普通高等教育“十二五”规划教材

TMS320LF240X系列
DSP原理及应用

杨风开　编著

華中科技大學出版社
http://www.hustp.com

普通高等教育“十二五”规划教材

TMS320LF240X系列

DSP原理及应用

杨风开 编著

华中科技大学出版社
中国·武汉

内 容 简 介

本书从教学和工程应用的角度出发，全面、系统地介绍了 TMS320LF240X 系列 DSP 的基本知识。主要内容包括硬件基本原理和结构、指令系统及汇编语言设计方法、片内外设的功能及应用等。全书注重结合工程应用的实际，阐述 TMS320LF240X 系列 DSP 硬件电路设计和指令系统设计的工程意义，并给出了大量的应用实例。读者通过原理的学习和应用实例的训练，可以从为什么要这样设计的角度来理解和记忆 TMS320LF240X 系列 DSP 的有关概念和知识。

本书既可以作为高等工科院校自动化、电气工程自动化以及机电一体化等电气类专业的教学用书，也可以供相关专业的工程技术人员参考。

图书在版编目(CIP)数据

TMS320LF240X 系列 DSP 原理及应用/杨风开　编著. —武汉：华中科技大学出版社，2012.2

ISBN 978-7-5609-7601-3

Ⅰ. T…　Ⅱ. 杨…　Ⅲ. 数字信号处理　Ⅳ. TN911.72

中国版本图书馆 CIP 数据核字(2011)第 271205 号

TMS320LF240X 系列 DSP 原理及应用　　杨风开　编著

策划编辑：谢燕群
责任编辑：陈元玉
封面设计：阮志翔
责任校对：朱　玢
责任监印：张正林
出版发行：华中科技大学出版社(中国·武汉)
武昌喻家山　　邮编：430074　　电话：(027)87557437
录　　排：禾木图文工作室
印　　刷：湖北通山金地印务有限公司
开　　本：787mm×1092mm　1/16
印　　张：18
字　　数：457 千字
版　　次：2012 年 2 月第 1 版第 1 次印刷
定　　价：32.00 元

前　言

控制类 DSP 作为嵌入式微控制器，片上设计了用于电动机数字控制的外设。由于具有强大的数字信号处理能力和较高的运算速度，DSP 在工业测量控制系统、智能仪器中得到广泛的应用。TMS320LF240X 系列 DSP 是目前仍在广泛应用、能够反映 DSP 结构原理的一种主流机型，掌握 TMS320LF240X 系列 DSP 的结构原理和应用方法，对于学习控制类 DSP 的一般性原理和应用方法，具有很大的帮助作用。本书全面介绍了 TMS320LF240X 系列 DSP 的原理和应用知识，其特点：一是内容全面、适合教学或自学，二是面向工程应用实际。

全书共分 7 章。第 1 章在介绍 TMS320LF240X 系列 DSP 的原理及基本结构后，着重介绍 CPU 及其相关电路。第 2 章先介绍寻址方式，然后按字母顺序逐条详细解释指令功能。第 3 章介绍汇编语言程序开发的过程和方法。第 4 章详细介绍片上 I/O 口原理和使用方法，并结合外部中断的使用，介绍中断功能的一般编程方法。第 5 章首先介绍电动机控制的基本知识和有关概念，然后结合电动机控制的实际，详细说明事件管理器各部分电路的功能。第 6 章首先阐述片上 A/D 转换电路的原理和使用方法，然后介绍扩张外部 D/A 器件的接口后编程方法。第 7 章在具体说明片上 SPI、SCI 串行通信接口原理和使用方法后，介绍 CAN 串行总线的结构。

本书作者自 1996 年以来长期从事控制类 DSP 的应用，全书基于作者对 TMS320LF240X 系列 DSP 的理解来编写，书中全部应用实例均由作者亲自设计、编程并上机调试通过。本书曾经以打印讲义的形式，先后在华中科技大学电气工程自动化专业 3 个年级的学生中试用，在教学实践中反复修改完善成书。

华中科技大学电气与电子工程学院“DSP 原理及应用”课程组何俊佳、熊健、戴轲、裴雪军等老师及许多试用讲义的学生，对本书的修改完善提出了大量的宝贵意见和建议，在此一并表示衷心的感谢。

鉴于编者水平有限，书中难免有不完善、不足之处，恳请广大读者批评指正。

编　者

2012 年 1 月

目　录

第 1 章　TMS320LF240X 硬件结构

TMS320LF240X(以下简称 LF240X)是 TI 公司目前流行的一个控制类 DSP 系列,主要用于电动机的数字控制。所谓控制类 DSP,其实也是一种单片式微型数字计算机(单片机)系统。因此,要了解控制类 DSP 的原理,首先必须了解计算机及单片机的原理。

1.1　概　　述

单片机即单片式微型数字计算机,是一种具有微型数字计算机功能的集成电路芯片。单片机主要用于各种仪器和现场控制设备中,作为控制系统的核心器件,所以单片机又称为微控制器(MCU,Micro Controller Unit)。

1.1.1　计算机原理

简单地说,微型数字计算机就是一种由一块芯片或多块芯片构成的复合数字电路,这种复合数字电路能够持续地根据给定的数字信号输入,进行算术逻辑运算,得到预期的数字信号输出。

1. 程序存储器

程序存储器的原理在《数字电子技术基础》教材中已经介绍过,以下再从使用的角度,简单回顾程序存储器的使用原理,如图 1.1.1 所示。

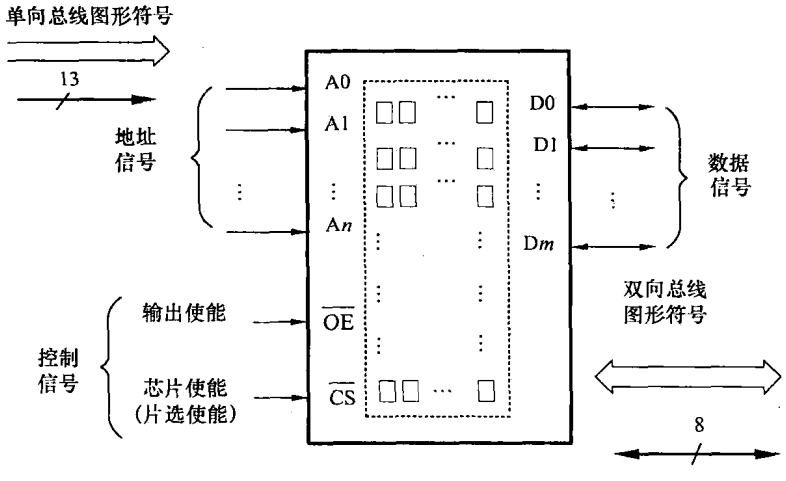

图 1.1.1　程序存储器使用原理

在程序存储器内部,有很多存储单元(如图 1.1.1 中虚线框内所示),每一个存储单元有一个地址编号,只有在有关控制信号(输出使能、片选使能)有效的前提下,才能根据地址信

号输出相应存储单元的数据。

每一个程序存储单元中存放的是一位位二进制数据，从物理状态来看，程序存储器中存放的是高、低电平信号。每个存储单元的数据位数称为程序存储器的字长。常见的程序存储器的字长有 8 位(1 字节)、16 位(1 个字)两种。也就是说，图 1.1.1 中的 m 通常为 8 或 16。

一块程序存储器芯片内部包含的存储单元数量称为程序存储器的容量。程序存储器的容量决定了图 1.1.1 中的 n 的值。常见的单片机程序存储器的容量有 8KB、16KB、32KB 和 64KB 等，因此，实际的程序存储器地址最高位(An)可以为 A12、A13、A14 和 A15 等。

在各种单片机系统中，常见的存储器容量单位是字节(B)和千字节(KB)，或者字(W)和千字(KW)。在个人计算机(PC，Personal Computer)中，存储单位还经常用到 MB(兆字节)、GB(吉字节)。必须注意，对存储器的单位来说，“千”和“兆”的概念有其特定的含义，1KB=2^{10}B=1024B，1MB=2^{10}KB=1024KB，1GB=1024MB。

存储在程序存储器中的二进制数据(高、低电平信号)才是真正的最终程序。通常所说的计算机软件，一般包括一定量的程序和数据。软件最终只有变成硬件的形式，也就是要变成高、低电平的物理信号，才能作为数字电路的输入信号，在硬件电路中发挥作用。

C 语言程序以及第 2 章即将学到的汇编语言程序如何变成这种二进制数据形式？这些二进制数据程序怎样才能烧写到程序存储器(EEPROM 或 FLASH)中去？这两个问题，将在第 3 章具体介绍。

目前大家只要明白：不管哪一种数字计算机系统，真正的最终程序都是存储在程序存储单元中的二进制数(高、低电平信号)；向程序存储器芯片发出控制信号和地址信号，就能得到相应存储单元的程序值信号输出。

为了便于画图，可以将若干根性质相同的信号线用一个总线图形符号来表示，例如，地址信号，通常是单向传送的，可以用图 1.1.1 左上角的单向总线图形符号来表示。控制信号(输出使能、片选使能)连线数量虽然较少，但一般也可用总线的图形符号来表示。数据信号通常是双向传送的，可以用图 1.1.1 右下角的双向总线图形符号来表示。有时为使图形更加简洁，也可以用一根稍粗一些的线条来表示总线，并且可以在总线旁边标注数字，表示实际连线数量。

通过《数字电子技术基础》一书的学习可知，目前的程序存储器一般为 EEPROM 或 FLASH；数据存储器一般为 SRAM，有时也直接将 SRAM 简称为 RAM。程序存储器和数据存储器的区别在于，程序存储器中的信号在系统停电后仍然能保持，数据存储器中的信号在系统停电后一般不能保持。

实际上，不管是 EEPROM(或 FLASH)还是 RAM，通电后内部存储的都是高、低电平的物理状态信号。如果将程序地址信号、控制信号加到 RAM 芯片上，则从 RAM 芯片中输出的物理状态信号就可以作为程序信号处理。例如，个人计算机启动的时候，先执行的是主板上 EEPROM 中的程序 BIOS(Based Input Output System)，执行 BIOS 程序，就可以将硬盘或光盘(统称外存储器)上的 Windows 等操作系统程序读到主板的 RAM 中(称为内部存储器或内存)，再执行 RAM 中的操作系统程序。鼠标、键盘和显示器等功能的实现都是运行内部 RAM 中操作系统程序的结果。用户要运行其他可执行程序，也要通过操作系统程序

(如 Windows)将硬盘上的用户可执行程序读到 RAM 中去执行。

因此,对于个人计算机等计算机,从使用的角度来说,程序存储器和数据存储器之间没有严格的界线。

2. 计算机的工作原理

在数字电路的学习中已经了解:

- 利用二极管、三极管(场效应管),可以构成基本的逻辑门电路;
- 利用基本的逻辑门电路可以构成各种复杂的数字逻辑运算和算术运算电路,也可以构成各种时序逻辑电路;
- 在程序存储器和数据存储器中存放的是表征二进制数值的高、低电平物理状态信号。

简单地说,计算机的工作原理就是:在时序逻辑电路控制下,利用各种复杂的逻辑和算术运算电路,不断地根据程序存储器输入信号及数据存储器、寄存器等输入信号,产生所需的数字信号输出。计算机的核心电路是中央处理单元(CPU,Central Progressing Unit),计算机的基本工作原理如图 1.1.2 所示。

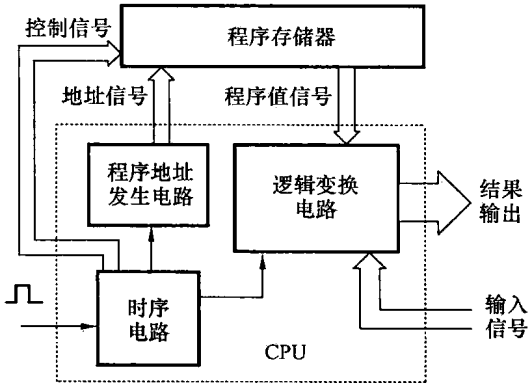

图 1.1.2 计算机的工作原理

从图 1.1.2 可以看出,CPU 内部程序地址发生电路首先向程序存储器发出地址信号,并通过时序电路向程序存储器发出控制信号(片选使能、输出使能),使得程序存储器相应地址单元的程序值信号输出到逻辑变换电路;然后内部时序控制电路控制逻辑变换电路对程序值信号和外部输入信号进行各种逻辑变换,从而输出结果。其基本过程如图 1.1.3 所示。

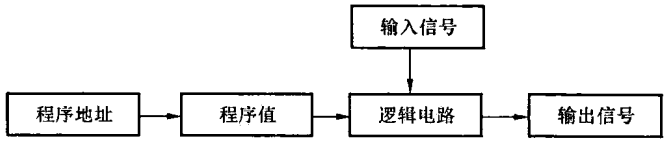

图 1.1.3 计算机工作原理的基本过程

持续不断地执行基本过程,就能够使 CPU 源源不断地根据程序存储器中的程序值信号和输入信号得到所需要的输出信号。各种算术运算电路归根结底也是由逻辑电路构成的。

每完成一次从程序值输入到结果输出的过程称为一次操作。完成一次操作的程序值称为一条指令。保存在程序存储器中的二进制数值称为指令的机器码。

指令所包括的数据位数与实际的计算机系统类型有关。在 DSP 系统中，有些指令由一个字(16 位)的二进制数值构成，有些指令由两个字的二进制数值构成。

一般来说，作为指令的二进制数值由两部分组成，前一部分称为操作码，后一部分称为操作数。操作码作为逻辑信号，没有数值含义，用于提供逻辑变换电路的控制信号，即设定逻辑变换电路进行何种操作；操作数具有数值含义，用于提供逻辑变换的原始数据或者数据的地址。

3. 指令的运行过程

每一次操作的过程又可以分为 4 个小过程，这 4 个小过程称为微操作。具体来说，程序指令的运行过程如图 1.1.4 所示，可以分为以下 4 个微操作。

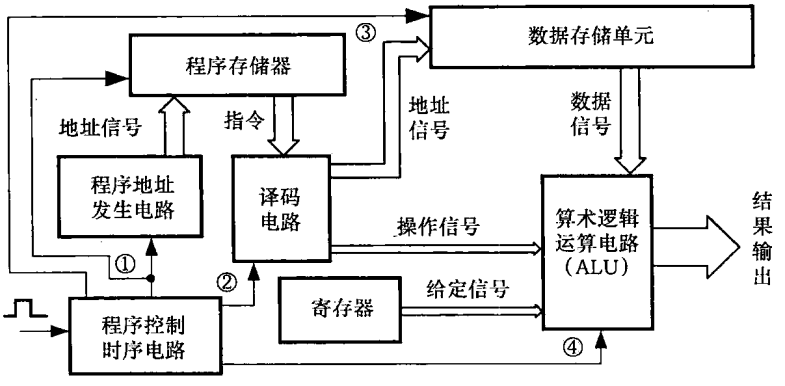

图 1.1.4　程序指令的运行过程

(1)取指(Fetch)。CPU 内部时序电路控制程序地址发生电路发出地址信号、控制信号，使得程序存储器输出程序值，这个程序值就是指令(机器码)。CPU 发出地址信号和控制信号，使程序存储器输出指令值的过程称为获取指令，简称取指。

(2)译码(Decode)。指令值一般包括操作码和操作数两部分，但是也有少数指令的操作数空缺，即只有操作码，没有操作数，CPU 必须对指令是否带有操作数进行甄别。

操作码并不是算术逻辑运算电路所需的实际操作信号(控制信号、设定信号)，而是操作信号的代码。CPU 必须把操作码通过逻辑变换电路变换成算术逻辑运算电路所需的实际操作信号。

对有操作数的指令，操作数有可能是参加算术逻辑运算的实际数值，也有可能是存放实际数值存储单元的地址信息。CPU 必须对操作数进行类型甄别，并把提供地址信息形式的操作数变换成实际的存储单元地址信号。

对指令操作码、操作数进行甄别、变换的过程称为指令的译码过程。实现译码功能的电路，称为译码电路。

(3)取操作数(Operand)。每条指令的实际功能不尽相同。当指令没有操作数时，CPU 跳过这一步，直接进行下一步。

当指令的操作数本身就是参与算术逻辑运算的原始数据时，CPU 直接将操作数信号输送到算术逻辑运算电路。

当指令操作数提供的是原始数据的存储单元地址时，CPU 必须发出对该存储单元操作的地址和控制信号，使该存储单元输出操作数信号给算术逻辑运算电路。

数据信号通过数据总线传送到 CPU 的过程称为取操作数。并不是所有指令都需要取操作数的过程。

(4)执行(Execute)。CPU 中的算术逻辑运算电路对来自译码电路的操作信号、数据存储单元的操作数信号(如果有)和对来自内部控制寄存器的给定信号进行算术逻辑运算，并将产生的结果信号输出到数据总线上。这一过程，称为指令的执行。

当指令不需要进行算术逻辑运算时，就不存在这一过程。

指令的操作过程习惯上也称为指令的执行过程。要注意指令的“执行”和微操作的“执行”在概念上的区别。每条指令的实际功能不同，有些指令的执行过程(操作)包括 4 个微操作过程(取指、译码、取操作数、执行)，有些指令的执行过程只包括两个微操作过程(取指、译码)，还有些指令的执行过程包括 3 个微操作过程(取指、译码、执行)。

CPU 内部电路可以分成两部分：一部分是算术逻辑运算电路，简称为运算器(ALU，Arithmetic Logic Unit)；另一部分是控制器(Control Unit)。ALU 能实现各种算术、逻辑运算功能。控制电路包括图 1.1.4 所示的程序控制时序电路、程序地址发生电路、译码电路、寄存器等，主要对程序的运行起控制作用。

4. 计算机的构成

总的来说，计算机系统由 CPU 和外部设备(简称外设)两大部分构成，如图 1.1.5 所示。

CPU 通过总线与各种外设相连。外设包括键盘、显示器、打印机等设备，也包括某些输入/输出(I/O)引脚。程序存储器和数据存储器通常也属于外设。

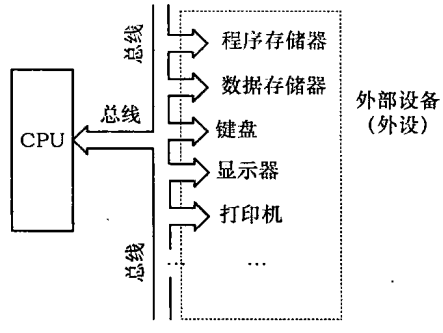

图 1.1.5 计算机系统构成

所谓总线，就是各种设备之间的多根公共连线。图 1.1.5 所示的总线符号代表的是一组总线，包括数据总线(DBUS，Data Bus)、地址总线(ABUS，Address Bus)和控制总线(CBUS，Control Bus)，习惯上称为“三总线”。普通个人计算机、单片机系统中，通常只有一组总线，DSP 等特殊计算机系统内部，则包括两组甚至三组总线，每组总线都由地址总线、控制总线、数据总线三种总线构成。

1.1.2 单片机结构

大型计算机、个人计算机的 CPU 和外设之间是分开的，是独立的芯片。单片机即单片式计算机，就是将 CPU 和某些外设功能集成在一个芯片上的计算机系统。

单片机的应用非常广泛，在日常生活中随处可见，例如，现在的彩电、空调、洗衣机等家用电器上一般都有单片机系统。有些外设，相对于计算机主机来说是外设，而从其自身内部来说又是独立的计算机系统，例如，个人计算机上除了主机外，还包括多个单片机系统，如键盘、鼠标、声卡、显卡和打印机等，都是由单片机系统组成的。

单片机的原理与普通计算机的原理基本相同，只不过普通计算机的外设功能都在 CPU 芯片以外的电路中，单片机的 CPU 芯片中则包括了部分外设功能，其原理结构如图 1.1.6 所示。

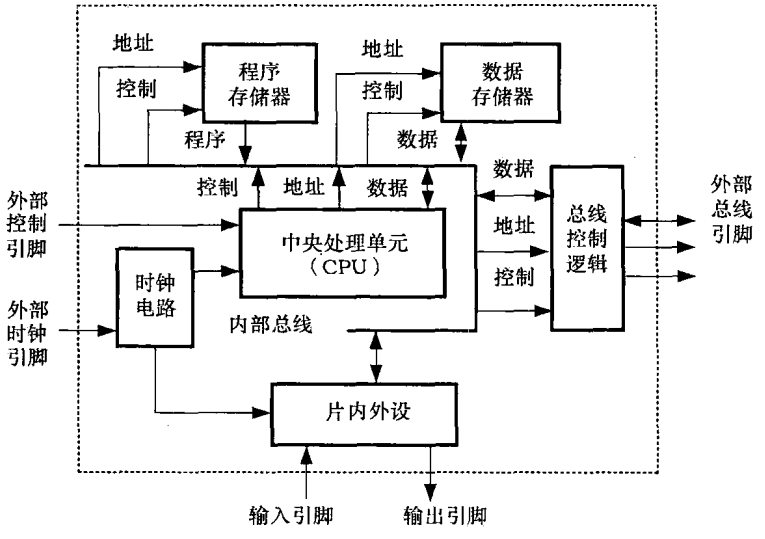

图 1.1.6　单片机的基本结构

CPU 是各种计算机的核心，通过内部总线与片内程序存储器、片内数据存储器及片内其他外设相连。

由于 CPU 内部控制信号线像人体的神经一样，遍布各个电路单元，每个电路单元的控制信号线都较少，并且各个电路单元的控制信号都存在区别，因此在论及 CPU 内部总线时，通常只包括地址总线和数据总线，而不把控制总线纳入总线范围。涉及外部总线时，则一般包括控制总线。实际上，外部总线中的控制总线的信号连线数量也很少，并且不同的设备使用的控制信号线也不尽相同。

为强调三总线的概念，图 1.1.6 中画出了 CPU 内部三总线。CPU 内部三总线通过总线控制逻辑电路与外部总线引脚相连，使得单片机能够通过外部总线引脚扩展各种其他外设。

CPU 的 ALU 的功能主要是利用硬件电路实现数字量的算术运算、逻辑运算、数据位处理和数据传送等操作。控制器是单片机的神经中枢，控制单片机的各个子系统完成所规定的各种操作任务。

时钟电路为 CPU 和外设电路提供必要的工作时钟，CPU 和片内外设按照外部控制引脚设定的方式工作。

比较图 1.1.6 和图 1.1.2 可以看出，对单片机来说，CPU 输出信号可能包括：数据存储器的地址、控制、数据信号，外部总线引脚的地址、控制、数据信号，外部(I/O)引脚的信号等；CPU 的输入信号可能包括：数据存储器的数据信号，外部总线引脚的数据信号，外部(I/O)引脚的输入信号等。

1.1.3 控制类 DSP

DSP 是一种单片机，是一种比较特殊的单片机。

1. 何谓 DSP

DSP(Digital Signal Processors)特指数字信号处理器。“数字信号处理”本身是一门专业课程，其英文缩写也可以为“DSP”，但是在专业领域，DSP 特指数字信号处理器。

2. 数字信号处理

从广义上说，只要是对数字信号进行处理(变化、提取信息等)，就是数字信号处理。从狭义上说，是指用特定的数学方法(FFT、数字滤波、卷积、相关等)对数字信号进行处理。

“数字信号处理”这门课程，研究的就是狭义意义上的“数字信号处理”。通常所说的“数字信号处理”也是狭义意义上的概念，即用特定的数学方法对数字信号进行处理。

3. 经典的数字信号处理方法

经典的数字信号处理方法有两个，一个是时域里的数字滤波，另一个是时域、频域的快速傅里叶变换(FFT)。

对于没有学过“数字信号处理”课程的读者，要理解 DSP 的某些部分有一定的难度。本书将通过实验详细介绍 DSP 数字滤波和 FFT 的实现方法及其实际应用，让大家掌握经典的数字信号处理的实现方法，理解数字信号处理的算法特点。

4. 数字信号处理算法特点

数字信号处理的算法有两个显著的特点，其一是包含众多的乘积累加($\sum A_i x_i$)计算，例如，在 FFT 计算中，通常包括数千次的乘积累加计算过程；其二是高速实时，例如，数字滤波时，必须实时地根据 A/D 转换得到输入数据，计算出输出数据，并通过 D/A 转换器输出。

因此，作为数字信号处理器的 DSP，必须具有适应数字信号处理算法特点的硬件设计和指令集功能。

5. DSP 与单片机

所谓单片机，就是指将 CPU 和外设功能集成在一块芯片上的计算机系统。DSP 符合单片机的定义，属于单片机中的一种，是一种特殊的单片机。其特殊性在于其硬件功能设计和指令集设计符合数字信号处理的算法特点。

DSP 的运算速度一般都很高，但是在某些方面的运算速度不一定比其他单片机的高，例如，做除法运算时，DSP 就没有优势。DSP 片内的外设很丰富，但这也不是其区别单片机的标志，很多 DSP 上有的外设功能，普通单片机上也有，而且各具特色。

区别 DSP 和普通单片机的标志是乘积累加类指令(有时也称加连乘指令)，即一条指令能够同时实现一个加法运算和一个乘法运算。凡是指令集当中具有乘积累加类指令(如 MAC 等)的，称为 DSP，否则称为单片机。

6. TI 公司 DSP 系列

生产 DSP 器件的厂家众多，TI 公司是占有市场份额最大的企业之一。目前 TI 公司得到最广泛应用的 DSP 器件有 TMS320C2000、TMS320C5000、TMS320C6000 等三大类，每

个类别都有繁多的系列和品种，新的品种层出不穷，更新的速度也非常快。TMS320C5000系列主要用于手机、调制解调器等图像处理领域，TMS320C6000系列主要用于视频、无线基站等无线通信领域，TMS320C2000系列为微控制器，主要用于数字控制领域。

7. LF240X系列

LF240X系列是TMS320C2000系列目前正在广泛应用的一个分支，是继TMS320C20X、TMS320C2X、TMS320C5X、TMS320C24X之后出现的一种性能优越、价格低廉的定点DSP芯片。

LF240X系列DSP包括LF2407、LF2406、LF2402等几种芯片，其中以LF2407芯片上的资源最多，应用最广泛。LF2406、LF2402为在LF2407的基础上删除部分功能而形成的芯片，因此，掌握了LF2407的应用，就掌握了LF240X系列DSP的应用。本书以LF2407为基本机型介绍DSP的原理和应用。

为了适应数字信号处理的算法特点，LF240X系列DSP设计了独特的CPU结构，使之具有各种DSP器件所共有的适合进行数字信号处理计算的特性。

LF240X是为电动机的数字化而设计的，其独特的外设事件管理器特别适合进行电动机的数字控制，这是LF240X作为DSP控制器所具有的特性。

学习LF240X系列DSP，应着重掌握其两种特性，一种是特别适合进行数字信号处理运算的特性，这种特性是各种DSP所共有的；另一种是其特别适合进行电动机的数字控制的特性，这种特性是该系列DSP所特有的。

1.2 CPU

DSP与普通计算机、单片机的区别，主要在于CPU。普通计算机、单片机的CPU一般由ALU、控制单元组成，乘法运算也由算术逻辑运算单元完成。

DSP的CPU与普通计算机、单片机的CPU相比较，具有以下几个特点。

(1)采用增强的改进哈佛总线结构和流水线作业方式，提高取程序指令和数据存取操作的速度。

(2) 为了提高乘法运算速度，增加了一个专门的乘法器。

(3) 为了适应数字信号处理算法的特点，增加了三个移位定标寄存器。

(4)为了增加寻址的灵活性和快捷性，增加了一个专门的辅助寄存器算术运算单元。

这些电路部分，使得DSP在指令集设计和应用上具有自身的特色。LF240X的CPU的结构如图1.2.1所示。以下介绍图1.2.1中主要部分的功能。

1.2.1 内部总线结构

所谓总线，就是与多个部件连接、能够供多个部件使用的一系列性质相同的连线。普通计算机、单片机通常只有一组总线，包括数据总线、地址总线和控制总线三种，习惯上称为三总线。

程序存储器和数据存储器各有其自己的地址，这种存储器结构称为哈佛结构。如果程

序存储器和数据存储器共用一套地址，则称为普林斯顿结构（也称为冯·诺依曼结构）。例如，MCS-96 型单片机和个人计算机，采用的是普林斯顿结构，MCS-51 单片机、各种 DSP 采用的是哈佛结构。

LF240X 系列 CPU 内部，采用的是增强的改进哈佛结构，这种结构具有多组总线，可以把程序总线和数据总线分开，能够在读取程序指令的同时，进行数据存、取的操作，具有很快的存储操作速度。

对外部存储器和 I/O 空间来说，DSP 与普通计算机、单片机一样，只有一组总线。对外部程序存储器、数据存储器和 I/O 空间的操作，必须分时进行。图 1.2.1 左上角的$\overline{IS}$、R/$\overline{W}$、$\overline{STB}$等，为外设控制总线；A15～A0 为外设地址总线；D15～D0 为外设数据总线。CPU 的外设控制总线、外设地址总线和外设数据总线，分别与芯片相应的引脚相连。外设地址总线、外设数据总线与内部相应总线，经 MUX（Multiplexer，多路选择器）相连。

习惯上，一般把外设总线处于芯片内部的部分，称为外设总线，而将外设总线处于芯片引脚以外的部分称为外部总线，两者其实并无本质区别。

CPU 内部的总线与外部总线不同，外部总线只有一组，内部总线由三组总线构成。每组内部总线主要包括三种总线，即地址总线、数据总线和控制总线。由于内部控制总线涉及的连线数量较少，并且因连接部件的不同而异，因此在介绍 CPU 内部总线时通常不列出。CPU 内部的三组总线如下。

1. 程序读总线

（1）程序读总线（PRDB，Program Read Bus）：用于 CPU 从程序存储器中读取指令值及各种表格数据等。

（2）程序地址总线（PAB，Program Address Bus）：用于提供对程序存储器操作的地址信号。

2. 数据读总线

（1）数据读总线（DRDB，Data Read Bus）：CPU 用它来从数据存储器读取数据。

（2）数据读地址总线（DRAB，Data Read Address Bus）：为 CPU 对数据存储器进行读操作提供地址信号。

3. 数据写总线

（1）数据写总线（DWEB，Data Write Bus）：CPU 用它来向数据存储器写数据。

（2）数据写地址总线（DWAB，Data Write Address Bus）：为 CPU 对数据存储器进行写操作提供地址信号。

LF240X 系列 CPU 可以通过内部三组并行总线访问内部程序和数据存储空间，同时进行三个不同的操作。由于总线工作是分离的，所以可以同时访问程序空间和数据空间。在一个给定的机器周期内，CPU 可以执行多达三次的并行存储器操作，从而 DSP 在内部存储器操作上能够具有很快的速度。

为了使图形简洁，图 1.2.1 中标注的“Data Bus”总线，实际上包括数据读总线和数据写总线两种（共 32 根信号线）。