



南京大學
NANJING UNIVERSITY



计算机系统层次结构

南京大学

计算机科学与技术系

袁春风

email: cfyuan@nju.edu.cn

2015.6

开发和运行程序需什么支撑？

- 最早的程序开发很简单（怎样简单？）
 - 直接输入指令和数据，启动后把第一条指令地址送PC开始执行
- 用高级语言开发程序需要复杂的支撑环境（怎样的环境？）
 - 需要编辑器编写源程序
 - 需要一套翻译转换软件处理各类源程序
 - 编译方式：预处理程序、编译器、汇编器、链接器
 - 解释方式：解释程序
 - 需要一个可以执行程序的界面（环境）
 - GUI方式：图形用户界面
 - CUI方式：命令行用户界面

语言
处理
程序

语言处理系统 +

语言的运行时系统

操作系统内核

指令集体系结构

计算机硬件

人机
接口

+
操作
系统

支撑程序开发和运行的环境由系统软件提供

最重要的系统软件是操作系统和语言处理系统

语言处理系统运行在操作系统之上，操作系统利用指令管理硬件

早期计算机系统的层次

- 最早的计算机用机器语言编程

机器语言称为第一代程序设计语言 (First generation programming language , 1GL)

应用程序

指令集体系结构

计算机硬件

- 后来用汇编语言编程

汇编语言称为第二代程序设计语言 (Second generation programming language , 2GL)

应用程序

汇编程序

操作系统

指令集体系结构

计算机硬件

现代（传统）计算机系统的层次

- 现代计算机用高级语言编程

第三代程序设计语言（3GL）为过程式语言，编码时需要描述实现过程，即“如何做”。

第四代程序设计语言（4GL）为非过程化语言，编码时只需说明“做什么”，不需要描述具体的算法实现细节。

可以看出：语言的发展是一个不断“抽象”的过程，因而，相应的计算机系统也不断有新的层次出现

应用程序

语言处理系统

操作系统

指令集体系结构

计算机硬件

语言处理系统包括：各种语言处理程序（如编译、汇编、链接）、运行时系统（如库函数，调试、优化等功能）

操作系统包括人机交互界面、提供服务功能的内核例程

回顾：计算机系统抽象层的转换

程序执行结果

不仅取决于

算法、程序编写

而且取决于

语言处理系统

操作系统

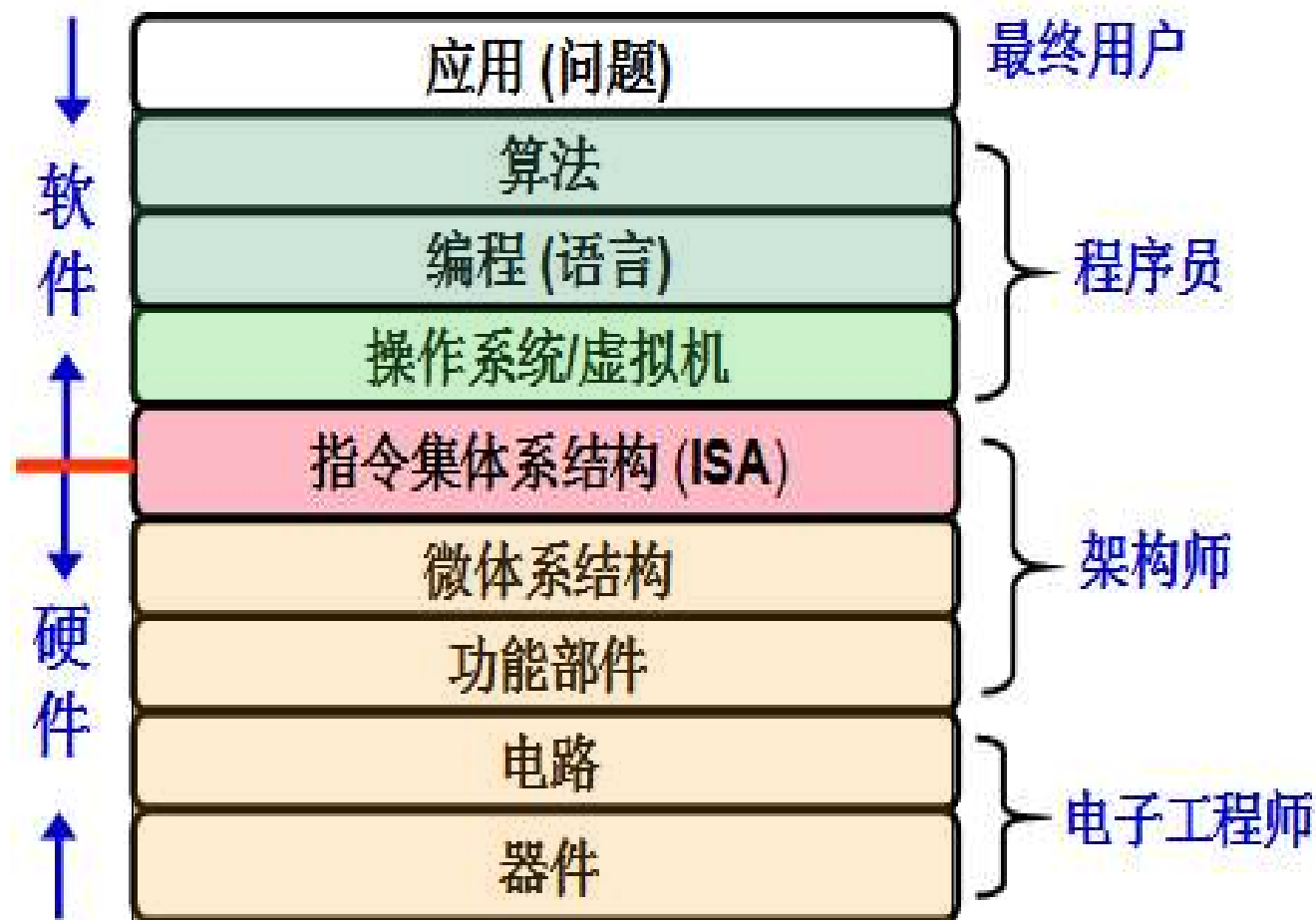
ISA

微体系结构

不同计算机课程
处于不同层次

必须将各层次关
联起来解决问题

功能转换：上层是下层的抽象，下层是上层的实现
底层为上层提供支撑环境！



最高层抽象就是点点鼠标、拖拖图标、敲敲键盘，但这背后有多少层转化啊！

计算机系统的不同用户

最终用户工作在由应用程序提供的最上面的抽象层

系统管理员工作在由操作系统提供的抽象层

应用程序员工作在由语言处理系统（**主要有编译器和汇编器**）的抽象层

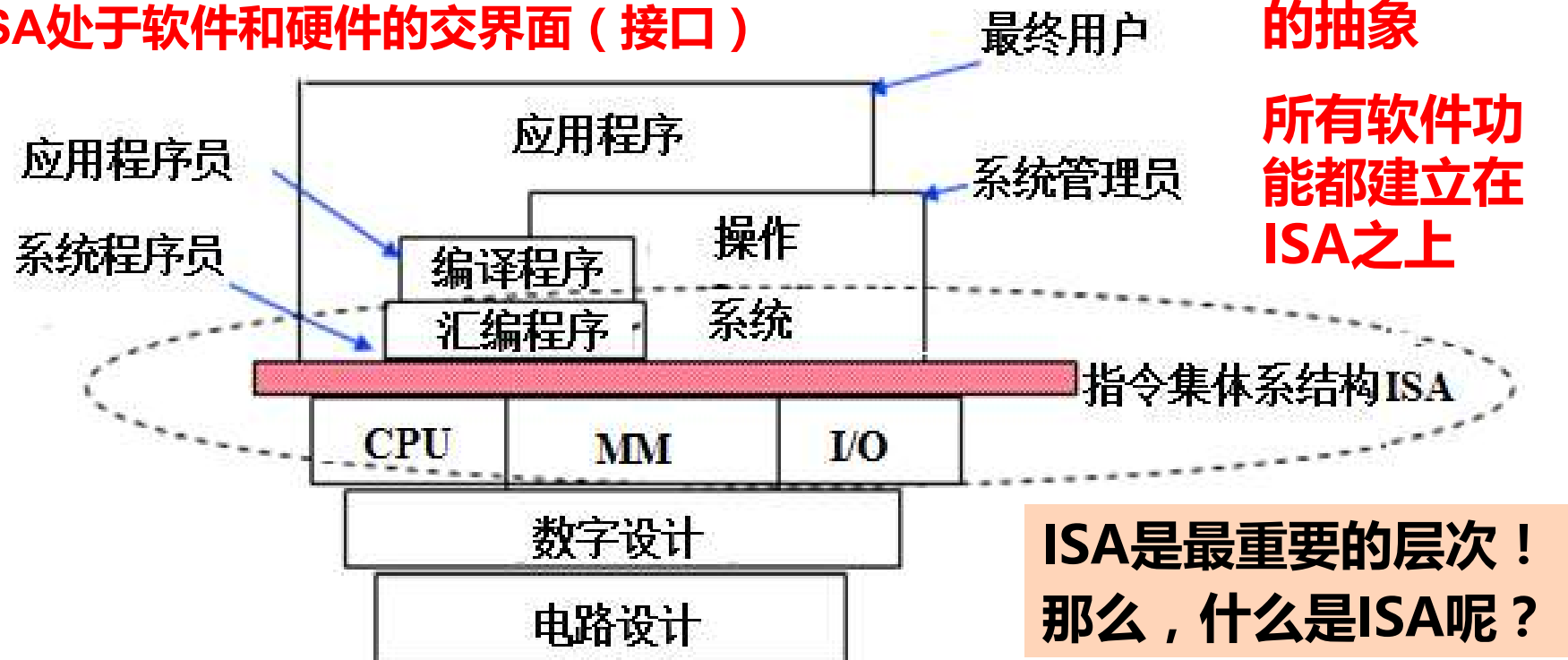
语言处理系统建立在**操作系统**之上

系统程序员（实现系统软件）工作在ISA层次，必须对ISA非常了解

编译器和汇编器的目标程序由**机器级代码**组成

操作系统通过指令直接对硬件进行编程控制

ISA处于软件和硬件的交界面（接口）



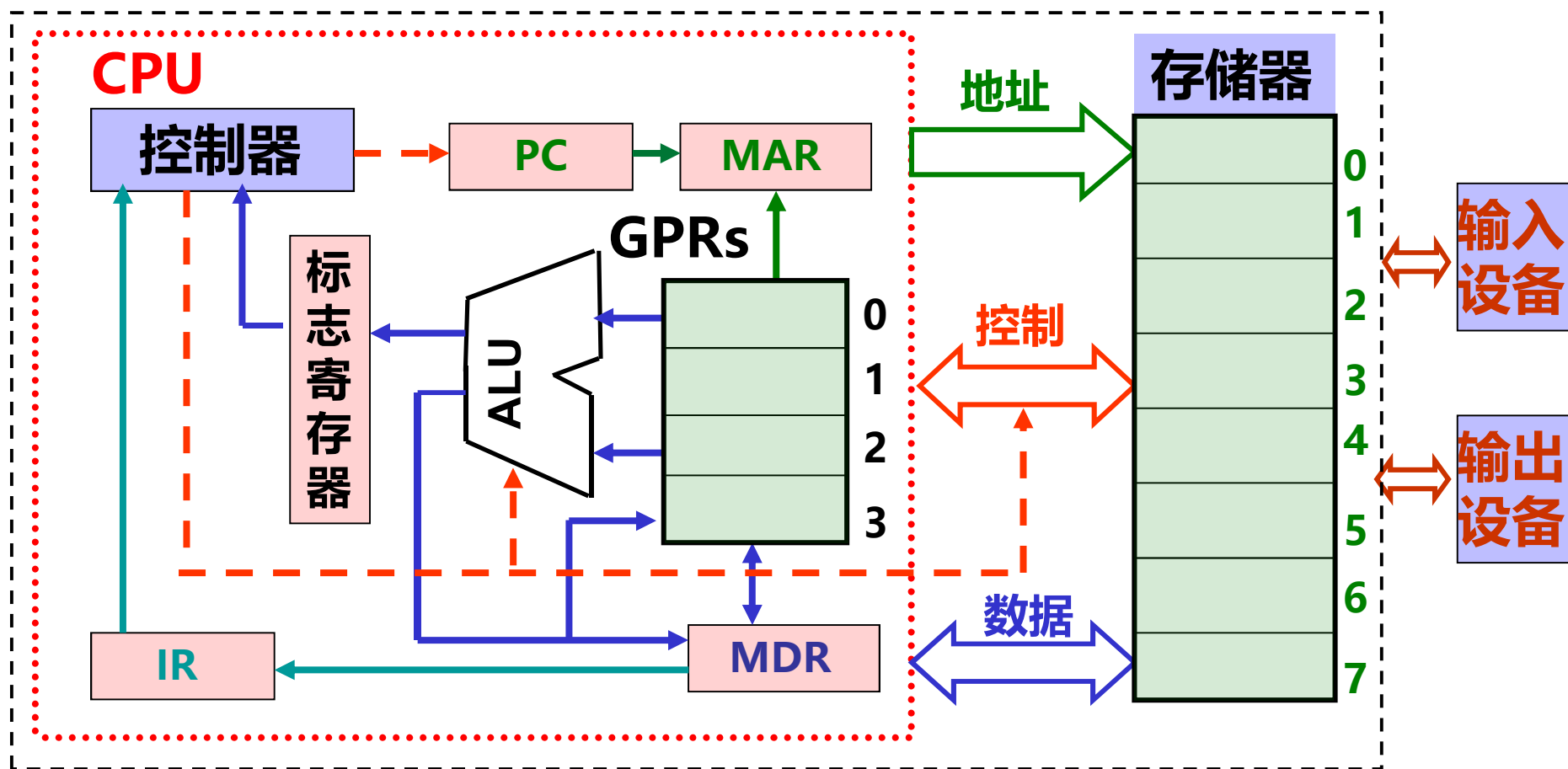
指令集体系结构（ISA）

- ISA指Instruction Set Architecture，即**指令集体系结构**，有时简称为**指令系统**
- ISA是一种规约（Specification），它规定了**如何使用硬件**
 - 可执行的指令的集合，包括**指令格式、操作种类以及每种操作对应的操作数的相应规定**；
 - 指令可以接受的**操作数的类型**；
 - 操作数所能存放的寄存器组的结构，包括每个**寄存器的名称、编号、长度和用途**；
 - 操作数所能存放的**存储空间的大小和编址方式**；
 - 操作数在存储空间存放时按照**大端还是小端方式存放**；
 - 指令获取操作数的方式，即**寻址方式**；
 - 指令执行过程的控制方式，包括**程序计数器（PC）、条件码定义等**。
- ISA在**通用**计算机系统中是必不可少的一个抽象层，Why？
 - 没有它，软件无法使用计算机硬件！
 - 没有它，一台计算机不能称为“通用计算机”

微体系结构

ISA和计算机组成（Organization，即MicroArchitecture）是何关系？

ISA和计算机组成（微结构）之间的关系



ISA是计算机组成的抽象

不同ISA规定的指令集不同，如，IA-32、MIPS、ARM等
计算机组成必须能够实现ISA规定的功能，如提供GPR、标志、运算电路等
同一种ISA可以有不同的计算机组成，如乘法指令可用ALU或乘法器实现