

“计算机组成原理”学习指导

马 永 强

西 南 交 通 大 学
信息科学与技术学院

2015 年 11 月

“学习指导”总论

1. “计算机组成原理”学什么？

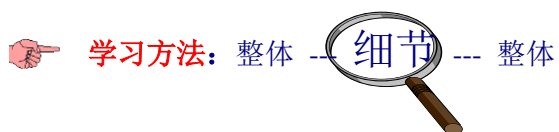
- 学计算机单机系统组成结构；
- 学计算机各组成部件内部的运行机制以及相关的基本理论；
- 学硬件分析和设计的基本技能和方法。

2. “计算机组成原理”的学习目的是什么？

- 理解和掌握计算机的基本组成、各部件内部的工作原理及逻辑实现；
- 提高硬件方面的专业素质和专业知识的学习能力；
- 为后续相关课程的学习及今后的科研工作奠定基础。

3. “计算机组成原理”怎么学？

- 抓住一条主线：以计算机完整地执行一条指令的基本过程，把各章的知识点串成一条线，由粗到细层层深入；
- 注意整体配合：以一条指令的执行过程为例，理解各部件之间的关系，理解信息的表述、存储、加工以及对存储和加工过程的控制原理，建立计算机整机概念；
- 弄清部件原理：各部件的每一功能是如何分解为一步步的微操作来实现的。

 **学习方法：**整体 --- 细节 --- 整体

4. 学习要求

- 勤动脑，多动手，认真做作业和实验；
- 课前预习，课后复习，不无故旷课；
- 及时答疑，及时反馈教学意见。

“学习指导”第 1 章（概论）

本章主要是概念性的知识。

应抓住最主要的两个概念：

1. 计算机系统的概念（硬件和软件的综合体）

由此概念，可以引申到：计算机的层次结构及虚拟机器概念、软硬件的逻辑等价、软硬件界面等概念。

2. 诺依曼体制的要点及存储程序控制的含义

要点有三个，其中最主要的一点就是存储程序工作方式，必须搞清其含义。由硬件的基本组成可以引申到计算机硬件的典型组成结构等知识。

应抓住计算机整体工作的基本过程：

初步了解计算机执行一条指令的基本过程、执行程序的过程。

本章主要概念可通过做练习自测题加以巩固。注意计算机主要指标中，MIPS 和 MFLOPS 的含义，字节的二进制位数和字的二进制位数哪个与机型有关。

附：本章学习要求

- ✓ 了解：存储程序概念
- ✓ 理解：CPU 和主机的含义
- ✓ 了解：总线概念和总线分时共享的特点
- ✓ 理解：计算机系统的含义
- ✓ 了解：计算机中主要性能指标（基本字长、数据通路宽度、存储容量等）

“学习指导”第 2 章 （数据的机器层次表示）

数据在计算机中用二进制编码形式表示，称为机器数。

机器数按什么编码规则来表示呢？是用原码还是补码还是 XX 码？原则上，机器数的表示方式不外乎是为了：

- （1）便于表示（规则别太怪太费解）
- （2）便于存储和传输（二进制位数合理）
- （3）便于处理（处理的算法要尽量简单，使得处理电路简单）

所以，计算机中最常用的是补码！

补码的编码规则大家都比较熟悉，但数学定义式觉得比较难理解，它主要是用于后续章节推倒运算规则。用模（计量器容量）的概念去理解补码定义式最方便，即，最高位的进位（溢出位）的值就是被切掉的值，就是取模。

补码还应特别注意：

- ① 最小的补码——很特殊，没有对应的原码和反码，可单独记；
- ② “变补”和“求补码”概念的不同；
- ③ 补码乘以 2（算术左移）和乘以 1/2（算术右移）的结果。

浮点数的规格化概念，真值 \longleftrightarrow 浮点数表示形式；

奇偶校验码的校验原理及校验位形成方法。

提高指导：

数据的表示，可以引申为一般的信息表示，甚至可进一步引申为对问题的描述。所以，本章和第 4 章中关于

“码制（数据表示）——运算规则（算法）——硬件实现”

与软件设计中

“问题描述（表示）——算法设计——程序实现”

道理是相通的，这是因为硬件和软件原本都是面向算法的！所以，学习硬件也培养计算思维能力，学习怎样把问题变成计算机可求解的形式。

附：本章学习要求

- ✓ 了解：无符号数与带符号数，真值和机器数等概念
- ✓ 掌握：原码、补码、反码表示法以及三种码制与真值之间的转换方法
- ✓ 掌握：定点数和浮点数的表示范围
- ✓ 理解：浮点数阶码的移码
- ✓ 了解：IEEE754 浮点数标准
- ✓ 了解：常见的字符编码方法（ASCII 码）、汉字国标码、区位码、机内码
- ✓ 掌握：8241 码和余 3 码
- ✓ 掌握：奇偶校验位及其形成方法

“学习指导”第 3 章（指令系统）

指令系统是软硬件的主要界面。

寻址方式要搞清其寻址原理和对应的寻址名称。在今后学习或科研中遇到的许多技术资料里（如单片机或 DSP 芯片资料等），一般都是说明提供哪些寻址方式而不会再详细解释寻址过程了！所以这部分内容很重要。

本章对“指令”部分只要求理解指令的基本格式、基本操作种类及扩展操作码方法，重点是学习各种寻址方式。关于各种指令的功能和用法将在“汇编语言程序设计”课（或“微机接口技术”课）中学习。

本课程与“汇编语言程序设计”课对寻址方式知识点的学习侧重点不同。前者注重各种寻址方式的实现过程，从而可以深刻理解不同指令、不同寻址方式的特点及时空效率（什么样的指令和什么样的寻址方式所需的二进制位数少、执行速度快），这些都是编写高效率程序所必备的知识；后者注重指令和寻址方式的应用，即如何用指令系统所提供的指令（包括各种寻址方式）编写程序完成指定的功能（也设计时空效率问题）。由此可见，本章对“汇编语言程序设计”及其它相关课程的学习是很有帮助的。

本章寻址方式实例（8086/8088 寻址方式）虽不是重点，但有利于本课程与汇编语言课及微机接口技术课的衔接，也有利于理解第 6 章模型机的工作原理；

本章给出的寻址方式的应用例子，是为了使大家对常见寻址方式的应用有个初步的了解，从而加深对寻址方式的印象，本课程对汇编语言编程部分不作要求。

最后，还应了解精简指令系统计算机（RISC）的基本思想和特点

附：本章学习要求

- ✓ 理解：指令的基本格式
- ✓ 理解：规整型指令和非规整型指令的特点
- ✓ 掌握：扩展操作码的方法
- ✓ 理解：编址单位和指令中地址码的位数与主存容量、最小寻址单位的关系
- ✓ 掌握：基本的数据寻址方式和有效地址 EA 的计算方法
- ✓ 了解：存储器堆栈的概念及堆栈的进、出栈操作
- ✓ 了解：RISC 的基本思想和特点

“学习指导”第 4 章 （数值的机器运算）

通过本章的学习，一方面要掌握运算器内部的机理机制，另一方面还要从中获得硬件分析设计基本技能，提高硬件方面的专业素质。前者是“鱼”，后者是“渔”。

定点数的加减乘除运算算式，是用来理解运算器的运算过程的，先学会算式计算再去理解运算过程学起来要容易些。

提高：注意把运算的规则（即运算方法、算法）与第 2 章所学的数据表示（如补码表示等）联系起来，与电路的实现联系起来。不同的数据表示会有不同复杂程度的处理算法，从而会有不同繁简的运算电路（对于软件来说会有不同繁简的程序段）。

积累硬件分析和设计的基本技能，可以从与门、与或门、三态门等器件的典型应用方法及寄存器的使用方法开始，再学习它们的组合应用方法。

微操作，是实现硬件各种功能的基础，应下功夫去理解。

微操是什么呢？微操作是“寄存器到寄存器的传送”；微操作是硬件一步可以完成的基本动作，该动作不存在数据通路上的障碍或冲突。数字系统的所有功能都是通过一序列的微操作来实现的。本章的每一运算步骤就是一个（或多个可同时执行的）微操作，要搞清每个微操作都需要些什么控制信号。第六章控制器将讨论这些控制信号怎么生成，时间先后顺序怎么控制等问题。

附：本章学习要求

- ✓ 掌握：定点补码加法和减法运算方法
- ✓ 理解：3 种溢出检测方法
- ✓ 理解：补码移位运算和常见的舍入操作方法
- ✓ 了解：串行加法器与并行加法器
- ✓ 理解：进位产生和进位传递
- ✓ 掌握：定点原码、补码乘法运算方法
- ✓ 掌握：定点原码、补码加减交替除法运算方法
- ✓ 理解：浮点加减乘除运算
- ✓ 理解：逻辑运算
- ✓ 了解：运算器的基本结构
- ✓ 理解：微操作，能用寄存器传送语言描述硬件的基本操作

“学习指导”第 5 章（存储系统和结构）

学习存储系统，从了解存储系统的层次结构开始。

注意 Cache、主存和辅存在计算机系统中的作用，存储体系中“Cache-主存”与“主存-辅存”这两级之间的异同点，以及虚拟存储器概念。

存储器的存储物理机制、存取方式是最基本的知识，应当了解；一些基本概念要搞清楚，比如：易失性、静态、动态、刷新、存取周期、位扩展、字扩展等。这些概念可以通过做配套的练习自测题来澄清。

对于半导体内存，其存储单元原理、存储芯片内部的存储阵列组织及译码方法要有一定的了解，但更重要的是要学会使用存储器芯片。对硬盘也要有所了解。

内存设计是本章的重点，关键是要掌握位扩展和字扩展方法、寻址逻辑（主要是片选信号逻辑电路）的设计方法以及与 CPU 之间读写控制信号的变换连接问题。第 7 章接口的寻址逻辑与本章的寻址逻辑原理相同，因此学好这部分内容也有利于后续内容的学习。

对于课程 A（4 学分），还应对高速缓存即虚拟存储器的机理机制有较深入的了解（如映射和抵制变换等），对磁表面存储器的磁记录格式有一定的了解。

此外，关于近年来提高主存读写速度的技术和提高存储器带宽的存储技术等，作一般性的了解。可以阅读些本课程所提供的课外阅读材料“内存和闪存”等。

附：本章学习要求

- ✓ 了解：存储器的分类方法和存储系统的层次
- ✓ 理解：主存储器的基本结构、存储单元和主存储器的主要技术指标
- ✓ 掌握：数据在主存中的存放方法
- ✓ 了解：半导体随机存储器（静态 RAM 和动态 RAM）的基本存储原理
- ✓ 理解：动态 RAM 的刷新
- ✓ 了解：RAM 芯片的基本结构
- ✓ 理解：各种不同类型的 ROM
- ✓ 掌握：主存储器容量的各种扩展方法
- ✓ 理解：主存储器和 CPU 的软连接
- ✓ 了解：Cache 存储系统和虚拟存储器的概念

“学习指导”第 6 章（中央处理器）

本章内容虽然较难，但只要抓住计算机执行指令的基本过程这一主线，由浅入深，逐层深入地理解和学习，便可事半功倍。

先回顾一下计算机执行一条指令的基本过程（即取指、分析指令和取数、执行），然后再以（模型机的）一条具体的指令为例，进一步搞清楚取指令、取操作数和执行这三个阶段的具体操作内容，并从中**理解计算机各部件的协调配合关系**（前面已分别学了运算器原理、寻址方式、内存工作原理等，**在此要把它们串到一起，回到“整体”**）。

再用 从寄存器到寄存器的信息传送（第 4 章学过）的微操作，来描述上述三个阶段中每一步的具体操作，并能指出每个微操作的实现分别需要哪些控制信号；

由于指令的执行过程是分阶段的，而每个阶段又分成若干动作（微操作）按先后顺序进行的，因此，如何用电信号（时序信号）的形式来表示计算机中的时间先后关系，成为控制器中的其中一个重要知识点，必须掌握；

在上述基础上，再来理解控制器是怎样根据指令码和时序信号等条件，来产生指令执行的每个阶段的每一步的控制信号的，从而控制计算机的各部件有条不紊、协调地工作；（内容包括组合逻辑控制器和微程序控制器）

至于指令执行过程中的另外两个 CPU 周期：INT 周期和 DMA 周期，在这一章学习时可以暂时先跳过，到第 7 章学了中断接口和 DMA 接口后便可轻而易举搞懂。

流水线技术及微处理器中的新技术作一般性了解，但对考研同学流水线技术较重要。

附：本章学习要求

- ✓ 理解：CPU 的功能和组成
- ✓ 理解：程序控制原理和控制器的基本组成
- ✓ 理解：时序系统中指令周期、机器周期的概念
- ✓ 了解：不同的控制方式（同步、异步、联合）
- ✓ 理解：指令执行的基本过程
- ✓ 掌握：取指周期的微操作序列（公共操作）
- ✓ 了解：组合逻辑控制器的设计方法
- ✓ 理解：微程序控制的基本概念
- ✓ 掌握：微指令编码法特点
- ✓ 理解：微程序控制器的组成和工作过程
- ✓ 了解：微程序入口地址和后继微地址的形成
- ✓ 了解：流水线技术

“学习指导”第 7 章 （输入输出系统）

本章的核心问题是 I/O 设备如何与主机进行信息交换。本章的主要学习内容是 I/O 接口技术及总线技术（主要是系统总线）。

本章给出的两个最简单输入输出接口的例子（补充例子），其原理是理解所有复杂接口原理的基础，应熟练掌握。接口中地址译码是很基本又很关键的，其译码原理与内存设计中的片选原理相似。

要搞清直接程序控制、程序中断、DMA 方式接口及通道的基本原理及各自的特点和应用场合。程序中断、DMA 方式着重理解基本概念和接口的一般原理，后续课“微机原理与接口技术”还要进一步地、深入地学习这部分的内容。

关于总线技术，理解其特点及总线数据通信的控制方式。

本章所涉及的一些概念，如：接口的编址方式、串行/并行接口、同步/异步接口、中断以及与中断相关的概念、DMA 等等，可通过做练习自测题（包括简答题）加以巩固。

附：本章学习要求

- ✓ 了解：接口的基本组成，接口和端口概念
- ✓ 了解：程序查询方式的特点和 workflow
- ✓ 理解：程序中断的基本概念，程序中断与调用子程序的区别
- ✓ 掌握：CPU 响应中断的条件和中断隐指令概念
- ✓ 理解：中断的各个过程
- ✓ 了解：DMA 方式的特点和 DMA 接口的组成
- ✓ 理解：DMA 传送方法和 DMA 传送过程。
- ✓ 了解：通道控制方式和通道控制的类型
- ✓ 了解：总线控制