

高等院校规划教材

数字信号处理技术 及其应用

刘丽钧 周英钢 刘本伟 编著

赠

电子教案



机械工业出版社
CHINA MACHINE PRESS

高等院校规划教材

数字信号处理技术及其应用

刘丽钧 周英钢 刘本伟 编著



机械工业出版社

本书以 TMS320C54x 系列 DSP 为描述对象, 系统全面地介绍了 TMS320C54x 的硬件结构、工作原理、指令系统、应用设计和开发技术。本书的特点是围绕 TMS320C54x 结合一些应用实例详细地介绍硬件实现方法和软件设计技巧。

本书可以作为高等院校及函授学院电子信息工程、通信工程等专业的教材或参考书, 也可供从事 DSP 应用与开发的技术人员参考。

本书配有免费电子教案, 可以辅助授课教师教学。如有需要, 请与北京机械工业出版社编辑联系 (电子邮箱: kongxijun@163.com)。

图书在版编目 (CIP) 数据

数字信号处理技术及其应用/刘丽钧等编著. —北京: 机械工业出版社, 2004.12

高等院校规划教材

ISBN 7-111-16016-9

I. 数… II. 刘… III. 数字信号—信号处理—高等学校: 技术学校—教材 IV. TN911.72

中国版本图书馆 CIP 数据核字 (2005) 第 003299 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 王世刚 孔熹峻

责任编辑: 孔熹峻 责任印制: 李 妍

北京蓝海印刷有限公司印刷·新华书店北京发行所发行

2004 年 12 月第 1 版第 1 次印刷

787mm×1092mm 1/16·18.5 印张·459 千字

定价: 26.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话 (010) 68993821、88379646

68326294、68320718

封面无防伪标均为盗版

前 言

信息技术的发展日新月异，数字化的浪潮正在迅速地席卷全球。数字信号处理技术作为数字化最重要的技术之一，在应用的深度和广度方面，正以前所未有的速度向前发展。

数字信号处理器（DSP）具有独特的硬件结构和快速实现各种数字信号处理算法的突出优点，在通信、雷达、声纳、语音合成和识别、图像处理、高速控制、家用电器等众多领域得到了广泛的应用。使用 DSP 处理器的产品已大批投放市场，随着微电子技术的不断发展，DSP 芯片的性能价格比将不断提高，可以预计 DSP 芯片将渗透到各个领域，得到更加广泛的应用。

当前市场上使用较多的 DSP 处理器是美国德州仪器（TI）公司的产品，该产品占到全球市场份额的 60% 左右。TI 公司的 DSP 主流产品包括 TMS320C2000 系列（其中有 TMS320C2x/C2xx）、TMS320C5000 系列（包括 TMS320C5x/C54x/C55x）、TMS320C6000 系列（包括 TMS320C62x/67x）。

本书主要介绍 TMS320C54x 的原理及应用，它采用改进的哈佛结构，8 条总线，高速并行的算术逻辑单元，片内存储器，片内外设和专业化的指令集，使该芯片速度高，操作灵活，应用广泛。

全书共分 7 章。第 1 章综述了 DSP 芯片的特点、应用及发展趋势；第 2 章详细系统地介绍了 TMS320C54x 的内部结构和工作原理；第 3 章剖析了 TMS320C54x 的指令系统；第 4 章介绍了汇编语言的开发工具；第 5 章讨论了汇编语言编程方法和技巧；第 6 章从应用的角度介绍了一些实际系统设计实例，包括常用的硬件设计方法和软件设计程序；第 7 章介绍了 CCS 集成开发环境的使用。

本书可以作为高等院校及函授学院电子信息工程、通信工程等专业的教材或参考书，也可供从事 DSP 应用与开发的技术人员参考。

本书第 1~3 章、第 5 章由刘丽钧执笔，第 4 章由刘本伟执笔，第 6 章由刘丽钧和周英钢执笔，第 7 章由周英钢执笔。全书由刘丽钧统稿。

由于作者水平有限，书中难免有不妥之处，恳请读者批评指正。

编者

2004 年 10 月

目 录

前言

第 1 章 绪论.....1

- 1.1 数字信号处理概述.....1
- 1.2 DSP 芯片及特点.....2
- 1.3 DSP 芯片的历史、现状、发展方
向及应用.....4

第 2 章 TMS320C54x 的结构原理.....6

- 2.1 TMS320C54x 的主要特点及组成框图.....6
 - 2.1.1 TMS320C54x 的主要特点.....6
 - 2.1.2 TMS320C54x 的组成框图.....8
- 2.2 总线结构.....9
- 2.3 内部存储空间.....10
 - 2.3.1 存储空间.....10
 - 2.3.2 程序存储空间.....13
 - 2.3.3 数据存储空间.....17
 - 2.3.4 I/O 空间.....20
- 2.4 中央处理单元 (CPU).....20
 - 2.4.1 算术逻辑运算单元 (ALU).....20
 - 2.4.2 累加器 A 和 B.....22
 - 2.4.3 桶形移位寄存器.....24
 - 2.4.4 乘法器/加法器单元.....25
 - 2.4.5 比较、选择和存储单元.....26
 - 2.4.6 指数编码器.....27
 - 2.4.7 CPU 状态和控制寄存器.....28
- 2.5 数据寻址方式.....30
 - 2.5.1 立即寻址.....31
 - 2.5.2 绝对寻址.....32
 - 2.5.3 累加器寻址.....33
 - 2.5.4 直接寻址.....33
 - 2.5.5 间接寻址.....34
 - 2.5.6 存储器映射寄存器寻址.....38
 - 2.5.7 堆栈寻址.....39
 - 2.5.8 数据类型.....39
- 2.6 程序存储器地址生成方式.....40
 - 2.6.1 程序计数器.....40

- 2.6.2 分支转移.....41
- 2.6.3 调用与返回.....41
- 2.6.4 条件操作.....42
- 2.6.5 重复操作.....43
- 2.6.6 复位操作.....44
- 2.6.7 中断.....45
- 2.6.8 省电方式.....50
- 2.7 流水线.....50
 - 2.7.1 流水线操作.....50
 - 2.7.2 延迟分支转移与流水线.....52
 - 2.7.3 条件执行与流水线.....53
 - 2.7.4 双寻址存储器与流水线.....54
 - 2.7.5 单寻址存储器与流水线.....56
 - 2.7.6 流水线的等待周期.....56
- 2.8 片内外围电路.....60
 - 2.8.1 电路映射寄存器.....60
 - 2.8.2 定时器.....61
 - 2.8.3 时钟发生器.....63
 - 2.8.4 主机接口.....67
- 2.9 串行口.....71
 - 2.9.1 串行口概述.....71
 - 2.9.2 串行口的组成框图.....72
 - 2.9.3 串行口寄存器.....74
- 2.10 外部总线.....76
 - 2.10.1 外部总线接口.....76
 - 2.10.2 外部总线的优先级.....77
 - 2.10.3 外部总线控制.....78
 - 2.10.4 分区转换逻辑.....80
 - 2.10.5 外部总线接口定时.....81
 - 2.10.6 复位和 IDLE3 省电工作方式.....84
 - 2.10.7 保持方式.....86
- 2.11 TMS320C54x 引脚信号说明.....89

第 3 章 汇编语言指令系统.....94

- 3.1 指令系统中的常用符号.....94
 - 3.1.1 指令系统中的缩写字符.....94

3.1.2 指令系统中的记号和运算符	96	5.6 双操作数乘法	222
3.2 指令系统	97	5.7 长字运算和并行运算	223
3.2.1 算术运算指令	97	5.8 小数运算	225
3.2.2 逻辑运算指令	101	5.9 除法运算	227
3.2.3 程序控制指令	102	5.10 浮点运算	228
3.2.4 加载和存储指令	105	第 6 章 TMS320C54X 应用系统设计	232
3.2.5 重复执行单条指令	108	6.1 复位电路	232
3.3 指令介绍及举例	109	6.2 时钟电路	233
第 4 章 汇编语言的开发工具	197	6.3 外部存储器与并行 I/O 接口电路	233
4.1 TMS320C54x 的常用软件工具	197	6.3.1 程序存储器扩展应用	234
4.1.1 常用软件工具及应用软件 的开发流程	197	6.3.2 静态数据存储器扩展	235
4.1.2 常用开发工具的功能	197	6.3.3 I/O 扩展应用	236
4.1.3 常用的调试方法	199	6.4 串行 I/O 接口电路	242
4.2 源程序的结构	199	6.5 TMS320C54x DSP 的以太网 接口设计	244
4.2.1 指令行的格式	200	6.5.1 RTL8019AS 介绍	244
4.2.2 常用汇编命令以及数据类型	200	6.5.2 DSP 与 RTL8019AS 的接口	247
4.3 源程序的编辑、汇编和链接过程	201	6.6 TMS320C54x 系列 DSP 与 PC 间串 行通信的实现	248
4.4 COFF 文件的格式	202	6.6.1 扩展方案	248
4.4.1 COFF 文件中的段	203	6.6.2 SPI 接口协议及 DSP 的 多通道缓冲串行接口	249
4.4.2 汇编器对段的处理	204	6.6.3 MAX3111 通用异步收发器	250
4.4.3 链接器对段的处理	205	6.6.4 DSP 与 MAX3111 的接口设计	250
4.4.4 COFF 文件中的符号	206	6.6.5 DSP 的异步通信软件的设计	251
4.5 汇编	207	6.7 中断系统应用	253
4.5.1 汇编过程	207	6.8 定时器/计数器应用	257
4.5.2 列表文件	208	6.8.1 方波发生器	257
4.5.3 交叉引用列表	209	6.8.2 脉冲频率监测	260
4.5.4 宏定义和宏调用	209	6.8.3 周期信号检测	262
4.6 链接	210	第 7 章 CCS 的使用	265
4.6.1 运行链接程序	210	7.1 CCS 概述	265
4.6.2 链接器选项	211	7.1.1 CCS 组成	265
4.6.3 链接器命令文件	211	7.1.2 CCS 开发流程	266
4.6.4 多个文件的链接	216	7.2 CCS 软件的安装	266
第 5 章 汇编语言的程序设计	217	7.3 CCS 的设置	266
5.1 数据块传送	217	7.3.1 软件仿真的设置	266
5.2 加、减法和乘法运算	218	7.3.2 手动配置 CCS 运行环境	267
5.3 重复操作	218	7.3.3 Memory Map 的设置	267
5.4 程序的控制与转移	220		
5.5 堆栈的使用方法	221		

7.4 CCS 使用初步	268	7.5.4 图形显示	278
7.4.1 建立一个新的工程	268	7.5.5 运行程序和自动图形显示	279
7.4.2 为项目添加文件	269	7.5.6 调整算法因子	280
7.4.3 查看代码	270	7.5.7 观察不同作用域的变量	280
7.4.4 建立和运行程序	271	7.5.8 使用 GEL 文件	281
7.4.5 改变程序设置和修改语法错误	272	7.5.9 调整并剖析处理器的负载	281
7.4.6 使用断点和查看窗口	272	7.5.10 更多的尝试	282
7.4.7 观察结构变量	273	7.6 FIR 滤波器的基本原理和设计方法	282
7.4.8 剖析程序的运行状况	273	7.6.1 FIR 滤波器的特点	282
7.4.9 更多的尝试	274	7.6.2 FIR 滤波器的设计方法	283
7.5 CCS 高级功能的使用	274	7.6.3 FIR 滤波器的定点 DSP 实现	284
7.5.1 打开并检查工程项目	274	附录	287
7.5.2 浏览源程序代码	275	附录 A TMS320 系列 DSP 的命名方法	287
7.5.3 添加 Probe Point 断点用于 文件读/写	277	附录 B 等待周期表	288
		参考文献	290

第1章 绪 论

1.1 数字信号处理概述

信号是指某种变化的物理量，从中可以传输并提取所需的信息，因而信号是传输信息的载体。信号按时间的函数可分为时域连续信号和时域离散信号，我们通常所说的数字信号是指时域离散信号。DSP 的定义狭义的理解，英文是 Digital Signal Processor，译为数字信号处理器；广义的理解，英文是 Digital Signal Processing，译为数字信号处理技术；我们讨论的 DSP 是指广义的理解。

信号的数字处理，包括对信号的分析、传输和处理，具体地讲就是对信号进行采样、变换、滤波、估值、增强、压缩、识别等，是 20 世纪 60 年代前后发展起来的技术，现已广泛应用于各个学科，特别是一些新兴学科。随着计算机及大规模集成电路，特别是微处理器的迅猛发展，数字信号处理已成为目前发展最快的学科之一。数字信号处理就是指利用计算机或专用处理设备，以数字形式对信号进行采集、变换、滤波、估值、增强、压缩、识别等处理，以得到符合人们需求的信号形式。

数字信号处理是以众多学科为理论基础的，它所涉及的范围极其广泛，例如在数学领域、微积分、概率统计、随机过程、数值分析等都是数字信号处理的基础工具，与网络理论、信号与系统、控制论、通信理论、故障诊断等也密切相关。近来新兴的一些学科，如人工智能、模式识别、神经网络等，都与数字信号处理密不可分。可以说，数字信号处理是把许多经典的理论体系作为自己的理论基础，同时又使自己成为一系列新兴学科的理论基础。

数字信号处理技术之所以发展得这样快，应用得这样广，是与它的突出优点分不开的，它具有精度高、灵活性大、可靠性高、时分复用等特点。

数字信号处理的实现方法一般有以下几种：

- 1) 在通用的计算机（如 PC）上用软件（如 C 语言）实现。
- 2) 在通用的计算机系统中加上专用的加速处理机实现。
- 3) 用通用的单片机（如 MCS-51、96 系列等）实现，这种方法可用于一些不太复杂的现场控制等。
- 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。
- 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，例如专用于 FFT、数字滤波、卷积、相关等算法的 DSP 芯片，这种芯片将相应的信号处理算法在芯片内部用硬件实现，无需进行编程。

在上述几种方法中，第 1) 种方法的缺点是速度较慢，一般可用于 DSP 算法的模拟；第 2) 种和第 5) 种方法专用性强，应用受到很大的限制，第 2) 种方法也不便于系统的独立运行；第 3) 种方法只适用于实现简单的 DSP 算法；只有第 4) 种方法才使数字信号处理的应用打开了新的局面。

虽然数字信号处理的理论发展迅速,但在 20 世纪 80 年代以前,由于实现方法的限制,数字信号处理的理论还得不到广泛的应用,直到 20 世纪 80 年代以后第一片数字信号处理器芯片 (Digital Signal Processors, 简称 DSP) 问世后, DSP 就以数字器件特有的稳定性、可重复性、大规模集成,特别是可编程性和易于实现自适应处理特性,给数字信号处理 (DSP) 的发展带来了巨大机遇,并使得信号处理手段更灵活,功能更复杂,应用更广泛。而且随着 DSP 运算速度的提高,能够实时处理的信号带宽也大大增加,数字信号处理的研究重点也由最初的非实时应用转向了高速实时应用。

实时指的是系统必须在有限的时间内对外部输入信号完成指定的处理,即信号处理的速度必须大于等于输入信号更新的速度;而且从信号输入到处理后的信号输出的延迟必须足够小,如一个制导系统的输出延迟要求在几毫秒以内。

1.2 DSP 芯片及特点

第一代 DSP 以 AMD2900、NEC7720 和 TMS32010 为代表。其中 TI 公司的 TMS32010 第一次使用了哈佛总线结构和硬件乘法器。

由于开发工具的问题,最初的 DSP 开发非常困难,要设计并实现一个基于 DSP 的系统是一个专业性很强的工作。美国 TI 公司给 DSP 引入许多通用计算机微处理器特点,并为其产品开发了汇编语言和 C 语言代码产生工具以及各种软硬件调试工具,使得 DSP 的开发难度大大降低, DSP 在 20 世纪 80 年代末和 90 年代初进入了快速发展的时期。现在 TI 公司的 DSP 包括了定点、浮点、多处理器三个类型的产品,每个类型又有不同性能和价格的具体系列可以供用户选择。

DSP 按照所支持的数据类型不同分为定点产品和浮点产品两大类。定点 DSP 进行算术操作时,使用的是小数点位置固定的有符号数或无符号数。浮点 DSP 进行算术操作时,使用的是带有指数的小数,小数点的位置随着具体数据的不同进行浮动。

定点器件在硬件结构上比浮点器件简单,具有价格低、速度快的特点,因而用得最多;而浮点器件的优点是精度高,不需要进行定标和考虑有限字长效应,但是其成本、功耗相对较高,速度较慢,适合于对数据动态范围和精度要求高的特殊场合。

除了定点和浮点的划分外,各个 DSP 厂家还根据 DSP 的 CPU 结构和性能,把自己的产品划分了不同系列。例如, TI 公司的定点系列 DSP 有 C20x、C24x、C5x、C54x、C62xx; 浮点系列 DSP 有 C3x、C4x、C67xx。不同系列 DSP 的 CPU 结构不同,性能和价格也有很大的差异。

在同一系列的 DSP 产品中还划分有很多个具体型号。各个具体型号的 CPU 结构完全相同,差别之处只在于 DSP 片内存储器和外设接口的配置不同。例如, TI 公司的 C5x 系列中的 C50 和 C52,它们的 CPU 结构完全相同,不同之处在于 C50 的片内 ROM 为 2K 字,片内 RAM 为 10K 字,具有标准串口和 TDM 串口各一个;而 C52 的片内 ROM 为 4K 字,片内 RAM 为 1K 字,只有一个标准串口。同一系列不同型号产品具有丰富多样的外设接口和存储器配置,在不同的应用场合可以最大程度减少 DSP 的外围器件,缩小电路板面积,从而提高系统性价比。

DSP 代码兼容,是指为某种 DSP 产品开发的代码可以不加修改或只作很小修改后,可以在其他 DSP 上执行,称为完全兼容或部分兼容。不同厂商在生产自己的 DSP 产品时,所

采用的代码兼容策略是不一样的。但是基本上每一个 DSP 厂家同一系列 DSP 产品中不同型号之间代码是兼容的。有的 DSP 厂家还尽量保持不同系列 DSP 产品之间的兼容性,例如, TI 公司的定点系列 DSP 的 C1x、C2x、C5x 之间是向下兼容的, C1x 和 C2x 的代码可以不加修改地在 C5x 上运行。此外 TI 的 C6000 的定点系列 C62xx 的代码可以不加修改地在浮点系列 C67xx 上运行。除了代码兼容外,还有管脚兼容,例如, TI 的 C62xx 与 C67xx 的某些片种之间保持管脚兼容。

DSP 主要特点是:

1. 哈佛结构

早期的微处理器内部大多采用冯·纽曼 (Von-Neuman) 结构,其片内程序空间和数据空间是合在一起的,取指令和取操作数都是通过一条总线分时进行的。当高速运算时,不但不能同时取指令和取操作数,而且还会造成传输通道上的瓶颈现象。而 DSP 内部采用的是程序空间和数据空间分开的哈佛 (Harvard) 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)。而且,还允许在程序空间和数据空间之间相互传送数据,即改进的哈佛结构。

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。例如 TMS320C54x 内部有 P、C、D 和 E 共 4 条总线 (每条总线包括地址总线 and 数据总线),可以在一个机器周期内从程序存储器取 1 条指令、从数据存储器读 2 个操作数和向数据存储器写 1 个操作数,大大提高了 DSP 的运行速度。因此,对 DSP 来说,内部总线是个十分重要的资源,总线越多,可以完成的功能就越复杂。

3. 流水线结构

DSP 执行一条指令,需要通过取指、译码、取操作数和执行等几个阶段。在 DSP 中,采用流水线结构,在程序运行过程中它的这几个阶段是重叠的,这样,在执行本条指令的同时,还依次完成了后面 3 条指令的取操作数、译码和取指,将指令执行时间降低到最小值。利用这种流水线结构,加上执行重复操作,就能保证数字信号处理中用得最多的乘法累加运算:

$$y = \sum_{i=1}^n a_i x_i$$

可以在单个指令周期内完成。

4. 多处理单元

DSP 内部一般都包括有多个处理单元,如算术逻辑运算单元 (ALU)、辅助运算单元 (ARAU)、累加器 (ACC) 以及硬件乘法器 (MUL) 等。它们可以在一个指令周期内同时进行运算,例如,当执行一次乘法和累加的同时,辅助寄存器单元已经完成了下一个地址的寻址工作,为下一次乘法和累加运算做好了充分的准备。因此, DSP 在进行连续的乘加运算时,每一次乘加运算都是单周期的。DSP 的这种多处理单元结构,特别适用于 FIR 和 IIR 滤波器。此外,许多 DSP 的多处理单元结构还表现在将一些特殊的算法,例如 FFT 的位码倒置寻址和取模运算等做成硬件,以提高运行速度。

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。例如, TMS320C25 中的 MACD (乘法、累加和数据移动) 指令, 具有执行 LT (加载)、DMOV (数据移动)、MPY (乘法) 和 APAC (累加) 等 4 条指令的功能; TMS320C54x 中的 FIRS 和 LMS 指令, 则专门用于对称 FIR 滤波器和 LMS 算法。

6. 指令周期短

早期的 DSP 的指令周期约 200 ns, 采用 4 μm NMOS 制造工艺, 其运算速度为 5 MIPS (每秒执行 500 万条指令)。随着集成电路工艺的发展, DSP 广泛采用亚微米 CMOS 制造工艺, 其运行速度越来越快, 以 TMS320C54x 为例, 其运行速度可达 100 MIPS 以上。

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、32 位。为防止运算过程溢出, 有的累加器达到 40 位。此外, 一批浮点 DSP, 例如 TMS320C3x、TMS 320C4x、ADSP21020 等, 则提供了更大的动态范围。

8. 硬件配置强

新一代 DSP 的接口功能愈来愈强, 片内具有串行口、主机接口 (HPI)、DMA 控制器、软件控制的等待状态产生器、锁相环时钟产生器以及实现在片仿真符合 IEEE 1149.1 标准的测试访问口, 更易于完成系统设计。许多 DSP 芯片都可以工作在省电方式, 使系统功耗降低。

DSP 芯片的上述特点, 使其在各个领域得到越来越广泛的应用。

DSP 的性能指标为: MIPS (每秒百万条指令), MOPS (每秒百万次操作), MFLOPS (每秒百万次浮点操作), MBPS (每秒百万位)。

1.3 DSP 芯片的历史、现状、发展方向及应用

第一个 DSP 芯片诞生于 20 世纪 70 年代末, 以 AMI 公司的 S2811 和 Intel 公司的 2920 为代表的第二代 DSP 芯片, 其片内都没有单周期硬件乘法器。

自从 1980 年以后, DSP 芯片取得了突飞猛进的发展, 主要表现在以下几个方面:

1. 制造工艺

早期 DSP 采用 4 μm 的 N 沟道 MOS (NMOS) 工艺, 现在的 DSP 则普遍采用亚微米 CMOS 工艺, 达到 0.25 μm 或 0.18 μm 。DSP 芯片的引脚数量从 40 个左右增加到 200 个以上, 需要设计的外围电路越来越少, 每 MIPS 的成本、体积和功耗都有很大的下降。

2. 存储器容量

20 世纪 80 年代初的 DSP, 片内的程序存储器和数据存储器只有几百个单元, 有的片内没有 ROM。目前, DSP 片内的数据和程序存储器可达几十千字节。此外, 对片外程序存储器和数据存储器的寻址能力也大大增强, 可分别达到 16M $\times$ 48 位和 4G $\times$ 40 位。

3. 运行速度

经过 20 年的发展, 使 DSP 的指令周期从 400 μs 缩短到 10 μs 以下, 相应的运行速度从 2.5 MIPS 提高到 100 MIPS 以上。具有代表性的是, AD 公司的 ADSP2106x 和 TI 公司的 TMS320C6x DSP, 它们执行 1024 点 FFT 运算的时间分别只有 90 μs 和 66 μs 。

4. 内部结构

目前, DSP 芯片内部广泛采用多总线、多处理单元和多级流水线结构。加上完善的接口功能, 使 DSP 的系统功能、数据处理能力以及与外部设备的通信功能大大增强。

5. 运算精度和动态范围

由于输入信号动态范围以及迭代算法可能产生误差积累问题，因此对单片 DSP 的精度提出了较高的要求。DSP 的字长从 8 位增加到 16 位、24 位、32 位，累加器的长度也增加到 40 位。高性能的浮点 DSP 的出现扩大了数据处理的动态范围。

6. 开发工具

20 世纪 90 年代推出的 DSP，都有较为完善的软件和硬件开发工具，其中包括 Simulator 软件仿真器、Emulator 在线仿真器、C 编译器等，给开发应用带来很大的方便。

随着现代通信技术、计算机技术以及超大规模集成电路工艺的不断发展，DSP 芯片将会有进一步的发展，预计其发展趋势可概括为：

1. 发展高速、高性能 DSP 器件

预计到 2010 年，DSP 芯片的运行速度将达到 50000 MIPS 以上。

2. 高度集成化

集滤波、A/D、D/A、ROM、RAM 和 DSP 内核于一体的模拟数字混合式 DSP 芯片将有较大的发展和应用。

3. 低功耗低电压

进一步降低功耗，开发低电压 DSP 内核（目前有的 DSP 内核电压已降到 3.3V 和 2.5V），使其更适用于个人通信机、便携式计算机和便携式仪器仪表。

4. 开发专用 DSP 芯片

为了满足系统级芯片的设计，开发基于 DSP 内核的 ASIC（专用集成电路）会有较大的发展。

5. 提供更加完善的开发环境

特别是开发效率更高的、优化的 C 编译器和代数式指令系统，以克服汇编语言程序可读性和可移植性较差的不足，缩短开发周期。

6. 扩大应用领域

DSP 芯片将向航空、航天、雷达、声纳、图像、影视、医疗设备、家用电器等众多领域渗透，进一步扩大应用范围。DSP 芯片的应用主要在以下一些方面：

1) 信号处理：数字滤波、快速 FFT、相关运算、谱分析、自适应滤波、卷积、模式匹配、加窗、波形产生等。

2) 通信：调制解调器、数据压缩、回波抵消、多路复用、传真、自适应均衡、数据加密、扩频通信、纠错编码、可视电话等。

3) 语音：语音邮件、语音存储、语音编码、语音合成、语音识别、语音增强、说话人辨认、说话人确认等。

4) 图形/图像：图像增强、动画、机器人视觉、二维/三维图形处理、图像压缩与传输等。

5) 军事：导航、雷达处理、声纳处理、导弹制导等。

6) 医学：病人监控、修复手术、超声设备等。

7) 控制：机器人控制、发动机控制、激光打印控制。

8) 汽车控制：自适应驾驶控制、导航、振动分析等。

9) 家用电器：数码相机、音乐合成、移动电话等。

第 2 章 TMS320C54x 的结构原理

2.1 TMS320C54x 的主要特点及组成框图

TMS320C54x(简称 C54x)定点系列是 TMS320 通用数字信号处理器家族的一员。C54x 系列适用于实时嵌入式系统的开发。例如,无线通信、语音处理等。C54x 的中央处理单元(CPU)具有哈佛结构、低功耗设计和高度并行性等特点。除此之外,通用的寻址模式和指令集可以全面地发挥系统性能。在此我们以 TMS320C54x 为主,系统介绍 TMS320C54x 结构原理及其应用。

2.1.1 TMS320C54x 的主要特点

C54x 具有高度的灵活性和速度,它的设计采用了一系列新技术组成:增强的哈佛结构(一条程序总线、三条数据总线和四条地址总线),CPU 中适用于特殊应用的硬件逻辑单元,片上存储器,片上外设和高效的专用指令集。

C54x 具有以下优点:

- 由一条程序总线、三条数据总线和四条地址总线构成的增强的哈佛结构提供了更高的速度和灵活性。
- CPU 中适用于特殊应用的硬件逻辑可以提高 CPU 处理性能。
- 高效的指令集能够快速地实现算法和优化的高级语言。
- 模块化的结构设计可以快速地推出新的系列产品。
- 高级的 IC 工艺提供了更好的性能价格比和更低的功耗。
- 使用新的静态设计技术,降低了芯片的电磁辐射。

下面对上述特性进行详细介绍。

1. CPU(中央处理单元)

- 1) 一条程序总线、三条数据总线和四条地址总线组成的增强多总线结构。
- 2) 40 位的算术逻辑单元(ALU)包括 40 位的桶形移位器和两个独立的 40 位累加器。
- 3) 17×17 位并行乘法单元和专用的 40 位加法器用于无等待状态的单周期乘/加操作。
- 4) 比较、选择和存储单元(CSSU)能够完成维特比的加法/比较选择操作。
- 5) 指数译码器可以在单周期内对 40 位累加器进行指数运算。
- 6) 两个地址发生器,包括 8 个辅助寄存器(AR0~AR7)和 2 个辅助寄存器算术单元(ARAU0、ARAU1)。
- 7) C5420 还包括一个双 CPU 的结构。

2. 存储器

- 1) $192\text{K} \times 16$ 位可寻址存储空间(64K 程序存储空间、64K 数据存储空间和 64K I/O 空间),其中, C548、C549、C5402、C5410 和 C5420 的程序存储空间还可以扩展到 8M。
- 2) 片上存储器配置见表 2-1:

表 2-1 片上存储器配置

型 号	ROM	双存储 RAM (DARAM)	单存取 RAM (SARAM)
C541/LC541	28K	5K	0
C542/LC542	2K	10K	0
LC543	2K	10K	0
LC545	48K	6K	0
LC546	48K	6K	0
LC548	2K	8K	24K
LC549/VC549	16K	8K	24K
VC5402	8K	16K	0
VC5410	16K	8K	56K
VC5420	0	32K	168K

3. 指令集
- 1) 单指令重复和块指令重复。

2) 可以更好地管理程序存储器和数据存储器的块移动指令。

3) 32 位长整数操作指令。

4) 指令同时读取 2 或 3 个操作数。

5) 条件保存指令。

6) 快速中断返回。
4. 片上外设 (见表 2-2)

表 2-2 片上外设

型 号	并 行 口	同 步 串 口	缓 冲 串 口	多通道缓冲串口	时分复用串口
C541/LC541	0	2	0	0	0
C542/LC542	1	0	1	0	1
LC543	0	0	1	0	1
LC545	1	1	1	0	0
LC546	0	1	1	0	0
LC548	1	0	2	0	1
LC549/VC549	1	0	2	0	1
VC5402	1	0	0	2	0
VC5410	1	0	0	3	0
VC5420	1	0	0	6	0

- 1) 可编程软件等待状态发生器。
- 2) 可使用内部振荡源或外部振荡源的锁相环。当使用外部振荡源时，内部允许使用多个值对芯片倍频。
- 3) 外部总线控制可以禁止或允许总线和控制信号的输出。
- 4) 数据总线支持总线挂起的特征。
- 5) 可编程定时器。

6) 串/并行口。

5. 速度

执行单周期定点指令时间为 25/20/15/12 (40MIPS/50MIPS/66MIPS/80MIPS/100MIPS)。表 2-3 给出了 C54x 有关参数。

表 2-3 C54x 电源、速度及封装

型 号	电源提供 (V)	速 度	封 装
C541	5V	25ns	100-pin TQFP
LC541	3V/3.3V	25ns/20ns	100-pin TQFP
LC541B	3V/3.3V	15ns	100-pin TQFP
C542	5V	25ns	144-pin TQFP
LC542	3V/3.3V	25ns/20ns	144-pin TQFP/128-pin TQFP
LC543	3V/3.3V	25ns/20ns	100-pin TQFP
LC545	3V/3.3V	25ns/20ns	128-pin TQFP
LC545A	3V/3.3V	15ns	128-pin TQFP
LC546	3V/3.3V	25ns/20ns	100-pin TQFP
LC546A	3V/3.3V	15ns	100-pin TQFP
LC548	3.3V	25ns/20ns	144-pin TQFP
LC549	3.3V	15ns/125ns	144-pin TQFP/144-pin BGA
VC549	3.3V(core2.5V)	10ns	144-pin TQFP/144-pin BGA
VC5402	3.3V(core1.8V)	10ns	144-pin TQFP/144-pin BGA
VC5410	3.3V(core1.8V)	10ns	144-pin TQFP/144-pin BGA
VC5420	3.3V(core1.8V)	10ns	144-pin TQFP/144-pin BGA

6. 电源

- 1) 由 IDLE1、IDLE2 和 IDLE3 指令配置电源管理和低功耗模式。
- 2) 控制 CLKOUT 引脚的输出。

7. 仿真

符合 IEEE149.1 边界扫描逻辑接口标准的片上边界扫描仿真逻辑。

2.1.2 TMS320C54x 的组成框图

C54x 的体系结构，包含中央处理机 (CPU)、存储器和片上外设三个部分。C54x 使用高级的增强型哈佛结构，8 条内部总线使芯片的处理能力发挥到最大程度，程序空间和数据空间允许同时存取程序指令和数据。例如，3 次读操作和 1 次写操作能在单周期内执行。并行指令与特殊的应用指令能完全地利用这个体系结构。另外，数据能在数据存储空间和程序存储空间中移动。这种并行结构使得算术、逻辑和位操作能在单机器周期内全部被执行。

同时，所有的 C54x 芯片都包含硬件控制部分，以用于管理中断、重复操作和子程序调用。图 2-1 是 TMS320C54x 的功能框图，它包含了片上外设和总线结构。

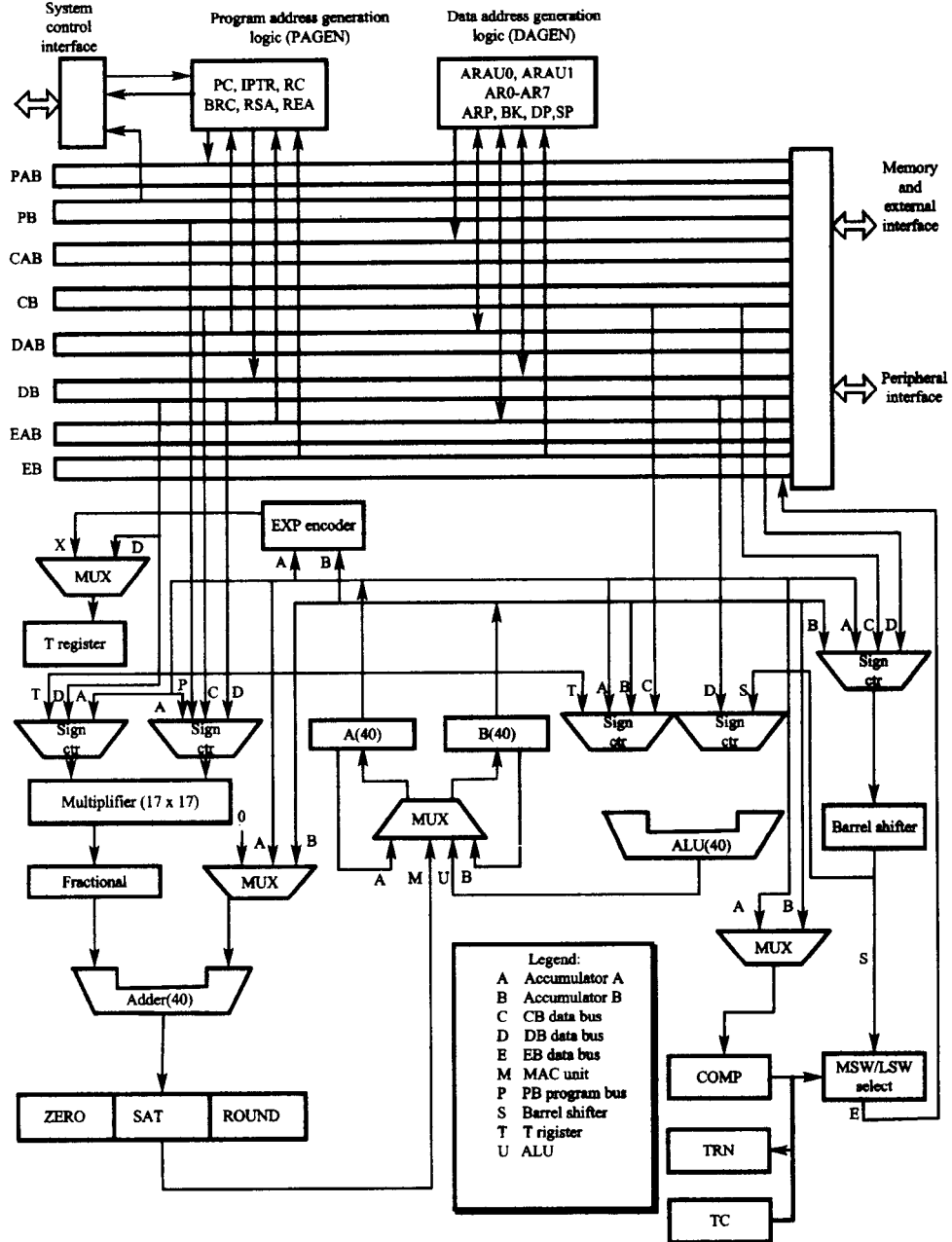


图 2-1 TMS320C54x DSP 的内部硬件组成框图

2.2 总线结构

C54x 由 8 条主要的 16 位总线（4 条程序/数据总线和 4 条地址总线）组成：

- 1) 程序总线 (PB)：从程序存储器装载指令码和立即操作数。
- 2) 3 条数据总线 (CB、DB、EB)：负责将片上的各个不同的部分相互连接，例如 CPU，

数据地址产生逻辑，程序地址产生逻辑，片上外设和数据存储器之间可以传送数据。

CB 和 DB：从数据存储器读取操作数。

EB：把操作数写到数据存储器。

3) 4 条地址总线 (PAB, CAB, DAB 和 EAB)：负责装载指令执行所需要的地址。

C54x 能利用两个辅助寄存器算术单元 (ARAU0 和 ARAU1) 在同一个周期内生成两个数据存储器地址。

PB 能加载保存于程序存储空间的操作数 (例如系数表) 到乘法器和加法器进行乘 / 加操作或利用数据移动指令 (MVPD 和 READA) 把操作数移到数据存储空间的目的地中。这种性能与双操作数读取的特征一起，使 C54x 支持单周期三操作数指令。例如 FIRS 指令。

C54x 也有一条双向的片上总线用于访问片上外设。这条总线轮流使用 DB 和 EB 与 CPU 连接。访问者使用这条总线进行读写操作需要两个或更多的周期，具体周期取决于片上外设的结构。表 2-4 给出了各种不同寻址类型所用到的总线。

表 2-4 各种读写方式用到的总线

读/写方式	地 址 总 线				数 据 总 线			
	PAB	CAB	DAB	EAB	PB	CB	DB	EB
程序读	√				√			
程序写	√							√
单数据读			√				√	
双数据读		√	√			√	√	
长数据 (32 位) 读		√ (hw)	√ (lw)			√ (hw)	√ (lw)	
单数据写				√				√
数据读/数据写			√	√			√	√
双数据读/系数读	√	√	√		√	√	√	
外设读			√				√	
外设写				√				√

注：hw=高 16 位， lw=低 16 位

2.3 内部存储空间

本节介绍 C54x 的存储器组织和操作。通常情况，C54x 芯片有一个总共 192K×16 位的存储空间，这个存储空间被分为独立编址的 3 个部分：64K 的程序存储空间、64K 的数据存储空间和 64K 的 I/O 空间。在某些时候，如 C548 存储空间还能进行更大的扩充。C54x 并行的体系结构和对片上存储器的双存取能力，使得 C54x 能够在一个指定的周期内完成 4 个并发的存储器操作：1 个取指操作、2 个数据读操作和 1 个数据写操作。

使用片上存储器有几个优点：

- 1) 高速执行而不需等待状态。
- 2) 节约成本。
- 3) 比外部存储器的功耗低。

使用外部存储器的主要优点是：可以访问一个很大的存储空间。

2.3.1 存储空间

C54x 的存储空间由 3 个独立可选的存储空间组成。它们是：程序存储空间、数据存储空间和 I/O 空间。在任何一个存储空间内，ROM、RAM、EPROM 和 EEPROM 以及芯片中的