原因,GPP 目前还不能完全取代 DSP 在实时信号处理方面的地位,但是 GPP 速度方面的优势也的确对 DSP 造成了相当的压力。相信正是由于这种压力,促成近年来 DSP 发生了明显变化,包括大幅度提高主频,采用复杂的流水线,以及超标量结构和动态分支预测技术等等。超标量结构和动态分支预测等技术都是 GPP 的典型技术,以至有人认为 GPP 的

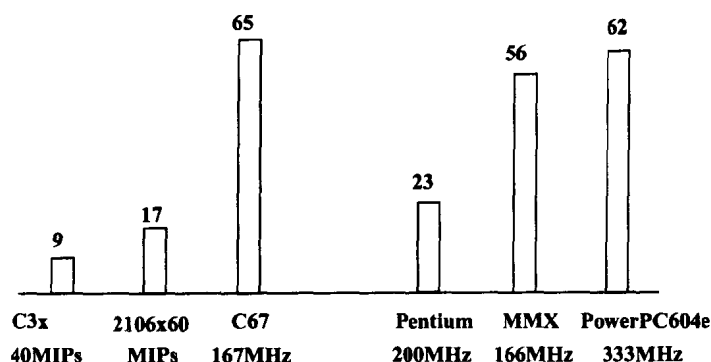


图 1-4 几种 DSP 与 GPP 的性能比较(单位:BDTI 指标)

发展方向就是 DSP 的发展方向,只不过 DSP 滞后了若干年而已。尽管这种看法过于绝对,但是 DSP 和 GPP 无疑在不断吸收对方的优点,二者的界限正在趋于模糊。1999 年 3 月,Intel 公司和 AD 公司宣布要联合开发下一代 DSP 内核,相信这一举措一定会加速 GPP 与 DSP 之间的相互渗透。

提高时钟频率、采用并行结构等措施在提高处理能力的同时也削弱了 DSP 功耗低、使用简单等传统特色,因此在高端 DSP 向 GPP 靠拢的同时,为了适应那些对体积、功耗有苛刻要求的便携电子设备的需要,一些低端 DSP 也在努力减小体积功耗,另外还增加控制功能。因为有些场合,如移动电话,需要同时具有控制功能和信号处理能力。当然一些微控制器的生产商也看到了这点,例如 Hitachi 公司就在 SH-2 微控制器中增加了 DSP 功能,形成了所谓 SH-DSP。这些都说明在高端 DSP 与 GPP 融合的同时,低端 DSP 与微控制器也在相互渗透。

1.2 SHARC 简介

下面介绍 AD 公司的 SHARC 系列 DSP。SHARC 是 Super Harvard Architecture Computer (超级哈佛结构计算机)的简称,目前包括 4 种产品:ADSP21060、ADSP21061、ADSP21062 和 ADSP21065。这里要介绍的是 SHARC 家族的代表:ADSP21060。由于 ADSP21062 和 ADSP21061 是 21060 的简化版本,所以这些内容对它们也同样适用。

1.2.1 SHARC 的基本结构

ADSP21060 是一种高性能的 32 位浮点 DSP,它的基本的特点有:

- 最高工作频率为 40MHz,时钟周期 25ns。
- 数据线有 48 根,地址线有 32 根,地址范围 4G。
- 所有指令都是单周期指令,指令长度均为 48bit。
- 32-bit IEEE 浮点运算单元,内含乘法器,ALU 和移位器,并支持 40bit 的扩展精度浮点运算。
- 10 个 DMA 通道。
- 4M bit 双口片内存储器。
- 有两个同步串口和 6 个 LINK 端口。

- 支持多处理器共享总线。

图 1-5 是 ADSP21060 的结构框图：

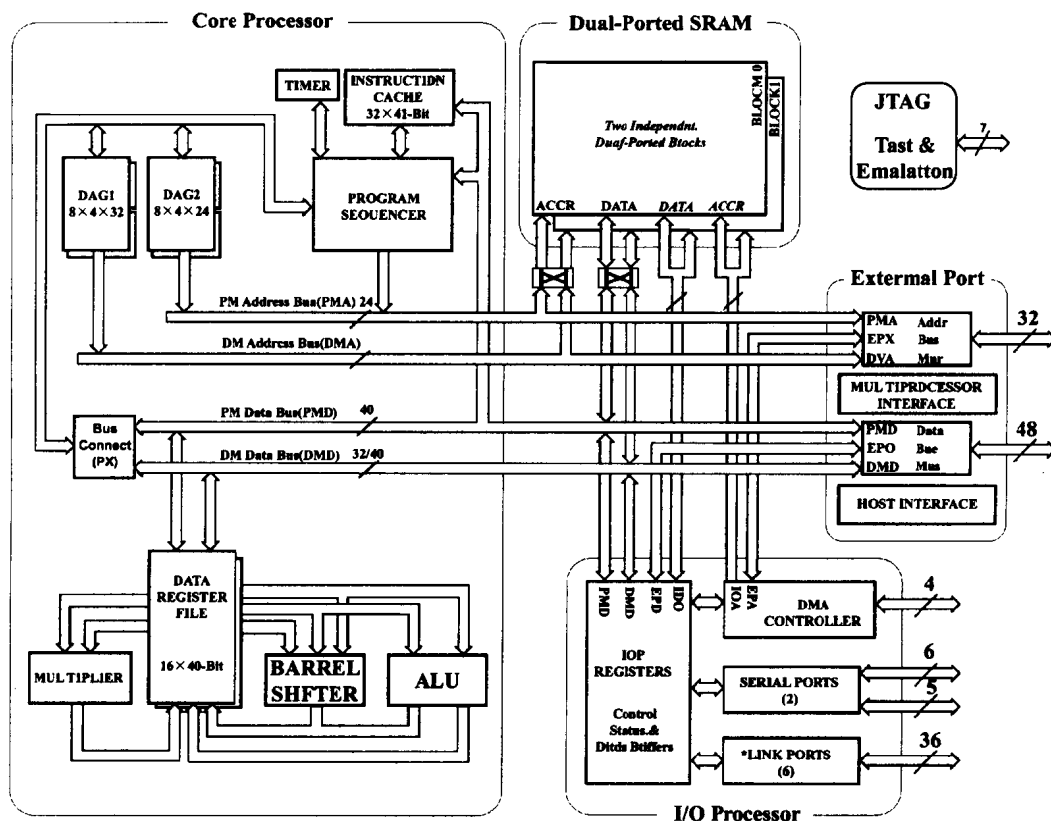


图 1-5 SHARC 的结构

1.2.1.1 内核处理器

SHARC 的内核处理器(Core Processor)简称内核,是指令执行的场所,其作用相当于微机的 CPU。SHARC 内核包括运算单元、DAG(Data Address Generator:数据地址产生器)、程序流控制器和指令 CACHE 等部分。运算单元可以进行 32 位定点和 32 位或 40 位浮点的加减运算和乘法运算,有丰富的逻辑和移位操作,并且支持对数、倒数和开方运算。

1.2.1.2 总线和内存

SHARC 有 3 套独立的片内总线,它们是 PM 总线(程序存储器总线)、DM 总线(数据存储器总线)和 IO 总线(输入、输出总线),每套总线都有独立的数据总线和地址总线。DM 的数据总线有 40bit,地址总线 32bit,可以寻址全部 4G 地址空间。PM 数据总线有 48bit,地址总线有 24bit,只能寻址低端的 12M 地址空间。PM 和 DM 总线都是由 SHARC 的内核处理器控制的,取指令只能通过 PM 总线,而访问数据既可以用 DM 总线也可以用 PM 总线。IO 总线是由 IO 处理器管理的,它的数据总线是 48bit,地址总线是 17bit,SHARC 的 DMA 控制器以及多处理器系统的其它处理器可以通过 IO 总线访问 SHARC 的内存。

ADSP21060 有 4Mbit 的片内双端口存储器,分成 Block0 和 Block1 两块,每块都可以被独立访问。如此大的片内存储器在 DSP 芯片中是不多见的,这也是 ADSP21060 的一个重要特点。