

哈尔滨工程大学

HARBIN ENGINEERING UNIVERSITY

计算机系统结构

计算机科学与技术学院

第3章 存储、中断、总线与输入/输出系统



- **理解**并行主存系统的组织形式、极限频宽和实际频宽的关系；
- **理解**通过使用主存的组成技术提高主存实际频宽的可能性、局限性和发展存储体系的必要性；
- **理解**中断源的分类和分级的原因。理解中断响应次序、中断处理次序和中断处理完成次序的概念。
- **熟练掌握**按中断处理次序设置各中断处理程序中中断屏蔽位的状态；
- **熟练掌握**并正确画出多种中断申请时，CPU执行程序时的状态转移过程示意图；
- **掌握**专用和非专用总线的定义、优缺点和适用场合。
- **理解**非专用总线的3种总线控制方式的总线分配过程、优缺点及所需增加的辅助控制总线数。
- **理解**总线采用同步和异步通信方式的通信过程、优缺点和适用场合。
- **理解**I/O系统的三种方式中处理机的两种处理方式。
- **掌握**通道方式I/O处理机进行输入/输出工作的全过程及通道处理机的工作原理。
- **掌握**字节多路、数组多路 and 选择三种通道各自采用的数据宽度及适用的场合。
- **掌握**通道处理机和I/O系统的流量计算和分析，字节多路通道流量的计算、通道周期的设计。
- **熟练掌握**并准确画出通道处理机响应和处理完各外设请求的时空示意图。
- **理解**外围处理机的工作原理以及它与通道处理机的不同。

3.1 存储系统的基本要求和并行主存系统

3.1.1 存储系统的基本要求

3.1.2 并行主存系统

3.2 中断系统

3.2.1 中断的分类和分级

3.2.2 中断的响应次序与处理次序

3.2.3 中断系统的软、硬件功能分配

3.3 总线系统

3.3.1 总线的分类

3.3.2 总线的控制方式

3.3.3 总线的通信技术

3.3.4 数据宽度与总线线数

3.4 输入/输出系统

3.4.1 输入/输出系统概述

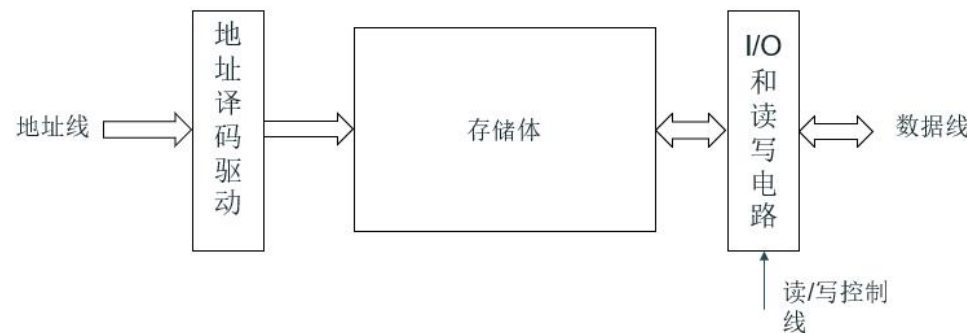
3.4.2 通道处理机的工作原理和流量设计

3.4.3 外围处理机

3.1.1 存储系统的基本要求



✓ 存储体



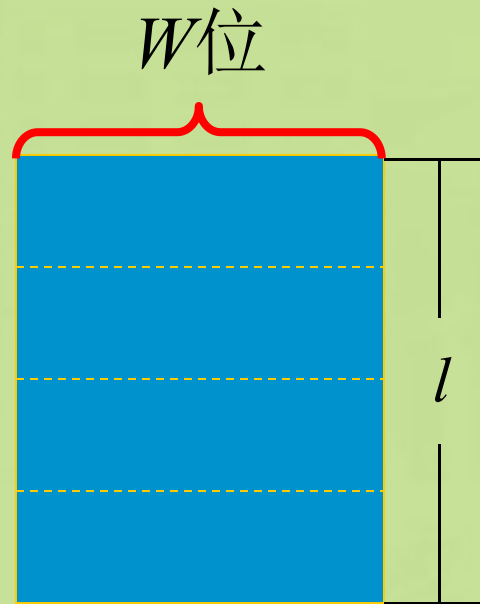
- ✓ **存储体**是存储单元的集合，用来存放数据；
- ✓ **地址译码驱动电路**包含**译码器**和**驱动器**两部分：
 - ✓ (1) **译码器**将地址总线输入的地址码转换成与之对应的译码输出线上的有效电平，以表示选中了某一存储单元；
 - ✓ (2) 由**驱动器**提供驱动电流去驱动相应的读写电路，完成对被选中存储单元的读写操作；
- ✓ **I/O和读写电路**包括读出放大器、写入电路和读写控制电路，用以完成被选中存储单元中各位的读出和写入操作。

3.1.1 存储系统的基本要求

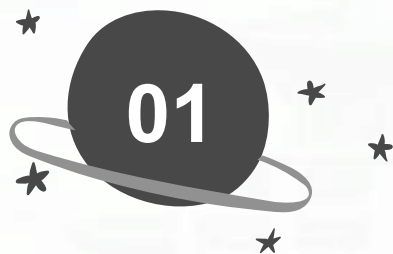


- 存储器的要求：大容量、高速度和低价格
- 存储器的容量
 - W : 存储体的字长（位、字节）
 - l : 每个存储体的字数
 - m : 并行工作的存储体个数

$$S_M = W * l * m$$



3.1.1 存储系统的基本要求



同步动态随机存取内存
(synchronous dynamic
random-access memory,
简称SDRAM) 是有一个**同
步接口**的**动态随机存取内存
(DRAM)**



SDRAM的一种存储芯片,
HY57V561620。其容量
为 $4\text{M} \times 16\text{bit} \times 4\text{bank}$ 什
么意思?

3.1.1 存储系统的基本要求



同步动态随机存取内存
(synchronous dynamic
random-access memory,
简称SDRAM) 是有一个**同步接口**的**动态随机存取内存**
(**DRAM**)



SDRAM的一种存储芯片,
HY57V561620。其容量
为4M×16bit×4bank什
么意思?

存储体的字长×每个存储单元的位数×并行工作的存储体个数

3.1.1 存储系统的基本要求



存储器的价格

$$c = C / S_M$$

总价格 C

每位价格 c



存储器的价格、速度和容量是互相冲突的

- 在存储器所用器件一定的情况下，容量越大，会因延迟增大而使速度越低；
- 容量越大，存储器的价格会越高；
- 存储器速度越高，价格也越高。



主存速度与CPU的速度不匹配，一条途径是在组成上引入**并行和重叠技术**，构成**并行主存系统**，在保持每位价格不变的情况下，提高主存的频宽。

第3章 存储、中断、总线与输入 / 输出系统



3.1 存储系统的基本要求和并行主存系统

3.1.1 存储系统的基本要求

3.1.2 并行主存系统

3.2 中断系统

3.2.1 中断的分类和分级

3.2.2 中断的响应次序与处理次序

3.2.3 中断系统的软、硬件功能分配

3.3 总线系统

3.3.1 总线的分类

3.3.2 总线的控制方式

3.3.3 总线的通信技术

3.3.4 数据宽度与总线线数

3.4 输入/输出系统

3.4.1 输入/输出系统概述

3.4.2 通道处理机的工作原理和流量设计

3.4.3 外围处理机

3.1.2 并行主存系统

- 存储器的存取速度（单字）

- **访问时间** T_A 是存储器从接到访存读申请，到信息被读到数据总线上所需的时间
- **存储周期** T_M 是一个存储体连续启动所需要的间隔时间(连续启动两次读操作所需间隔的最小时间)
- **最大频宽（带宽）** B_m 存储器连续访问时的频宽，一般用每秒钟传送的信息位数（或字节数）来衡量
 - 单体的频宽 $B_m = W / T_M$
 - m个分体的最大频宽 $B_m = W * m / T_M$
 - 实际频宽 < 最大频宽

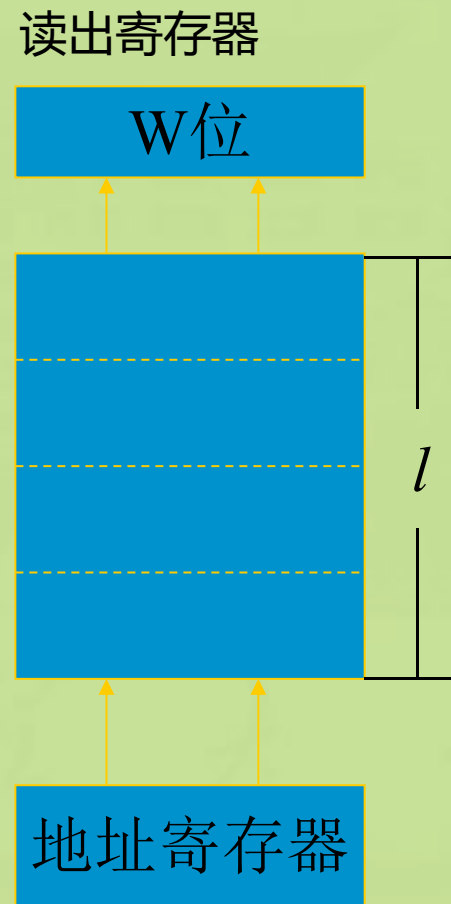
3.1.2 并行主存系统——单体单字存储器



一个字长为 W 位的单体主存，
一次可以访问一个存储器字，
所以主存最大频宽为：

$$B_m = W / T_M$$

假设，存储器字长 W 与CPU所要访问的字
（数据或指令）的字长 W 相同，
则CPU从主存获得信息的速度为 W/T_M ，
称这种主存是**单体单字存储器**



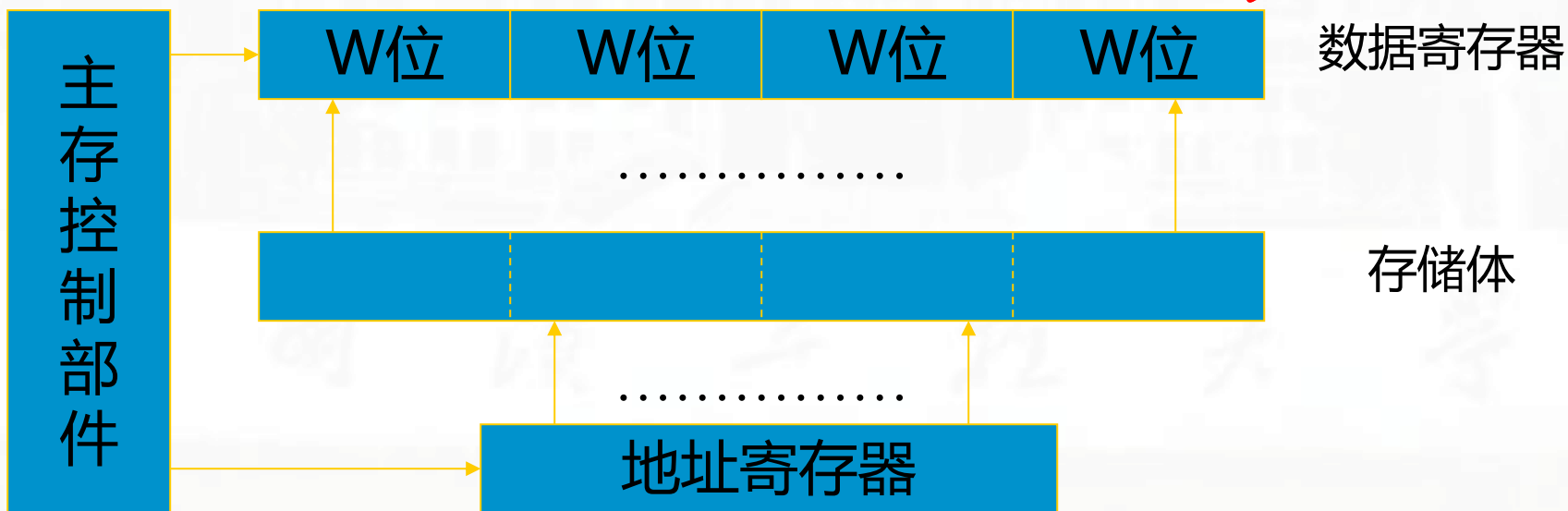
3.1.2 并行主存系统——单体多字存储器



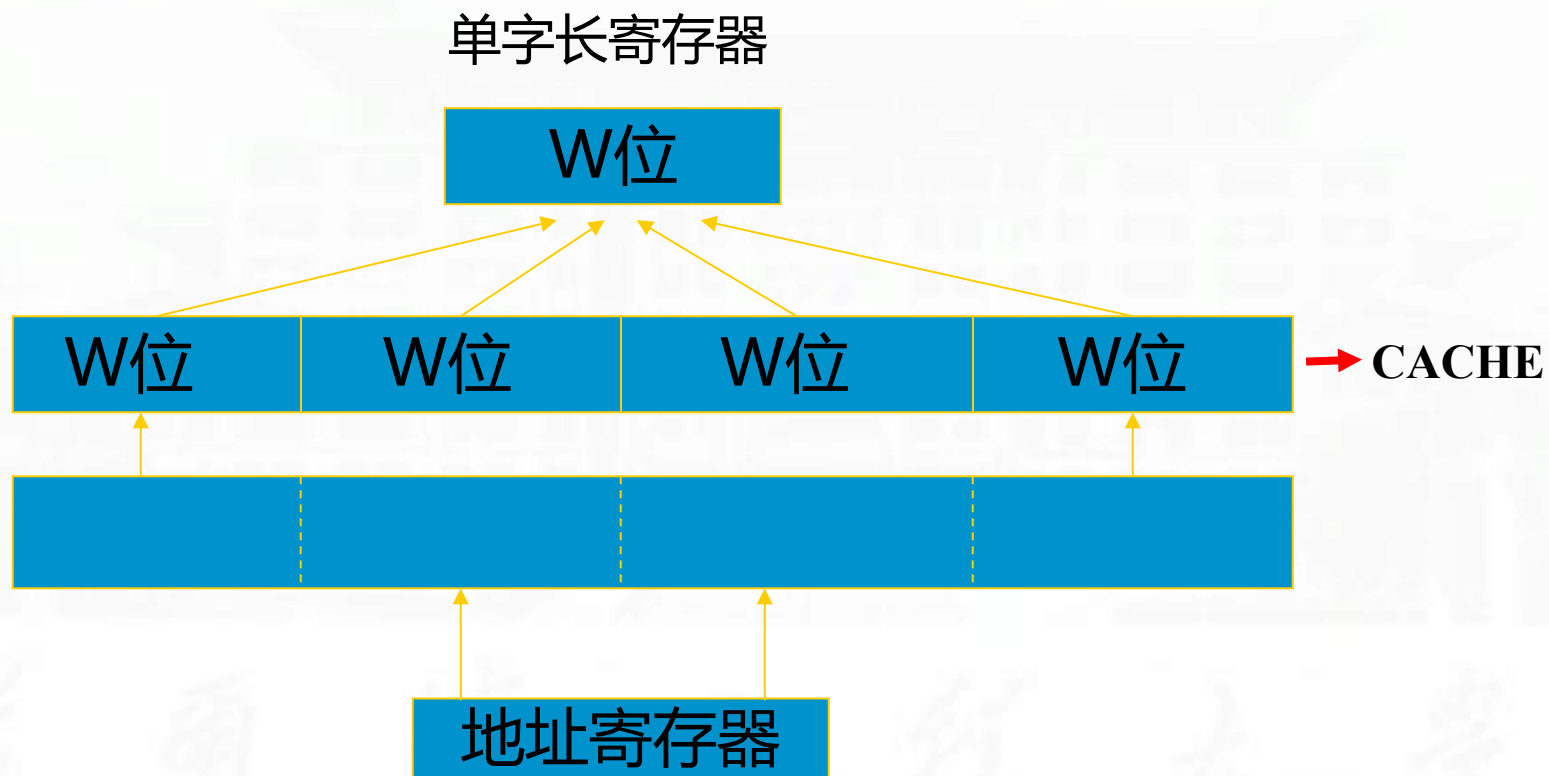
前提：指令和数据在主存中必须是连续存放的，一旦遇到转移指令或者操作数不能连续存放，这种方法的效果就不明显

$$B_m = 4 * W / T_M$$

提高存储器
字长



3.1.2 并行主存系统——单体多字存储器



3.1.2 并行主存系统——单体多字存储器

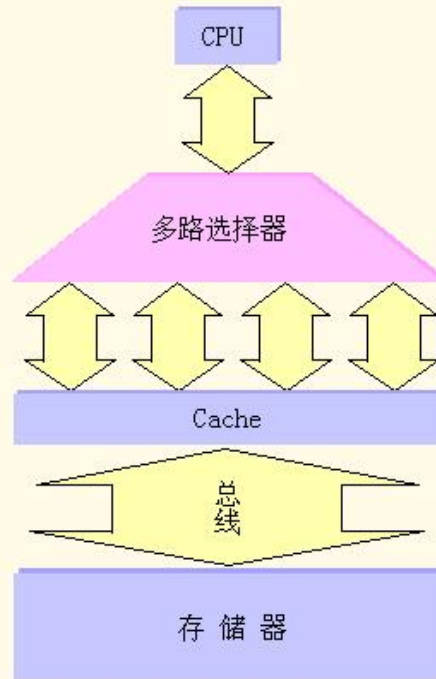
由于CPU的大部分访存都是单字宽的，所以第一级Cache的宽度通常为一个字。在不具有第二级Cache的计算机系统中，主存的宽度一般与Cache的宽度相同。

增加存储器的宽度

单字宽存储器



多字宽存储器



3.1.2 并行主存系统——单体多字存储器



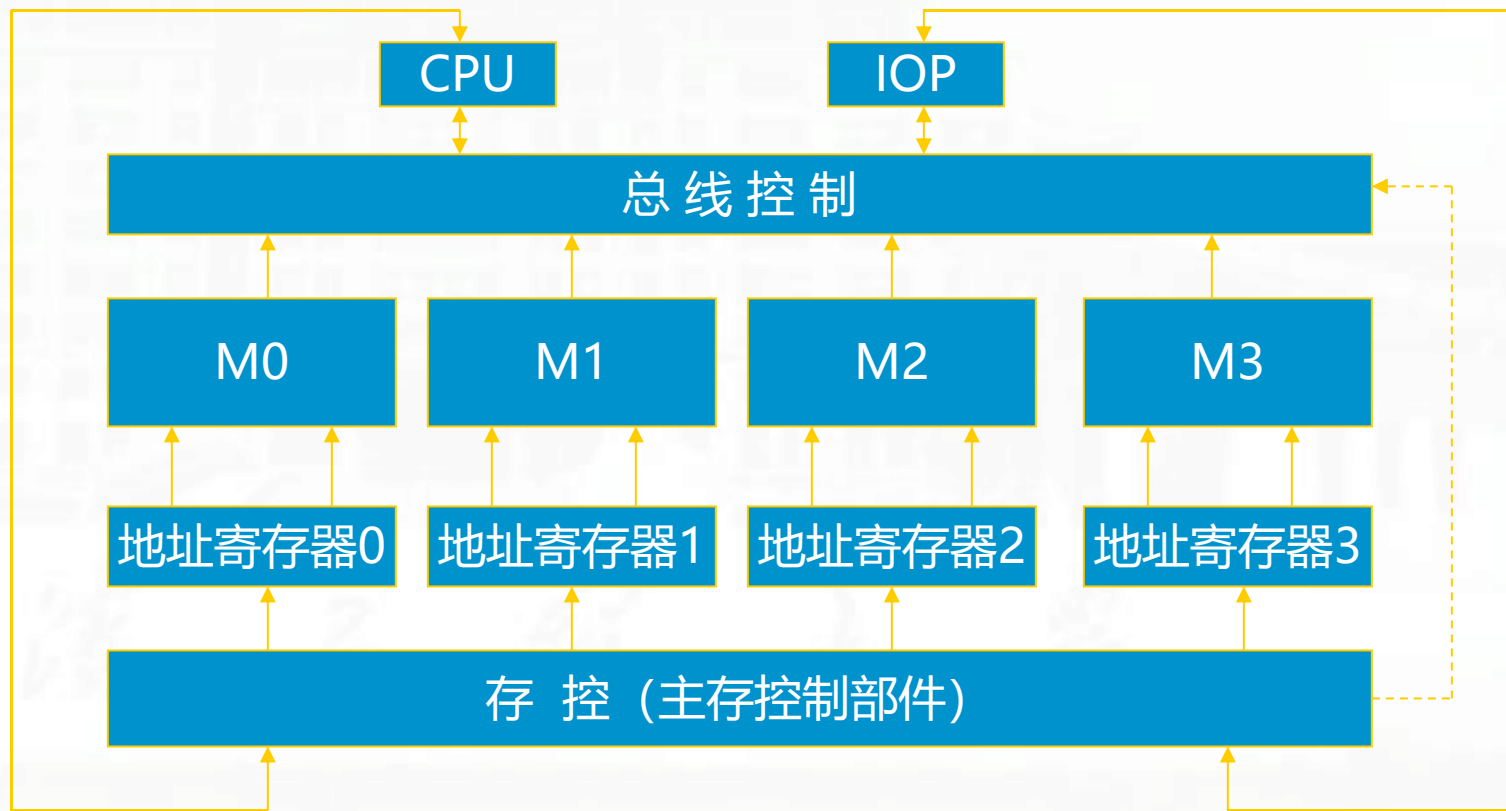
● 存在的问题

- 增加CPU和存储器之间的连接通路(通常称为存储器总线)的宽度,使其实现**代价提高**
 - 由于CPU访问Cache是每次访问一个字,所以CPU和Cache之间需要有一个**多路选择器**
- 在具有纠错功能的存储器中实现对一行(**一次可并行读出的数据**)中**部分数据的写入比较复杂**
 - 当进行这种写入时,相应行中其余的数据也必须读出,以便在写入新数据后,重新计算纠错码,并写回存储器中
- **取指令冲突**,遇到转移指令且转移成功,一个存储周期中读出的n条指令中,后面的指令都将无用
 - 绝对跳转、条件跳转
 - **导致无法一个存储周期读出**

3.1.2 并行主存系统——多体单字交叉存取存储器

一个大容量的半导体主存往往由许多容量较小、字长较短的**存储器片子**组搭而成，每个存储器片子都有自己的**地址译码、读/写驱动**等**外围电路**。

$$B_m = m * W / T_M$$



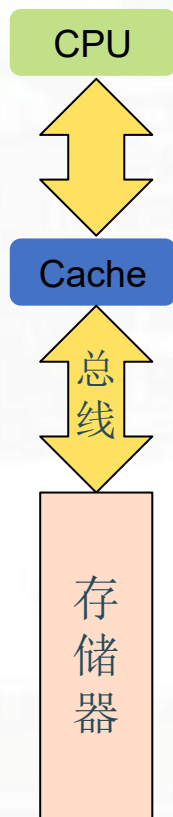
3.1.2 并行主存系统——多体单字交叉存取存储器



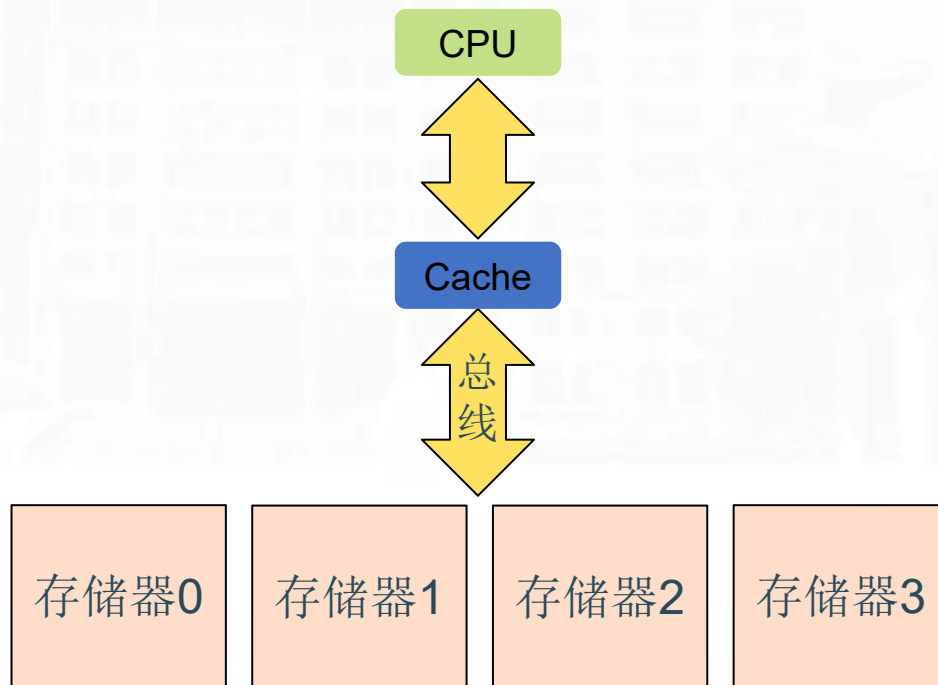
哈尔滨工程大学
HARBIN ENGINEERING UNIVERSITY

多体交叉存储器

单字宽存储器



多体交叉存储器



3.1.2 并行主存系统——多体交叉存取存储器



1

计算机中大容量的主存，可由多个存储体组成，每个存储体都具有自己的读写线路、地址寄存器和数据寄存器，称为“存储模块”。这种多模块存储器可以实现重叠与交叉存取。

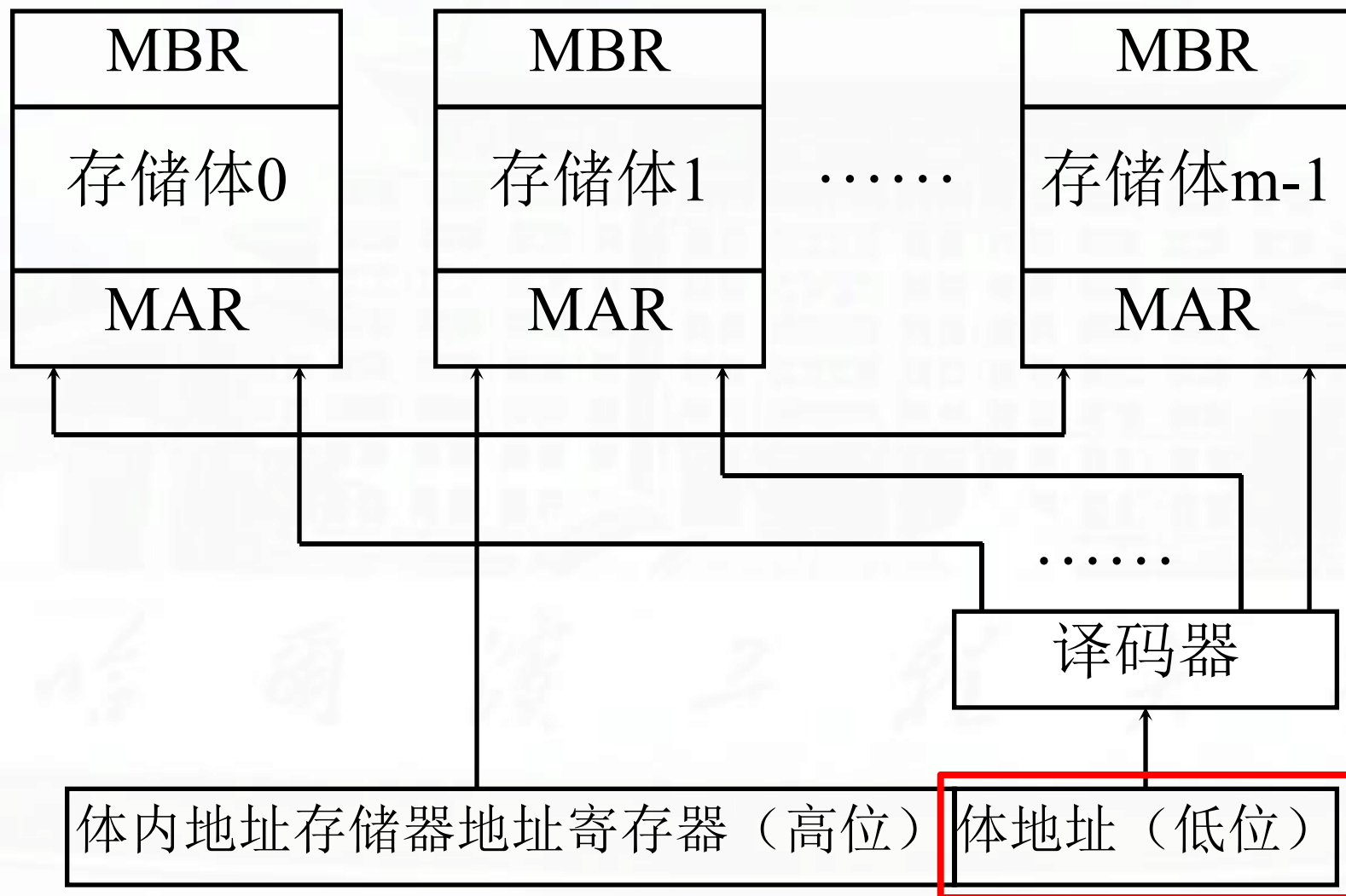
2

地址按并行分体横向顺序编号，地址序号连续的两个存储单元依次分布在相邻两个存储分体中，而不是在一个体内排序，称为多体交叉编址

3

如果在 M 个模块上交叉编址($M = 2^m$)，则称为模 M 交叉编址。

模 m 低位多体交叉编址（低位选体，高位选体内寄存器）



3.1. 并行主存系统——多体交叉存取存储器

- 单体容量为L 的 个分体，其M_j 体的编制模式为

$$m \times i + j$$

- 其中， $i=0, 1, 2, \dots, l-1$ 。 $j=0, 1, 2, \dots, m-1$

$m=4$ 时

模体	地址的模4低位交叉编制	对应二进制地址码最末两位的状态
M ₀	0, 4, 8, 12, ... , 4i+0, ...	0 0
M ₁	1, 5, 9, 13, ... , 4i+1, ...	0 1
M ₂	2, 6, 10, 14, ... , 4i+2, ...	1 0
M ₃	3, 7, 11, 15, ... , 4i+3, ...	1 1

模m低位多体交叉编址



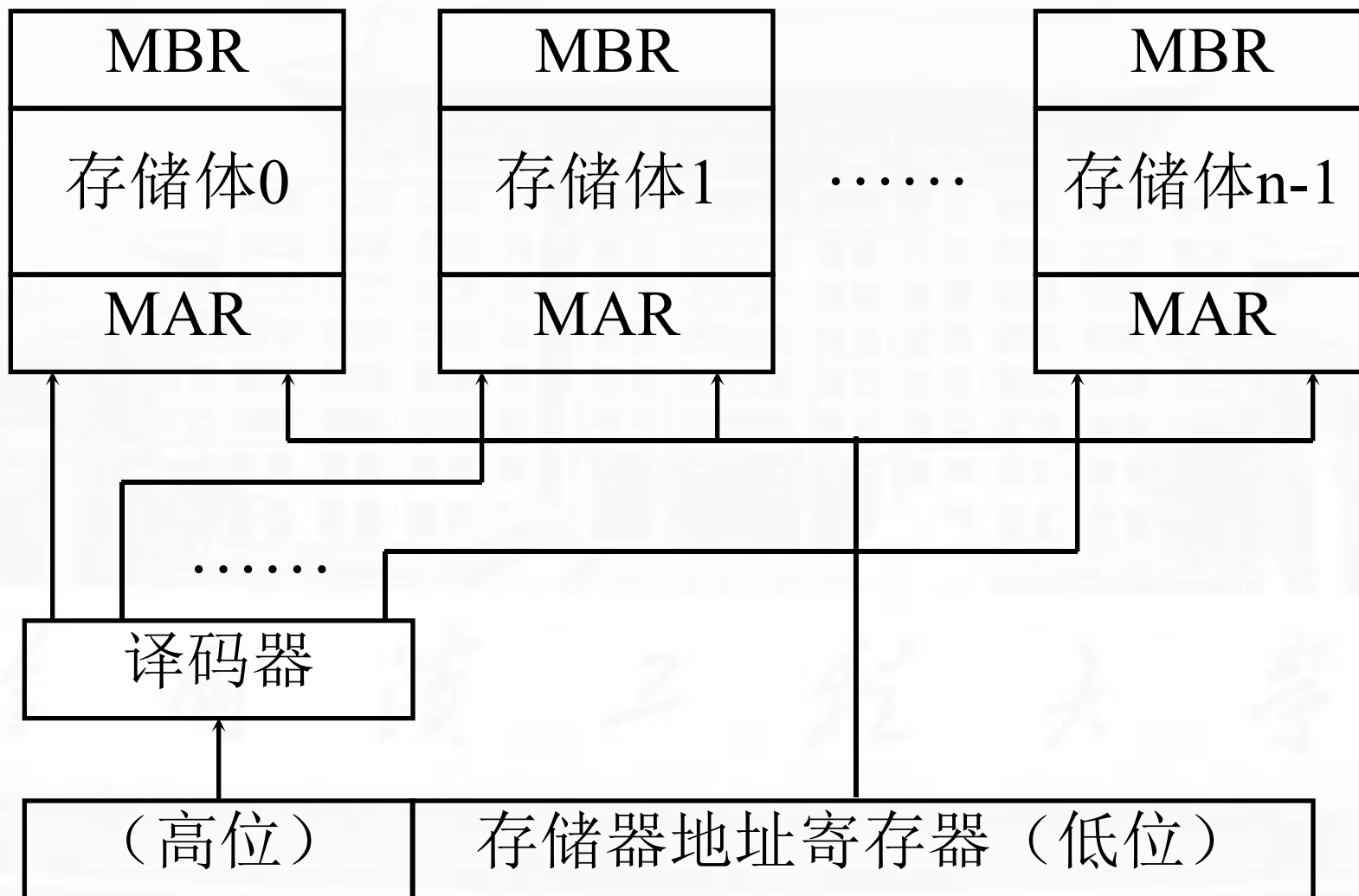
	M ₀		M ₁		M ₂		M ₃
000000		000001		000010		000011	
000100		000101		000110		000111	
001000		001001		001010		001011	
001100		001101		001110		001111	
010000		010001		010010		010011	
010100		010101		010110		010111	
011000		011001		011010		011011	
011100		011101		011110		011111	
100000		100001		100010		100011	
100100		100101		100110		100111	
101000		101001		101010		101011	
101100		101101		101110		101111	
110000		110001		110010		110011	
110100		110101		110110		110111	
111000		111001		111010		111011	
111100		111101		111110		111111	

模m低位交叉编址



- 地址 $A = m * i + j$, $i = 0, 1, 2, \dots, l-1, j = 0, 1, 2, \dots, m-1$
- 寻址规则：
 - 体地址 $j = A \bmod m (A \% m)$
 - 体内地址 $i = A / m$
 - $M_0 : 0, m, 2m, \dots, m(l-1)+0$
 -
 - $M_i : i, m+i, 2m+i, \dots, m(l-1)+i$
- 适合于流水、向量或阵列机
- 适合于单处理机内的数据存取和带Cache的主存
 - 将数据存放在不同模块内, 可以降低访存冲突

模m高位交叉编址（更广义）



模m高位交叉编址（更广义）

	M ₀		M ₁		M ₂		M ₃
000000		010000		100000		110000	
000001		010001		100001		110001	
000010		010010		100010		110010	
000011		010011		100011		110011	
000100		010100		100100		110100	
000101		010101		100101		110101	
000110		010110		100110		110110	
000111		010111		100111		110111	
001000		011000		101000		111000	
001001		011001		101001		111001	
001010		011010		101010		111010	
001011		011011		101011		111011	
001100		011100		101100		111100	
001101		011101		101101		111101	
001110		011110		101110		111110	
001111		011111		101111		111111	

- 地址 $A = i * l + j$

$$i=0, 1, 2, \dots, m-1; j=0, 1, 2, \dots, l-1$$

- 寻址规则:

- 体地址 $i = A / l$
- 体内地址 $j = A \bmod l$
- 这里, A/l 计算的是跨过了多少个存储体的长度,确定了这个地址属于哪个存储体,而 $A \bmod L$ 给出了在该存储体内的具体位置。

例1 当A=18 (地址码为18, 第19个元素), $l=16, m=4$

- 体地址 $i = 18 / 16 = 1$ M1存储体 (第2个存储体)
- 体内地址 $j = 18 \bmod 16 = 2$ M1存储体第3个元素
- 当A=33 (地址码是33, 第34个元素)
- 体地址 $i = 33 / 16 = 2$ M2存储体 (第3个存储体)
- 体内地址 $j = 33 \bmod 16 = 1$ M2存储体第2个元素

- 地址 $A = i * l + j$
 $i=0, 1, 2, \dots, m-1; j=0, 1, 2, \dots, l-1$
- 寻址规则:
 - 体地址 $i = A // l$
 - 体内地址 $j = A \bmod l$
- 适合于共享存储器的多机系统，适用于指令和数据分别存于不同分体中
- 在单任务系统中，采用高位交叉访问方式的主要目的是扩大存储器容量
- 在多任务和多用户系统中，可以通过把不同的任务分配给不同的存储体来提高存储器的访问速度



按高位分为模块，
模块内按低位交叉

控制复杂，不常
采用

3.1.2 并行主存系统



- **多体单字方式**组成只要 m 个地址不发生分体冲突，即使地址之间不是顺序的，仍可并行读出。（主存可采用并行读出）
- 主存采用**单体多字方式**组成要求可并行读出的 m 个字必须是地址顺序且处于同一主存单元。
- 可以将两者结合构成**多体多字**交叉存储器。

并行存储器系统

能并行读取多个CPU字的单体多字、多体单字或多体多字的交叉存储主存系统。

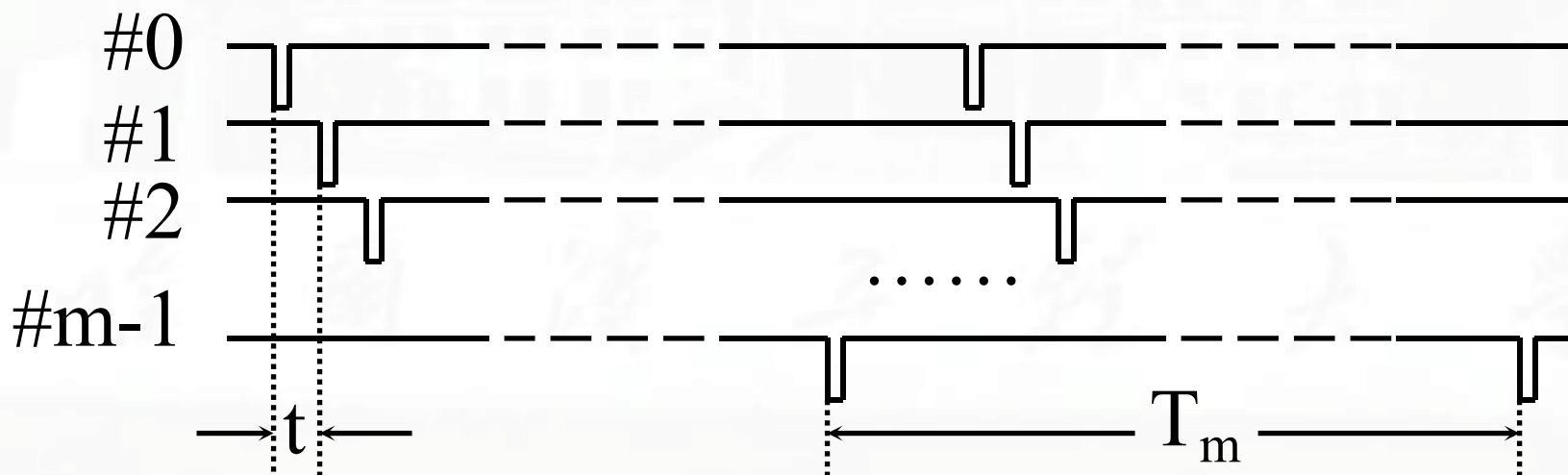
m个存储体分时启动

实际上是一种采用流水线方式工作的并行存储器,理论上,存储器的速度可望提高n倍
每存储体的启动间隔t为:

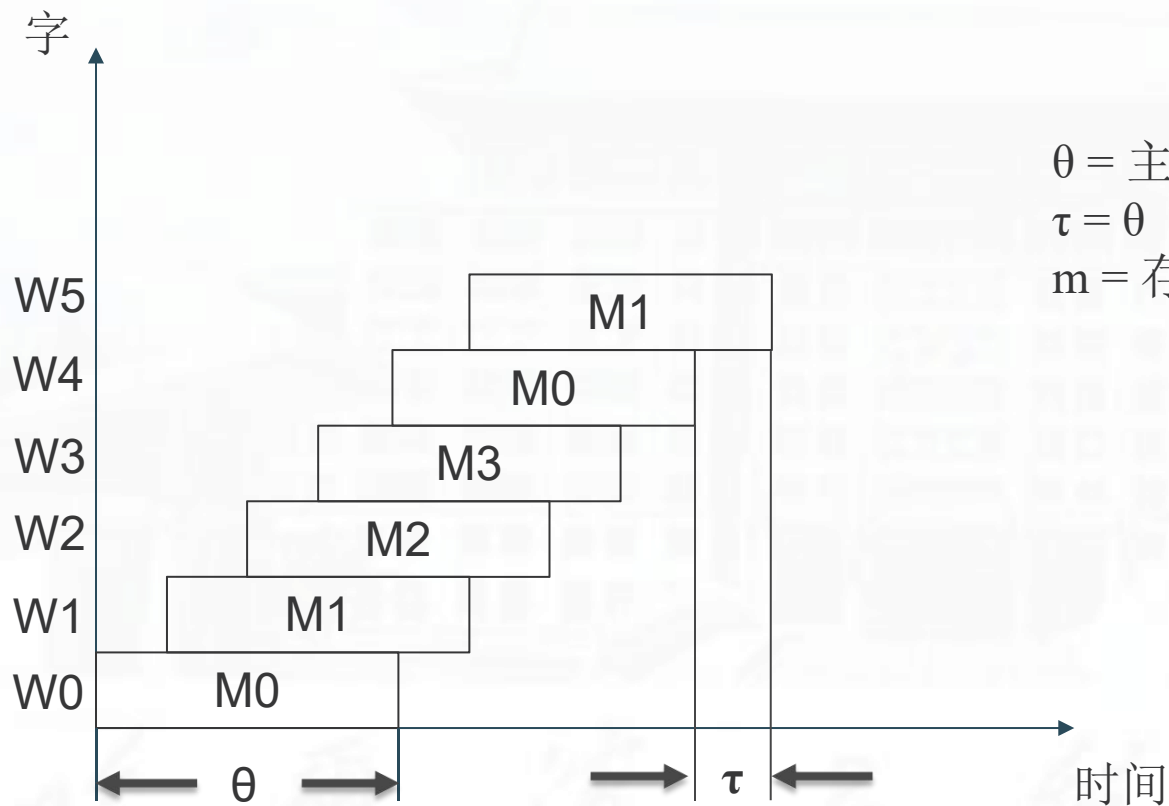
其中: n为存储体个数

T_m 为每个存储体的访问周期

$$t = \left\lfloor \frac{T_m}{n} \right\rfloor$$



m个存储体分时启动



θ = 主存储器的一个存储周期

$$\tau = \theta / m = \theta / 4$$

m = 存储体的个数

模四多体交叉存取存储器的流水存取示意图

定量分析主存频宽与分体数 m 、转移概率 λ 的关系



哈尔滨工程大学
HARBIN ENGINEERING UNIVERSITY

设对具有 m 路独立模体的并行主存系统，CPU发出地址 $A_1, A_2, \dots, A_k, \dots, A_q$ 的访存申请队列，假定：

- (1) 在每一个主存周期之前，扫描队列，**截取** A_1, A_2, \dots, A_k
- (2) A_1, A_2, \dots, A_k 是在 k 个地址中**没有两个或两个以上的地址处于同一模体中**，显然
 $k \leq m$
- (3) 截取的 k 个地址的队列，能同时访问 k 个模体， $k = 1, 2, \dots, m$

定量分析主存频宽与分体数 m 、转移概率 λ 的关系



另设： $P(k)$ 表示申请序列长度为 k 的概率。 k 的平均值，即每个主存周期所能访问到的平均字数，用 B 表示：

$$B = \sum_{k=1}^m k * p(k)$$



k 越接近于 m ,效率就会越高



当访存申请队列都是指令的话，那么影响最大的是**转移指令概率 λ** ，给定指令的下一条指令地址为非顺序地址的概率。

定量分析主存频宽与分体数m、转移概率 λ 的关系

申请队中如果第一条就是转移指令且转移成功，与第一条指令同时读出的其它m-1条指令就没有用，相当于k=1,也就是

$$k=1 \quad p(1) = \lambda$$

$$k=2 \quad p(2) = [1-p(1)] * \lambda = (1-\lambda) * \lambda$$

$$k=3 \quad p(3) = [1-p(1)-p(2)] * \lambda = (1-\lambda)^2 * \lambda$$

$$k=m-1 \quad p(m-1) = [1-p(1)-p(2)-\dots-p(m-2)] * \lambda = (1-\lambda)^{m-2} * \lambda$$

$$k=m \quad p(m) = [1-p(1)-p(2)-\dots-p(m-1)] * \lambda = (1-\lambda)^{m-1}$$

定量分析主存频宽与分体数 m 、转移概率 λ 的关系

01

带入上式：

$$B_m = \sum_{k=1}^m k * p(k) = \sum k(1-\lambda)^{k-1} \lambda + m(1-\lambda)^{m-1}$$

02

由数学归纳法得：

$$B_m = \sum_{i=0}^{m-1} (1-\lambda)^i$$

$$\sum_{k=1}^{m-1} k (1-\lambda)^{k-1} \lambda + m (1-\lambda)^{m-1} = \sum_{i=0}^{m-1} (1-\lambda)^i$$

$m = 1$ 时：左边 $= 1$ ；右边 $= 1$

$m = 2$ 时：左边 $= \lambda + 2 - 2\lambda = 2 - \lambda$

右边 $= 1 + 1 - \lambda = 2 - \lambda$

设： $m = n$ 时，两边等式成立，即 $\sum_{k=1}^{n-1} k (1-\lambda)^{k-1} \lambda + n (1-\lambda)^{n-1} = \sum_{i=0}^{n-1} (1-\lambda)^i$:

$$\sum_{k=1}^{n-1} k (1-\lambda)^{k-1} \lambda + n (1-\lambda)^{n-1} = \sum_{i=0}^{n-1} (1-\lambda)^i$$

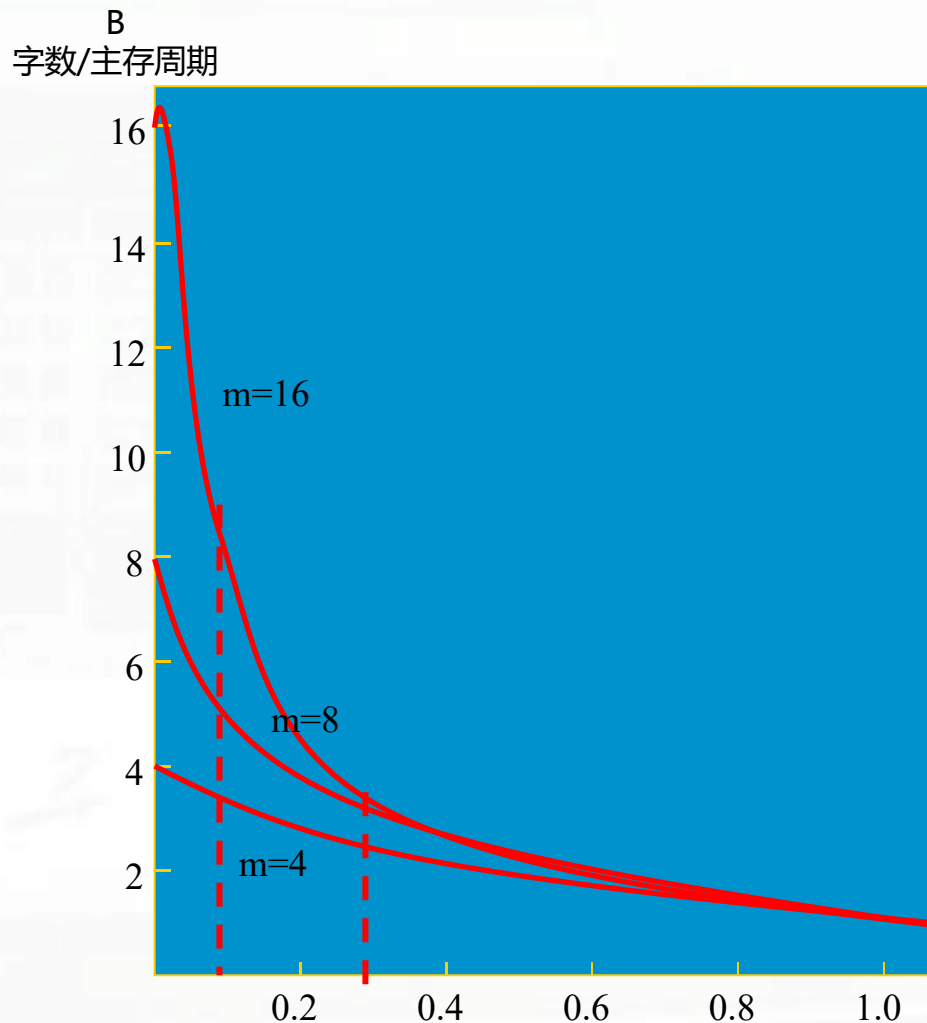
证明： $m = n + 1$ 时成立。

用归纳法证明 (续)



$$\begin{aligned} & \sum_{k=1}^n k (1 - \lambda)^{k-1} \lambda + (n+1)(1 - \lambda)^n \\ &= \sum_{k=1}^{n-1} k (1 - \lambda)^{k-1} \lambda + n (1 - \lambda)^{n-1} \lambda + (n+1)(1 - \lambda)^n \\ &= \sum_{k=1}^{n-1} k (1 - \lambda)^{k-1} \lambda + n (1 - \lambda)^{n-1} (\lambda + 1 - \lambda) + (1 - \lambda)^n \\ &= \sum_{k=1}^{n-1} k (1 - \lambda)^{k-1} \lambda + n (1 - \lambda)^{n-1} + (1 - \lambda)^n \\ &= \sum_{i=0}^{n-1} (1 - \lambda)^i + (1 - \lambda)^n \\ &= \sum_{i=0}^n (1 - \lambda)^i \end{aligned}$$

- $\lambda=1$ 时, $B=1$
- $\lambda=0$ 时, $B=m$, 效率最高
- $\lambda > 0.3$ 时, $m=4、8、16$ 的 B 差别不大, m 再大, 对系统的效率也不会带来多大的好处
- $\lambda < 0.1$ 时, m 值的大小对 B 的改进会有显著影响
- 对数据来讲, 由于随机性大, 因此靠加大 m 不一定满足要求。
- 单纯靠增大 m 来提高并行主存系统的频宽是有限的, 性能价格比随 m 的增大而下降, 采用并行主存系统仍不能满足速度上的要求必须从体系结构上改进



3.1 存储系统的基本要求和并行主存系统

3.1.1 存储系统的基本要求

3.1.2 并行主存系统

3.2 中断系统

3.2.1 中断的分类和分级

3.2.2 中断的响应次序与处理次序

3.2.3 中断系统的软、硬件功能分配

3.3 总线系统

3.3.1 总线的分类

3.3.2 总线的控制方式

3.3.3 总线的通信技术

3.3.4 数据宽度与总线线数

3.4 输入/输出系统

3.4.1 输入/输出系统概述

3.4.2 通道处理机的工作原理和流量设计

3.4.3 外围处理机

3.2.中断系统



01

中断：

CPU中止正在执行的程序，转去处理随机提出的请求，待处理完后，再回到原先被打断的程序继续恢复执行的过程。

02

中断系统：

响应和处理中断的软、硬件总体。

3.2.1 中断的分类和分级



01

引起中断的**各种事件**称为**中断源**

02

中断系统的复杂性实际上是由**中断源的多样性**引起的

03

中断源可以来自系统外部，也可以来自机器内部，甚至处理机本身

04

中断可以是硬件引起的，也可以是软件引起

05

把各种各样的中断源分类、分级组织好，是中断系统的关键之一

3.2.1 中断的分类和分级

1、中断源的种类（常见的中断源有如下几种类别）：

1

由**外围设备**引起的中断：低速外围设备；高速外围设备的前处理和后处理；

2

由**处理机**产生的中断：算术运算溢出，除数为零，数据校验错，非法数据格式等；

3

由**存储器**产生的中断：地址越界、地址不存在、写ROM、页面失效、数据或地址校验错、访问主存超时等；

4

由**控制器**产生的中断：非法指令、未定义操作码、用户程序执行了特权指令、堆栈溢出、分时系统中时间片到、用户态与特权态的切换等。

5

由**总线**产生的中断。输入输出总线出错，存储器总线出错等

6

实时过程控制产生的中断

7

实时钟的定时中断

8

多处理机系统中，从其它处理机**发送来的中断**

9

程序调试过程中，由**断点**产生的中断

10

硬件故障中断

11

电源故障中断

3.2.1 中断的分类和分级



2、中断源的分类组织

01

中断源分类组织的目的：在响应中断后处理机能够尽快找到中断入口。

02

根据中断事件的紧迫程度、中断源工作速度、中断源的性质等进行分类。

03

为每一**类**中断源分配一个**硬件的中断入口**，在进入这个入口之后，再通过**软件分支转入**相应的中断处理部分。

3.2.1 中断的分类和分级

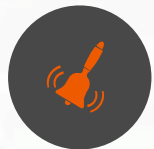
IBM公司的机器，把中断源分为7类：



重新启动中断。处理机不能禁止这类中断 ⑦



紧急机器检验出错中断。由硬件故障时产生 ①



程序性错误引起的中断 ③



访问管理程序中断。当用户程序执行访管指令引起的中断 ②



可以抑制的机器检验错误引起的中断 ④



外部事件中断 ⑤



输入输出中断 ⑥

3.2.1 中断的分类和分级



3、中断分级（为什么要分级？分成哪几级？）

由于中断源很多，它们相互独立并随机地发出中断请求，因此，同一时刻常常会同时发生多个中断请求，而对同一类型中的各中断请求，其响应和处理仍有优先次序，当然这种优先次序通常不是由中断系统硬件来安排，而是由软件或通道等来管理的。而对于不同类型的各个中断源就要根据终端的性质、紧迫性、重要以及软件处理的方便性上把它们分为几级。中断系统按中断源的级别高低进行响应。通常优先级最高的中断定为第一级，依次类推。

3.2.1 中断的分类和分级



3、中断优先级

安排中断优先顺序主要由下列因素来决定：

(1) 中断源的急迫性

(2) 设备的工作速度。优先级从高到低的次序一般如下：

- 实时钟
- 磁盘存储器，包括软磁盘
- 行式打印机
- 控制台终端输出
- 控制台键盘输入

(3) 数据恢复的难易程度

(4) 要求处理机提供的服务量

3.2.1 中断源的组织



例如：在IBM 370系列机中，把7类中断分为5级，从高到低分别是：

- 高
- ① 紧急的**机器检验**错误引起的中断 硬件故障：电源故障、运算电路误动作、主存出错、通道动作故障、处理器的各种硬件故障；
- ② **调用管理程序，程序性错误，可以抑制的机器检验**错误引起的中断 访管中断；非法指令、各种溢出、除数为0、有效位为0、实时程序监测中断；
- ③ **外部事件**引起的中断 定时器（计时、计费、控制）中断、外部信号中断（与其系统的通信）、中断键中断（操作员操作）；
- ④ **外围设备**的中断 I/O操作完成、I/O通道或者设备故障；
- 低
- ⑤ **重新启动**引起的中断 为操作员或另一台cpu要启动一个程序；故障后恢复；用户请求或预定任务；软件更新或安装；

**当同时发生多个不同类中断时，中断系统是如何
响应和处理次序？**

3.1 存储系统的基本要求和并行主存系统

3.1.1 存储系统的基本要求

3.1.2 并行主存系统

3.2 中断系统

3.2.1 中断的分类和分级

3.2.2 中断的响应次序与处理次序

3.2.3 中断系统的软、硬件功能分配

3.3 总线系统

3.3.1 总线的分类

3.3.2 总线的控制方式

3.3.3 总线的通信技术

3.3.4 数据宽度与总线线数

3.4 输入/输出系统

3.4.1 输入/输出系统概述

3.4.2 通道处理机的工作原理和流量设计

3.4.3 外围处理机

3.2.2 中断的响应次序与处理次序



01

中断的响应次序是靠中断响应的**排队器硬件**事先**固定好的**;

02

中断的响应次序是在同时发生多个不同中断类的中断请求时，**中断响应硬件中的排队器**所决定的**响应次序**;

03

为了能动态地调节中断处理程序**实际执行完的次序**，即**中断处理次序**，在中断级请求源与中断响应排队器的入口端之间又加设了一个**中断级屏蔽字寄存器**和相应的控制门电路硬件，中断级屏蔽字寄存器中的每一个中断级屏蔽位可以控制让相应等级的中断请求能否进入**中断响应排队器中参加排队**;

04

操作系统可以通过修改各中断级处理程序的中断级屏蔽位的状况，来使中断处理（完）的次序符合所希望的次序。

3.2.2 中断的响应次序与处理次序

01

当执行用户程序时，如果有中断请求发生时，先判断现行程序状态字PSW的中断级屏蔽位（“1”表示对该级中断开放，“0”表示对该级中断屏蔽）；

02

根据屏蔽位情况，判断哪些中断请求可以进入中断响应排队器；

03

只要进入中断响应排队器的总是让高级别的优先响应；

04

中断请求排队微命令到来时，优先响应高级别的中断；

05

中断用户程序，中断断点地址被压入返回地址堆栈，通过交换PSW实现程序切换，将旧的PSW保存到内存的指定单元，取出新的PSW建立新现场。

3.2.2 中断的响应次序与处理次序



例题：处理机共有4个中断源，中断优先级从高到低分别是：1级、2级、3级和4级，中断级屏蔽位设置如下表。当处理机在执行主程序时，同时有3级中断源和2级中断源向处理机发出中断服务的请求。当处理机为2级中断源服务时又有4级中断源发出中断服务请求，当处理机为4级中断源服务时又有1级中断发出中断服务请求。（0是屏蔽，1是开放）

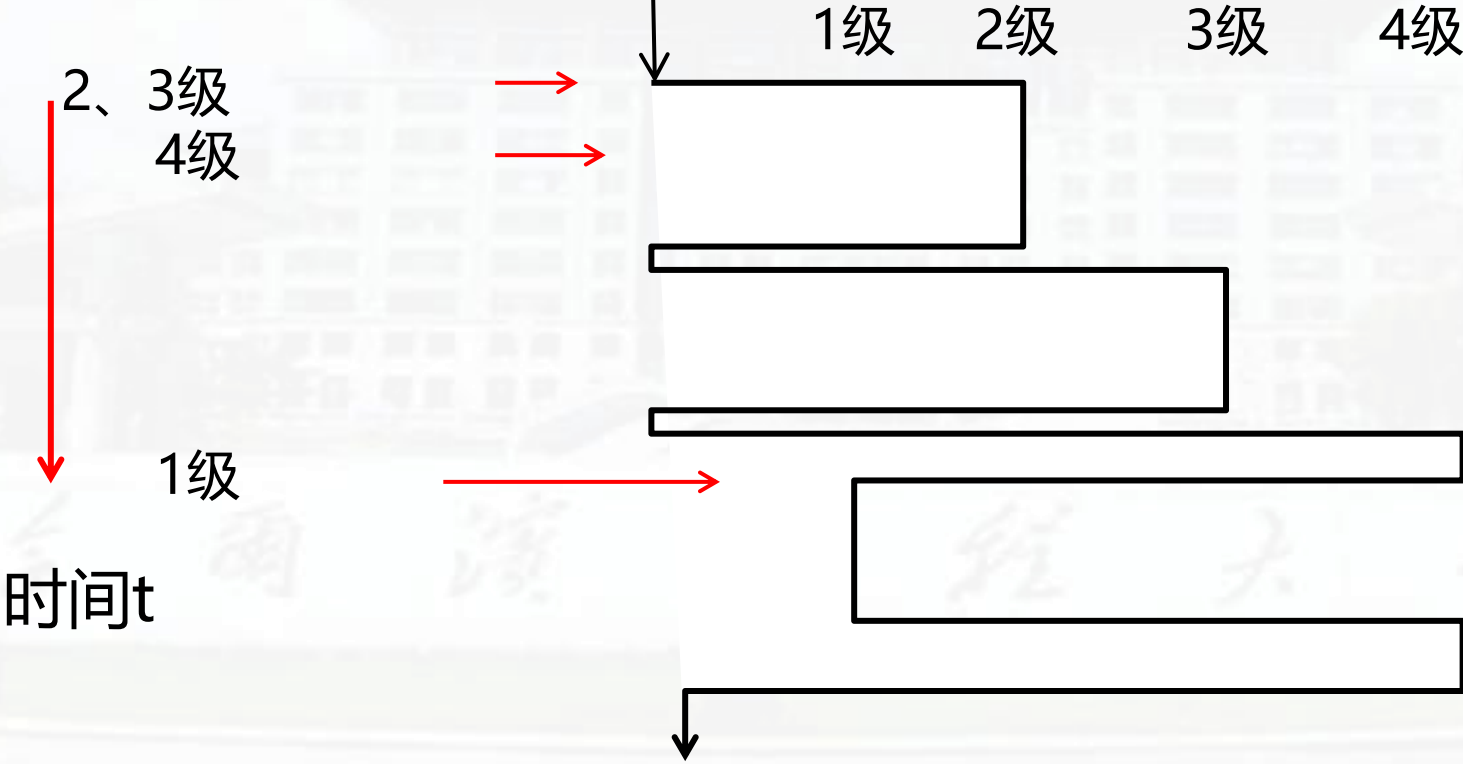
中断处理 程序级别	中断级屏蔽位			
	第1级	第2级	第3级	第4级
第1级	0	0	0	0
第2级	1	0	0	0
第3级	1	1	0	0
第4级	1	1	1	0

3.2.2 中断的响应次序与处理次序

中断请求

主程序

中断服务程序



按照中断优先级响应中断请求的例子



3.2.2 中断的响应次序与处理次序



例题：有四个中断源D1、D2、D3和D4，它们的中断优先级从高到低分别是1级、2级、3级和4级。这些中断源的正常中断屏蔽码和改变后的中断屏蔽码见下表。每个中断源一位，共4位屏蔽码。

(中断响应次序和中断处理次序相同或相反，“1”为中断屏蔽，“0”为开放)

中断源名称	中断优先级	正常中断屏蔽码	改变后的中断屏蔽码
		D ₁ D ₂ D ₃ D ₄	D ₁ D ₂ D ₃ D ₄
D1	1	1 1 1 1	1 0 0 0
D2	2	0 1 1 1	1 1 0 0
D3	3	0 0 1 1	1 1 1 0
D4	4	0 0 0 1	1 1 1 1

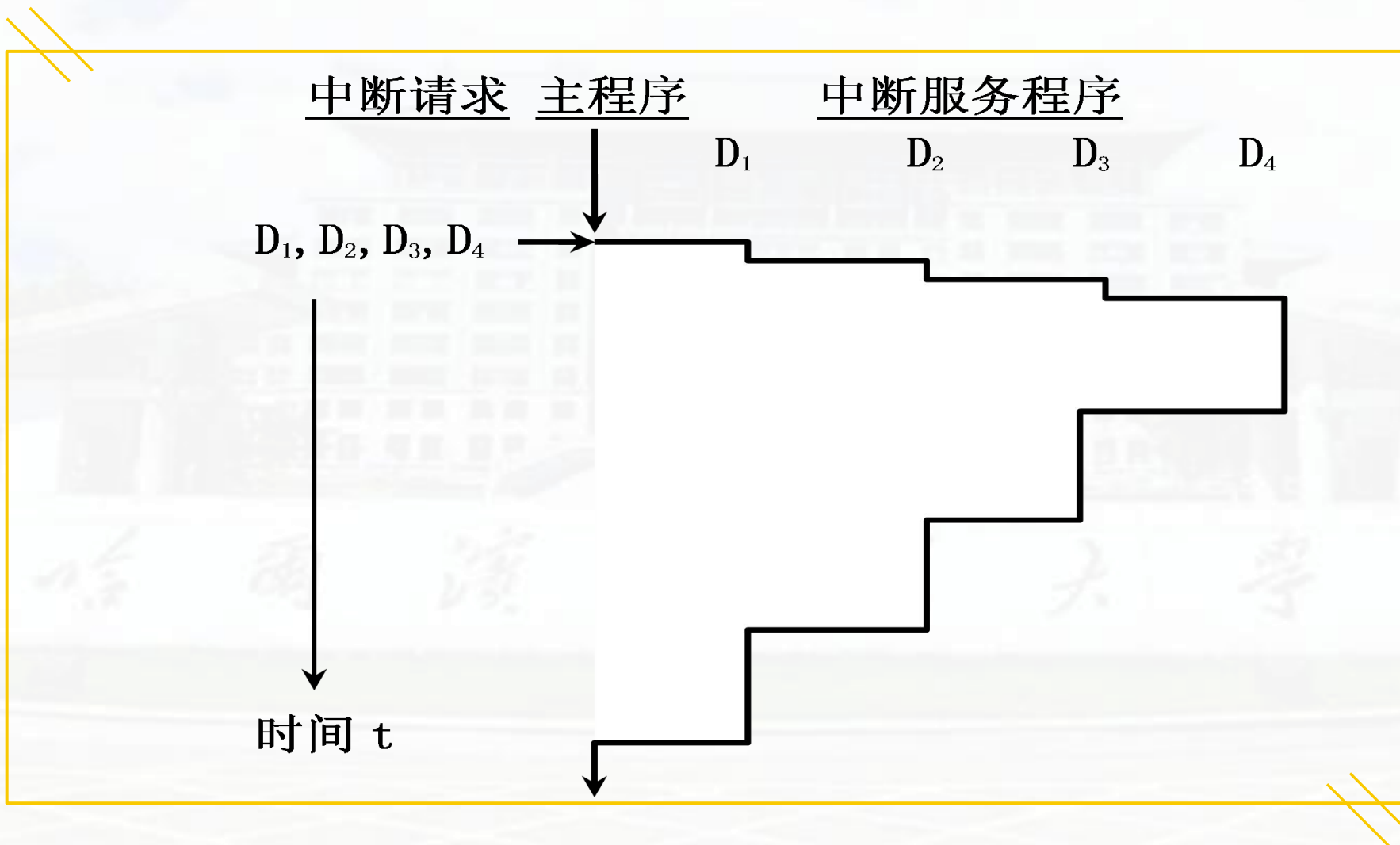
解：

- ✓ 如果4个中断源都使用正常的中断屏蔽码，处理机的中断服务顺序将严格按照中断源的中断优先级进行。
- ✓ 如果改变中断屏蔽码，当 D_1 、 D_2 、 D_3 和 D_4 这4个中断源同时请求中断服务时，处理机实际为各个中断源服务的先后次序就会改变。
- ✓ 处理机响应的顺序是 D_1 、 D_2 、 D_3 、 D_4
- ✓ 实际中断服务的顺序是 D_4 、 D_3 、 D_2 、 D_1

中断屏蔽



哈尔滨工程大学
HARBIN ENGINEERING UNIVERSITY



3.2.2 中断的响应次序与处理次序

中断响应与处理次序

★ 硬件的工作

中断响应次序是靠中断响应的硬件排队器事先固定好的

★ 软件的工作

为了能动态地调节中断处理程序**实际执行完的次序**，即**中断处理次序**，在中断级请求源与中断响应排队器的入口端之间又加设了一个**中断级屏蔽字寄存器**和相应的控制门电路硬件，中断级屏蔽字寄存器中的每一个中断级屏蔽位可以控制让相应等级的**中断请求能否进入中断响应排队器中参加排队**（这个相当于无形中也修改了中断响应次序）



3.1 存储系统的基本要求和并行主存系统

3.1.1 存储系统的基本要求

3.1.2 并行主存系统

3.2 中断系统

3.2.1 中断的分类和分级

3.2.2 中断的响应次序与处理次序

3.2.3 中断系统的软、硬件功能分配

3.3 总线系统

3.3.1 总线的分类

3.3.2 总线的控制方式

3.3.3 总线的通信技术

3.3.4 数据宽度与总线线数

3.4 输入/输出系统

3.4.1 输入/输出系统概述

3.4.2 通道处理机的工作原理和流量设计

3.4.3 外围处理机

3.2.3 中断系统的软硬件功能分配



有些功能必须用硬件来实现，有的功能必须用软件来实现，而大部分功能既可以用硬件实现，也可以用软件实现。

恰当分配中断系统的软件与硬件功能，是中断系统最关键问题。



3.2.3 中断系统的软硬件功能分配

①主要考虑的两个因素

- ▶ 中断响应时间
 - 从中断源向处理机发出中断服务请求开始，到处理机开始执行这个中断源的中断服务程序时为止，这一段时间称为**中断响应时间**。
 - 中断响应时间是一个非常重要的指标
- ▶ 灵活性
 - 用硬件实现速度快，但灵活性差
 - 用软件实现灵活性好，但速度慢

3.2.3 中断系统的软硬件功能分配



②中断处理过程

01

表示本功能一般用**硬件**来实现。

02

表示本功能一般用**软件**来实现。

03

表示本功能既可以用**硬件**来实现，也可以用**软件**来实现。

3.2.3 中断系统的软硬件功能分配

中断处理过程开始



现行指令执行结束，且没有更紧急的服务请求

关CPU中断

保存中断点，PC中的内容

撤消中断请求

保存硬件现场，PSW及SP等

识别中断源

改变设备的屏蔽状态

进入中断服务程序入口

保存软件现场，通用寄存器等

打开CPU中断，CPU可以响应更高级别的中断请求

中断服务，执行中断服务程序

关CPU中断

恢复软件现场

恢复屏蔽状态

恢复硬件现场

打开CPU中断

返回到中断点

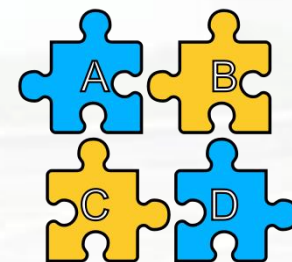


中断处理过程全部结束

3.2.3 中断系统的软硬件功能分配



- 必须用**硬件**实现的有：保存中断点和进入中断服务程序入口
 - 这两个功能相当于执行一条转子程序指令，因为中断发生在现行程序的什么地方是不确定的，不能由程序员来安排。
 - 第一次关CPU中断一般也用硬件来实现，因为非常简单。
- 必须用**软件**实现的有：中断服务和返回到中断点
 - 中断服务程序，实施具体的中断处理逻辑。
 - 返回到中断点，通过执行一条中断返回指令来实现。





■ 主存出错引起的中断是 ()

- A.访管中断 B.外中断 C.机器校验中断 D.程序性中断

■ 通道程序执行结束后引起的中断是 ()

- A.机器校验中断 B.I / O中断 C.程序性中断 D.外中断

■ 指令执行结果出现异常引起的中断是 ()

- A.I / O中断 B.机器校验中断 C.程序性中断 D.外中断

■ 数据通路出错引起的中断是 ()

- A.机器校验中断 B.访管中断 C.外中断 D.程序性中断

■ IBM 370系统中断响应优先级级别最低的是 ()

- A.机器校验中断 B.外中断 C.I / O中断 D.重新启动中断



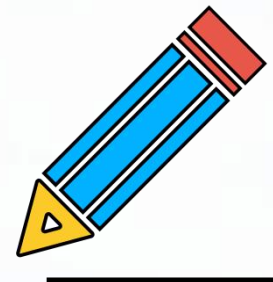


■ IBM 370系统中，中断响应优先级级别最高的中断类型是（）。

A.程序性中断 B.重新启动中断 C.紧急机器校验中断 D.输入输出中断

■ 在IBM 370系统中，当用户程序需调用外设进行输入输出时会发生（）。

A.访管中断 B.输入 / 输出中断 C.程序性中断 D.外中断



■ 中断响应由高到低的优先次序宜用（）。

A.访管→程序性→机器故障 B.访管→程序性→重新启动 C.外部→访管→程序性 D.程序性→I / O→访管

■ 在IBM 370机器的中断系统中，分配给中断处理软件的功能应当是（）

A.保存程序断点 B.保存通用寄存器 C.保存程序状态字 D.转向中断处理程序总入口

■ 在IBM 370系统中，中断响应硬件分工保存的是（）

A.通用寄存器内容 B.条件码等状态信息 C.各种软件状态和标志 D.作业名称和优先级等

中断系统要求掌握内容(2)



- ◆ 某系统共有4个中断级，相应地每个中断处理程序的现行状态字中就设有4个中断屏蔽位。若中断响应的次序是1->2->3->4，现要求其实际的中断处理次序为1->4->3->2，回答下面问题：
- ◆ (1)设计各级中断处理程序的中断级屏蔽位（令“1”对应于屏蔽，“0”对应于开放）；
- ◆ (2)若在运行用户程序时，同时出现第1、2、3、4级中断请求，请画出此程序运行过程示意图。



- 机器共有4级中断，中断响应优先次序为1->2->3->4，现要求其实际的中断处理次序为1->4->3->2，回答下面问题：

(1)设计各级中断处理程序的中断级屏蔽位

(令“1”对应于屏蔽，“0”对应于开放)

【解答】

中断处理程序级别	中断级屏蔽位			
	第1级	第2级	第3级	第4级
第1级	1	1	1	1
第2级	0	1	0	0
第3级	0	1	1	0
第4级	0	1	1	1



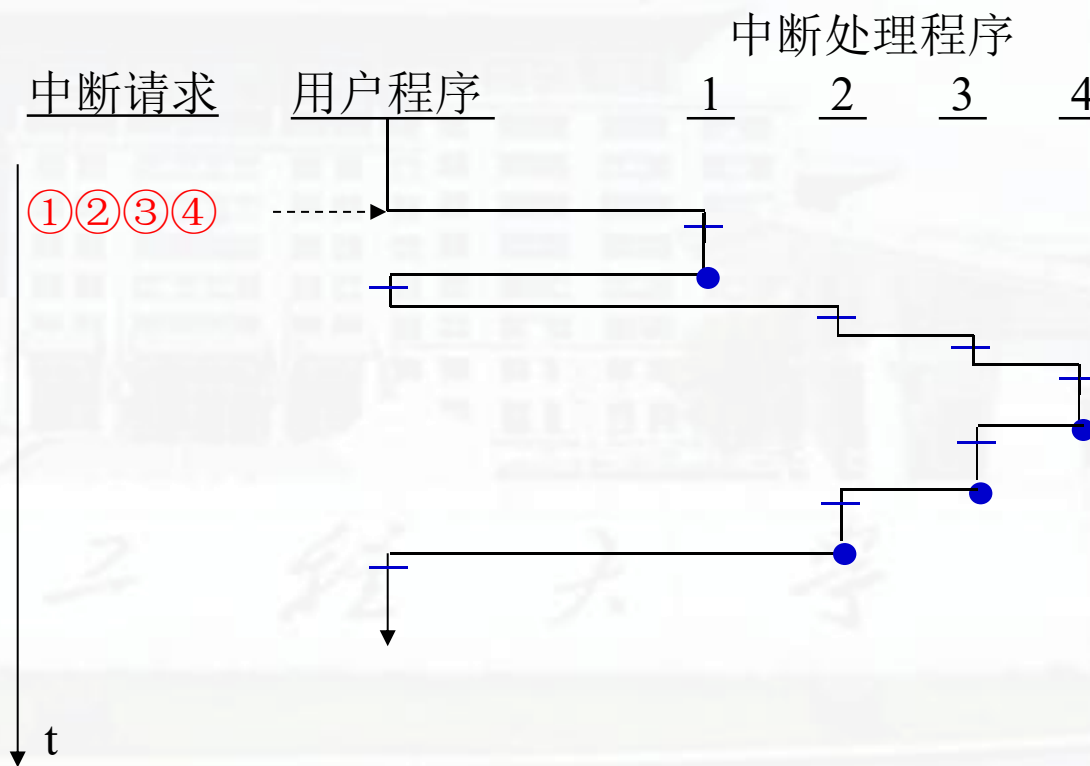
若机器共有4级中断，中断响应优先次序为

1-→2-→3-→4，现要求其实际的中断处理次序为

1-→4-→3-→2，回答下面问题：

(2)若在运行用户程序时，同时出现第1、2、3、4级中断请求，请画出此程序运行过程示意图。

【解答】



中断系统要求掌握内容(3)



- ◆ 若机器共有5级中断，中断响应优先次序为1->2->3->4->5，现要求其实际的中断处理次序为1->4->5->2->3，回答下面问题：
- ◆ (1)设计各级中断处理程序的中断级屏蔽位（令“1”对应于屏蔽，“0”对应于开放）；
- ◆ (2)若在运行用户程序时，同时出现第4、2级中断请求，而在处理第2级中断未完成时，又同时出现第1、3、5级中断请求，请画出此程序运行过程示意图。



- 若机器共有5级中断，中断响应优先次序为1->2->3->4->5，现要求其实际的中断处理次序为1->4->5->2->3，回答下面问题：

(1)设计各级中断处理程序的中断级屏蔽位

(令“1”对应于屏蔽，“0”对应于开放)

【解答】

中断处理 程序级别	中断级屏蔽位				
	第1级	第2级	第3级	第4级	第5级
第1级	1	1	1	1	1
第2级	0	1	1	0	0
第3级	0	0	1	0	0
第4级	0	1	1	1	1
第5级	0	1	1	0	1



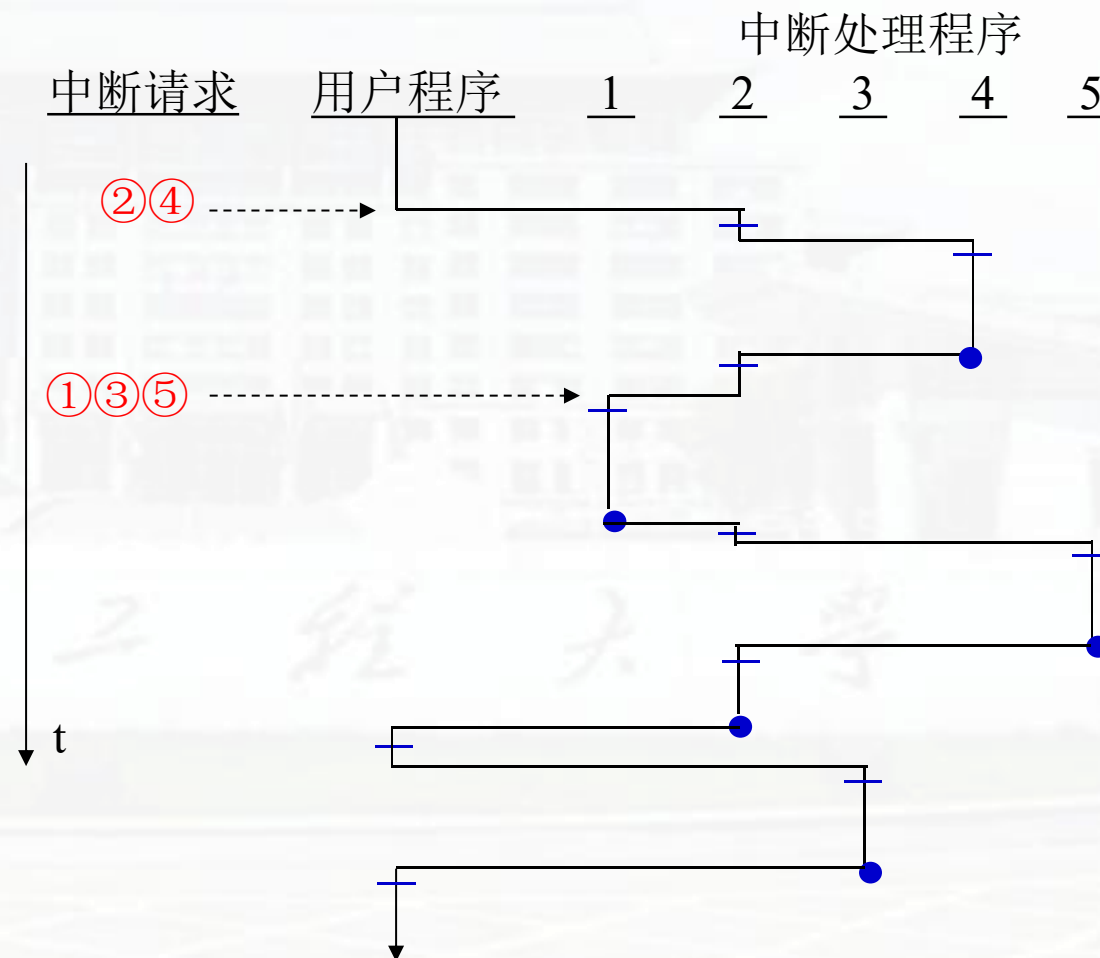
若机器共有5级中断，中断响应优先次序为

1->2->3->4->5，现要求其实际的中断处理次序为

1->4->5->2->3，回答下面问题：

(2)若在运行用户程序时，同时出现第4、2级中断请求，而在处理第2级中断未完成时，又同时出现第1、3、5级中断请求，请画出此程序运行过程示意图。

【解答】



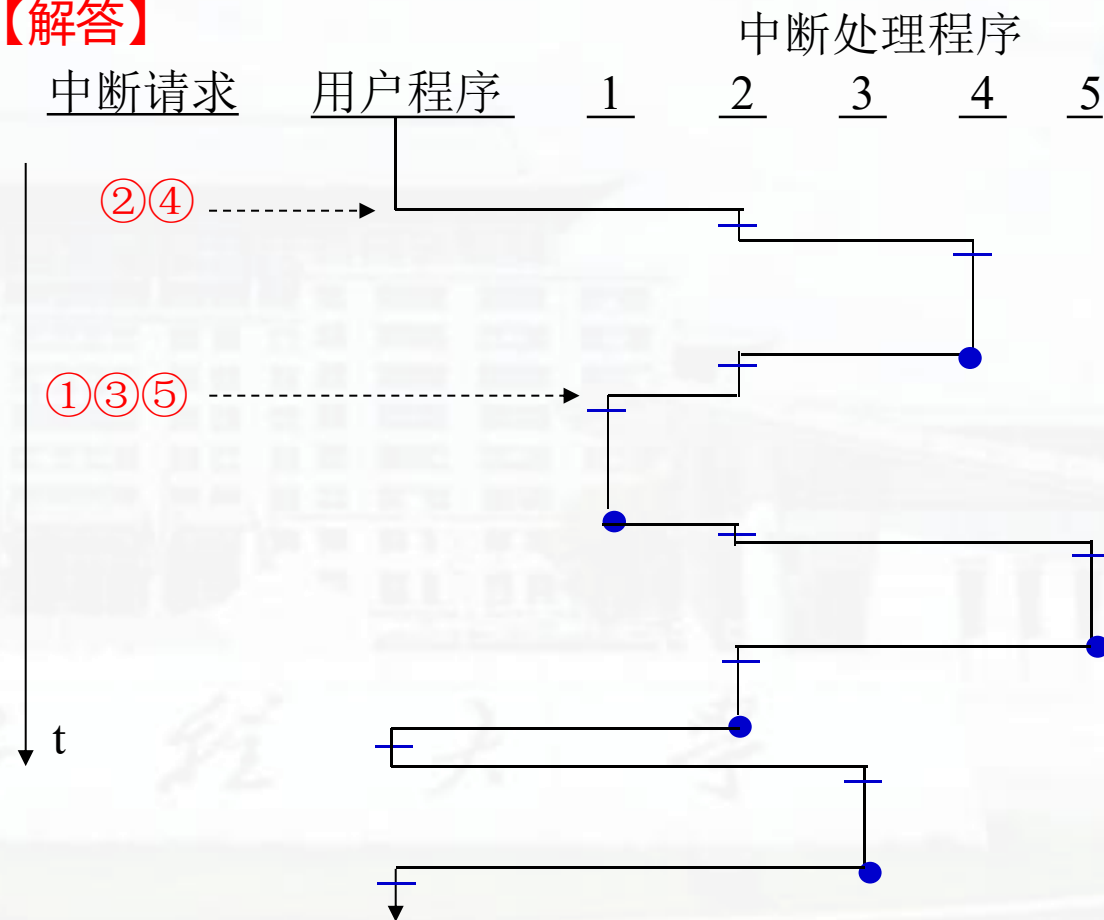


若机器共有5级中断，中断响应优先次序为1->2->3->4->5，现要求其实际的中断处理次序为1->4->5->2->3，回答下面问题：

(2)若在运行用户程序时，同时出现第4、2级中断请求，而在处理第2级中断未完成时，又同时出现第1、3、5级中断请求，请画出此程序运行过程示意图。

解析：当发生第4、2级中断请求时，用户程序的现行PSW中的中断屏蔽位是00000，因此中断用户程序。

【解答】



按照中断响应优先次序，先响应第2级中断请求，第4级中断请求进入排队器。又因为第2级中断请求PSW中的屏蔽位是01100，对第4级是开放的，所以中断第2级，响应第4级。依次类推。