

18-19学年秋季学期计算概论课后辅导讲义Lecture 5

陈婉雯

2018 年 11 月 3 日

1 本讲主题

- 冯诺依曼结构
- 计算机硬件介绍

学习编程语言，首先需要了解计算机硬件组成结构，因为编程语言编写的程序，最终需要在计算机中运行。计算机与程序间属共生关系，二者相互依存，互利共赢，没有程序，计算机如同没有生命一堆废铜烂铁，没有计算机，程序像是无用武之地的英雄，失去存在的意义。所以，了解计算机硬件组成结构是学习编程语言的基础。

2 冯诺依曼结构

2.1 存储程序的工作方式

任何要计算机完成的工作要先被编写成程序，然后将程序和原始数据送入主存并启动执行。一旦程序被启动，计算机应在不需操作人员干预之下，自动完成逐条取出指令和执行指令的任务。为了完成这种工作方式，我们需要：

主存（存放程序、数据）、自动逐条取出指令的部件、具体执行指令的部件、程序和原始数据输入计算机的部件、输出运算结果的部件。

2.2 基本部件

存储器：存放数据、指令；形式上没有区别，但计算机可以区分数据和指令

控制器：自动取出指令来执行

运算器：基本算术运算、逻辑运算、附加运算

通信：输入设备、输出设备

计算机内部以二进制表示指令和数据，每条指令由操作码和地址码两部分组成。操作码指出操作类型，地址码指出操作数的地址。一串指令组成程序。

2.3 CPU

中央处理器（CPU，Central Processing Unit），是一块超大规模集成电路，是计算机的运算和控制核心，其作用类似于人脑，负责对数据的加工处理和各种控制信号的下发。功能主要是解释执

行计算机指令来完成计算。CPU从逻辑上可以划分成3个模块，分别是控制单元、运算单元和存储单元，这三部分由CPU内部总线连接起来。

- 控制单元控制单元是整个CPU的指挥控制中心，由指令寄存器IR(Instruction Register)、指令译码器ID(Instruction Decoder)和操作控制器OC(Operation Controller)等，对协调整个电脑有序工作极为重要。它根据用户预先编好的程序，依次从存储器中取出各条指令，放在指令寄存器IR中，通过指令译码(分析)确定应该进行什么操作，然后通过操作控制器OC，按确定的时序，向相应的部件发出微操作控制信号。操作控制器OC中主要包括节拍脉冲发生器、控制矩阵、时钟脉冲发生器、复位电路和启停电路等控制逻辑。说通俗一点就是控制器读取完指令后就会协调把数据在存储器上的地址通知给运算器从而运算器可以取数据然后控制器再根据指令通知运算器在做什么样的运算、然后运算器就可以得出结果、而生成的结果需要由控制器操作保存到存储器上（控制器知道存储器上哪些单元是空闲的）这一切的操作都是在计算器、运算器和存储器三者之间操作的。
- 运算单元可以执行算术运算(包括加减乘数等基本运算及其附加运算)和逻辑运算(包括移位、逻辑测试或两个值比较)。相对控制单元而言，运算器接受控制单元的命令而进行动作，即运算单元所进行的全部操作都是由控制单元发出的控制信号来指挥的，所以它是执行部件。
- 存储单元包括CPU片内“缓存”和“寄存器”组，是CPU中暂时存放数据的地方，里面保存着那些等待处理的数据，或已经处理过的数据，CPU访问寄存器所用的时间要比访问内存的时间短。采用寄存器，可以减少CPU访问内存的次数，从而提高了CPU的工作速度。但因为受到芯片面积和集成度所限，寄存器组的容量不可能很大。寄存器组可分为专用寄存器和通用寄存器。专用寄存器的作用是固定的，分别寄存相应的数据。而通用寄存器用途广泛并可由程序员规定其用途，通用寄存器的数目因微处理器而异。

再具体看CPU内部的结构和工作原理。CPU内部主要结构有算术逻辑运算单元（ALU，Arithmetic Logic Unit）、程序计数器（PC，Program Counter）、寄存器组、高速缓冲存储器（Cache）以及用于传输数据、控制及状态的总线（Bus）。计算机从开机开始，不停地从程序计数器中获取指令、分析指令、执行指令，然后更新程序计数器，指向下一条指令。CPU常见的操作包括：

- 加载：如从主存复制一个字节或一个字到寄存器，用于后续运算；
- 存储：如从寄存器赋值一个字节或一个字到主存；
- 操作：把两个寄存器的内容复制到ALU进行算术运算，并将结果存放到另一个寄存器中；
- 从指令本身中抽取一个字，并将这个字复制到程序计数器，完成更新。

在计算机执行程序前，会将数据、指令存放在存储器中，每条指令、数据都有地址；指令按序存放。程序起始地址放在PC。计算机执行过程可以为：根据PC取指令- \rightarrow 指令译码- \rightarrow 取操作数- \rightarrow 执行指令- \rightarrow 回写结果- \rightarrow 修改PC值- \rightarrow 下一道指令

- 取指令：CPU的控制器从内存读取一条指令并放入指令寄存器。指令包括操作码和操作数地址。操作码就是汇编语言里的mov,add,jmp等符号码；操作数地址说明该指令需要的操作数所在的地方，是在内存里还是在CPU的内部寄存器里。

- 指令译码：指令寄存器中的指令经过译码，决定该指令应进行何种操作(就是指令里的操作码)、操作数在哪里(操作数的地址)。
- 执行指令：分两个阶段“取操作数”和“进行运算”。
- 修改指令计数器，决定下一条指令的地址。

2.4 计算机存储结构

顶层的存储器（寄存器）读写速度较高，但是空间较小。底层的读写（比如硬盘）速度较低，但是空间较大。

- 寄存器：寄存器是CPU内部用来存放数据的一些小型存储区域，用来暂时存放参与运算的数据和运算结果，寄存器虽然也能存储数据，但同内存等存储器相比，寄存器又有自己独有的用途，主要是CPU对内存中的数据进行处理时，往往先把数据取到寄存器中，而后再作处理，加快直接从内存读取指令和读写数据的速度。寄存器和cpu速度相当，空间比较小在kb级别。
- 高速缓存：高速缓冲存储器是存在于内存与CPU之间的存储器，容量比较小但速度比内存高得多，接近于CPU 的速度。高速缓存是加速读取速度的一个桥梁（工具）。CPU向内存读取数据时，首先查询缓存区是否有对应数据，如果有则直接读取，没有再从内存中读取。高速缓存中存储的都是内存中的数据，这部分数据是cpu访问比较频繁的部分。系统也会动态管理缓存中的数据，如果有数据访问频率降低到一定值，就从缓存中移除，而将内存中访问更加频发的数据替换进去。（高速缓存可以不只一级，可以有多级缓存区）高速缓存比寄存器要慢1倍左右，但是空间可以达到MB级别。
- 内存（主存）：计算机运行过程中的存储主力，用于存储指令（编译好的代码段），运行中的各个静态，动态，临时变量，外部文件的指针等等。前面提到的寄存器和高速缓存只是加速存储速度的中间部件，原始运行文件肯定都是先加入到内存中的，因此内存的大小决定了一个可运行程序的最大大小。如果内存大小是4G，一个程序装入内存需要大于4G空间的话，程序是无法运行的，当然实际是还要去掉操作系统占用的内存，因此连4G都无法满足的。（当然现在有了虚拟内存，也就是将硬盘的一部分映射为内存空间，将不经常访问的指令放到虚拟内存中，这样其实是支持大于4G的程序的，这里只是在假设没有虚拟内存的情况下的举例说明）。内存比缓存要慢10倍左右，但是空间可以达到GB级别，当前个人电脑一般都不小于4G。
- 硬盘：硬盘这个东西，优点是空间大，价格便宜，并且掉电数据不丢失。常常用来存储需要永久存储的文件。速度更慢，比内存要慢上万 千万倍级别，但是价格便宜，空间也很大。

计算机的存储设计的这么复杂，并不是设计者故意要搞得麻烦，而是要兼顾性能和价格综合考量的一个因素，如果未来有一个存储产品，能达到当前寄存器的读取性能，而又可以做的空间极大，并且价格和硬盘一样，或者说比硬盘还便宜，且数据是掉电不丢失的话，那么计算机的存储结构肯定就被简化了，直接就只有一个存储器。

2.5 I/O设备

I/O设备（Input/Output Device）是计算机用于输入和输出的设备，主要分为字符设备、块设备和网络通信设备。

- 字符设备（Character Device），又称为人机交互设备。用户通过这些设备实现与计算机系统的通信。它们大多以字符为单位传输数据，通信速度较慢。常见的有键盘、鼠标、显卡、显示器、打印机和扫描仪等，还有早期的卡片和纸带输入输出机。
- 块设备（Block Device），又称为外部存储器。用户通过这些设备实现程序和数据的长期保存。与字符设备相比，它们以块为单位进行传输，传输速度较快。常见的有磁盘、U盘、磁带和光盘等。
- 网络通信设备。这类设备主要有网卡、调制解调器等，主要用于与远程设备间的通信。这类设备的传输速度比字符设备快，比块设备慢。

I/O设备是计算机与外界通信的通道，通过控制器或适配器与I/O总线连接。控制器与适配器的主要区别在于封装方式，控制器是I/O设备或计算机主板上的芯片组，用于翻译CPU指令为设备可识别的控制信号。而适配器是一个独立的外部设备，用于I/O设备与计算机间数据协议的转换。二者在功能上是相同的，通过I/O总线，实现I/O设备与计算机间的为通信。

通俗来讲，I/O设备与人体的某些器官很类似。人每天工作生活，其实很像计算机的工作过程，从外界获取信息、加工，然后作出反应。计算的工作过程也是如此，通过输入设备获取信息，计算处理后，将结果输出。所以，计算的鼠标、键盘、扫描仪，类似与人体的眼、鼻、耳，来获取信息，打印机类似于人体的手脚，将信息加工后塑造出来。比如画家看到一幅美丽的山水景色，用双手创作出惊艳的图画。

2.6 总线

总线（Bus），又称为系统总线（System Bus），是计算机各种功能部件之间传送信息的公共通信干线，是由导线组成的传输线束。按照计算机所传输的信息种类，总线可以划分为数据总线（Data Bus）、地址总线（Address Bus）和控制总线（Control Bus），分别用来传输数据、数据地址和控制信号。

总线是一种内部结构，它是CPU、内存、I/O设备传递信息的公用通道，主机的各个部件通过总线相连接，外部设备通过相应的接口电路再与总线相连接，从而形成了计算机硬件系统。通俗来讲，如果将计算机主板比作一座城市，那么总线就是这座城市的交通线路，交通线路根据运行的不同类型的交通工具，可分为汽运公路、航运水路、铁路干线等。

总线的技术指标包括：

- 位宽总线的位宽指的是总线能同时传送的二进制数据的位数，或数据总线的位数，即32位、64位等总线宽度的概念。总线的位宽越宽，每秒钟数据传输率越大，总线的带宽越宽。例如，每条交通线路同时能够运行的交通工具的数量是有限的，比如拥有4个车道的公路，则同时并行运行的车辆最多为4辆。
- 带宽（总线数据传输速率）总线的带宽指的是单位时间内总线上传送的数据量，即每秒钟传送MB的最大稳态数据传输率。与总线密切相关的两个因素是总线的位宽和总线的工作频率，它们之间的关系： $\text{总线的带宽} = \text{总线的工作频率} \times \text{总线的位宽} / 8$ 或者 $\text{总线的带宽} = (\text{总线的位宽} / 8) / \text{总线周期}$
- 工作频率总线的工作时钟频率以MHZ为单位，工作频率越高，总线工作速度越快，总线带宽越宽。