

本节主题

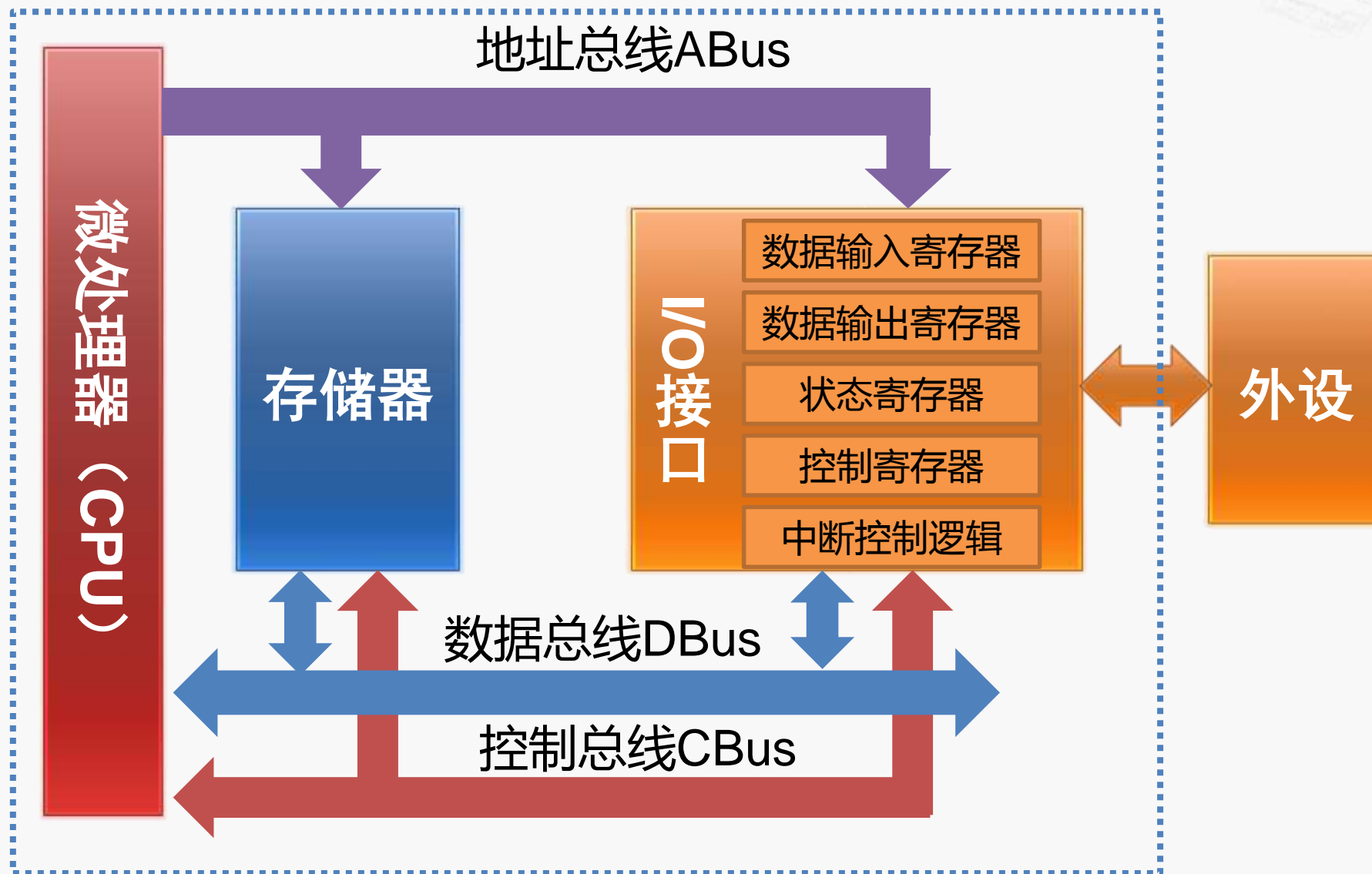


输入输出接口 的编址方式

北京大学·慕课
计算机组成
制作人：陆俊林



输入输出接口（I/O接口）的基本结构



I/O端口及其编址方式

I/O端口

- I/O接口内部包含一组称为I/O端口的寄存器
- 每个I/O端口都需有自己的端口地址（或称端口号），以便CPU访问

I/O端口的编址方式

- 在计算机系统中，如何编排I/O接口的端口地址？



常见的I/O端口编址方式



④ I/O端口和存储器分开编址

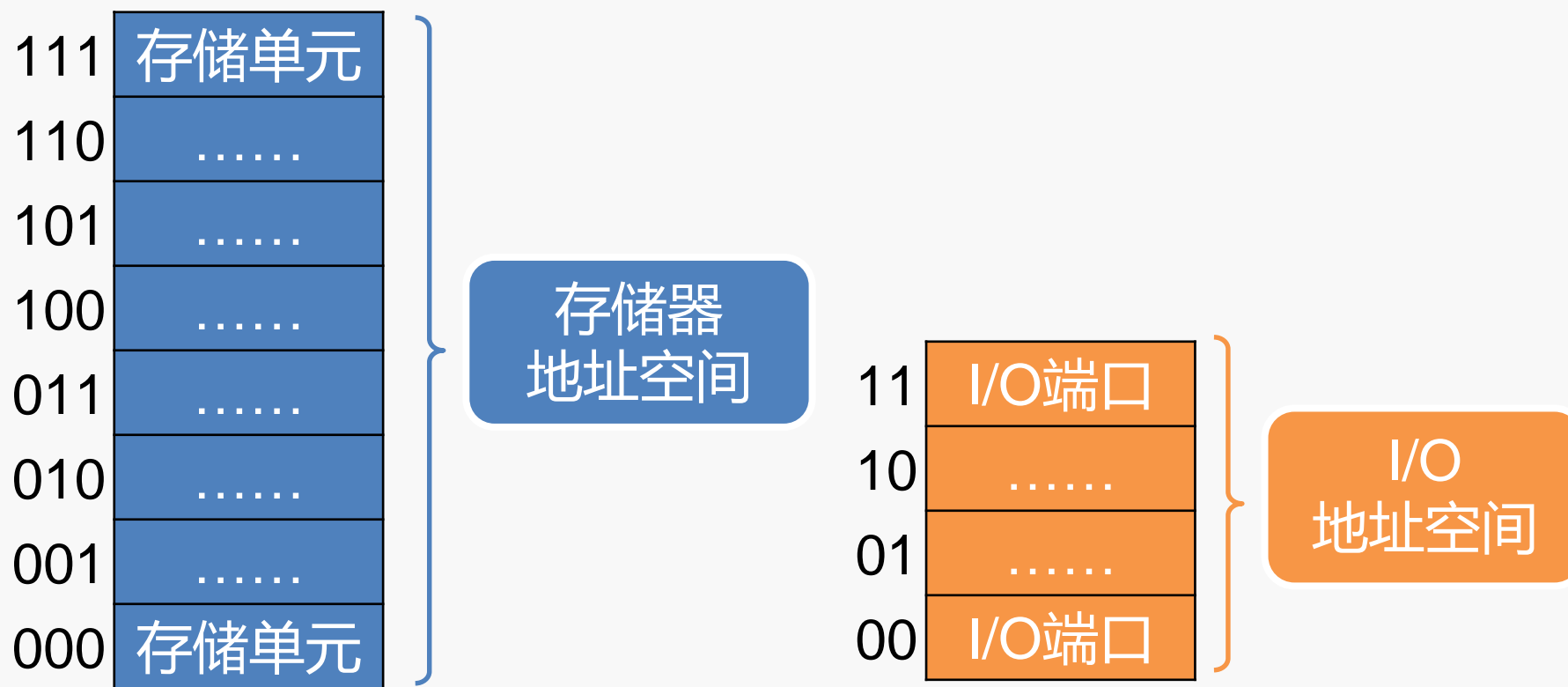
- I/O映像的I/O方式，I/O Mapped I/O
- x86体系结构采用该方式

④ I/O端口和存储器统一编址

- 存储器映像的I/O方式，Memory Mapped I/O
- ARM、MIPS、PowerPC等体系结构采用该方式

I/O端口和存储器分开编址

- 假设地址宽度为3，一个分开编址的地址空间划分示例



I/O指令说明

IN指令（输入）

- 格式：IN AC, PORT
- 操作：把外设端口的内容输入到AL或AX

OUT指令（输出）

- 格式：OUT PORT, AC
- 操作：把AL或AX的内容输出到外设端口



IN/OUT指令的寻址方式和示例

▶ 端口地址为0~255

- 直接寻址：用一个字节立即数指定端口地址
- 间接寻址：用DX的内容指定端口地址

▶ 端口地址大于255

- 间接寻址：用DX的内容指定端口地址

	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
IN：直接寻址	1 1 1 0 0 1 0 w	port
IN：间接寻址	1 1 1 0 1 1 0 w	
OUT：直接寻址	1 1 1 0 0 1 1 w	port
OUT：间接寻址	1 1 1 0 1 1 1 w	

IN AL, 80H

IN AX, 80H

OUT 80H, AL

OUT 80H, AX

MOV DX, 288

IN AL, DX

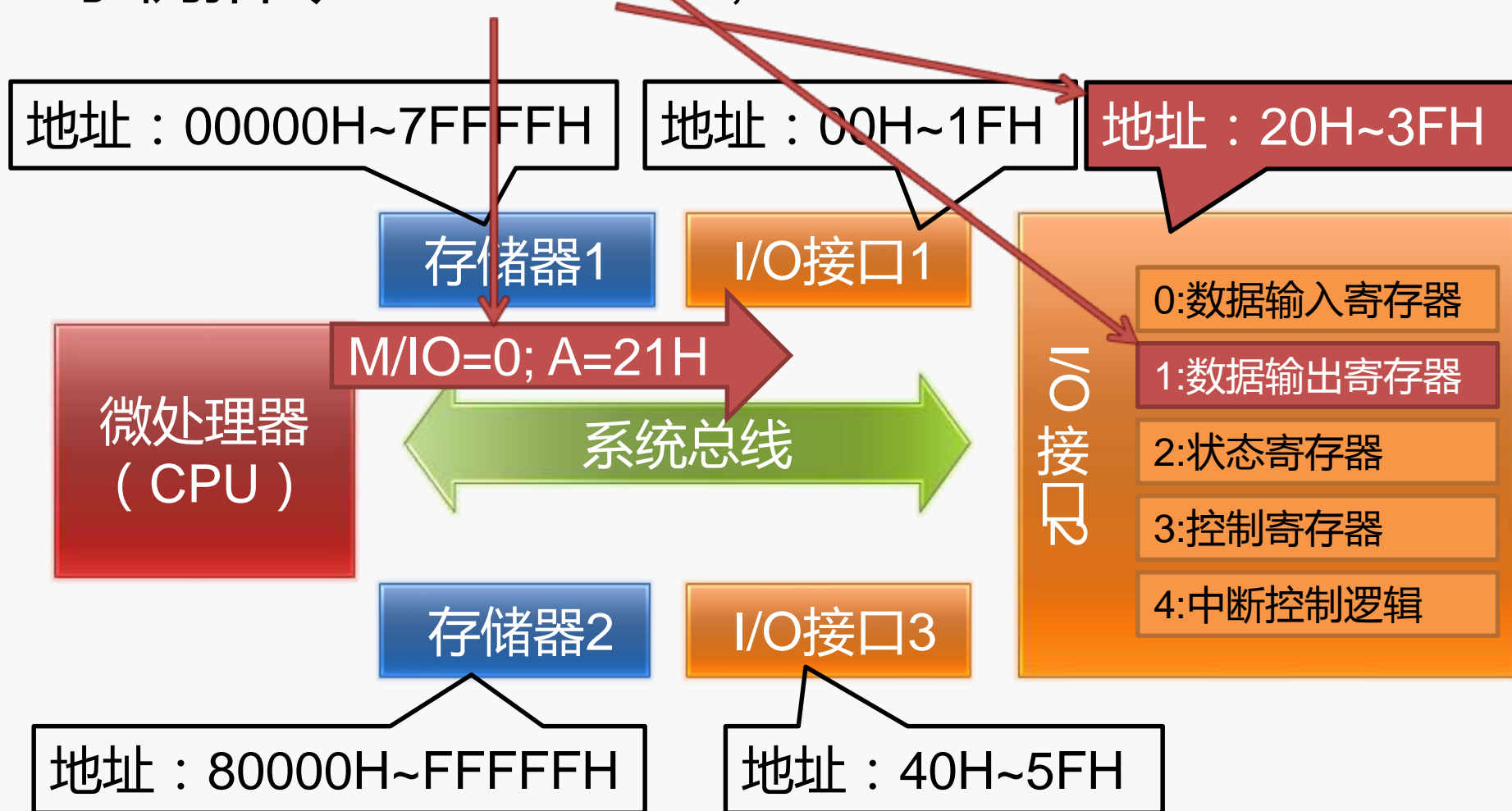
IN AX, DX

OUT DX, AL

OUT DX, AX

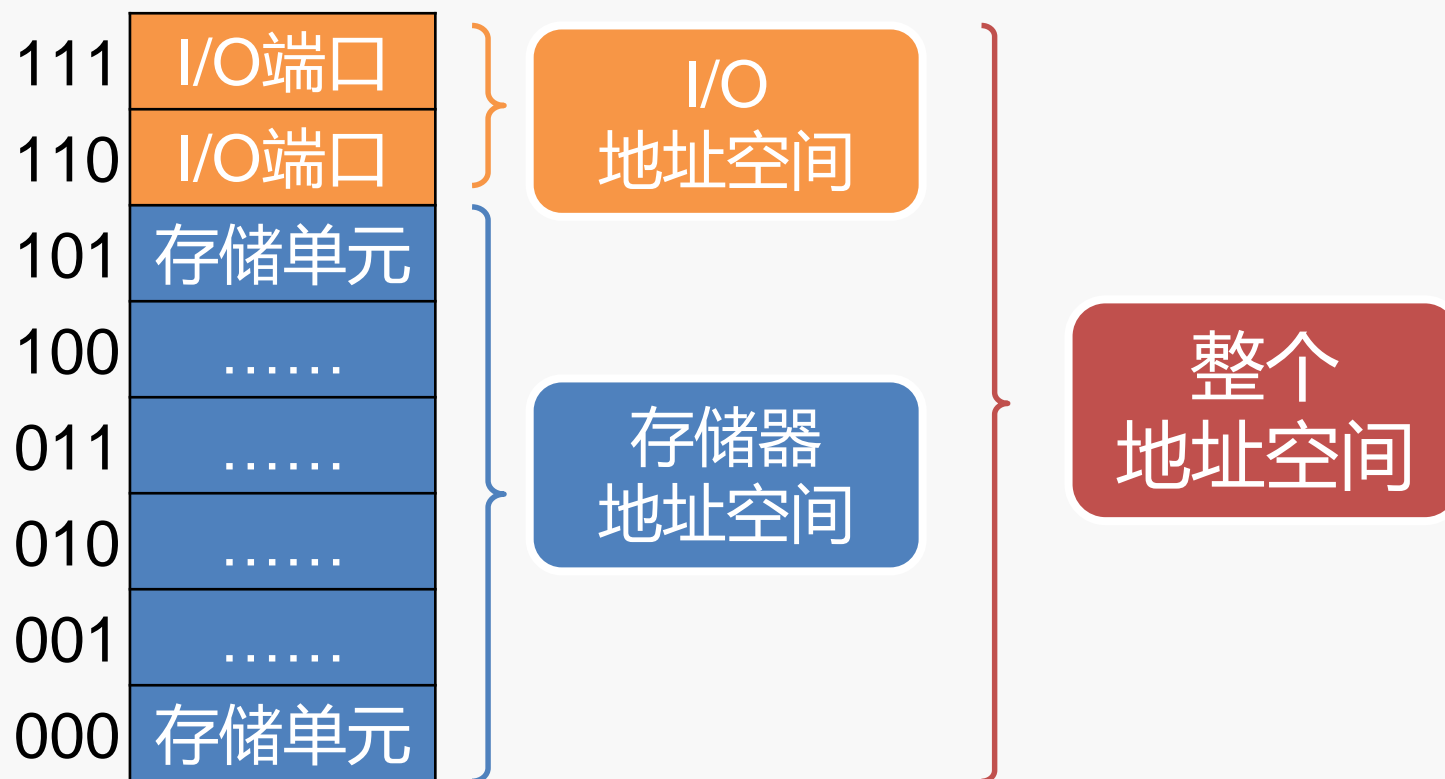
I/O指令的地址译码过程示例

示例指令：OUT 21H, AL

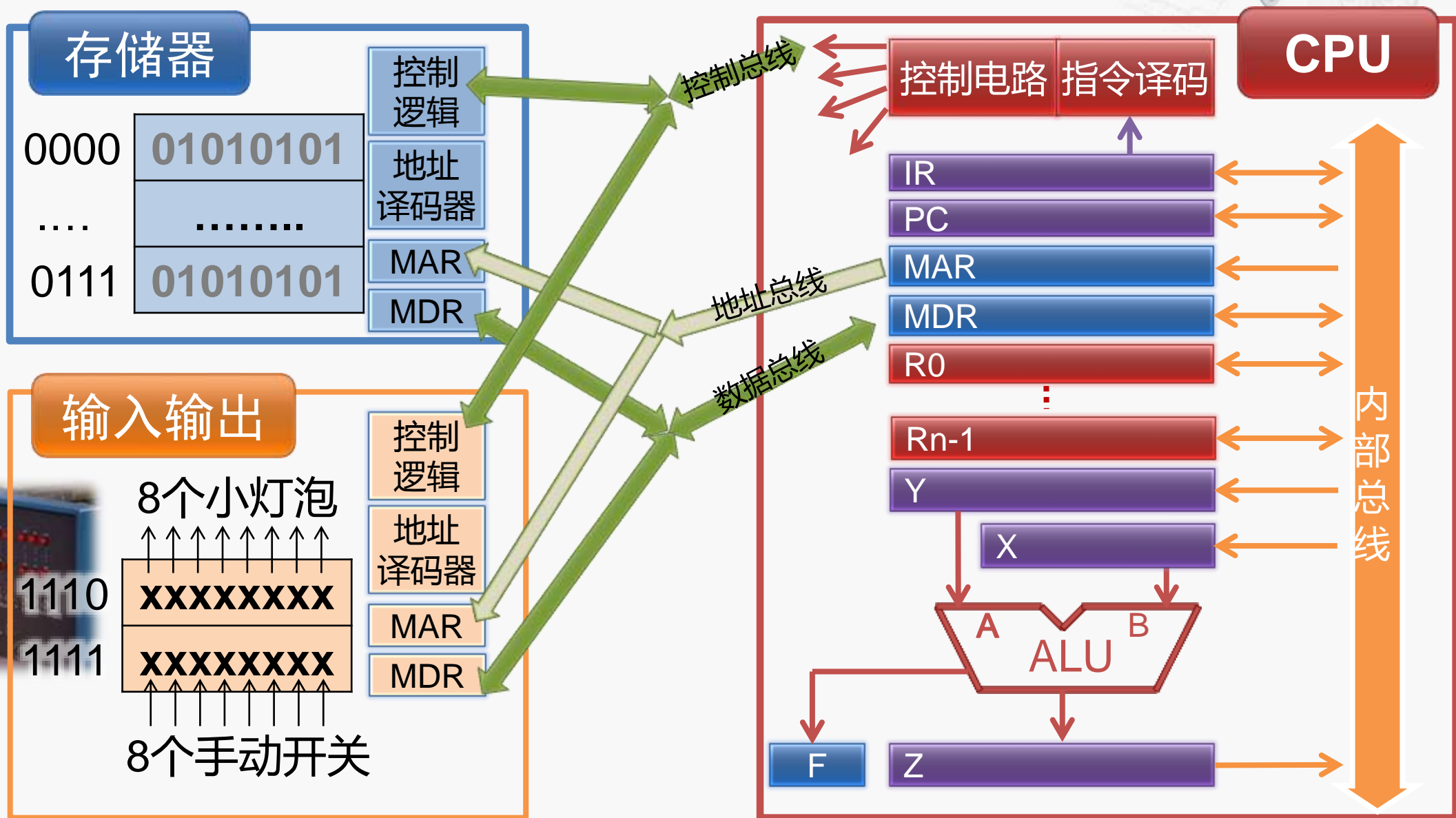


I/O端口和存储器统一编址

- 假设地址宽度为3，一个统一编址的地址空间划分示例



模型机采用了统一编址的方式



统一编址的特点



④ 优点

- 可以用访问存储器的指令来访问I/O端口，访问存储器的指令功能比较齐全，可以实现直接对I/O端口内的数据进行处理
- 可以将CPU中的I/O操作与访问存储器操作统一设计为一套控制逻辑，简化内部结构，同时减少CPU的引脚数目

④ 缺点

- 由于I/O端口占用了一部分存储器地址空间，因而使存储地址空间减小
- 由于利用访问存储器的指令来进行I/O操作，指令的长度通常比单独I/O指令要长，因而指令的执行时间也较长

分开编址的特点



▶ 优点

- I/O端口不占用存储器地址，不会减少用户的存储器地址空间
- I/O指令编码短，执行速度快
- I/O指令的地址码较短，地址译码方便
- 采用单独的I/O指令，使程序中I/O操作和其他操作层次清晰，便于理解

▶ 缺点

-

本节小结



输入输出接口 的编址方式

北京大学·慕课
计算机组成
制作人：陆俊林

