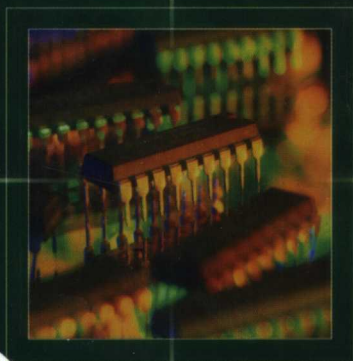
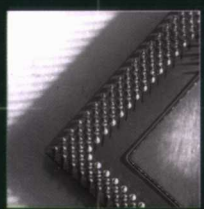


TI-DSP系列开发应用技巧丛书

TMS320C5000系列DSP 开发应用技巧

——重点与难点剖析

李绍胜 赵振涛 编著



中国电力出版社

www.infopower.com.cn

TI-DSP系列开发应用技巧丛书

TMS320C5000系列DSP 开发应用技巧

——重点与难点剖析

李绍胜 赵振涛 编著



中国电力出版社

www.infopower.com.cn

内容提要

本书是 TI-DSP 系列开发应用技巧丛书之一。全书系统介绍了目前最通用、最流行的德州仪器的 TMS320C5000 系列芯片的系统结构、内部硬件资源和软件资源, DSP 系统与外界接口及 CCS 编程等, 并配有大量的应用系统实际开发案例。本书本着易于领会、由浅入深的原则, 根据作者多年一线研发经验, 重点解析了开发 TMS320C5000 系列芯片过程中遇到的难点和分析问题的方法, 并采用了 C5509/C5510 和 C54xx 对比描述的方式, 加深读者对 DSP 的深入了解。

本书实例典型、内容丰富新颖, 具有很强的实用性和指导性, 特别适合于 DSP 工程开发人员作为工作参考书, 也适合作高等院校教材, 满足本科生、研究生对 DSP 系统学习和提高的需求。

图书在版编目(CIP)数据

TMS320C5000 系列 DSP 开发应用技巧——重点与难点剖析 / 赵振涛编著. —北京: 中国电力出版社, 2007
(TI-DSP 系列开发应用技巧丛书)

ISBN 978-7-5083-5366-1

I. T… II. 赵… III. 数字信号—信息处理系统 IV. TN911.72

中国版本图书馆 CIP 数据核字 (2007) 第 061366 号

丛 书 名: TI-DSP 系列开发应用技巧丛书

书 名: TMS320C5000 系列 DSP 开发应用技巧——重点与难点剖析

出版发行: 中国电力出版社

地 址: 北京市三里河路 6 号

邮政编码: 100044

电 话: (010) 68362602

传 真: (010) 68316497, 88383619

服务电话: (010) 58383411

传 真: (010) 58383267

E-mail: infopower@cepp.com.cn

印 刷: 北京市同江印刷厂印刷

开本尺寸: 185×260

印 张: 22

字 数: 538 千字

书 号: ISBN 978-7-5083-5366-1

版 次: 2007 年 8 月北京第 1 版

印 次: 2007 年 8 月第 1 次印刷

印 数: 0001—4000 册

定 价: 36.00 元

敬告读者

本书封面贴有防伪标签, 加热后中心图案消失

本书如有印装质量问题, 我社发行部负责退换

版 权 专 有 翻 印 必 究

前 言

自 1982 年美国 TI 公司诞生第一代数字信号处理器 (DSP) 以来, 该产品系列已经得到迅速发展, 并在信号处理、通信、控制、医疗、多媒体处理等领域得到了广泛应用。TI 公司的 C5000 是最新一代 16 位定点 DSP 系列, 它在芯片技术方面具有高性能、低功耗、低成本等优势, 同时与上一代芯片也具有较好的代码兼容性。其内核架构与 ARM 体系结合, 衍生出的 OMAP 双核系统在便携式移动终端中也得到了广泛应用。

本书是作者近几年来从事 TMS320 系列 DSP 系统设计和开发的技术结晶, 本着易于领会、由浅入深的原则, 凭借在 DSP 开发过程中积累的独特经验, 全面系统地介绍了 TMS320C5000 系列 DSP, 并结合具体实例讲解设计中的技巧。在本书的编写过程中, 我们还将 DSP 及相关的软硬件基础知识作为知识点, 以问答的方式提出问题并作相应解答以提升读者对本书的阅读兴趣, 而且我们还将相关领域的知识进行了总结并与新内容融合、贯通, 进一步充实、完善了专业知识体系, 也在一定程度上丰富了本书的内容。

全书共分为 8 章, 内容包括 DSP 特性及其选型方法、TMS320C5000 系列硬件结构、内部功能模块、汇编和 C/C++ 程序设计及混合编程和算法优化、DSP/BIOS 程序设计、RTDX 的应用、片级支持库 (CSL) 的使用、Boot Loader 程序设计以及应用程序设计。

对于开发者最迫切需要掌握的重点章节, 本书投入了大量篇幅的应用例子来讲述。阅读本书, 将会对 TI 公司其他系列 DSP 的设计与应用具有直接的指导意义。

本书内容兼顾 DSP 知识的系统化和开发流程, 每个重点知识均作为专题讲解, 可作为高等院校电子信息、通信、自动控制、仪器仪表类专业本科生和研究生学习 DSP 的高级进阶教材和参考书, 也可供从事 DSP 开发与应用的广大工程技术人员参考。因此, 在阅读本书之前, 建议读者先了解 DSP 芯片的基本原理和常用 DSP 芯片的应用, 熟悉 DSP 芯片开发工具及使用, 掌握 DSP 系统的软硬件设计的基本方法。

本书由李绍胜、赵振涛执笔主编, 李绍胜组织编写、审校, 赵振涛负责统稿工作。在本书编写过程中, 李巧红、胡频、许雪燕、刘渠等在资料收集、整理等方面做了大量的准备工作; 此外还得到了北京邮电大学信息工程学院赵振纲教授的悉心指点和帮助。在此一并表示衷心的感谢!

望本书对读者的工作和学习有所裨益! 限于作者的水平, 难免在编写过程中出现错误和不妥之处, 恳请广大读者批评指正!

作 者
2007 年 5 月

目 录

前 言

第 1 章 DSP 概述.....	1
1.1 数字信号处理技术.....	1
1.2 DSP 特点概述.....	2
1.3 DSP 的系统设计过程.....	3
1.4 DSP 处理器.....	4
1.5 DSP 芯片的应用领域.....	7
1.6 TI 的 C5000 系列中 C54x 和 C55xx 的比较	7
第 2 章 TMS320C5000 系统结构.....	10
2.1 TMS320C55xx 体系结构综述	10
2.2 存储空间结构	16
2.3 寻址方式	22
2.4 流水线操作	35
2.5 CPU 寄存器	53
2.6 OMAP 简介	82
第 3 章 TMS320VC55xx 系列 DSP 的外设.....	87
3.1 TMS320VC55x DSP 外设概述	87
3.2 时钟发生器	88
3.3 外部存储器接口 (EMIF)	91
3.4 直接存储器访问 (DMA) 控制器.....	108
3.5 主机接口 (HPI) 和增强型主机接口 (EHPI)	124
3.6 多通道缓冲器接口 (McBSP)	131
3.7 多媒体卡 (MMC) 控制器	137
3.8 通用串行总线 (USB)	145
3.9 实时时钟	155
3.10 中断和定时器	160
第 4 章 汇编语言程序和.cmd 文件设计	174
4.1 TMS320C55x 汇编指令系统	174
4.2 汇编语言源程序格式	195
4.3 汇编源程序中的表达式	197
4.4 .cmd 文件和汇编伪指令	200
4.5 MEMORY 伪指令及其使用	214
4.6 SECTIONS 伪指令及其使用	216

4.7	汇编器的内部函数.....	223
4.8	汇编与优化.....	225
第 5 章	C/C++语言设计和混合编程设计.....	226
5.1	C/C++程序设计数据结构及语法.....	226
5.2	C/C++语言编程的注意事项.....	230
5.3	汇编语言与 C/C++混合编程.....	241
5.4	设计与优化.....	251
5.5	关于程序的优化或提高运算速度的建议.....	256
第 6 章	CCS 的使用进阶及 DSP/BIOS	258
6.1	CCS 简介	258
6.2	简单应用程序调试实例.....	258
6.3	GEL 的应用.....	266
6.4	DSP/BIOS	285
6.5	片级支持库 CSL	294
6.6	基于 RTDX 技术的数据通信	307
6.7	CCS 与优化	313
第 7 章	DSP 系统的电路设计	317
7.1	JTAG 接口	317
7.2	C55xx 的电源电路.....	319
7.3	复位电路与看门狗.....	320
7.4	时钟信号的接入简介.....	322
7.5	存储器的接口.....	322
7.6	异步串口	328
7.7	音频接口	331
第 8 章	Bootloader	336
8.1	概述	336
8.2	DSP 系统的上电程序加载	337
	参考文献.....	344

第 1 章 DSP 概 述

人们通常所说的 DSP 有两种解释，一种是数字信号处理技术 (Digital Signal Process)，另一种是数字信号处理器 (Digital Signal Processor)。

1.1 数字信号处理技术

作为信号数字化技术的基石，数字信号处理技术是一个不可或缺的角色。数字信号处理是一门涉及到许多学科而又广泛应用于许多领域的新兴学科。数字信号处理技术已广泛用于数字通信、雷达、遥感、声纳、语音合成、图像处理、测量与控制、HDTV、数字音响、多媒体技术、地球物理学、生物医学工程及机器人等领域。随着数字信号处理技术的发展，其研究范围和应用领域还在不断地发展和扩大。

数字信号处理是指人们利用计算机或专用处理设备，以数字的形式对信号进行采集、变换、滤波、估值、增强、压缩、识别等处理，以得到符合人们需要的信号形式。

图 1.1 是数字信号处理系统的简化框图。



图 1.1 数字信号处理系统的简化框图

数字信号处理技术日趋完善成熟，应用得越来越广泛，这是与它的突出优点分不开的，和模拟技术相比，它有以下优点：

- (1) 便于计算机处理控制。
- (2) 避免模拟信号的畸变失真。
- (3) 减小杂散信号的干扰 (提高抗干扰能力)。
- (4) 便于软件包调试及遥感、遥测和遥调。
- (5) 便于自诊断、容错等技术的植入。

数字信号处理技术的实现方法可以分为以下 5 类：

- (1) 在通用的计算机 (如 PC) 上用软件 (如 C 语言) 实现，一般可用于 DSP 算法的模拟与仿真。缺点是速度较慢。
- (2) 在通用计算机系统中加上专用的加速处理机实现。缺点是系统不便于独立运行，并且专用性太强，应用受到很大的限制。
- (3) 用通用的单片机 (如 MCS-51 系列) 实现，这种方法只适用于实现简单的 DSP 算法，如数字控制。
- (4) 用通用的可编程 DSP 芯片实现，与单片机相比，DSP 芯片具有更加适合于数字信号

处理的软件和硬件资源,可用于复杂的数字信号处理算法,这种方法也是应用最多的。

(5) 用专用的 DSP 芯片实现。在一些特殊的场合,要求的信号处理速度极高,用通用 DSP 芯片很难实现,如专用于 FFT、数字滤波、卷积、相关等算法的 DSP 芯片,它是把相应的处理算法在芯片内部用硬件资源实现,无需编程。

数字信号处理的主要缺点是处理量随处理精度、信息量的增加而成倍增长,解决这一问题的方法是研究高速运行的数字信号处理系统。而数字信号处理系统中最核心的数字信号处理器。

1.2 DSP 特点概述

数字信号处理器 (Digital Signal Processor, DSP) 是一种专门为实时、快速实现各种数字信号处理算法而设计的、具有特殊结构的微处理器。

自 20 世纪 60 年代以来,随着信息技术的飞速发展,数字信号处理技术应运而生,并且得到迅速发展。20 世纪 80 年代以来,由于方法的限制,数字信号处理技术还处于理论研究阶段,没有得到广泛的应用。在此阶段,人们利用通用计算机进行数字滤波、频谱分析等算法的研究和数字信号处理系统的模拟和仿真。20 世纪 80 年代初,世界上第一片可编程 DSP 芯片的诞生为数字信号处理理论的实际应用开辟了道路,而低成本数字信号处理器的不断推出,更加促进了这一进程。

DSP 作为一种微处理器,其设计的出发点和通用 CPU 以及 MCU 等处理器是不同的。DSP 是为完成实时数字信号处理任务而设计的,如何实现算法的高效是 DSP 器件的设计核心。DSP 在体系结构设计方面的很多考虑都可以追溯到算法自身的特点。它的主要特点是:

(1) 哈佛结构 (Harvard Structure)。早期的微处理器内部大多采用冯·诺依曼结构,其片内程序空间和数据空间是合在一起的,取指令和取操作数是通过一条总线分时进行的。而 DSP 芯片内采用的是程序空间和数据空间分开的哈佛结构,允许同时取指令 (来自程序存储器) 和取操作数 (来自数据存储器)。

(2) 多总线结构。许多 DSP 芯片内部都采用多总线结构,这样可以保证在一个机器周期内可以多次访问程序空间和数据空间。

(3) 流水线结构。在 DSP 中,采用流水线结构,在程序运行过程中的取指、译码、取操作数和执行是重叠运行的。

(4) 多处理单元。DSP 内部一般都包括多个处理单元,如 ALU、ARAU、ACC 及 MUL 等。

(5) 特殊的 DSP 指令。为了更好地满足数字信号处理应用的需要,在 DSP 的指令系统中,设计了一些特殊的 DSP 指令。

(6) 指令周期短。早期的 DSP 的指令周期约为 400ns,其运算速度为 5MIPS (每秒执行 5 百万条指令),随着集成电路的发展,其运行速度越来越快,TMSC320C55xx 的运行速度可达 200MIPS,TMS320C6203 的运行速度达到 2400MIPS。

(7) 运算精度高。早期 DSP 的字长为 8 位,后来逐步提高到 16 位、24 位和 32 位。

(8) 硬件配置强。新一代 DSP 的接口功能越来越强,片内具有串行口、主机接口 (HPI)、DMA 控制器、锁相环时钟产生器等,更易于完成系统设计。

【问】相比传统的 MCU, DSP 有哪些优点?

【答】(1) DSP 的速度比 MCU 快, 主频较高。

(2) DSP 适合于数据处理, 数据处理的指令效率较高。

(3) DSP 均为 16 位以上的处理器, 不适合于低档的场合。

(4) DSP 可以同时处理的事件较多, 系统级成本有可能较低。

(5) DSP 的灵活性较好, 大多数算法都可以通过软件实现。

(6) DSP 的集成度较高, 可靠性较好。

(7) DSP 有乘法器, 且乘法器是独立的, 可以和加法器并行工作, 提示了 CPU 的处理速率。

(8) DSP 有片内 PLL, 使片内高频、片外高频工作, 有利于系统的稳定。

1.3 DSP 的系统设计过程

DSP 系统的设计过程主要包括确定系统的性能指标、核心算法的模拟和验证、依据要达到的性能指标进行 DSP 芯片的选型、系统的硬件设计和软件设计以及系统的集成和测试。图 1.2 所示是 DSP 系统设计的流程图。

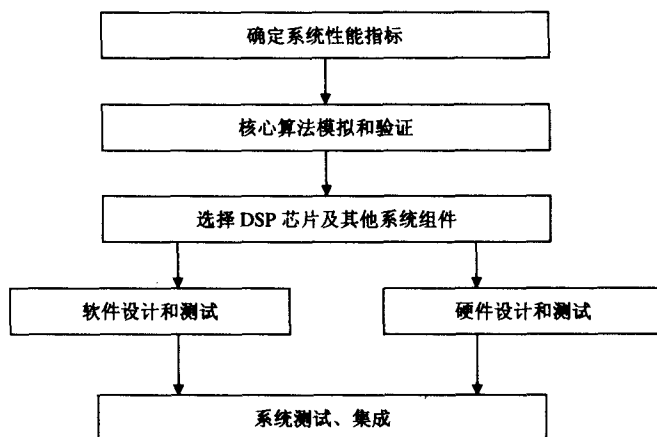


图 1.2 DSP 系统设计的流程图

在设计 DSP 系统之前, 首先必须根据应用系统的目标确定系统的性能指标及信号处理的要求, 通常可用数据流程图、数学运算序列、正式的符号或自然语言来描述。

第二步是根据系统的要求进行高级语言的模拟。一般来说, 为了实现系统的最终目标, 需要对输入的信号进行适当的处理, 而处理方法的不同会导致不同的系统性能, 要得到最佳的系统性能, 就必须在这一步确定最佳的处理方法, 即数字信号处理的算法 (Algorithm), 因此这一步也称算法模拟阶段。例如, 语音压缩编码算法就是要在确定的压缩比条件下, 获得最佳的合成语音。算法模拟所用的输入数据是实际信号经采集而获得的, 通常以计算机文件的形式存储为数据文件。如, 语音压缩编码算法模拟时所用的语音信号就是实际采集而获得并存储为

计算机文件形式的语音数据文件。有些算法模拟时所用的输入数据并不一定要是实际采集的信号数据,只要能够验证算法的可行性,输入假设的数据也是可以的。

在完成第二步之后,接下来就可以设计实时 DSP 系统,实时 DSP 系统的设计包括硬件设计和软件设计两个方面。硬件设计首先要根据系统运算量的大小、对运算精度的要求、系统成本的限制以及体积、功耗等要求选择合适的 DSP 芯片,然后设计 DSP 芯片的外围电路及其他电路。软件设计和编程主要根据系统要求和所选的 DSP 芯片编写相应的 DSP 汇编程序,若系统运算量不大且有高级语言编译器支持,也可用高级语言(如 C 语言)编程。由于现有的高级语言编译器的效率还比不上手工编写汇编语言的效率,因此在实际应用系统中常常采用高级语言和汇编语言的混合编程方法,即在算法运算量大的地方,用手工编写的方法编写汇编语言,而运算量不大的地方则采用高级语言。采用这种方法,既可缩短软件开发的周期,提高程序的可读性和可移植性,又能满足系统实时运算的要求。

DSP 硬件和软件设计完成后,就需要进行硬件和软件的调试。软件的调试一般借助于 DSP 开发工具,如软件模拟器、DSP 开发系统或仿真器等。调试 DSP 算法时一般采用比较实时结果与模拟结果的方法,如果实时程序和模拟程序的输入相同,则两者的输出应该一致。应用系统的其他软件可以根据实际情况进行调试。硬件调试一般采用硬件仿真器进行调试,如果没有相应的硬件仿真器,且硬件系统不是十分复杂,也可以借助于一般的工具进行调试。

系统的软件和硬件分别调试完成后,就可以将软件脱离开发系统而直接应用系统上运行。当然,DSP 系统的开发,特别是软件开发是一个需要反复进行的过程,虽然通过算法模拟基本上可以知道实时系统的性能,但实际上模拟环境不可能做到与实时系统环境完全一致,而且将模拟算法移植到实时系统时必须考虑算法是否能够实时运行的问题。如果算法运算量太大不能在硬件上实时运行,则必须重新修改或简化算法。

1.4 DSP 处 理 器

为描述方便,本书后面所述的 DSP 均指数字信号处理器(Digital Signal Processor)。

1.4.1 DSP 概述

世界上第一个单片 DSP 芯片是 1978 年 AMI 公司宣布的 S2811,1979 年美国 Intel 公司发布的商用可编程器件 2920 是 DSP 芯片的一个主要里程碑。这两种芯片内部都没有现代 DSP 芯片所必须的单周期芯片。1980 年,日本 NEC 公司推出的 μ PD7720 是第一个具有乘法器的商用 DSP 芯片。第一个采用 CMOS 工艺生产浮点 DSP 芯片的是日本的 Hitachi 公司,它于 1982 年推出了浮点 DSP 芯片。1983 年,日本的 Fujitsu 公司推出的 MB8764,其指令周期为 120ns,且具有双内部总线,从而处理的吞吐量发生了一个大的飞跃。而第一个高性能的浮点 DSP 芯片应是 AT&T 公司于 1984 年推出的 DSP32。

在这么多的 DSP 芯片种类中,最成功的当数美国 TI(Texas Instruments)公司的一系列产品。TI 公司在 1982 年成功推出启迪一代 DSP 芯片 TMS32010 及其系列产品 TMS32011、TMS32C10/C14/C15/C16/C17 等,之后相继推出了第二代 DSP 芯片 TMS32020、TMS320C25/C26/C28;第三代 DSP 芯片 TMS32C30/C31/C32;第四代 DSP 芯片 TMS32C40/C44;第五代 DSP 芯片 TMS32C50/C51/C52/C53 和 TMS32C5000 系列,以及集多个 DSP 于一体的高

性能 DSP 芯片 TMS32C80/C82 和 OMAP, 还有目前方兴未艾的 Davinci 系列芯片等。

【问】多媒体通信领域为什么要用 DSP?

【答】3G 技术和 Internet 技术的发展, 要求处理器的速度越来越高, 体积越来越小, DSP 的发展正好能满足这一发展的要求。因为, 传统的其他处理器都有不同的缺陷: MCU 的速度较慢; CPU 体积较大, 功耗较高; 嵌入 CPU 的成本较高。DSP 的发展, 使得在许多速度要求较高, 算法较复杂的场合取代 MCU 或其他处理器, 而成本也可能会更低。

自 1980 年以来, DSP 芯片得到了突飞猛进的发展, DSP 芯片的应用越来越广泛。从运算速度来看, MAC(一次乘法和一次加法)时间已经从 20 世纪 80 年代初的 400ns(如 TMS32010)降低到 40ns(如 TMS32C40), 处理能力提高了 10 多倍。DSP 芯片内部关键的乘法器部件速度从 1980 年的 40ns 左右下降到 5ns 以下, 片内 RAM 增加一个数量级以上。从制造工艺来看, 1980 年采用 4 μ 的 N 沟道 MOS 工艺, 而现在则普遍采用亚微米 CMOS 工艺。DSP 芯片的引脚数量从 1980 年的最多 64 个增加到现在的 200 个以上, 引脚数量的增加, 意味着结构灵活性的增加。此外, DSP 芯片的发展, 使 DSP 系统的成本、体积、重量和功耗都有很大程度的下降。

下面简要介绍一下美国 TI 公司的一系列产品。

(1) TMS320C2000。TMS320C2000 系列包括 C24x 和 C28x 系列。C24x 系列建议使用 LF24xx 系列替代 C24x 系列, LF24xx 系列的价格比 C24x 便宜, 性能高于 C24x, 而且 LF24xxA 具有加密功能。C28x 系列主要用于大存储设备管理, 高性能的控制场合。

(2) TMS320C3x。TMS320C3x 系列包括 C3x 和 VC33, 主要推荐使用 VC33。C3x 系列是 TI 浮点 DSP 的基础, 不可能停产, 但价格不会进一步下调。

(3) TMS320C5x。TMS320C5x 系列已不推荐使用, 建议使用 C24x 或 C5000 系列替代。

(4) TMS320C5000。TMS320C5000 系列包括 C54x 和 C55xx 系列。其中 VC54xx 还不断有新的器件出现, 如 TMS320VC5471 (DSP+ARM7)。C55xx 系列是 TI 的第三代 DSP, 功耗为 VC54xx 的 1/6, 性能为 VC54xx 的 5 倍, 是一个正在发展的系列。C5000 系列是目前 TI DSP 的主流 DSP, 它涵盖了从低档到中高端的应用领域, 目前也是用户最多的系列。

(5) TMS320C6000。TMS320C6000 系列包括 C62xx、C67xx 和 C64xx。此系列是 TI 的高档 DSP 系列。其中 C62xx 系列是定点的 DSP, 系列芯片种类较丰富, 是主要的应用系列。C67xx 系列是浮点的 DSP, 用于需要高速浮点处理的领域。C64xx 系列是新发展, 性能是 C62xx 的 10 倍。

(6) OMAP 系列。该系列是 TI 专门用于多媒体领域的芯片, 它是 C55xx+ARM9, 性能卓越, 非常适合于手持设备、Internet 终端等多媒体应用。

(7) DM 系列。该系列的最新处理器是功能强大的 DM6446, 其采用了 TI 的达芬奇 (DaVinci) 技术。DM6446 集成了时钟频率达 594MHz 的 C64x+DSP 内核与 ARM926EJ-S 内核。新一代 c64x+DSP 是 TMS320C6000 平台中性能最高的定点 DSP, 并建立在 TI 开发的第二代高性能高级 VLIW 架构的增强版之上。C64x+与前代 C6000 DSP 平台代码兼容。专门针对高端视频系统的需求而设计。

TI 的 DSP 最高主频可以从芯片的型号中获得, 但每一个系列不一定相同。

1) TMS320C2000 系列:

- TMS320F206 最高主频为 20MHz。
- TMS320C203/C206 最高主频为 40MHz。
- TMS320F24x 最高主频为 20MHz。
- TMS320LF24xx 最高主频为 30MHz。
- TMS320LF24xxA 最高主频为 40MHz。
- TMS320LF28xx 最高主频为 150MHz。

2) TMS320C3x 系列:

- TMS320C30 最高主频为 25MHz。
- TMS320C31PQL80 最高主频为 40MHz。
- TMS320C32PCM60 最高主频为 30MHz。
- TMS320VC33PGE150 最高主频为 75MHz。

3) TMS320C5000 系列:

- TMS320VC54xx 最高主频为 160MHz。
- TMS320VC55xx 最高主频为 300MHz。

4) TMS320C6000 系列:

- TMS320C62xx 最高主频为 300MHz。
- TMS320C67xx 最高主频为 230MHz。
- TMS320C64xx 最高主频为 720MHz。

随着技术的进步,会有越来越多更高性能的 DSP 出现。

1.4.2 DSP 芯片的选型

设计 DSP 应用系统中,选择 DSP 芯片是非常重要的一个环节。只有选定了 DSP 芯片,才能进一步设计其外围电路及系统的其他电路。总的来说,DSP 芯片的选型应根据实际的应用系统需要而确定。不同的 DSP 应用系统由于应用场合、应用目的等不尽相同,对 DSP 芯片的选择也是不同的。一般来说,选择 DSP 芯片时应考虑到如下诸多因素:

(1) 速度。DSP 速度一般用 MIPS 或 FLOPS 表示,即百万次/秒。根据您对处理速度的要求选择适合的器件。一般选择处理速度不要过高,速度高的 DSP,系统实现也较困难。

(2) 精度。DSP 芯片分为定点、浮点处理器,对于运算精度要求很高的处理,可选择浮点处理器。定点处理器也可完成浮点运算,但精度和速度会有影响。

(3) 寻址空间。不同系列 DSP 程序、数据、I/O 空间大小不一,与普通 MCU 不同,DSP 在一个指令周期内能完成多个操作,所以 DSP 的指令效率很高,程序空间一般不会有问题,关键是数据空间是否满足。数据空间的大小可以通过 DMA 的帮助,借助程序空间扩大。

(4) 成本。一般定点 DSP 的成本会比浮点 DSP 的要低,速度也较快。要获得低成本的 DSP 系统,尽量用定点算法,用定点 DSP。

(5) 实现方便。浮点 DSP 的结构实现 DSP 系统较容易,不用考虑寻址空间的问题,指令对 C 语言支持的效率也较高。

(6) 内部部件。根据应用要求,选择具有特殊部件的 DSP,如,C2000 适合于电机控制、OMAP 适合于多媒体等。

(7) 其他。除了上述因素外,选择 DSP 芯片还应考虑到封装的形式、质量标准、供货情况、生命周期等。有的 DSP 芯片可能有 DIP、PGA、PLCC、PQFP 等多种封装形式。有些 DSP 系统可能最终要求的是工业级或军用级标准,在选择时就需要注意到所选的芯片是否有工业级或军用级的同类产品。如果所设计的 DSP 系统不仅仅是一个实验系统,而是需要批量生产并可能有几年甚至十几年的生命周期,那么需要考虑所选的 DSP 芯片供货情况如何,是否也有同样甚至更长的生命周期等。

【问】如要设计生物图像处理系统,选用哪种型号较好(高性能和低成本)?

【答】可采用 C54x 或 C55xx 平台,如果需要更高性能,就可以采用 C6x 系列。

1.5 DSP 芯片的应用领域

自从 20 世纪 70 年代末 DSP 芯片诞生以来, DSP 芯片得到了飞速的发展。DSP 芯片的高速发展,一方面得益于集成电路技术的发展,另一方面也得益于巨大的市场。目前, DSP 芯片的价格越来越低,性能价格比日益提高,具有巨大的应用潜力。DSP 芯片的应用主要有:

- (1) 信号处理。如数字滤波、自适应滤波、快速傅立叶变换、相关运算、谱分析、卷积、模式匹配、加窗、波形产生等。
 - (2) 通信。如调制解调器、自适应均衡、数据加密、数据压缩、回波抵消、多路复用、传真、扩频通信、纠错编码、可视电话等。
 - (3) 语音。如语音编码、语音合成、语音识别、语音增强、说话人辨认、说话人确认、语音邮件、语音存储等。
 - (4) 图形/图像。如二维和三维图形处理、图像压缩与传输、图像增强、动画、机器人视觉等。
 - (5) 军事。如保密通信、雷达处理、声纳处理、导航、导弹制导等。
 - (6) 仪器仪表。如频谱分析、函数发生、锁相环、地震处理等。
 - (7) 自动控制。如引擎控制、声控、自动驾驶、机器人控制、磁盘控制等。
 - (8) 医疗。如助听、超声设备、诊断工具、病人监护等。
 - (9) 家用电器。如高保真音响、音乐合成、音调控制、玩具与游戏、数字电话/电视等。
- 随着 DSP 芯片性能价格比的不断提高,可以预见 DSP 芯片将会在更多的领域内得到更为广泛的应用。

1.6 TI 的 C5000 系列中 C54x 和 C55xx 的比较

1. 结构特点对比

- (1) 均为 16bit 定点 DSP。
- (2) C55xx 有双 MAC 单元; C54x 有单 MAC 单元。
- (3) C55xx 的指令长度可变,且没有排队的限制; C54x 的指令长度固定。
- (4) C55xx 有 12 组总线; C54x 有 8 组总线。

(5) C55xx 提供了 EMIF 外部存储器扩展接口, 可以直接使用 SDRAM, 而 C54xx 则不能直接使用。

2. 内部结构对比

C5000 是 16bit 定点 DSP 系列, 包括旧有的 C5x、当前主流的 C54x 和最新的 C55xx。C55xx 和 C54x 源代码兼容, 而 C5x 和 C2x 源代码兼容。C54x 关注于低功耗, 而 C55xx 则将低功耗提到一个新水平: 300MHz 的 C55xx 和 120MHz 的 C54x 相比, 性能提高 5 倍, 而功耗则降到 1/6。C5x 已经停产, TI 公司已经将新设计转向 C54x 和 C55xx。C54x 和 C55xx 采用改进的哈佛结构。

C55xx 具有 12 组独立的总线, 而 C54x 仅有 8 组。它们都有一组程序总线和相应的程序地址总线。C54x 总线的宽度为 16bit, 而 C55xx 总线的宽度为 32bit。C55xx 有三组数据读总线和两组数据写总线, 而 C54x 有两组数据读总线和一组数据写总线。每组数据总线都有其相应的地址总线。C55xx 的数据地址总线的宽度为 24bit, 而 C54x 的数据地址总线的宽度为 16bit。

C54x 使用两个辅助寄存器算术单元, 在每个周期内产生一个或两个数据存储地址。C55xx 的地址-数据流单元 (ADFU) 包含了专门的硬件来管理五组数据总线。该 ADFU 也可以作为通用的 16bit ALU, 用于简单的算术运算。该 ALU 从指令缓冲单元 (IU) 接收立即数, 和存储器、ADFU 寄存器、数据计算单元 (DCU) 寄存器、程序流单元 (PFU) 寄存器作双向通信。无论是 ALU, 还是 3 个地址寄存器 ALU (ARAU) 中的一个, 都可以修改作间接寻址的 9 个地址寄存器。这 3 个 ARAU 为 C55xx 的三组数据读总线提供独立的地址。这种并行性保证了在每个 CPU 周期内 DCU 去读两个 16bit 的操作数和一个 16bit 的系数。

C55xx 的 DCU 包含了两个 MAC 单元, 在单周期内作两个 17×17 bit 的 MAC 运算。它还包含了一个 40bit 的 ALU 和 4 个 40bit 的累加器寄存器、一个桶型移位器以及专门的 Viterbi 算法硬件。每个 MAC 单元包含一个乘法器和带 32bit 或 40bit 饱和逻辑的加法器。3 个数据读总线将两个数据流和一个公共系数流送给两个 MAC 单元。用户可以用 ALU 作 32bit 的运算, 或分开作两个 16bit 的运算。除开接受从 DCU 的 40bit ACC 寄存器来的输入外, ALU 还从 IU 接受立即数, 并和存储器、ADFU 寄存器、PFU 寄存器作双向通信。

C54x 有一个单 17×17 bit 的 MAC 乘法器、一个 40bit 的加法器、两个 40bit 的 ACC 和一个分开的 40bit 的 ALU。与 C55xx 相类似, C54x 的 ALU 也可以做成两个 16bit 的配置, 完成两个单周期运算。乘法器输出处的 40bit 的加法器允许作非流水的 MAC 运算, 以及并行的两个加法和乘法。单周期归一化和指数编码支持浮点数运算。

两个系列的结构都支持一个桶型移位器, 可将 40bit 的 ACC 的值左移或右移最多达 31bit。该桶型移位器将移位后的值送给 DCU 的 ALU, 以便作进一步的运算。指令集中关于二操作数、三操作数和 32bit 操作数的指令, 支持结构的并行性。8 个可以独立寻址的辅助寄存器和软件堆栈提高了 C 编译器的效率。

C55xx 可以执行可变长度的指令, 这和 C54x 有显著的不同。C54x 的指令长度为固定的 16bit, 而 C55xx 的指令长度则为 8~48 bit。C55xx 的 IU 缓存 64 byte 的代码, 且有一个解码逻辑来确认可变长度指令中各指令的区别。局部循环指令使用指令缓冲队列来循环执行代码块。指令缓冲队列还可以在条件程序流控制指令的条件测试时, 推测性地提取指令。指令解码器按排列顺序对指令解码, 而不是执行动态时序, 从而可以在预定的时间得到结果。

C55xx 的 PFU 跟踪程序的执行点, 并为多达 16Mbyte 的程序存储器产生 24bit 的地址。

(2) k23 绝对寻址。使用 k23 绝对寻址的指令，其操作数为* (#k23)，其中 k23 是一个 23 位的无符号常数。如图 2.27 所示，使用这种寻址方式的指令将常数编码为 3 字节（去掉最高位），与 k16 绝对寻址一样，使用该模式寻址的指令不能与其他指令并行执行。

k23	数据空间
000 0000 0000 0000 0000 0000 ⋮ 000 0000 1111 1111 1111 1111	第 0 主数据页： 00 0000h~00 FFFFh
000 0001 0000 0000 0000 0000 ⋮ 000 0001 1111 1111 1111 1111	第 1 主数据页： 01 0000h~01 FFFFh
000 0010 0000 0000 0000 0000 ⋮ 000 0010 1111 1111 1111 1111	第 2 主数据页： 02 0000h~02 FFFFh
⋮	⋮
111 1111 1111 0000 0000 0000 ⋮ 111 1111 1111 1111 1111 1111	第 127 主数据页： 7F 0000h~00 FFFFh

图 2.27 k23 绝对寻址

(3) I/O 绝对寻址。对于 I/O 绝对寻址模式，如果使用代数指令，其操作数是*port (#k16)，其中 k16 是一个 16 位的无符号常数；如果使用助记符指令，其操作数是 port (#k16)（操作数前没有*）。如图 2.28 所示，使用该模式的指令将常数编码为两个字节。同样，该指令不能与其他指令并行执行。

k16	I/O 空间
0000 0000 000 0000 ⋮ 1111 1111 1111 1111	0000h~FFFFh

图 2.28 I/O 绝对寻址

2. 直接寻址模式

直接寻址有以下几种方式：数据页指针（DP）直接寻址、堆栈指针（SP）直接寻址、寄存器位直接寻址和外设数据页指针（PDP）直接寻址。其中，DP 直接寻址和 SP 直接寻址与状态寄存器 ST1-55 的 CPL 位有关：

- 当 CPL=0 时，采用 DP 直接寻址模式。
- 当 CPL=1 时，采用 SP 直接寻址模式。

(1) DP 直接寻址。在 DP 直接寻址方法中，23 地址的高 7 位由 DPH 提供，用来确定主数据页，其余 16 位由两部分组成：

1) DP。DP 确定在主数据页内长度为 128 字节的局部数据页的起始地址，该起始地址可以是主数据页内的任何地址。

2) 由编译器计算出的 7 位偏移量（Doffset）：偏移量的计算与访问的是数据空间还是存储器映射寄存器（限定词是 mmap()）有关。

由 DPH 和 DP 构成扩展数据页寄存器 XDP，可以将 DPH 和 DP 分别载入，也可以用一条

指令载入 XDP。

(2) SP 直接寻址。当一条指令采用 SP 直接寻址模式, 其 23 位地址的高 7 位由 SPH 确定, 其余 16 位由 SP 和 7 位偏移量决定, 偏移量的范围是 0~127。由 SPH 和 SP 构成扩展数据堆栈指针 XSP。

由于在第 0 主数据页, 地址 00 0000h~00 005Fh 为存储映射寄存器保留, 所以如果数据栈位于该主数据页, 则可用的地址范围是 00 0060h~00 00FFFh。

(3) 寄存器位寻址。使用寄存器位寻址的指令, 其操作数是 @bitoffset。该操作数是从寄存器的最低位开始的偏移值。例如, 如果 bitoffset 为 0, 那么可以访问寄存器的最低位; 如果 bitoffset 为 3, 那么可以访问寄存器的低 3 位。

只有寄存器的位测试、置位、清零、取反指令支持这种寻址模式。

(4) PDP 直接寻址。当一条指令使用 PDP 间接寻址模式时, 64KB 的 I/O 空间分为 512 个外设数据页, 用 9 位的外设数据页指针 PDP 表示, 其中每一页有 128 个字, 由指令中指定的 7 位偏移值来表示, 如图 2.29 所示。

PDP	Poffset	数据空间
0000 0000 0	000 0000 ⋮ 111 1111	第 0 主数据页: 00 0000h~00 FFFFh
0000 0000 1	000 0000 ⋮ 111 1111	第 1 主数据页: 01 0000h~01 FFFFh
0000 0010 0	000 0000 ⋮ 111 1111	第 2 主数据页: 02 0000h~02 FFFFh
⋮	⋮	⋮
1111 1111 1	000 0000 ⋮ 111 1111	第 127 主数据页: 7F 0000h~7F FFFFh

图 2.29 PDP 直接寻址

3. 间接寻址模式

CPU 支持的间接寻址模式有 AR 间接寻址、双 AR 间接寻址、CDP 间接寻址和系数间接寻址。利用这些模式可以进行线性或循环寻址。

(1) AR 间接寻址模式。AR 间接寻址模式通过一个辅助寄存器 AR_n (n=0, 1, 2, 3, 4, 5, 6 或 7) 访问数据空间。而 ST2-55 的 ARMS 位决定了 AR 间接寻址模式的操作类型。

1) ARMS=0, DSP 模式: CPU 提供 DSP 增强型应用的高效执行功能。

2) ARMS=1, 控制模式: 针对控制系统的应用, CPU 能够优化代码的长度。

(2) 双 AR 间接寻址模式。双 AR 间接寻址模式可以通过 8 个辅助寄存器 (AR0~AR7) 同时访问两个数据存储单元, 与单个 AR 间接访问数据空间一样, CPU 使用一个扩展辅助寄存器产生 23 位地址。双 AR 间接寻址可以实现以下功能:

1) 执行一条可完成两个 16 位数据空间访问的指令。在这种情况下, 两个数据存储操作在指令中为 Xmem 和 Ymem。例如:

```
ADD Xmem, Ymem, ACx
```

2) 并行执行两条指令。在这种情况下, 必须每条指令访问一个存储数据, 操作数在指令中是 Smem 或 Lmem。

```
MOV Smem, dst
||AND Smem, src, dst
```

双 AR 间接寻址模式是 AR 间接寻址模式操作数的子集, 而 ARMS 状态位不影响双 AR 间接寻址的操作。

(3) CDP 间接寻址模式。CDP 间接寻址模式使用系数数据指针 (CDP) 对数据空间、寄存器位和 I/O 空间进行访问。

(4) 系数间接寻址模式。系数间接寻址模式的地址产生过程与使用 CDP 间接寻址数据空间的地址产生过程一样。CDP 间接寻址模式支持以下算术指令:

- 1) FIR 滤波。
- 2) 乘法。
- 3) 乘加。
- 4) 乘减。
- 5) 双乘加或双乘减。

2.4 流水线操作

与哈佛结构相关, DSP 芯片广泛采用 2~6 级流水线, 以减少指令执行时间, 如 C54xx 采用 4 级流水, C55xx 采用 6 级流水。因为计算机在执行一条指令时, 总要经过取指令、译码、访问操作数、执行等几个步骤, 需要若干个机器周期才能完成。DSP 的流水线结构是指它的这几个阶段在执行过程中是重叠的, 如图 2.30 所示。流水线大大降低了指令的平均执行时间。8031 单片机至少 12 个机器周期才能完成一条指令操作, 而利用流水线机制的 DSP 可以在一个机器周期内完成一条单周期指令操作。

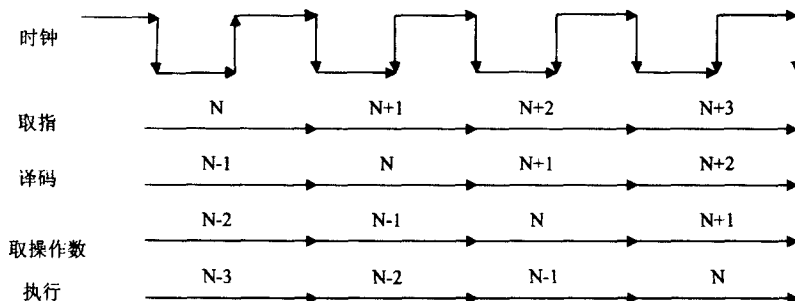


图 2.30 4 级流水线操作

在 4 级流水线中, 任意给定的周期内可能有 1~4 条不同的指令是激活的, 每一条指令都处于不同的阶段。也就是说在执行本条指令时, 下面的 3 条指令已依次完成了取操作数、译码、取指令的操作。尽管每一条指令的执行时间仍然是几个机器周期, 但由于指令的流水作业, 使得每条指令基本都是单周期指令。