

王道考研——组成原理

WWW.CSKAOYAN.COM

第三章 存储系统

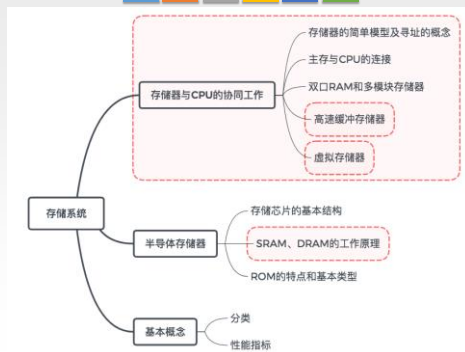
本节内容

存储系统

主存简单模型
及寻址的概念

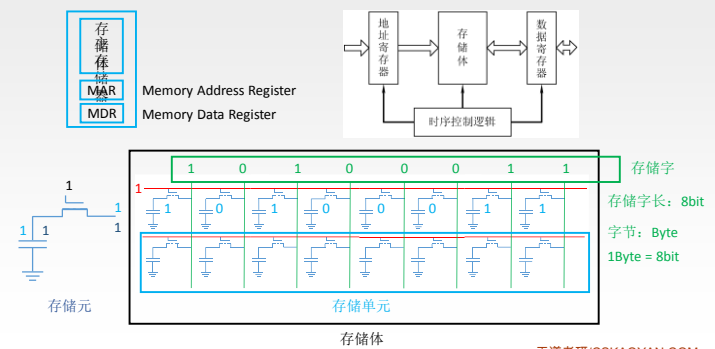
王道考研/CSKAOYAN.COM

本章总览



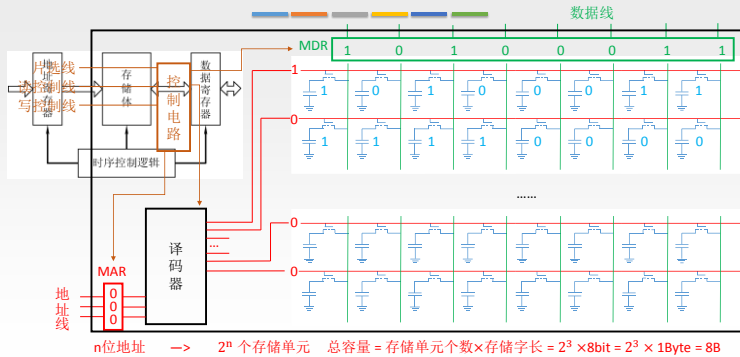
王道考研/CSKAOYAN.COM

主存储器



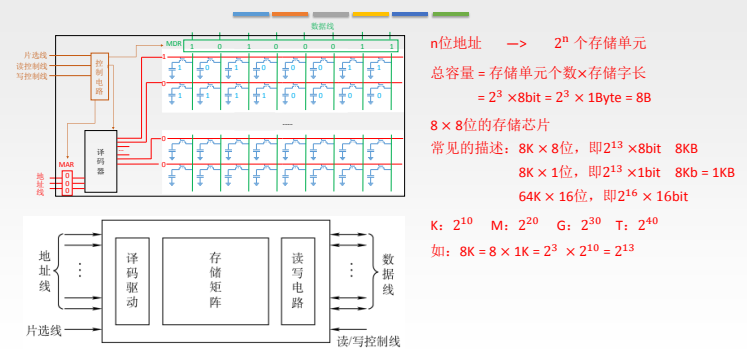
王道考研/CSKAOYAN.COM

主存储器的简单模型



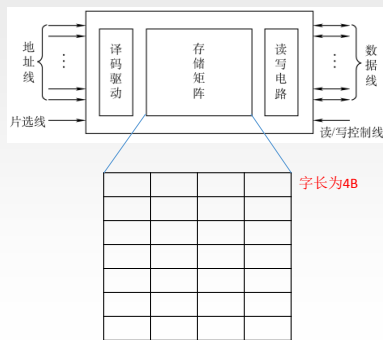
王道考研/CSKAOYAN.COM

存储器芯片的基本结构



王道考研/CSKAOYAN.COM

寻址

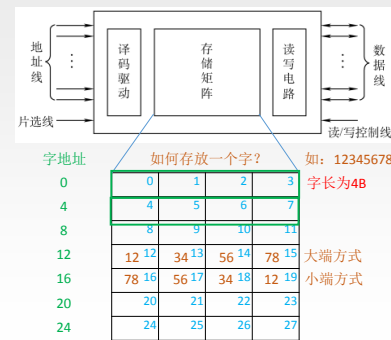


十进制:
地址: 00 0000 0000 0
00 0000 0001 1
00 0000 0010 2
00 0000 0011 3
00 0000 0100 4
00 0000 0101 5
00 0000 0110 6
.....

总容量为1KB 地址线: 10根
按字节寻址: 1K个单元, 每个单元1B
按字寻址: 256个单元, 每个单元4B
按半字寻址: 512个单元, 每个单元2B
按双字寻址: 128个单元, 每个单元8B

王道考研/CSKAOYAN.COM

寻址

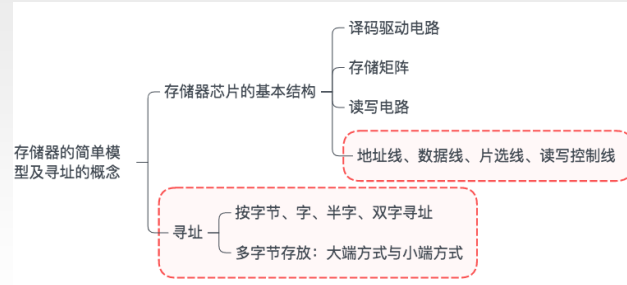


十进制:
地址: 00 0000 0000 0
00 0000 0001 1
00 0000 0010 2
00 0000 0011 3
00 0000 0100 4
00 0000 0101 5
00 0000 0110 6
.....

总容量为1KB 地址线: 10根
按字节寻址: 1K个单元, 每个单元1B
按字寻址: 256个单元, 每个单元4B
按半字寻址: 512个单元, 每个单元2B
按双字寻址: 128个单元, 每个单元8B

王道考研/CSKAOYAN.COM

本节回顾



王道考研/CSKAOYAN.COM

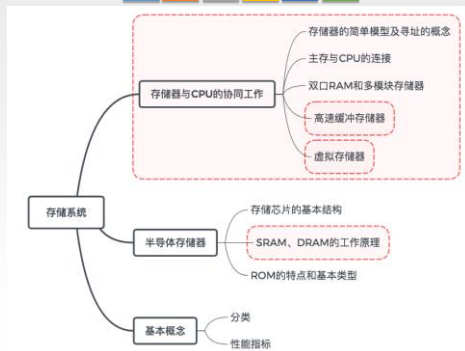
本节内容

存储系统

主存与CPU的连接

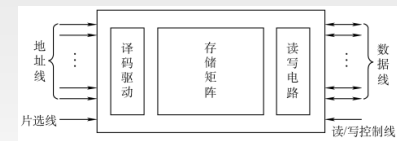
王道考研/CSKAOYAN.COM

本章总览



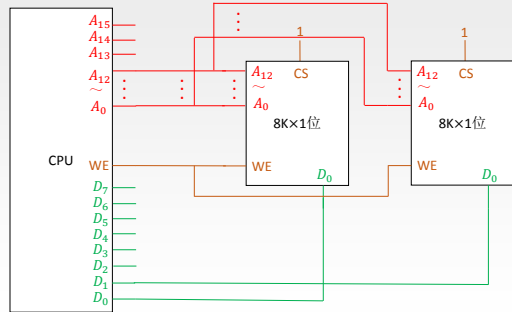
王道考研/CSKAOYAN.COM

存储器芯片的基本结构



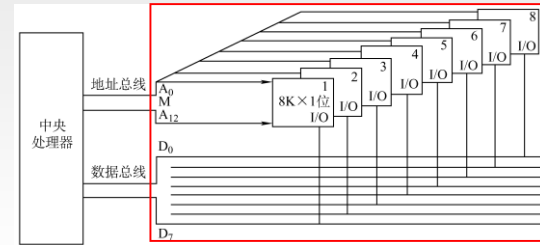
王道考研/CSKAOYAN.COM

主存容量扩展-位扩展



王道考研/CSKAOYAN.COM

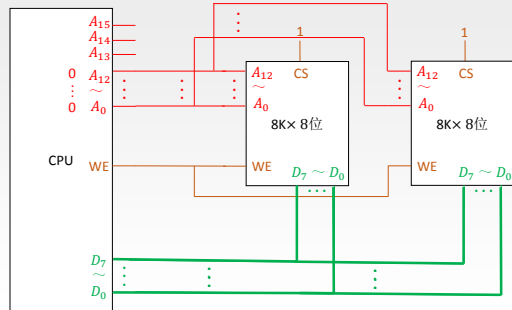
主存容量扩展-位扩展



8片 $8K \times 1$ 位的存储芯片
 → 1个 $8K \times 8$ 位的存储器, 容量8KB

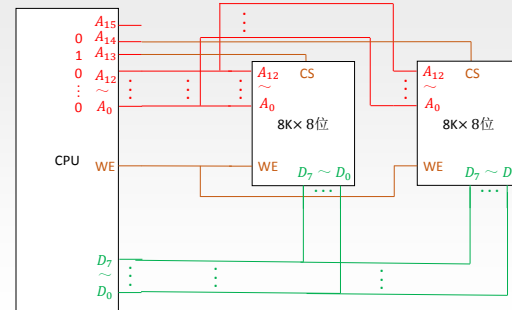
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



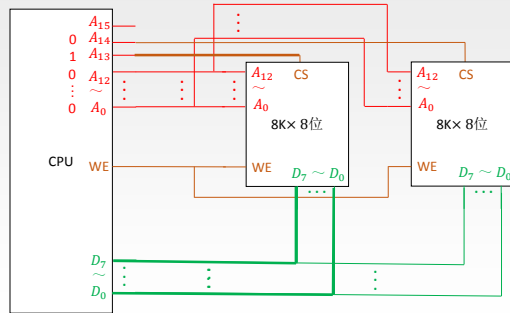
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



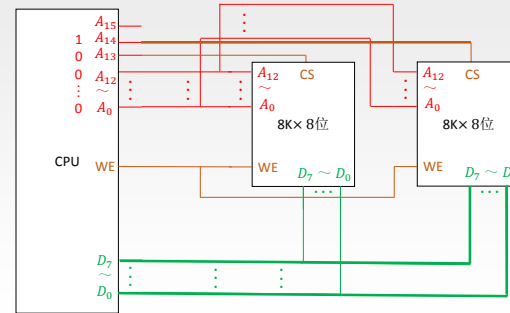
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



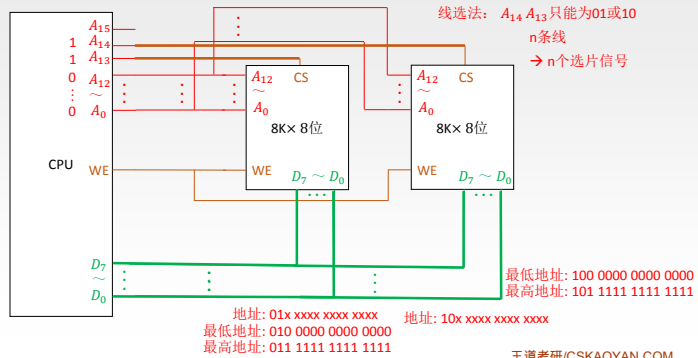
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



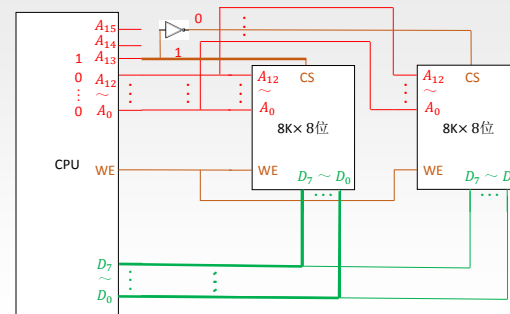
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



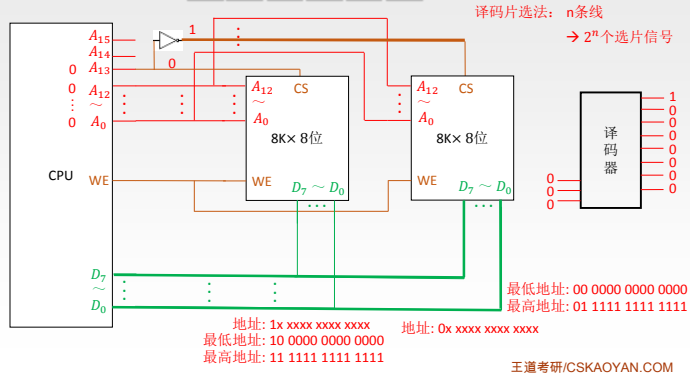
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展

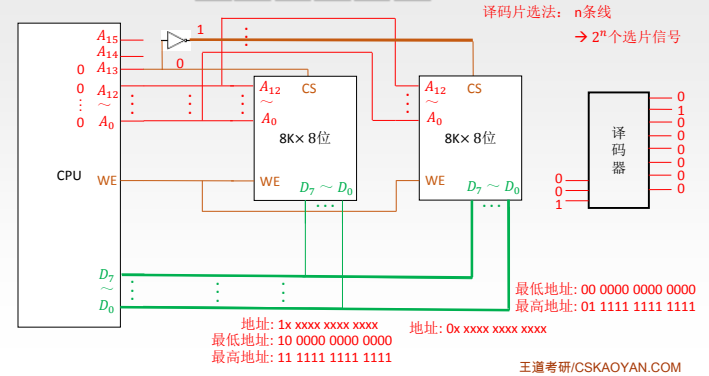


王道考研/CSKAOYAN.COM

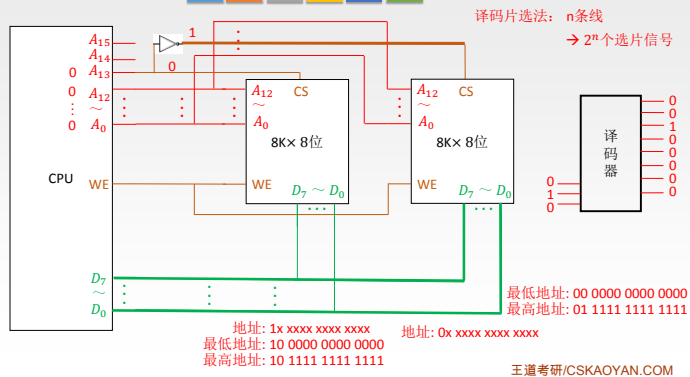
主存容量扩展-字扩展



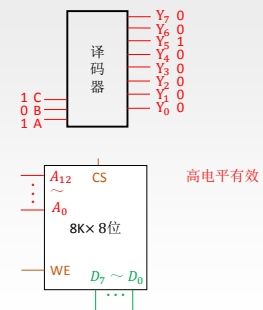
主存容量扩展-字扩展



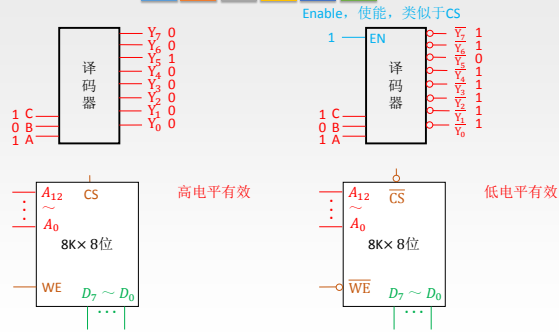
主存容量扩展-字扩展



主存容量扩展-译码器

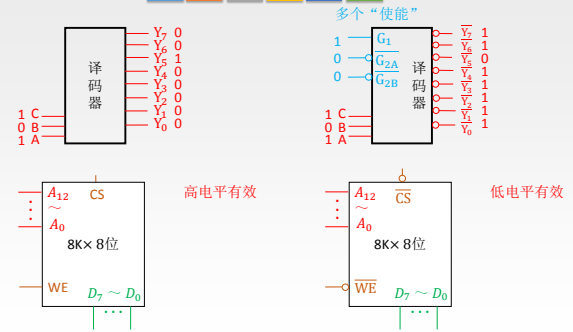


主存容量扩展-译码器



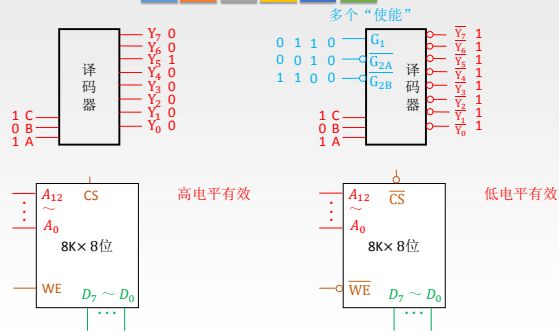
王道考研/CSKAOYAN.COM

主存容量扩展-译码器



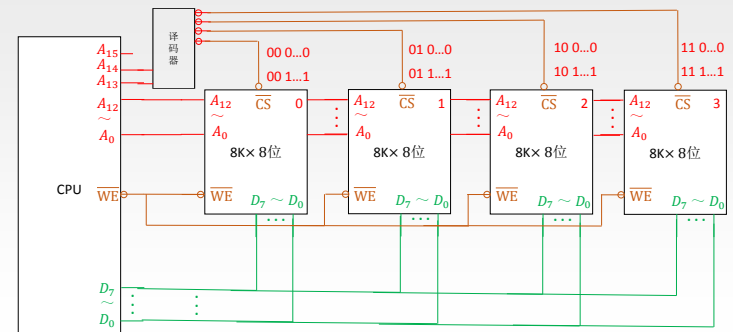
王道考研/CSKAOYAN.COM

主存容量扩展-译码器



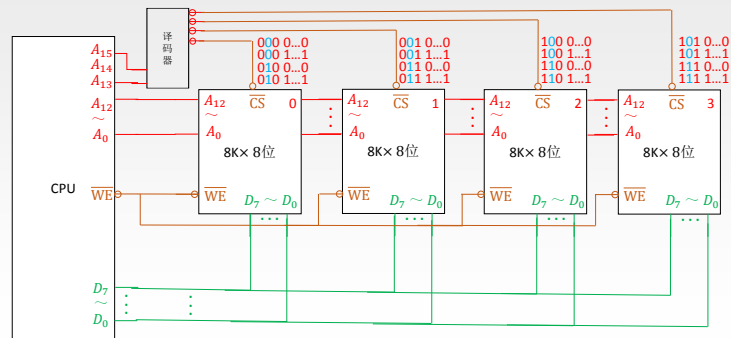
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



王道考研/CSKAOYAN.COM

主存容量扩展-字扩展



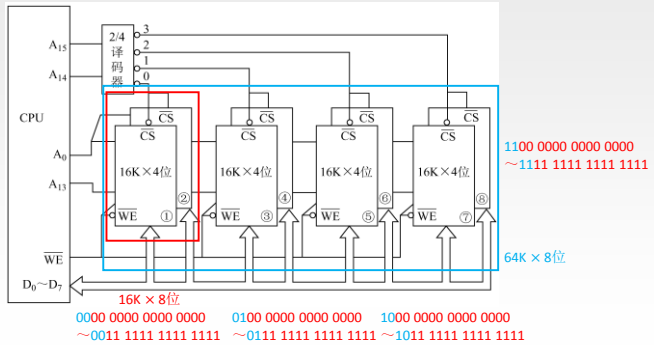
王道考研/CSKAOYAN.COM

主存容量扩展-字扩展

线选法	译码片选法
n条线→n个选片信号	n条线→2 ⁿ 个选片信号
电路简单	电路复杂
地址空间不连续	地址空间可连续, 可以增加逻辑设计

王道考研/CSKAOYAN.COM

主存容量扩展-字位同时扩展



王道考研/CSKAOYAN.COM

本节回顾



王道考研/CSKAOYAN.COM

本节内容

存储系统

主存与CPU
的连接-例题

王道考研/CSKAOYAN.COM

主存与CPU的连接

设CPU有16根地址线，8根数据线，并用MREQ作为访存控制信号（低电平有效），用WR作为读/写控制信号（高电平为读，低电平为写）。现有下列存储芯片：1K×4位RAM，4K×8位RAM，8K×8位RAM，2K×8位ROM，4K×8位ROM，8K×8位ROM及74LS138译码器和各种门电路。画出CPU与存储器的连接图，要求：

- 1) 主存地址空间分配：6000H~67FFH为系统程序区；6800H~6BFFH为用户程序区。
- 2) 合理选用上述存储芯片，说明各选几片？
- 3) 详细画出存储芯片的片选逻辑图。

补充：系统程序区用ROM，用户程序区用RAM

1. 确认地址线、数据线，选择存储芯片

数据线：CPU数据线8根 → 存储器位数应扩展为8位

地址分配：6000H~67FFH → 67FFH - 6000H + 1 = 800H, $8 \times 16^2 = 2^3 \times 2^8 = 2^{11} = 2K$

→ 用1片2K×8位ROM **ROM地址线11根**

6800H~6BFFH → 6BFFH - 6800H + 1 = 400H, $4 \times 16^2 = 2^2 \times 2^8 = 2^{10} = 1K$

→ 用2片1K×4位RAM，位扩展 **RAM地址线10根**

王道考研/CSKAOYAN.COM

主存与CPU的连接

设CPU有16根地址线，8根数据线，并用MREQ作为访存控制信号（低电平有效），用WR作为读/写控制信号（高电平为读，低电平为写）。现有下列存储芯片：1K×4位RAM，4K×8位RAM，8K×8位RAM，2K×8位ROM，4K×8位ROM，8K×8位ROM及74LS138译码器和各种门电路。画出CPU与存储器的连接图，要求：

- 1) 主存地址空间分配：6000H~67FFH为系统程序区；6800H~6BFFH为用户程序区。

补充：系统程序区用ROM，用户程序区用RAM

1. 确认地址线、数据线，选择存储芯片

数据线：CPU数据线8根 → 存储器位数应扩展为8位

地址分配：6000H~67FFH → 67FFH - 6000H + 1 = 800H, $8 \times 16^2 = 2^3 \times 2^8 = 2^{11} = 2K$

→ 用1片2K×8位ROM **ROM地址线11根**

6800H~6BFFH → 6BFFH - 6800H + 1 = 400H, $4 \times 16^2 = 2^2 \times 2^8 = 2^{10} = 1K$

→ 用2片1K×4位RAM，位扩展 **RAM地址线10根**

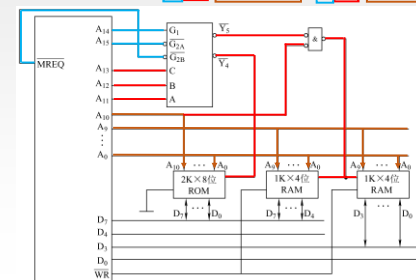
地址线：6000H~67FFH → **0110 0000 0000 0000** ~ **0110 0111 1111 1111** \bar{V}_4
6800H~6BFFH → **0110 1000 0000 0000** ~ **0110 1011 1111 1111** \bar{V}_5

王道考研/CSKAOYAN.COM

主存与CPU的连接

补充：系统程序区用ROM，用户程序区用RAM

地址线：6000H~67FFH → **0110 0000 0000 0000** ~ **0110 0111 1111 1111** \bar{V}_4
6800H~6BFFH → **0110 1000 0000 0000** ~ **0110 1011 1111 1111** \bar{V}_5 $A_{10}=0$ & $\bar{V}_5=0$



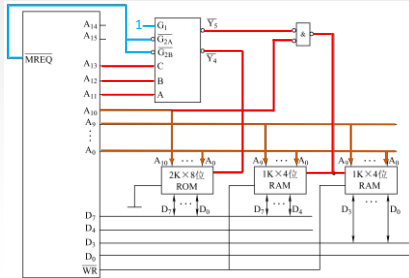
王道考研/CSKAOYAN.COM

主存与CPU的连接

补充：系统程序区用ROM，用户程序区用RAM

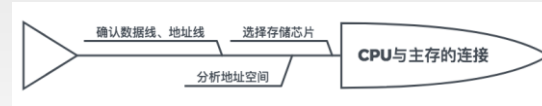
地址线：6000H~67FFH → 0110 000 0000 0000 ~ 0110 0111 1111 1111 \overline{V}_4

6800H~6BFFH → 0110 1000 0000 0000 ~ 0110 1011 1111 1111 \overline{V}_5 $A_{10}=0$ & $\overline{V}_5=0$

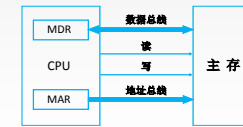


王道考研/CSKAOYAN.COM

本节回顾



系统程序区用ROM，用户程序区用RAM



王道考研/CSKAOYAN.COM

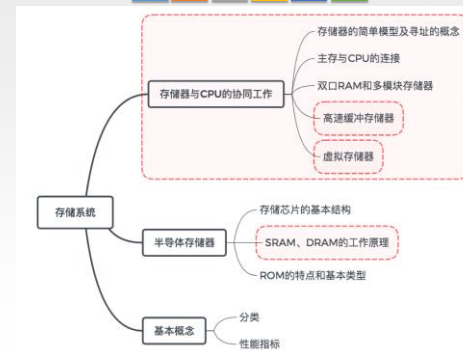
本节内容

存储系统

双口RAM
多模块存储器

王道考研/CSKAOYAN.COM

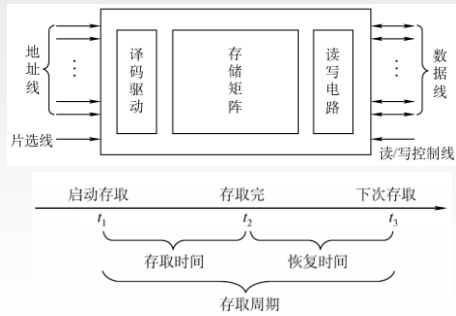
本章总览



提高存储器的工作速度

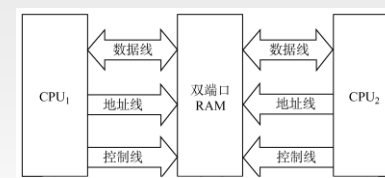
王道考研/CSKAOYAN.COM

存取周期



王道考研/CSKAOYAN.COM

双端口RAM

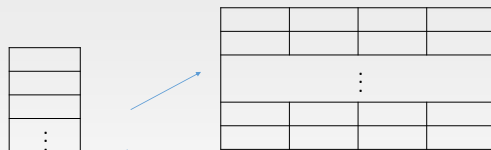


- 两个端口对同一主存操作有以下4种情况:
1. 两个端口同时对同一地址单元存取数据。⊗
 2. 两个端口同时对同一地址单元读出数据。⊗
 3. 两个端口同时对同一地址单元写入数据。⊗
 4. 两个端口同时对同一地址单元，一个写入数据，另一个读出数据。⊗

解决方法：置“忙”信号为0，由判断逻辑决定暂时关闭一个端口（即被延时），未被关闭的端口正常访问，被关闭的端口延长一个很短的时间后再访问。

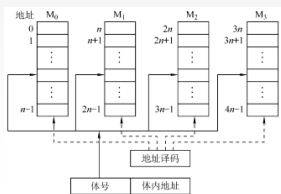
王道考研/CSKAOYAN.COM

多模块存储器



普通存储器：每行为1个存储单元

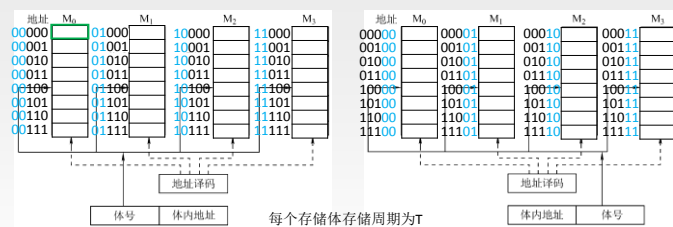
单体多字存储器
每个存储单元存储 m 个字
总线宽度也为 m 个字
一次并行读出 m 个字
⊗指令和数据在主存内必须是连续存放的



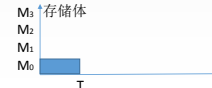
多体并行存储器
每个模块都有相同的容量和存取速度。
各模块都有独立的读写控制电路、地址寄存器和数据寄存器。
它们既能并行工作，又能交叉工作。

王道考研/CSKAOYAN.COM

多体并行存储器



高位交叉编址的多体存储器



连续访问：

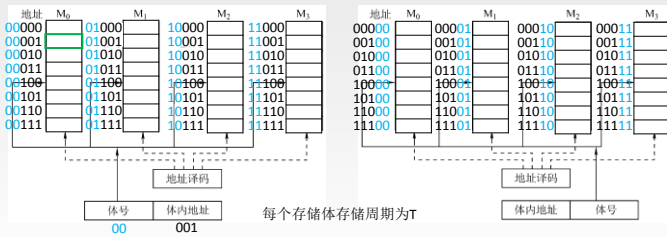


低位交叉编址的多体存储器

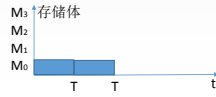


王道考研/CSKAOYAN.COM

多体并行存储器



高位交叉编址的多体存储器



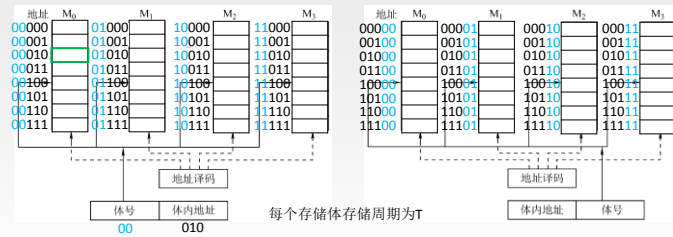
连续访问:

00000
00001
00010
00011
00100

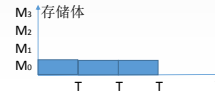
低位交叉编址的多体存储器

王道考研/CSKAOYAN.COM

多体并行存储器



高位交叉编址的多体存储器



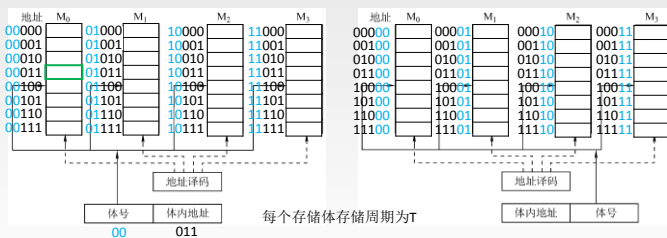
连续访问:

00000
00001
00010
00011
00100

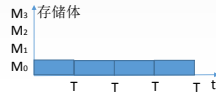
低位交叉编址的多体存储器

王道考研/CSKAOYAN.COM

多体并行存储器



高位交叉编址的多体存储器



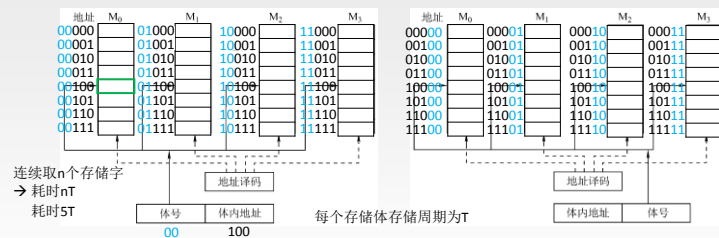
连续访问:

00000
00001
00010
00011
00100

低位交叉编址的多体存储器

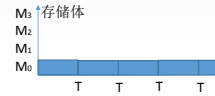
王道考研/CSKAOYAN.COM

多体并行存储器



连续取 n 个存储字
→ 耗时 nT
耗时 ST

高位交叉编址的多体存储器



连续访问:

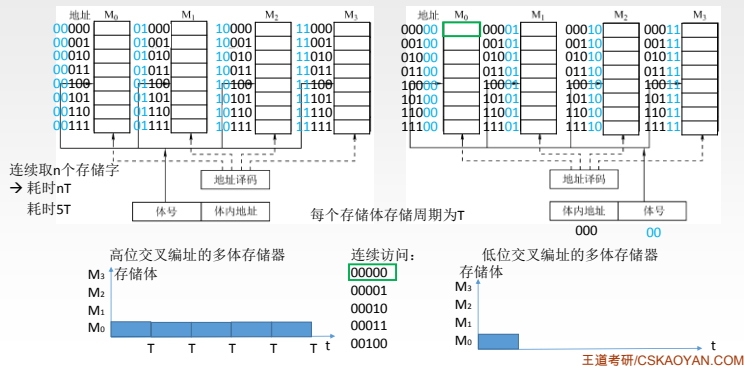
00000
00001
00010
00011
00100

低位交叉编址的多体存储器

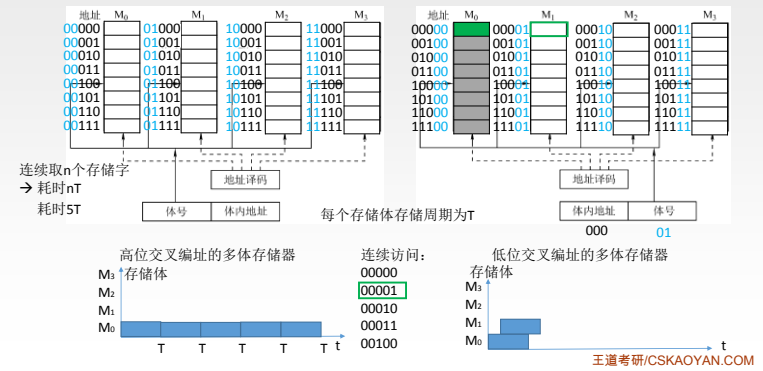


王道考研/CSKAOYAN.COM

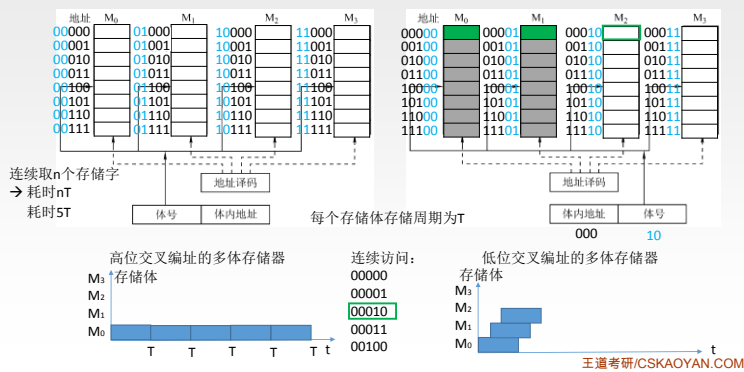
多体并行存储器



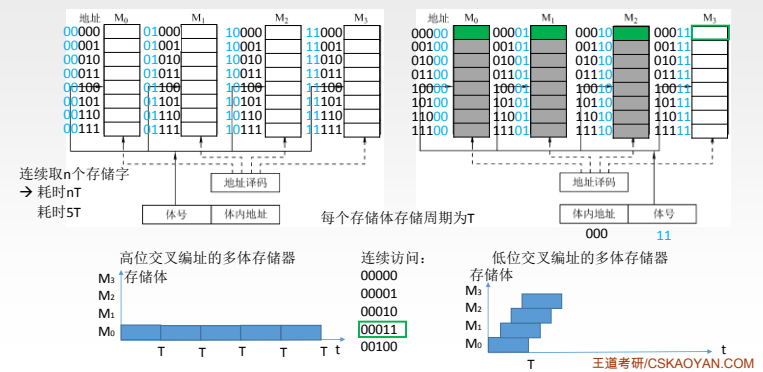
多体并行存储器



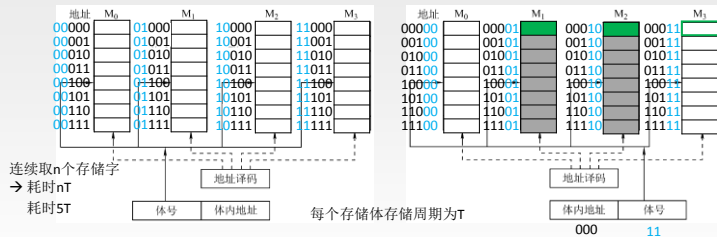
多体并行存储器



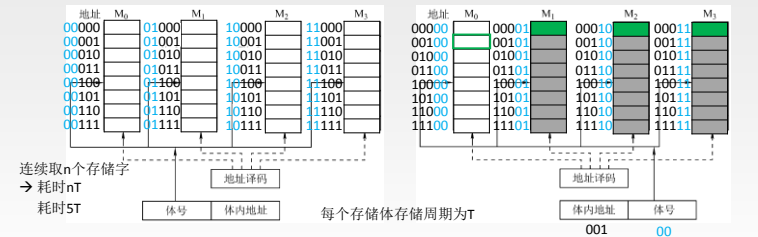
多体并行存储器



多体并行存储器



多体并行存储器



多体并行存储器

流水线

微观(计算题)

模块数 $m=4$, 存储周期为 T , 字长 W , 数据总线宽度为 W , 总线传输周期为 r , 连续存取 n 个字, 求交叉存储器的带宽。

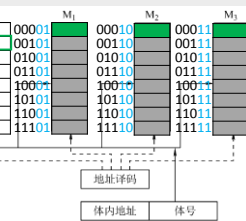
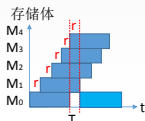
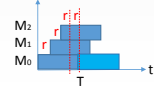
有 m 个存储体, 存储周期为 T , 字长 W , 每隔 r 时间启动下一个存储体, 连续存取 n 个字, 求存储器的存取速率。

连续存取 n 个字耗时 $=T+(n-1)r$ $m \geq T/r$

$m < T/r$

$m > T/r$

存储体



多体并行存储器

流水线

微观(计算题)

模块数 $m=4$, 存储周期为 T , 字长 W , 数据总线宽度为 W , 总线传输周期为 r , 连续存取 n 个字, 求交叉存储器的带宽。

有 m 个存储体, 存储周期为 T , 字长 W , 每隔 r 时间启动下一个存储体, 连续存取 n 个字, 求存储器的存取速率。

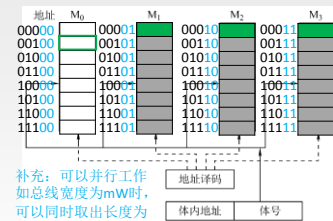
连续存取 n 个字耗时 $=T+(n-1)r$ $m \geq T/r$

带宽 $= \frac{n \times W}{T+(n-1)r}$

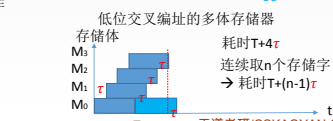
n 较大时, 带宽 $\rightarrow W/r$ 相当于 m 个存储体并行工作对比单个存储体的带宽: W/T

宏观(概念题)

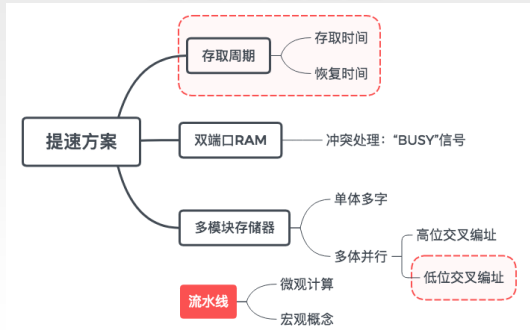
一个存储周期内, 交叉存储器可以提供的数据量为单个模块的 m 倍。



补充: 可以并行工作
如总线宽度为 mW 时,
可以同时取出长度为
 mW 的数据



本节回顾



王道考研/CSKAOYAN.COM