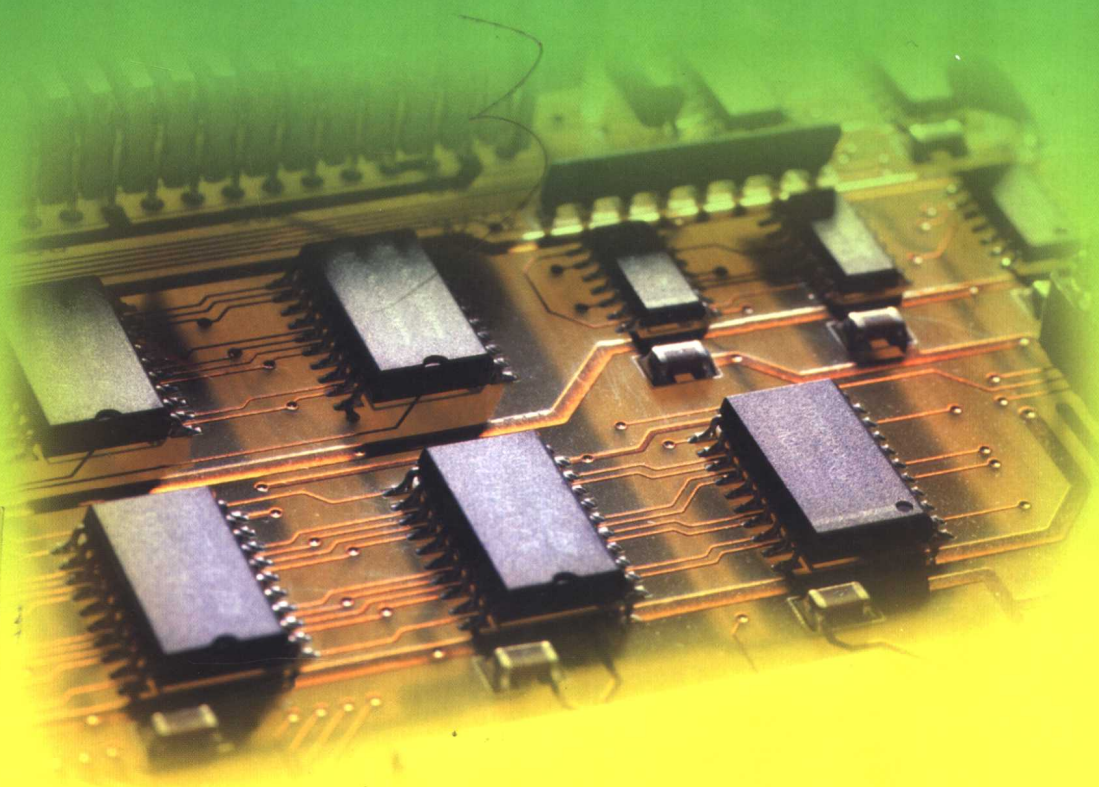


# DSP

## 开发技术实例与技巧

任建国 主编

张家祥 徐从启 孟志强 李益华 编著



国防工业出版社

National Defense Industry Press

## 内 容 简 介

本书以 TI 公司的 16 位定点处理器 TMS320C54x 系列 DSP(数字信号处理器)为例,系统地介绍了 DSP 的内部结构、外设、指令系统、软件开发及其可视化开发环境 CCS 的使用。全书共分为 7 章,内容包括:DSP CPU 的构成、串口的使用、外设扩展应用、汇编指令系统及 CCS 软件的使用说明。

本书的特点是理论和实践相结合,由浅入深、通俗易懂、实用性强,能帮助读者循序渐进地掌握 DSP 的工作原理 and 开发技能。

本书可作为电子信息、通信工程、自动化等专业高年级本科生和研究生的教材或参考工具书,也可供从事 DSP 芯片开发应用的工程技术人员参考。

### 图书在版编目(CIP)数据

DSP 开发技术实例与技巧/任建国主编;张家祥等编  
著. —北京:国防工业出版社,2007.1  
ISBN 7-118-04906-9

I. D... II. ①任...②张... III. 数字信号—信号  
处理 IV. TN911.72

中国版本图书馆 CIP 数据核字(2006)第 153972 号

※

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100044)

腾飞胶印厂印刷

新华书店经售

\*

开本 787×1092 1/16 印张 16 $\frac{3}{4}$  字数 382 千字

2007 年 1 月第 1 版第 1 次印刷 印数 1—4000 册 定价 28.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)68428422

发行邮购:(010)68414474

发行传真:(010)68411535

发行业务:(010)68472764

# 前 言

数字信号处理(Digital Signal Processing, DSP)是现代电子系统的核心和灵魂。随着越来越多的电子产品将 DSP 作为技术核心, DSP 已经成为推动数字化进程的动力。数字信号处理器(Digital Signal Processor, DSP)是现代电子技术、计算机技术和信号处理技术相结合的产物。DSP 独特的结构决定了它具有运算速度快,具有可编程特性和接口灵活的特点,能快速实现各种数字信号处理算法,随着信息和电子技术的飞速发展, DSP 在电子信息、通信、软件、无线电、自动控制、仪器仪表、信息家电等多个领域得到了广泛的应用。采用它来实现数字信号处理系统更是当前的发展趋势。

本书是作者参考国内外最新的教材和文献资料,结合近几年来学习开发 DSP 系统的体会编写而成,其目的是使读者了解 TMS320C54x 的体系结构和基本原理,熟悉 DSP 芯片的开发工具和使用,掌握 DSP 系统的软硬件设计和应用系统的开发方法,具备独立从事 DSP 应用开发的能力。

本书共分 7 章。

第 1 章为 DSP 开发基础知识,介绍了数字信号处理的发展、DSP 的结构特点、种类、选型和设计的一般流程,以及 TMS320 系列 DSP 的主要特点。

第 2 章讲的是 TMS320C54x DSP 的内部结构原理,详细描述了 CPU、存储器、中断控制以及寻址方式。

第 3 章全面介绍了 DSP 片内、片外外设的特点和应用。

第 4 章以 TMS320C54x 为例,介绍了数字信号处理器的汇编语言编程方法、指令系统和宏指令,并对 DSP 专用的 COFF 文件作了介绍。

第 5 章主要介绍用 C 语言开发 DSP 程序的方法,给出了 C 语言程序的一般框架以及 C 语言与汇编语言混合编程的方法。

第 6 章是 TMS320C54x 的开发实例,介绍了 Flash 烧写和语音处理两个例子。

第 7 章介绍了 TI 公司的可视化开发环境 CCS。

书中附录给出了 C54x DSP 指令系统列表、TMS320 系列产品的命名方法以及书中部分实例的源代码,这些都是学习与设计当中可参考使用的。

本书由任建国主编,张家祥、徐从启、孟志强、李益华、周林、孙志强、满军同志参与了编写,湖南大学的孟志强教授和长沙理工大学的李益华副教授还对本书进行了细致的指导和审校工作。

由于水平有限,书中难免存在错误和疏漏,不当之处恳请读者批评指正。

编 者

2006 年 9 月

# 目 录

|                                 |           |
|---------------------------------|-----------|
| <b>第 1 章 DSP 开发基础知识 .....</b>   | <b>1</b>  |
| 1.1 数字信号处理 .....                | 1         |
| 1.1.1 数字信号处理简介 .....            | 1         |
| 1.1.2 DSP 系统构成 .....            | 2         |
| 1.2 DSP 开发的一般流程 .....           | 2         |
| 1.3 DSP 芯片的基本结构特点 .....         | 3         |
| 1.4 DSP 芯片的分类、选择及应用 .....       | 5         |
| 1.4.1 DSP 芯片的分类 .....           | 5         |
| 1.4.2 DSP 芯片的选择 .....           | 5         |
| 1.4.3 DSP 芯片的应用 .....           | 7         |
| 1.5 DSP 芯片的发展及 TI 公司产品简介 .....  | 7         |
| 1.5.1 DSP 芯片的发展 .....           | 7         |
| 1.5.2 TI 公司 DSP 芯片简介 .....      | 8         |
| 1.6 定点 DSP 芯片的数据格式 .....        | 11        |
| 1.6.1 数的定标 .....                | 11        |
| 1.6.2 定点数据格式选择 .....            | 12        |
| 1.6.3 浮点数与定点数的转换关系 .....        | 12        |
| <b>第 2 章 DSP 芯片的结构原理 .....</b>  | <b>14</b> |
| 2.1 TMS320C54x 的内部结构及主要特性 ..... | 14        |
| 2.1.1 TMS320C54x 的内部结构 .....    | 14        |
| 2.1.2 TMS320C54x 的主要特性 .....    | 15        |
| 2.2 总线结构 .....                  | 17        |
| 2.3 存储器 .....                   | 18        |
| 2.3.1 存储器空间 .....               | 19        |
| 2.3.2 程序存储器 .....               | 21        |
| 2.3.3 数据存储器 .....               | 23        |
| 2.4 中央处理单元 .....                | 24        |
| 2.4.1 CPU 状态和控制寄存器 .....        | 25        |
| 2.4.2 算术逻辑单元 .....              | 29        |
| 2.4.3 累加器 A 和 B .....           | 31        |
| 2.4.4 桶形移位寄存器 .....             | 32        |
| 2.4.5 乘法器/加法器单元 .....           | 33        |

|            |                                 |            |
|------------|---------------------------------|------------|
| 2.4.6      | 比较、选择和存储单元 .....                | 35         |
| 2.4.7      | 指数编码器 .....                     | 37         |
| 2.4.8      | 地址发生器 .....                     | 37         |
| 2.5        | 寻址方式 .....                      | 39         |
| 2.5.1      | 立即数寻址 .....                     | 40         |
| 2.5.2      | 绝对地址寻址 .....                    | 41         |
| 2.5.3      | 累加器寻址 .....                     | 42         |
| 2.5.4      | 直接寻址 .....                      | 43         |
| 2.5.5      | 间接寻址 .....                      | 44         |
| 2.5.6      | 存储器映射寄存器寻址 .....                | 51         |
| 2.5.7      | 堆栈寻址 .....                      | 52         |
| 2.6        | 流水线结构 .....                     | 52         |
| 2.7        | 中断系统 .....                      | 54         |
| 2.7.1      | 中断系统概述 .....                    | 54         |
| 2.7.2      | 中断标志寄存器(IFR)及中断屏蔽寄存器(IMR) ..... | 55         |
| 2.7.3      | 中断处理流程 .....                    | 55         |
| 2.7.4      | 重新映射中断向量地址 .....                | 60         |
| <b>第3章</b> | <b>DSP 芯片的外围设备 .....</b>        | <b>61</b>  |
| 3.1        | 片内外围设备 .....                    | 61         |
| 3.1.1      | 通用 I/O 口 .....                  | 61         |
| 3.1.2      | 定时器 .....                       | 62         |
| 3.1.3      | 时钟发生器 .....                     | 64         |
| 3.1.4      | 可编程等待状态发生器 .....                | 69         |
| 3.1.5      | 分区切换逻辑 .....                    | 70         |
| 3.1.6      | HPI 接口 .....                    | 72         |
| 3.1.7      | McBSP 多通道缓冲串口 .....             | 76         |
| 3.1.8      | JTAG 接口 .....                   | 104        |
| 3.2        | 片外外围设备 .....                    | 106        |
| 3.2.1      | 外扩程序存储器 .....                   | 107        |
| 3.2.2      | 外扩数据存储器 .....                   | 109        |
| 3.2.3      | I/O 接口扩展 .....                  | 110        |
| 3.2.4      | A/D 和 D/A 接口设计 .....            | 112        |
| <b>第4章</b> | <b>汇编语言工具 .....</b>             | <b>113</b> |
| 4.1        | 汇编语言基本概念 .....                  | 113        |
| 4.1.1      | 汇编源程序语句格式 .....                 | 113        |
| 4.1.2      | 汇编语言常量 .....                    | 115        |
| 4.1.3      | 字符串 .....                       | 116        |
| 4.1.4      | 符号 .....                        | 116        |
| 4.1.5      | 表达式 .....                       | 117        |

|              |                                  |            |
|--------------|----------------------------------|------------|
| 4.2          | 汇编语言指令系统 .....                   | 118        |
| 4.2.1        | 指令系统中的符号和缩写 .....                | 118        |
| 4.2.2        | 指令系统中的记号和运算符 .....               | 121        |
| 4.2.3        | 指令系统分类 .....                     | 122        |
| 4.3          | 宏语言 .....                        | 123        |
| 4.4          | 汇编伪指令 .....                      | 125        |
| 4.4.1        | 常数初始化伪指令 .....                   | 125        |
| 4.4.2        | 段程序计数器定位指令 .align .....          | 128        |
| 4.4.3        | 输出列表格式指令 .drlst /.drnolist ..... | 128        |
| 4.4.4        | 引用其他文件的伪指令 .....                 | 129        |
| 4.4.5        | 条件汇编指令 .....                     | 129        |
| 4.4.6        | 汇编时的符号定义伪指令 .....                | 130        |
| 4.4.7        | 其他方面的汇编伪指令 .....                 | 130        |
| 4.5          | 汇编语言程序的编辑、汇编和链接过程 .....          | 131        |
| 4.5.1        | 汇编语言程序的开发流程 .....                | 131        |
| 4.5.2        | 汇编 .....                         | 132        |
| 4.5.3        | 链接 .....                         | 137        |
| 4.5.4        | 链接器命令文件 .....                    | 139        |
| 4.6          | 公共目标文件格式 .....                   | 142        |
| 4.6.1        | COFF 文件中的段 .....                 | 143        |
| 4.6.2        | 段定义伪指令 .....                     | 144        |
| 4.6.3        | 链接伪指令 .....                      | 147        |
| 4.6.4        | COFF 文件中的符号 .....                | 152        |
| <b>第 5 章</b> | <b>DSP 软件开发过程 .....</b>          | <b>154</b> |
| 5.1          | C 语言编程 .....                     | 154        |
| 5.1.1        | “C54x DSP 软件开发流程 .....           | 154        |
| 5.1.2        | C 程序设计框架 .....                   | 156        |
| 5.1.3        | “C54x DSP C 优化编译器 .....          | 158        |
| 5.1.4        | C 语言编程链接命令文件的设计 .....            | 163        |
| 5.2          | 用 C 语言和汇编语言混合编程 .....            | 165        |
| 5.2.1        | 混合编程规范 .....                     | 166        |
| 5.2.2        | 混合编程方法 .....                     | 167        |
| 5.3          | 引导方式设计 .....                     | 172        |
| 5.3.1        | 引导方式选择 .....                     | 173        |
| 5.3.2        | 并行引导方式 .....                     | 173        |
| 5.3.3        | 引导表生成 .....                      | 175        |
| <b>第 6 章</b> | <b>TMS320C54x 芯片开发实例 .....</b>   | <b>177</b> |
| 6.1          | 面向 DSP 的 C 程序设计流程及设计准则 .....     | 177        |
| 6.1.1        | 面向 DSP 的 C 程序设计流程 .....          | 177        |

|                                   |     |
|-----------------------------------|-----|
| 6.1.2 面向 DSP 的 C 程序设计准则 .....     | 177 |
| 6.2 实例一:“Hello world!” .....      | 179 |
| 6.3 实例二:Flash 烧写程序 .....          | 181 |
| 6.4 实例三:语音处理程序 .....              | 183 |
| <b>第 7 章 CCS 集成开发环境</b> .....     | 194 |
| 7.1 CCS 简介 .....                  | 194 |
| 7.1.1 CCS 的主要功能 .....             | 194 |
| 7.1.2 代码生成工具 .....                | 195 |
| 7.1.3 CCS 的安装及设置 .....            | 197 |
| 7.1.4 DSP 应用程序开发的一般步骤 .....       | 199 |
| 7.2 CCS 的基本操作 .....               | 199 |
| 7.2.1 创建工程文件 .....                | 199 |
| 7.2.2 向工程添加文件 .....               | 200 |
| 7.2.3 查看源代码 .....                 | 201 |
| 7.2.4 构建和运行程序 .....               | 202 |
| 7.2.5 修改构建选项和纠正语法错误 .....         | 205 |
| 7.2.6 使用断点和观察窗口 .....             | 206 |
| 7.2.7 使用观察窗口观察 structure 变量 ..... | 207 |
| 7.3 在 CCS 中调试应用程序 .....           | 208 |
| 7.3.1 查看工程及运行程序 .....             | 208 |
| 7.3.2 断点工具 .....                  | 210 |
| 7.3.3 探针点工具 .....                 | 212 |
| 7.3.4 图形显示 .....                  | 215 |
| 7.3.5 存储器/变量的查看与修改 .....          | 221 |
| 7.3.6 程序测试工具 .....                | 224 |
| <b>附录 A TMS320C54x 指令表</b> .....  | 229 |
| <b>附录 B Flash 烧写程序源代码</b> .....   | 238 |
| <b>附录 C TMS320 系列产品命名</b> .....   | 257 |
| <b>参考文献</b> .....                 | 258 |

# 第1章 DSP 开发基础知识

本章对 DSP 开发的基础知识进行简要介绍。

## 1.1 数字信号处理

### 1.1.1 数字信号处理简介

数字信号处理 (Digital Signal Processing, DSP), 顾名思义, 就是对信号的数字处理, 是20 世纪60 年代前后发展起来的并广泛应用于许多领域的新兴学科。现在大学课本里的数字信号处理是讲述信号数字化处理的基本理论、算法和应用, 而数字信号处理的实现则是理论和应用之间的桥梁。它是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, 以得到符合人们需要的信号形式。

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, 由于受实现方法的限制, 数字信号处理的理论还得不到广泛的应用。直到20世纪70年代末80年代初世界上第一片单片可编程DSP芯片的诞生, 才将理论研究结果广泛应用到低成本的实际系统中, 并且推动了新的理论和应用领域的发展。可以毫不夸张地说, DSP芯片的诞生及发展对近20年来通信、计算机、控制等领域的技术发展起到十分重要的作用。

数字信号处理的实现方法一般有以下几种:

#### 1. 软件实现法

在通用的计算机 (如PC机) 上用软件 (如Fortran、C语言) 实现; 速度较慢, 一般可用于DSP 算法的模拟。

#### 2. 硬件实现法

在通用计算机系统中加上专用的加速处理机实现; 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。

#### 3. 软硬件结合实现法

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数字信号处理, 如数字控制等; 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, 为数字信号处理的应用打开了新的局面。



1.1.2 DSP 系统构成

图1-1所示为一个典型的DSP系统。



图1-1 典型的DSP系统

图中的输入信号可以有各种各样的形式。例如，它可以是麦克风输出的语音信号或是电话线来的已调制数据信号，可以是编码后在数字链路上传输或存储在计算机里的摄像机图像信号等。

输入信号首先进行带限滤波和抽样，然后进行A/D（Analog to Digital）转换将信号转换成数字比特流。根据抽样定理，为保证信息不丢失，抽样频率至少必须是输入带限信号最高频率的2倍。

DSP芯片的输入是由模数转换器（ADC）转换后得到的以抽样形式表示的数字信号，DSP芯片对输入的数字信号进行某种形式的处理，如实现一系列的信号处理算法等。经过处理后的数字样值再经数模转换器（DAC）转换为模拟样值，之后再再进行内插和平滑滤波就可得到连续的模拟波形。

上面给出的DSP系统模型是一个典型模型，并不是所有的DSP系统都必须具有模型中的所有部件。如语音识别系统在输出端并不是连续的波形，而是识别结果，如数字、文字等；有些输入信号本身就是数字信号，如CD（Compact Disk）就不必进行模数转换。

1.2 DSP 开发的一般流程

图 1-2 所示是 DSP 系统设计的一般过程。

DSP 系统设计的步骤依次为：

(1) 在设计 DSP 系统之前，应根据目标系统的应用需求确定系统的性能指标和数据处理算法。

(2) 根据系统的要求进行高级语言的模拟。对于系统而言，不同的数据处理方法可能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。算法模拟所用的输入数据，由实际信号经采集而获得，通常以计算机文件的形式存储为数据文件。在有些情况下，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

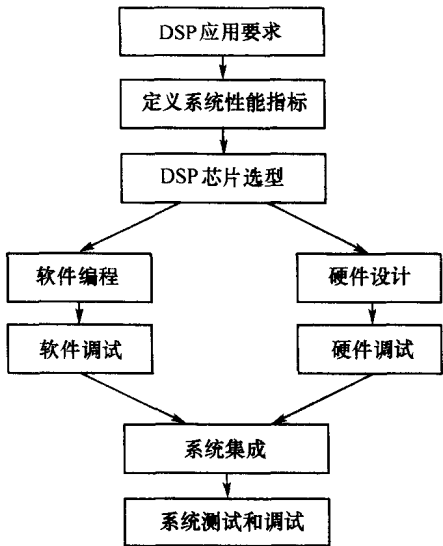


图1-2 DSP系统的设计流程

入假设的数据也是可以的。

(3) 根据系统运算量的大小、对运算精度的要求、系统成本限制以及体积、功耗等要求选择合适的 DSP 芯片。

(4) 设计实时 DSP 系统,包括硬件设计和软件设计两个方面。硬件设计包括设计 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。

(5) 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。

(6) 将软件直接搭载在应用系统上运行,反复调试使系统性能达到指定要求。需要注意的是,此时并没有大功告成。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。

## 1.3 DSP 芯片的基本结构特点

DSP 芯片,也称数字信号处理器(Digital Signal Processor),是一种特别适合于进行数字信号处理运算的微处理器,其主要应用是实时快速地实现各种数字信号处理算法。根据数字信号处理的要求,DSP 芯片一般具有如下主要特点。

### 1. 哈佛结构

哈佛结构是不同于传统的冯·诺依曼(Von Neuman)结构的并行体系结构,其主要特点是将程序和数据存储在不同的存储空间中,即程序存储器和数据存储器是两个相互独立的存储器,每个存储器独立编址,独立访问。与两个存储器相对应的是系统中设置了程序总线 and 数据总线两条总线,从而使数据的吞吐率提高了一倍。而冯·诺依曼结构则是将指令、数据、地址存储在同一存储器中,统一编址,依靠指令计数器提供的地址来区分是指令、数据还是地址。取指令和取数据都访问同一存储器,数据吞吐率低。

在哈佛结构中,由于程序和数据存储器在两个分开的空间中,因此取指和执行能完全重叠运行。为了进一步提高运行速度和灵活性,第二代哈佛结构在原先基础上作了改进,一是允许数据存放在程序存储器中,并被算术运算指令直接使用,增强了芯片的灵活性;二是指令存储在高速缓冲器(Cache)中,当执行此指令时,不需要再从存储器中读取指令,节约了一个指令周期的时间。

### 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、E 等 4 条总线(每条总线又包括地址总线和数据总线),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一个操作数,大大提高了 DSP 的运行速度。对 DSP 而言,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

与哈佛结构相关，DSP 芯片广泛采用流水线以减少指令执行时间，从而增强了处理器的处理能力。图 1-3 所示为一个 4 级流水线操作的例子。

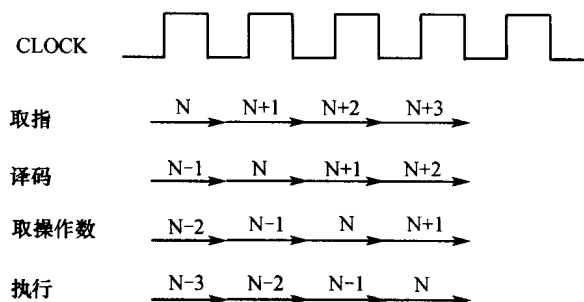


图1-3 4级流水线操作

在 4 级流水线操作中，取指、译码、取操作数和执行操作可以独立地处理，这可使指令执行能完全重叠。在每个指令周期内，4 个不同的指令处于激活状态，每个指令处于不同的阶段。例如，在第  $N$  个指令取指时，前一个指令即第  $N-1$  个指令正在译码，而第  $N-2$  个指令正在取操作数， $N-3$  个指令则正在执行。一般来说，流水线对用户是透明的。

### 4. 多处理单元

DSP 芯片内部一般都包括多个处理单元，如算术逻辑运算单元 (ALU)、辅助寄存器运算单元 (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操作的同时，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 芯片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 芯片的这种结构特别适用于 FIR 和 IIR 滤波器。此外，也使得 FFT 的位码倒置寻址和取模运算等一些特殊的算法，在芯片内部能用硬件实现，提高运行速度。

### 5. 特殊的 DSP 指令

DSP 芯片的另一个特征是采用特殊的指令。例如，TMS320C25 中的 MACD (乘法、累加和数据移动) 指令，具有执行 LT、DMOV、MPY 和 APAC 等 4 条指令的功能；TMS320C54x 中的 FIRS 和 LMS 指令，则专门用于系数对称的 FIR 滤波器和 LMS 算法。

### 6. 快速的指令周期

哈佛结构、流水线操作、专用的硬件乘法器、特殊的 DSP 指令再加上集成电路的优化设计，可使 DSP 芯片的指令周期在 200ns 以下。TMS320 系列处理器的指令周期已经从第一代的 200ns 降低到现在的 20ns 以下。快速的指令周期使得 DSP 芯片能够实时实现许多 DSP 应用。

### 7. 高的运算精度

早期 DSP 芯片的字长为 8 位，后来逐步提高到 16 位、24 位和 32 位。为防止运算过程中溢出，有的累加器达到 40 位。此外，一些浮点 DSP 芯片，例如 TMS320C3x、TMS320C4x、ADSP21020 等，则提供了更大的动态范围。

## 8. 丰富的硬件配置

新一代 DSP 芯片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、软件控制的等待状态发生器、锁相环时钟产生器以及实现在片仿真符合 IEEE1149.1 标准的测试仿真接口, 使系统更易于完成。另外, 许多 DSP 芯片都可以工作在省电方式, 大大降低了系统功耗。

# 1.4 DSP 芯片的分类、选择及应用

## 1.4.1 DSP 芯片的分类

DSP 芯片可以按照表 1-1 所列的 3 种方法进行分类。

表 1-1 DSP 芯片的分类

| 分类方法  | 类 型     | 特 点                                                 | 芯 片 举 例                                                                             |
|-------|---------|-----------------------------------------------------|-------------------------------------------------------------------------------------|
| 按基础特性 | 静态 DSP  | 在某时钟频率范围内的任何时钟频率上, DSP 芯片都能正常工作, 除计算速度有变化外, 没有性能的下降 | TMS320C2xx 系列芯片                                                                     |
|       | 一致性 DSP | 指令集和相应的机器代码机管脚结构相互兼容                                | TMS320C54X                                                                          |
| 按数据格式 | 定点 DSP  | 数据以定点格式工作                                           | TMS320C1X/C2X、TMS320C2xx/C5X、TMS320C54X/C62xx 系列, ADSP21xx 系列, DSP16/16A, MC56000 等 |
|       | 浮点 DSP  | 数据以浮点格式工作                                           | TMS320C3X/C4X/C8X, ADSP21xxX 系列, DSP32/32C, MC96002 等                               |
| 按用途   | 通用型 DSP | 适合普通的 DSP 应用                                        | TI 公司的一系列 DSP 芯片属于通用型 DSP 芯片                                                        |
|       | 专用型 DSP | 为特定的 DSP 运算而设计的, 更适合特殊的运算, 如数字滤波、卷积和 FFT            | Motorola 公司的 DSP56200, Zoran 公司的 ZR34881, Inmos 公司的 IMSA100 等                       |

需要注意的是, DSP 芯片所采用的浮点格式不完全一样, 有的 DSP 芯片采用自定义的浮点格式, 如 TMS320C3x, 而有的 DSP 芯片则采用 IEEE 的标准浮点格式, 如 Motorola 公司的 MC96002、FUJITSU 公司的 MB86232 和 Zoran 公司的 ZR35325 等。

本书主要讨论通用型 DSP 芯片。

## 1.4.2 DSP 芯片的选择

设计 DSP 应用系统, 选择 DSP 芯片是非常重要的一个环节。总的来说, DSP 芯片的选择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. DSP 芯片的运算速度**

运算速度是 DSP 芯片的一个最重要的性能指标,决定整个系统的实时性问题,可以用以下几种性能指标来衡量。

(1) 指令周期:执行一条指令所需的时间,通常以 ns(纳秒)为单位。

(2) MAC 时间:一次乘法加上一次加法的时间。

(3) FFT 执行时间:运行一个 N 点 FFT 程序所需的时间(由于 FFT 运算涉及的运算在数字信号处理中很有代表性,因此 FFT 运算时间常作为衡量 DSP 芯片运算能力的一个指标)。

(4) MIPS:每秒执行 100 万条指令。

(5) MOPS:每秒执行 100 万次操作。

(6) MFLOPS:每秒执行 100 万次浮点操作。

(7) BOPS:每秒执行 10 亿次操作。

### **2. DSP 芯片的运算精度**

一般的定点 DSP 芯片的字长为 16 位,如 TMS320 系列。但有的公司的定点芯片为 24 位,如 Motorola 公司的 MC56001 等。浮点芯片的字长一般为 32 位,累加器为 40 位。

### **3. DSP 芯片的硬件资源**

如片内 RAM、ROM 的数量,外部可扩展的程序和数据空间,总线接口,I/O 接口等。即使是同一系列的 DSP 芯片(如 TI 的 TMS320C54X 系列),系列中不同 DSP 芯片也具有不同的内部硬件资源,可以适应不同的需要。

### **4. DSP 芯片的功耗**

在某些 DSP 应用场合,功耗也是一个需要特别注意的问题。如便携式的 DSP 设备、手持设备、野外应用的 DSP 设备等都对功耗有特殊的要求。目前,1.8V、3.3V 供电的低功耗高速 DSP 芯片已大量使用。

### **5. DSP 芯片的开发工具**

在 DSP 系统的开发过程中,开发工具是必不可少的。有功能强大的开发工具的支持,开发的时间会大大缩短。现在的 DSP 都有较为完善的软件和硬件开发工具,如 TI 公司为它的 TMS320 系列 DSP 开发了专用的集成开发环境 CCS。

### **6. DSP 芯片的价格**

DSP 芯片的价格也是选择 DSP 芯片需要考虑的一个重要因素。价格昂贵的 DSP 芯片,即使性能再高,其应用范围也肯定会受到一定的限制,尤其是民用产品。因此根据实际系统的应用情况,需确定一个价格适中的 DSP 芯片。当然,由于 DSP 芯片发展迅速,DSP 芯片的价格往往下降较快。一般来讲,在开发阶段选用某种价格稍贵的 DSP 芯片,等到系统开发完毕,性价比已经有了较大的改善。

### **7. 其他**

除了上述因素外,选择 DSP 芯片还应考虑以下因素。

(1) 封装的形式:可能有 DIP、PGA、PLCC、PQFP、CQFP 等多种封装形式。

(2) 质量标准:有些 DSP 系统可能最终要求的是工业级或军用级标准,在开发阶段

选择时需要注意所选的芯片是否有工业级或军用级的同类产品。

(3) 供货情况、生命周期：如果所设计的 DSP 系统不仅仅是一个实验系统，而是需要批量生产且有很长的生命周期，那么需要考虑所选的 DSP 芯片的供货情况，是否也有同样甚至更长的生命周期等。

在上述诸多因素中，一般而言，定点 DSP 芯片的价格较便宜，功耗较低，但运算精度稍低。而浮点 DSP 芯片的优点是运算精度高，且 C 语言编程调试方便，但价格稍贵，功耗也较大。

### 1.4.3 DSP 芯片的应用

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。现代数字信号处理器是执行高速数字信号处理的 IC 电路，它恰好适应多媒体信息化社会需求，迅速发展壮大。如今，世界电子器件市场上，各种各样的 DSP 器件已相当丰富，大大小小封装形式的 DSP 器件，已广泛应用于各种产品的生产领域，而且 DSP 的应用领域仍在不断地扩大，发展迅速异常。DSP 的应用主要有以下方面。

- (1) 信号处理：自适应滤波器、快速傅里叶变换、卷积、相关、加窗、波形产生等；
- (2) 通信：如调制解调器、数据加密、数据压缩、扩频通信、纠错编码、可视电话等；
- (3) 图像处理：如三维图像处理、图像压缩与传输、动画、机器人视觉等；
- (4) 军事：如雷达处理、声呐处理、导航、导弹制导、保密通信等；
- (5) 自动控制：如引擎控制、声控、磁盘控制等；
- (6) 语音：如语音编码、语音识别、语音合成、文本语音转换等；
- (7) 医学：胎儿监控、助听器、修复手术、超声设备等；
- (8) 家用电器：高清晰数字电视、高保真音响、音乐合成、电动玩具等。

## 1.5 DSP 芯片的发展及 TI 公司产品简介

### 1.5.1 DSP 芯片的发展

世界上第一个单片 DSP 芯片应当是 1978 年 AMI 公司发布的 S2811，1979 年美国 Intel 公司发布了商用可编程器件 2920。这两种芯片内部都没有现代 DSP 芯片所必须有的单周期乘法器。1980 年，日本 NEC 公司推出的  $\mu$ PD7720 是第一个具有乘法器的商用 DSP 芯片。

其后，随着大规模集成电路技术提高，DSP 芯片得到了突飞猛进的发展，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 10ns 以下（如 TMS320C54x、TMS320C62x/67x 等），处理能力提高了几十倍。DSP 芯片内部关键的乘法器部件从 1980 年的占模片区（die area）的 40% 左右下降到 5% 以下，片内 RAM 数量增加一个数量级以上。从制造工艺来看，1980 年采用 4 $\mu$ m 的 N 沟道 MOS（NMOS）工艺，而现在则普遍采用亚微米（Micron）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,如外部存储器的扩展和处理器间的通信等。此外,DSP芯片的发展使DSP系统的成本、体积、质量和功耗都有很大程度的下降和减小。

目前市场上的主要DSP芯片生产商包括TI、ADI、Motorola、Lucent和Zilog等,其中TI占有最大的市场份额,达到近50%。

经过20多年的发展,随着DSP芯片性价比的不断提高,DSP产品的应用已经扩大到人们的学习、工作和生活的各个方面,并逐渐成为电子产品更新换代的决定因素。

未来DSP的发展趋势为:

- (1) 更高的运行速度和信号处理速度;
- (2) 多DSP协同工作;
- (3) 更方便的开发环境;
- (4) 大量专用DSP的出现(DSP核);
- (5) 更低的价格,或更高的性能价格比;
- (6) 更广泛的应用(每年以30%增长);
- (7) 更低的功耗(55x 0.05mV/MIPS)。

## 1.5.2 TI公司DSP芯片简介

作为第一片DSP产品TMS32010的生产商和DSP行业的领头者,TI公司的产品包括从低端低成本低速度的DSP到高端大运算量的DSP产品。目前,广泛使用的TI的DSP有三个系列:C2000,C5000,C6000,C3x也有使用,而其他型号都基本淘汰。需要注意的是,同一系列中不同型号的DSP一般都具有相同的DSP核,相同或兼容的汇编指令系统;而它们的差别仅在于片内存储器的大小,外设资源(如定时器、串口、并口等)的多少。不同系列的DSP的汇编指令系统不兼容,但汇编语言的语法非常相似。除了汇编语言外,TI还为每个系列都提供了优化C编译器,方便用户使用C(使用ANSI的标准C)语言进行开发,效率可以达到手工汇编的90%甚至更高。下面简单介绍上述几个常用系列。

### 1. C2000 系列

C2000系列是一个控制器系列,全部为16位定点DSP。该系列中的一些型号具有片内Flash RAM,如TMS320F24x,TMS320LF240x等。TI所有的DSP中,也只有C2000有Flash。作为控制器,C2000系列除了有一个DSP核以外,还有大量的外设资源,如A/D、定时器、各种串口(同步或异步)、WATCHDOG、CAN总线、PWM发生器、数字I/O脚等等。特别是C2000的异步串口可以与PC的UART相连,也是TI所有DSP中唯一具有异步串口的系列。C2000系列分布如图1-4所示。

### 2. C5000 系列

C5000系列是一个定点低功耗系列,特别适用于手持通信产品,如手机、PDA、GPS等。目前的处理速度一般在80MIPS~400MIPS。C5000系列主要分为C54xx和C55xx两个系列。两个系列在执行代码级是兼容的,但它们的汇编指令系统却不同。目前TMS320VC5402的零售价在¥60元~¥80元,性价比极高。C5000包含的主要外设有McBSP同步串口、HPI并行接口、定时器、DMA等。其中C55xx提供EMIF外部存储器扩展接口,允许用户直接使用静态存储器SDRAM、SBSRAM、SRAM、EPROM等各

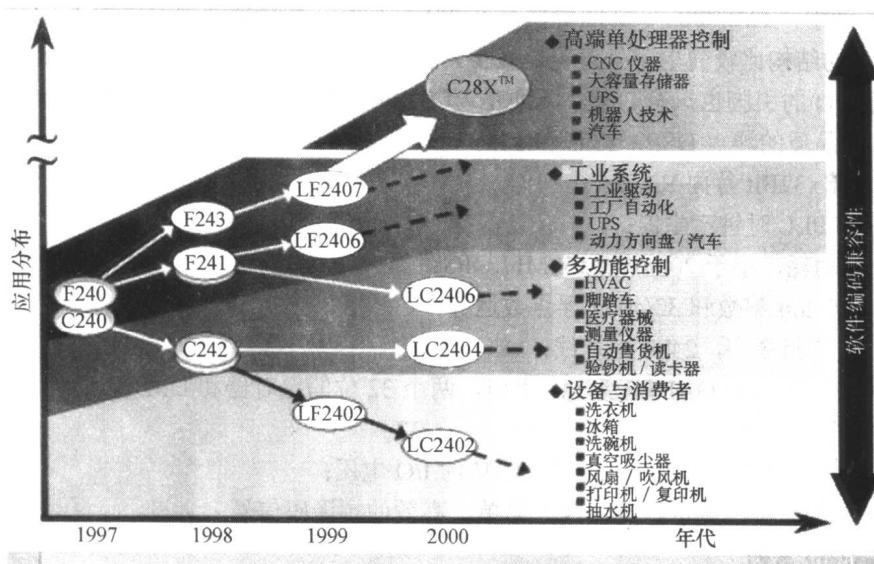


图1-4 C2000系列分布

种存储器。而 C54xx 没有提供 EMIF，所以只能使用静态存储器 SDRAM 和 EPROM。另外，C5000 系列一般都使用双电源供电，其 I/O 电压和核电压一般不同，而且型号也有差别。不过，TI 提供了全系列的 DC-DC 变换器可以解决 DSP 的电源问题。C5000 系列一般都提供 PGE 封装，便于 PCB 板的制作。C5000 系列是目前 TI DSP 的主流 DSP，它涵盖了从低档到中高档的应用领域，目前也是用户最多的系列。C5000 系列分布如图 1-5 所示。

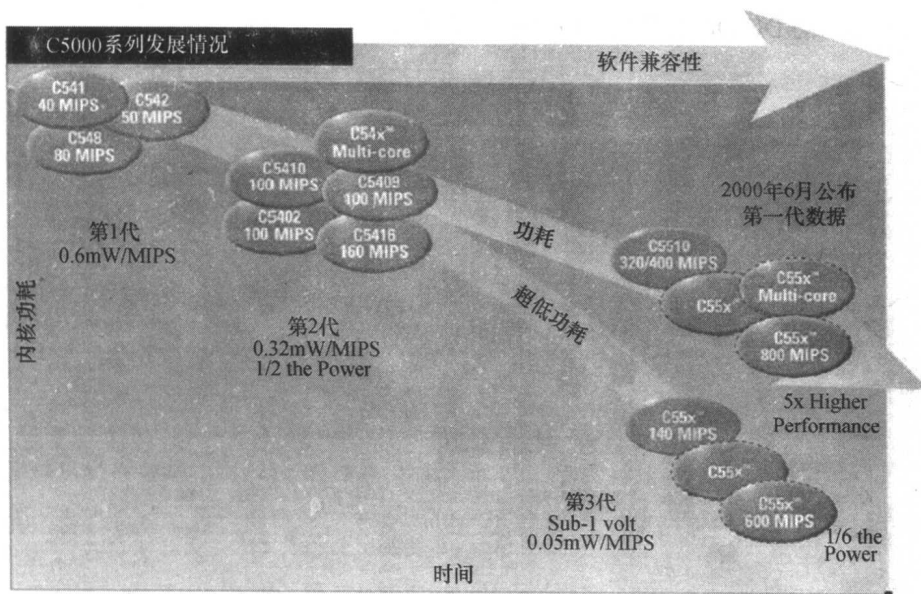


图1-5 C5000系列分布

### 3. C3X 系列

C3X 系列虽然不是 TI 的主流产品，但作为一个 32 位的低价位浮点 DSP，仍然被广



泛使用。其中，TMS320VC33 的价格在¥200 元左右，其最高处理速度为 150MFLOPS。C3X 系列的结构比较简单，外设也比较少，主要有同步串口、DMA 通道、定时器，能用于数字 I/O 的引脚也只有 2 条。下面是 VC33 的简单情况。

- (1) 高品质的浮点 DSP，13ns 和 17ns 指令周期；
- (2) 34Kx32Bit 片内 RAM；
- (3) X5 PLL 时钟产生器；
- (4) 低功耗，小于 200mV@150MFLOPS；
- (5) 16/32bit 整数和 32/40bit 浮点数运算；
- (6) 32 位指令字，24bit 地址线；
- (7) 具有 BOOTLOADER，一个串口，两个 32 位的定时器和 DMA；
- (8) 8 个扩展精度寄存器，R0，R1，…，R7；
- (9) 双电压供电，1.8V 核电压和 3.3V 的 I/O 电压；
- (10) 支持 JTAG 调试标准，四个简单、高效的预译码信号。

#### 4. C6000 系列

C6000 系列是一个 32 位的高性能的 DSP 芯片，目前处理速度从 800MIPS~2400MIPS，而且还在不断提高。其中，C62xx 为定点系列，C67xx 和 C64xx 为浮点系列。同 C55xx 一样，C6000 也提供 EMIF 扩展存储器接口，方便用户使用各种外部扩展存储器，如 SBSRAM、SDRAM、SRAM、EPROM。C6000 提供的主要外设 McBPS 同步串口，HPI 并行接口，定时器，DMA 等。另外，在 C6000 的一些型号中还提供了 PCI 接口。C6000 几乎都只提供 BGA 球形封装，在 PCB 板制作时需要多层板，增加了开发和调试的难度。另外，C6000 的功耗较大，需要仔细考虑 DSP 与系统其他部分的电力分配，选择适当的 DC-DC 转换器。C6000 系列分布如图 1-6 所示。

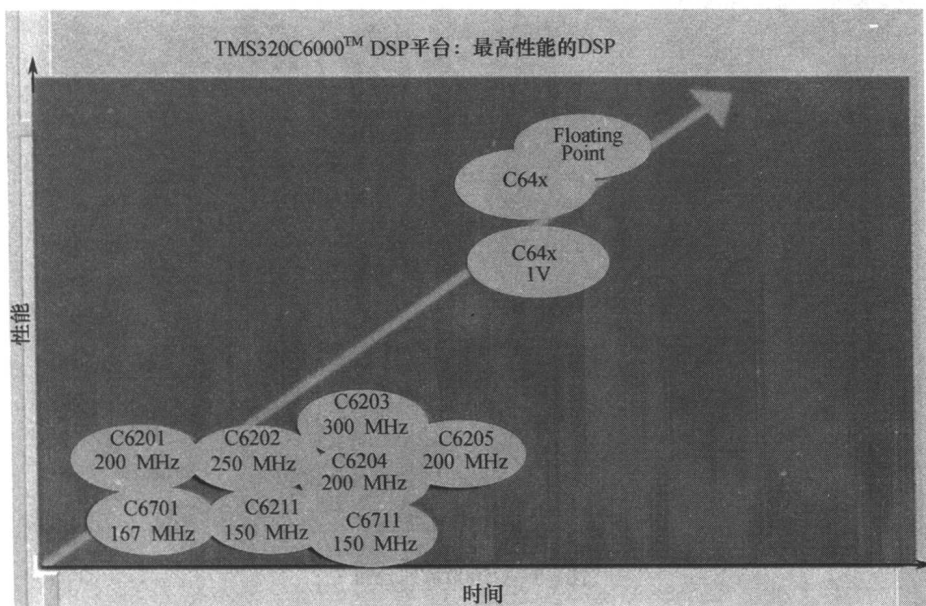


图1-6 C6000系列分布