1. 计算机系统概论

1.1 什么是计算机系统、计算机硬件和计算机软件？硬件和软件哪个更重要？

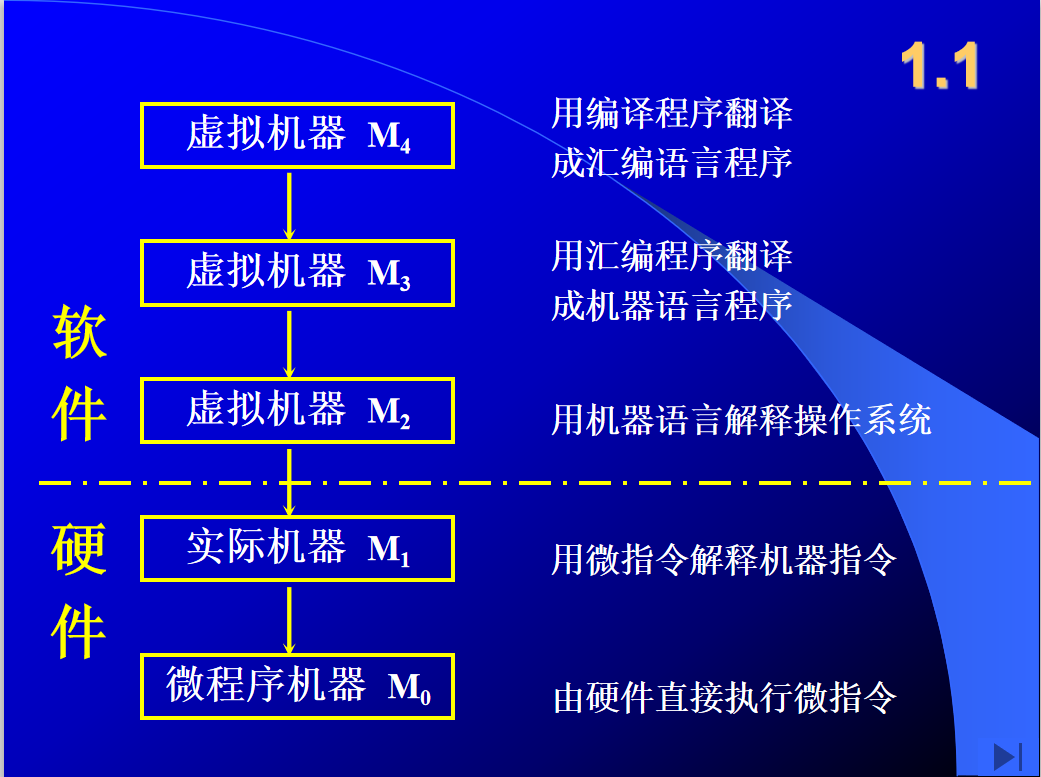
计算机系统：由硬件和软件两大部分组成。

计算机硬件：是指计算机的实体部分，它由看得见摸得着的各种电子元器件，各类光、电、机设备的实物组成，如主机、外部设备等。

计算机软件：看不见摸不着，由人们事先编制的具有各类特殊功能的程序组成。用来充分发挥硬件功能，提高机器工作效率，便于人们使用机器，指挥整个计算机硬件系统工作的程序集合。

计算机性能得好坏取决于“软”“硬”件功能的总和，所以软件和硬件都很重要。

1.2 如何理解计算机系统的层次结构？



机器语言

操作系统

微指令系统

汇编语言

高级语言

硬件层面分层

两个实体机

软件层面分层

三个虚拟机

1.3 说明高级语言、汇编语言和机器语言的差别及其联系？

机器语言：由0、1代码组成，是机器能够识别的一种语言

汇编语言：是面向机器的语言，它由一些特殊的符号表示指令

高级语言：是面向用户的语言，它是一种接近于数学的语言，直观，通用，与具体的机器无关

1.4 如何理解计算机组成和计算机体系结构

计算机体系结构：

指那些能够被程序员所见到的计算机系统的属性，即概念性的结构与功能特性。

“计算机系统的属性”通常是指用机器语言编程的程序员所看到的传统机器的属性，包括指令集、数据类型、存储器寻址技术、I/O机理等，大都属于抽象的属性。

计算机组成：

是指如何实现计算机体系结构所体现的属性，它包含了许多对程序员来说是透明的硬件细节

举例说明区别：

指令系统体现了机器的属性，这是属于计算机结构的问题。但是指令的实现，即如何取指令、分析指令、取操作数、运算、送结果等，这些都是属于计算机组成问题。

一台机器是否具备乘法指令的功能，这是一个结构问题；如何实现乘法指令（实现乘法指令）采用什么方式，则是一个组成问题。

1.5 冯·诺依曼计算的特点是什么？

特点：

①计算机由运算器、存储器、控制器、输入设备和输出设备五大部件组成。

②指令和数据以同等地位存放于存储器内，并可按地址寻访。

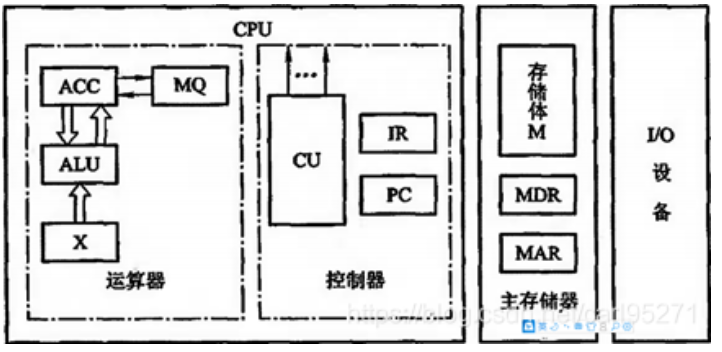
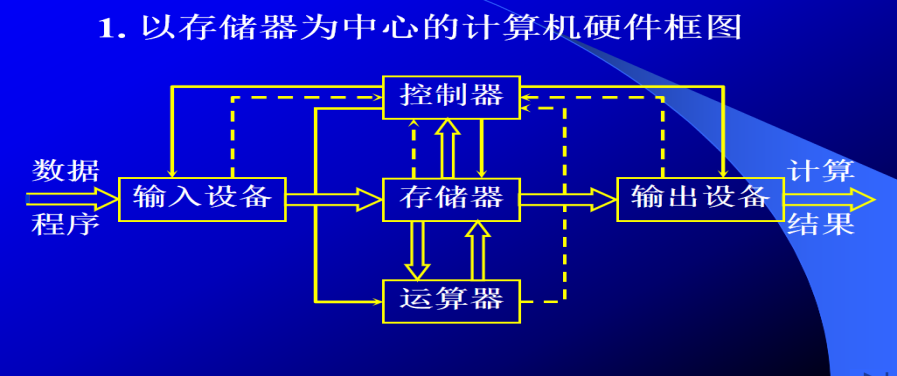
③指令和数据均以二进制数表示。

④指令由操作码和地址吗组成，操作码用来表示操作的性质，地址码用来表示操作数在存储器中的位置。

⑤指令在存储器内按顺序存放。通常，指令是顺序执行的，在特定条件下，可根据运算结果或根据设定的条件改变执行顺序。存储程序顺序执行（本质）

⑥机器以运算器为中心，输入输出设备与存储器间的数据传送通过运算器完成。

1.6 画出计算机硬件组成框图，说明各部件的作用及计算机硬件的主要技术指标。



①运算器：用来完成算数运算和逻辑运算，并将运算的中间结果暂存在运算器的寄存器内，包括3个寄存器和1个算逻运算单元

ALU：（Arithmetic Logic Unit），算逻部件，用来完成算数逻辑运算

ACC：（Accumulator），累加器，是寄存器，用来存放操作数，如被加数及和，被减数及差，乘积高位，被除数及系数。

MQ：（Multiplier-Quotient Register），乘商寄存器，在乘法中用来存放乘数及乘积低位，在除法中用来存放商

X：操作数寄存器，用来存放：加数、减数、被乘数、除数

控制器：用来控制、指挥程序和数据的输入、运行及处理运算结果

CU：（Control Unit），控制单元，用来解释存储器中的指令，并发出各种操作命令来执行指令

PC：（Program Counter），程序计数器，也是一个寄存器，用来存放当前欲执行指令的地址，它与主存的MAR之间有一条直接通路，且具有自动加1的功能，即可自动形成下一条指令的地址。

IR：（Instruction Register），指令寄存器，用来存放当前的指令，IR的内容来自主存的MDR。

②主存储器：用来存放数据和程序

MAR：（Memory Address Register），是存储地址寄存器，用来存放欲访问的存储单元的地址

MDR：（Memory Data Register），是存储器数据寄存器，用来存放从存储体某单元取出的代码或者准备往某存储单元存入的代码，其位数与存储字长相等。

③硬件的主要技术指标：

机器字长：指CPU一次能处理数据的位数，通常与CPU的寄存器位数有关

存储容量：包括主存储容量和辅存储容量，存放二进制代码的总数=存储单元个数 \* 存储字长

运算速度：

主频、Gibson法

MIPS（Million Instruction Per Second）每秒执行百万条指令

CPI（Cycle Per Instruction）执行一条指令所需时钟周期数

FLOPS（Floating Point Operation Per Second）每秒的浮点运算次数

1.7 解释概念：主机、CPU、主存、存储单元、存储元件、存储基元、存储元、存储字、存储字长、存储容量、机器字长、指令字长

主机：CPU和主存储器合起来又可称为主机

CPU：运算器和控制器合起来统称为中央处理器（CPU）

主存：是存储器子系统的一类，用来存放程序和数据，可以直接与CPU交换信息

存储单元：由若干个连续的存储元件构成，是组成存储体的单元

存储元件、存储基元、存储元：是构成存储单元的单位，每个存储（基）元只能寄存一位二进制代码0或1

存储字：一个存储单元可存储一串二进制代码，称这串二进制代码为一个存储字

存储字长：存储字所对应的二进制代码的个数称为存储字长

存储容量：存储单元个数 \* 存储字长，存储容量包括主存容量和辅存容量

机器字长：指CPU一次能处理数据的位数，通常与CPU寄存器位数有关

指令字长：一条指令在计算机中用一串二进制代码表示，此二进制代码的个数称为指令字长

1.8 解释英文代号：CPU、PC、IR、CU、ALU、ACC、MQ、X、MAR、MDR、I/O、MIPS、CPI、FLOPS

CPU：（Central Processing Unit），中央处理器

PC：（Program Counter），程序计数器（控制器中的一个寄存器）

IR：（Instruction Register），指令寄存器（控制器中的一个寄存器）

CU：（Control Unit），控制单元

ALU：（Arithmetic Logic Unit），算术逻辑运算单元

ACC：（Accumulator），累加器（运算器中的一个寄存器）

MQ：（Multiplier-Quotient Register），乘商寄存器（运算器中的一个寄存器）

X：操作数寄存器（运算器中的一个寄存器）

MAR：（Memory Address Register），是存储器地址寄存器

MDR：（Memory Data Register），是存储器数据寄存器

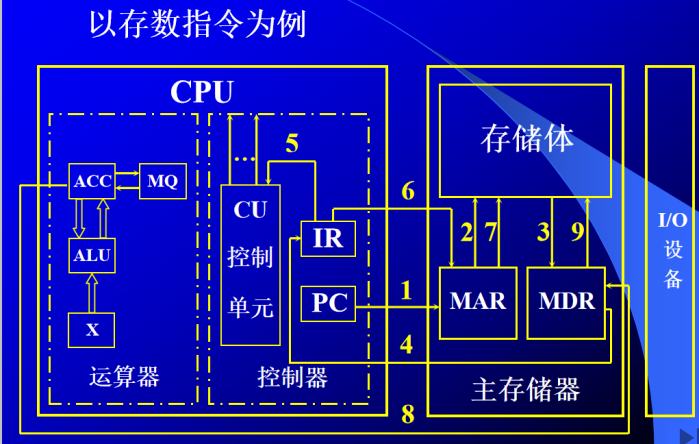
I/O：（Input / Output Equipment），输入输出设备

MIPS：（Million Instruction Per Second）每秒执行百万条指令

CPI：（Cycle Per Instruction）执行一条指令所需时钟周期数

FLOPS：（Floating Point Operation Per Second）每秒的浮点运算次数

1.9 画出主机框图，分别以存数指令“STA M”和加法指令“ADD M”（M均为主存地址）为例，在图中按序标出完成该指令（包括取指阶段）的信息流程。假设主存容量为256M\*32位，在指令字长、存储字长、机器字长相等的条件下，指出图中各寄存器的位数。

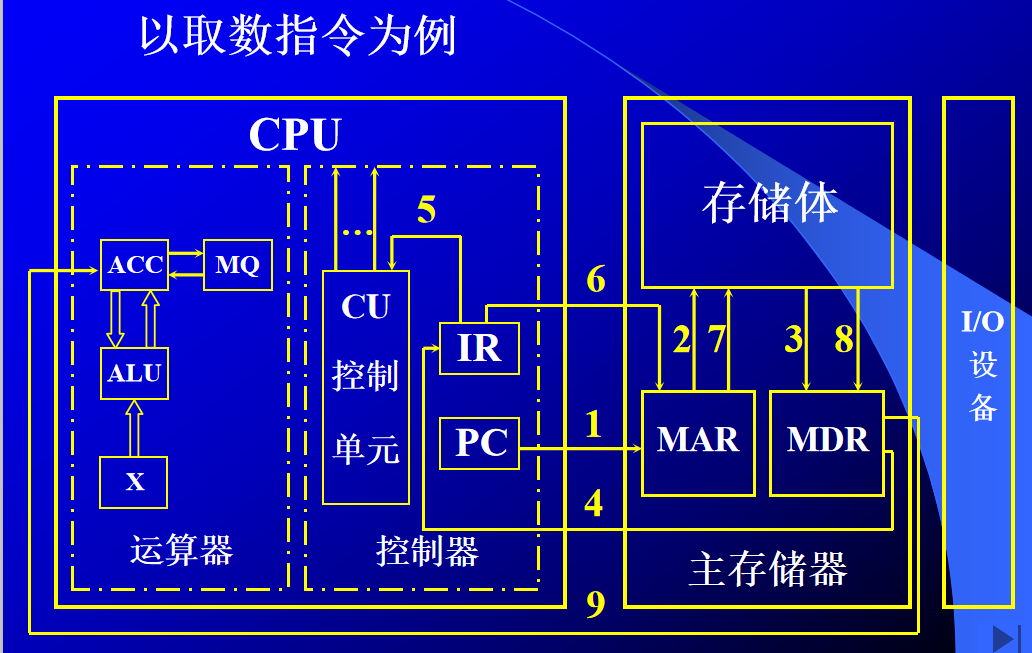


STA M指令：将ACC中的数据，存放进主存储器

取指令：PC→MAR，MAR→[M]，[M]→MDR，MDR→IR

分析指令：OP(IR)→CU

执行指令：Ad(IR)→MAR，ACC→MDR，MAR→[M]



ADD M指令：将地址为M的存储单元中的加数取出并送至运算器中，然后和存放在运算器的被加数通过ALU（算术逻辑单元）相加，结果仍放在运算器中。结合上面的主机框图，设运算器中ACC存放被加数，X存放加数，求和结果存放在ACC中  。故完成ADD M指令的信息流程为：

取指令：PC→MAR，MAR→[M]，[M]→MDR，MDR→IR

分析指令：OP(IR)→CU，

执行指令：

Ad(IR)→MAR，[M]→MDR，MDR→ACC

ACC→ALU，同时 X→ALU

ALU→ACC

假设主存容量256M\*32位，在指令字长、存储字长、机器字长相等的条件下，ACC、X、IR、MDR寄存器均为32位，256M=2^28，因此PC和MAR寄存器均为28位。

说明：主存容量=存储单元个数x 存储字长，MAR的位数反映了存储单元个数，此题存储单元个数为256M，即2^8\*2^20=2^28 (存储单元) 故而MAR为28位 ；MDR的位数反映了存储字长，故MDR的位数位32位

1.10（不会做，在网上截的图）





1.11 指令和数据都存于存储器中，计算机如何区分它们？

①通过不同的时间段来区分指令和数据，即在取指令阶段（或取指微程序）取出的为指令，在执行指令阶段（或相应微程序）取出的即为数据。

②通过地址来源区分，由PC提供存储单元地址的取出的是指令，由指令地址码部分提供存储单元地址的取出的是操作数。

1.12 什么是指令？什么是程序？

指令：认为输入计算机，由计算机识别并执行一步步操作的命令的形式称为指令。

程序：一系列指令的有序集合称为程序。