

计算机组成原理

(第四讲-1)



厦门大学软件学院 曾文华

2021年3月10日



第2篇 计算机系统的硬件结构

第3章 系统总线

第4章 存储器

第3章 输入输出系统



第4章 存储器

共87页

4.1 概述

4.2 主存储器

4.3 高速缓冲存储器

4.4 辅助存储器



4.1 概 述

- 一、存储器分类
- 二、存储器的层次结构

一、存储器分类

1. 按存储介质分类

(1) 半导体存储器

TTL、MOS

易失

(2) 磁表面存储器

磁头、载磁体

(3) 磁芯存储器

硬磁材料、环状元件

(4) 光盘存储器

激光、磁光材料

非易失

TTL: Transistor Transistor Logic
晶体管-晶体管逻辑

MOS: Metal Oxide Semiconductor
金属-氧化物-半导体

2. 按存取方式分类

(1) 存取时间与物理地址无关（随机访问）

RAM: Random Access Memory

随机存储器**RAM** 在程序的执行过程中可读可写

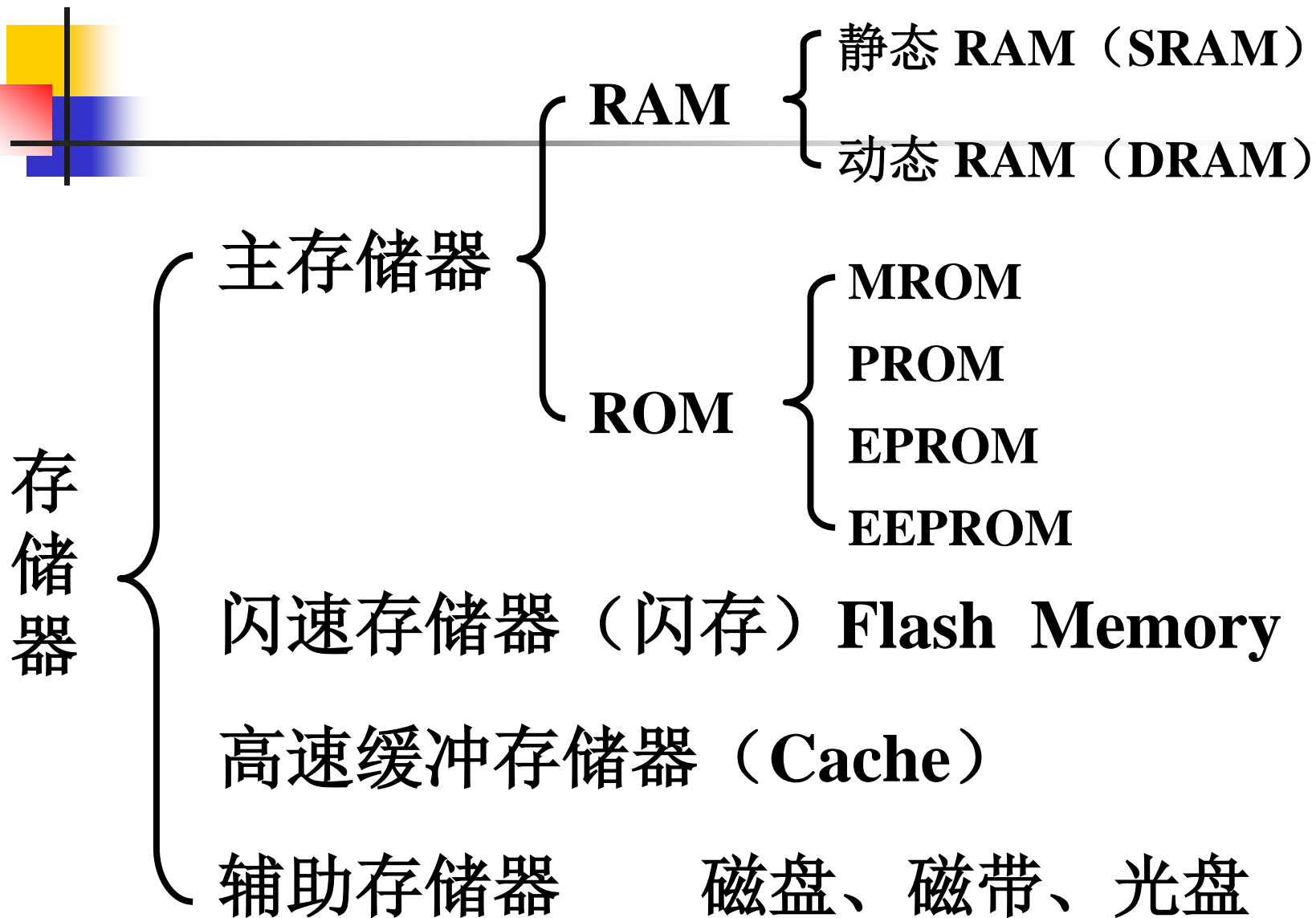
ROM: Read Only Memory

只读存储器**ROM** 在程序的执行过程中只读

(2) 存取时间与物理地址有关（串行访问）

顺序存取存储器	磁带	} 串行访问存储器
直接存取存储器	磁盘	

3. 按在计算机中的作用分类



闪存和闪存卡



闪存 (Flash Memory) 是一种非易失性存储器，即断电数据也不会丢失。因为闪存不像**RAM** (随机存取存储器) 一样以字节为单位改写数据，因此不能取代**RAM**。

闪存卡 (Flash Card) 是利用闪存技术达到存储电子信息的存储器，一般应用在数码相机，掌上电脑，**MP3**等小型数码产品中作为存储介质，所以样子小巧，有如一张卡片，所以称之为闪存卡。根据不同的生产厂商和不同的应用，闪存卡大概有**SmartMedia (SM卡)**、**Compact Flash (CF卡)**、**MultiMediaCard (MMC卡)**、**Secure Digital (SD卡)**、**Memory Stick (记忆棒)**、**XD-Picture Card (XD卡)**和**微硬盘 (MICRODRIVE)** 这些闪存卡虽然外观、规格不同，但是技术原理都是相同的。



CF卡



MMC卡



SD卡



SM卡



记忆棒



XD卡



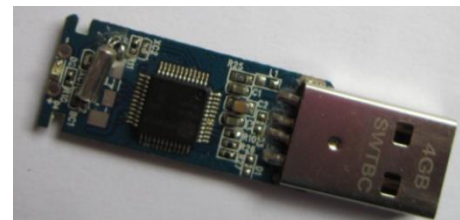
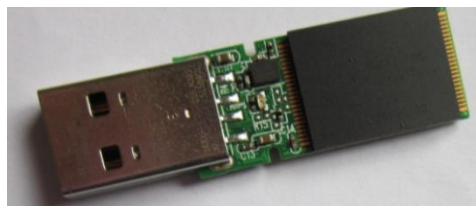
微硬盘

闪存和U盘

U盘，全称**USB闪存盘**，英文名“**USB flash disk**”。它是一种使用**USB**接口的无需物理驱动器的微型高容量移动存储产品，通过**USB**接口与电脑连接，实现即插即用。

U盘的组成很简单，主要由外壳+机芯组成，其中：

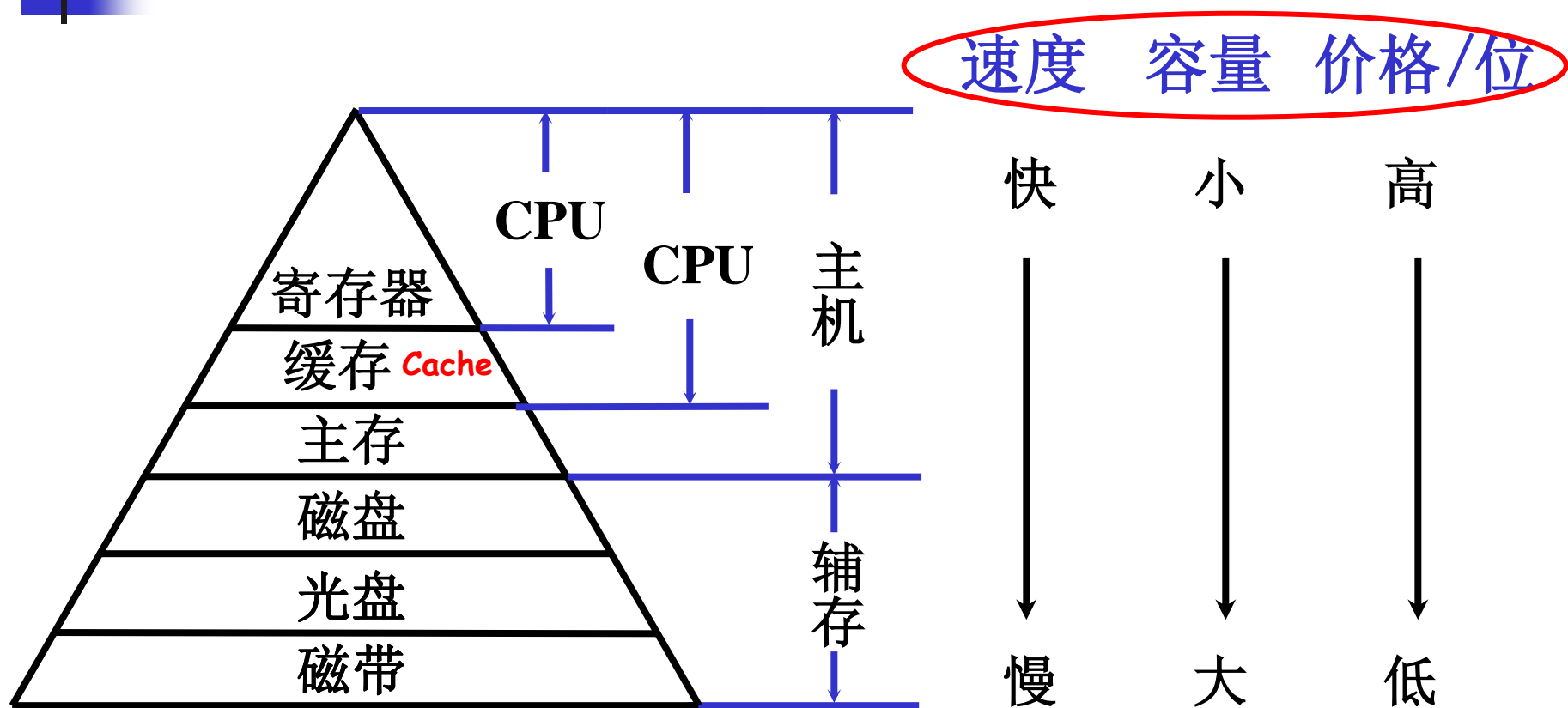
1. **机芯**：机芯包括一块**PCB+USB**主控芯片+晶振+贴片电阻、电容+**USB**接口+贴片**LED**（不是所有的U盘都有）+**FLASH（闪存）芯片**；



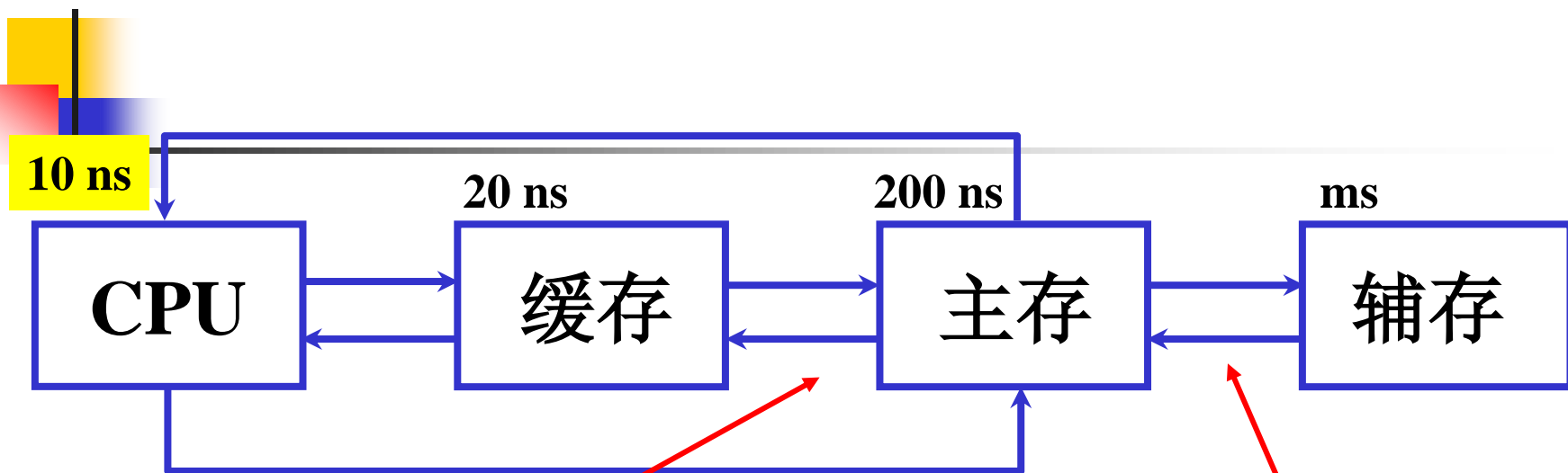
2. **外壳**：按材料分类，有**ABS**塑料、竹木、金属、皮套、硅胶、**PVC**软件等；按风格分类，有卡片、笔型、迷你、卡通、商务、仿真等；按功能分类，有加密、杀毒、防水、智能等。

二、存储器的层次结构

1. 存储器三个主要特性的关系



2. 缓存 — 主存层次和主存 — 辅存层次



主存和缓存之间的数据调动是由硬件自动完成的，对系统程序员和应用程序员都是透明的

(速度)
缓存 — 主存

(容量)
主存 — 辅存

主存储器

虚拟存储器

实地址

虚地址

物理地址

逻辑地址

主存和辅存之间的数据调动是由硬件和操作系统共同完成的，对系统程序员是不透明的、对应用程序员是透明的

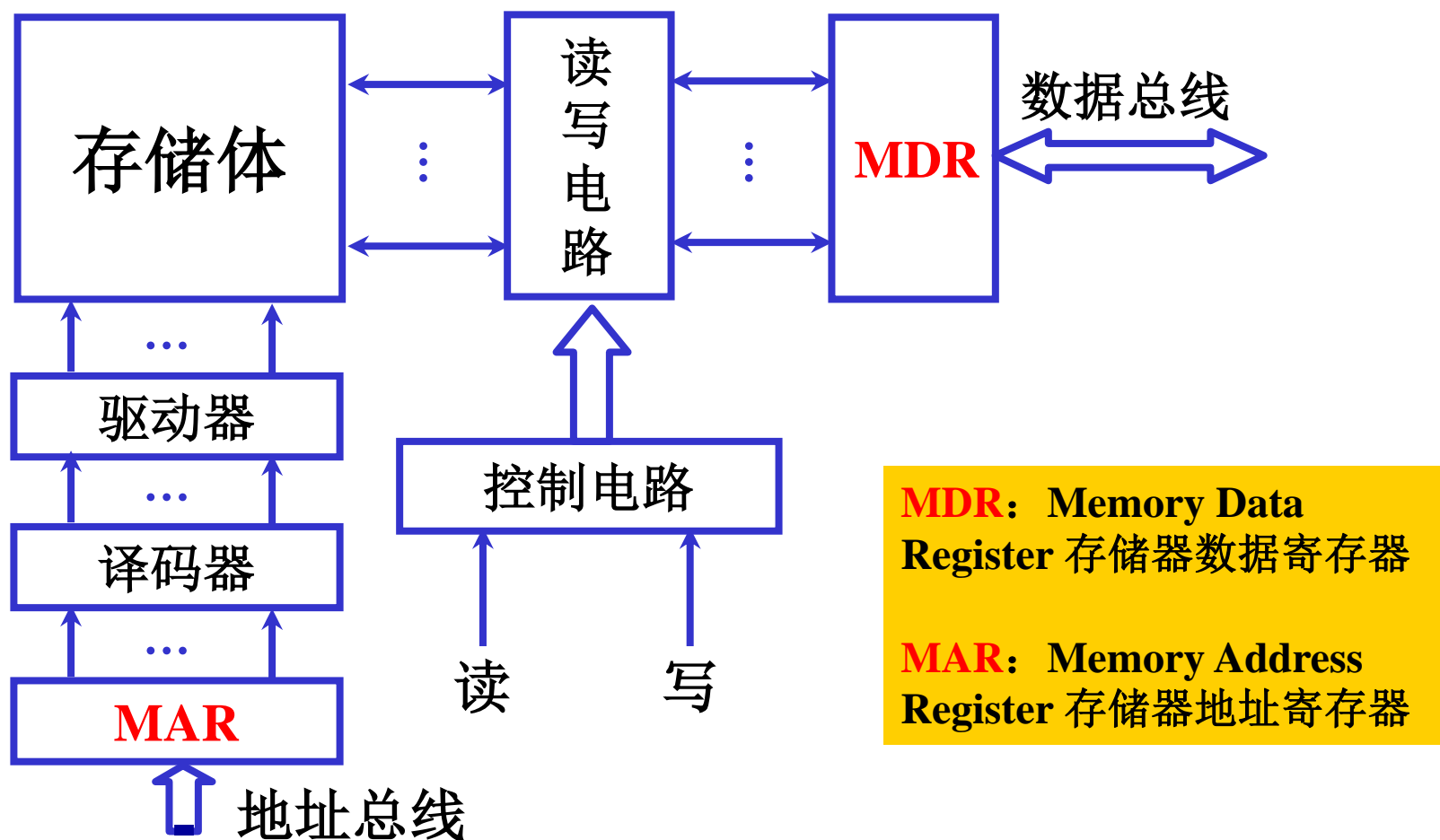


4.2 主存储器

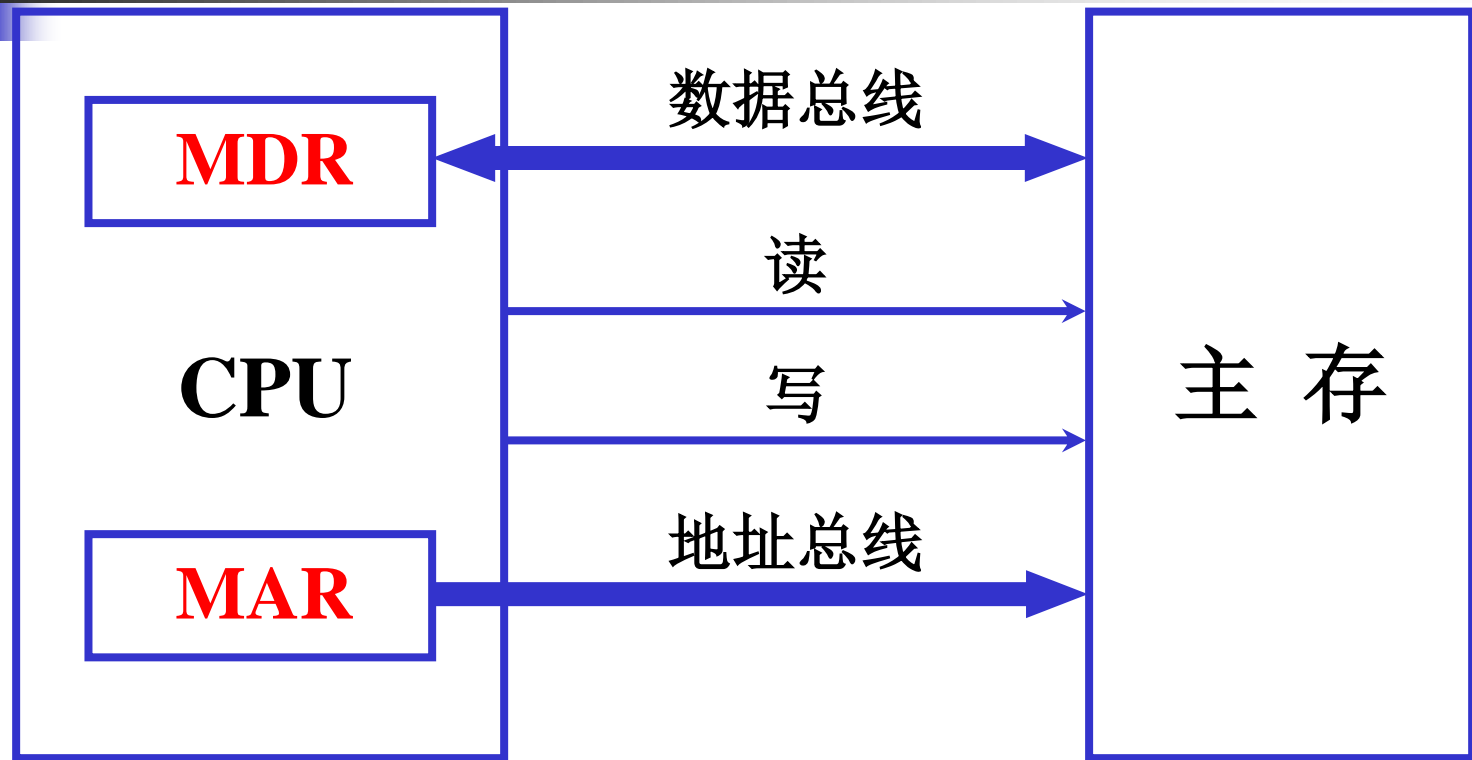
- 一、概述
- 二、半导体存储芯片简介
- 三、随机存取存储器 (**RAM**)
- 四、只读存储器 (**ROM**)
- 五、存储器与 CPU 的连接
- 六、存储器的校验
- 七、提高访存速度的措施

一、概述

1. 主存的基本组成



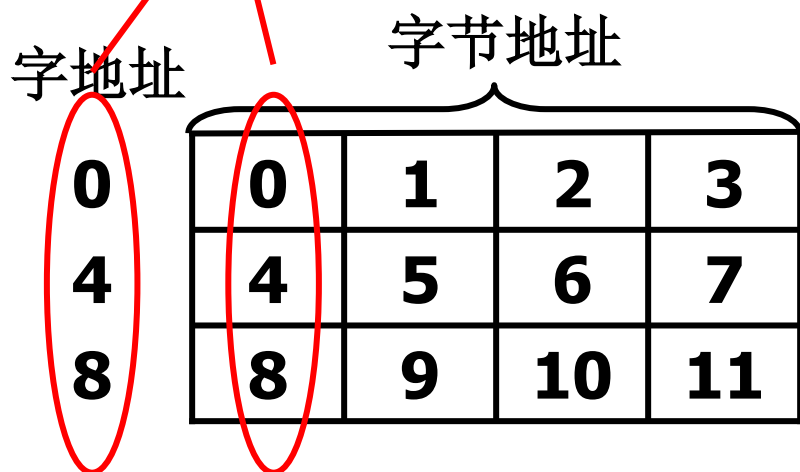
2. 主存和 CPU 的联系



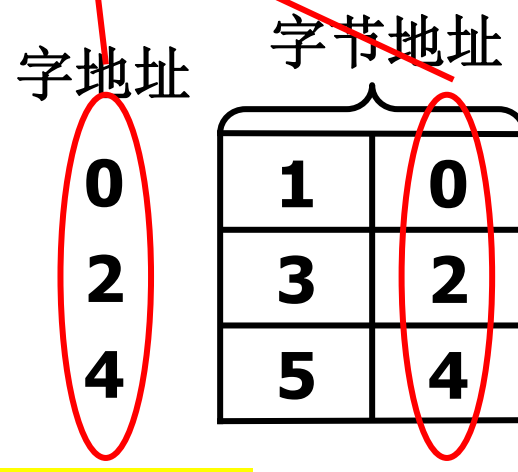
3. 主存中存储单元地址的分配

高位字节 地址 作为 字地址

低位字节 地址 作为 字地址



1字=4字节



1字=2字节

设地址线 24 根

按 字节 寻址 $2^{24} = 16 \text{ M}$

若字长为 16 位

按 字 寻址

8 M

1字=2字节

若字长为 32 位

按 字 寻址

4 M

1字=4字节

高位字节 地址作为字地址

字节寻址

字寻址
(1个字=16位)

字寻址
(1个字=32位)

0	11
1	22
2	33
3	44
4	55
5	66
6	77
7	88
8	

[illegible][illegible]

低位字节 地址作为字地址

字节寻址

字寻址
(1个字=16位)

字寻址
(1个字=32位)

0	11
1	22
2	33
3	44
4	55
5	66
6	77
7	88
8	

[illegible][illegible]

4. 主存的技术指标

(1) 存储容量

主存 存放二进制代码的总位数

(2) 存储速度

• 存取时间

存储器的 访问时间

MAT(Memory Access Time)

读出时间 写入时间

MCT > MAT

• 存取周期

连续两次独立的存储器操作

MCT(Memory Cycle Time)

(读或写) 所需的 最小间隔时间

读周期 写周期

(3) 存储器的带宽

位/秒

16b/500ns=32Mb/s

存取周期为500ns、每个存取周期可访问16位，则带宽为32Mb/s

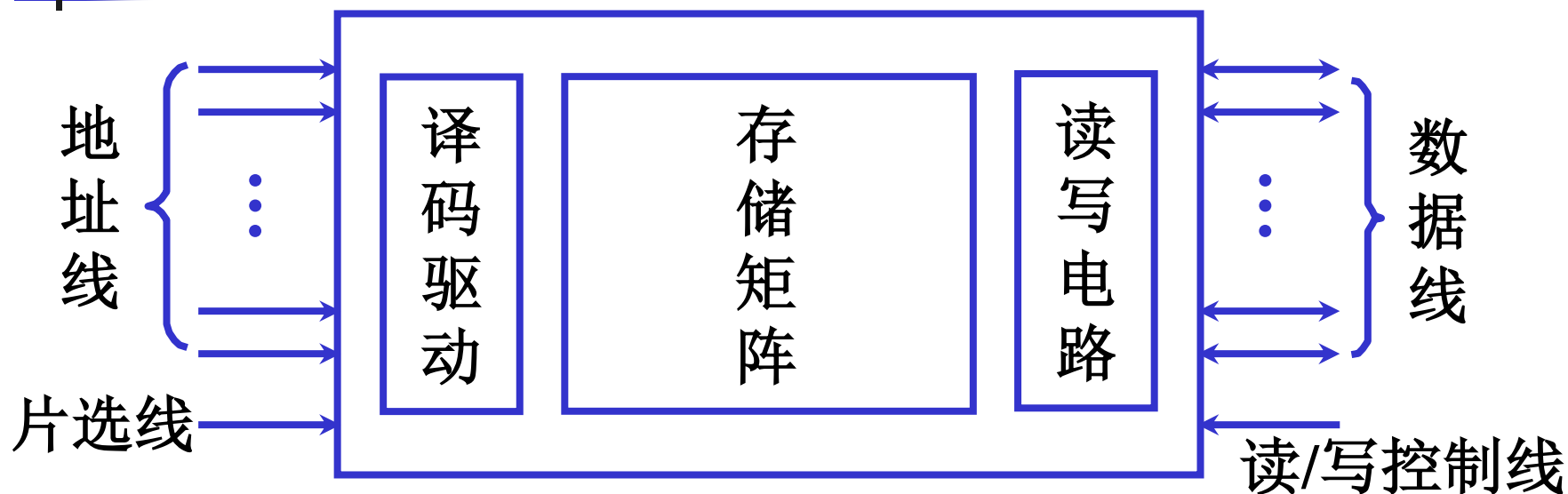


提高存储器带宽的措施:

- 缩短存取周期(**MCT**)
- 增加存储字长, 使每个存取周期可读/写更多的二进制位数 (**8位**、**16位**、**32位**、.....)
- 增加存储体 (见教材的**4.2.7**小节)

二、半导体存储芯片简介

1. 半导体存储芯片的基本结构



地址线（单向）

数据线（双向）

芯片容量

10

4

1K×4位

$2^{10}=1K$

14

1

16K×1位

$2^{14}=16K$

13

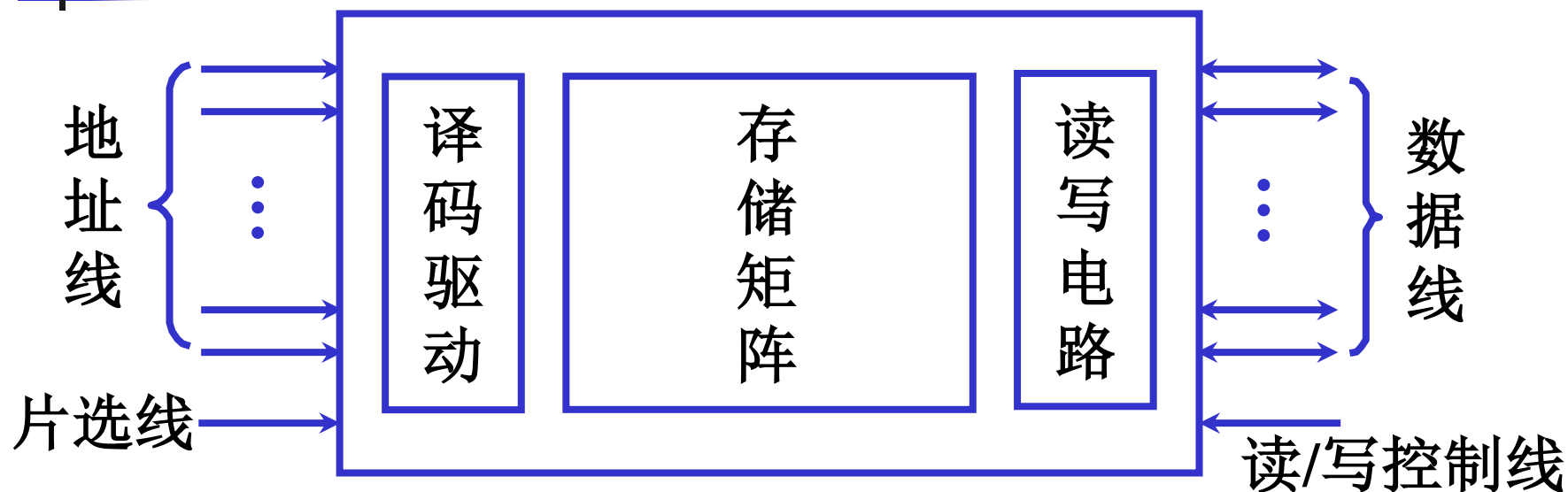
8

8K×8位

$2^{13}=8K$

二、半导体存储芯片简介

1. 半导体存储芯片的基本结构（续）



片选线

$\overline{\text{CS}}$

$\overline{\text{CE}}$

Chip Select

Chip Enable

读/写控制线

$\overline{\text{WE}}$

(低电平写、高电平读)

$\overline{\text{OE}}$

(允许读)

$\overline{\text{WE}}$

(允许写)

Output Enable

Write Enable

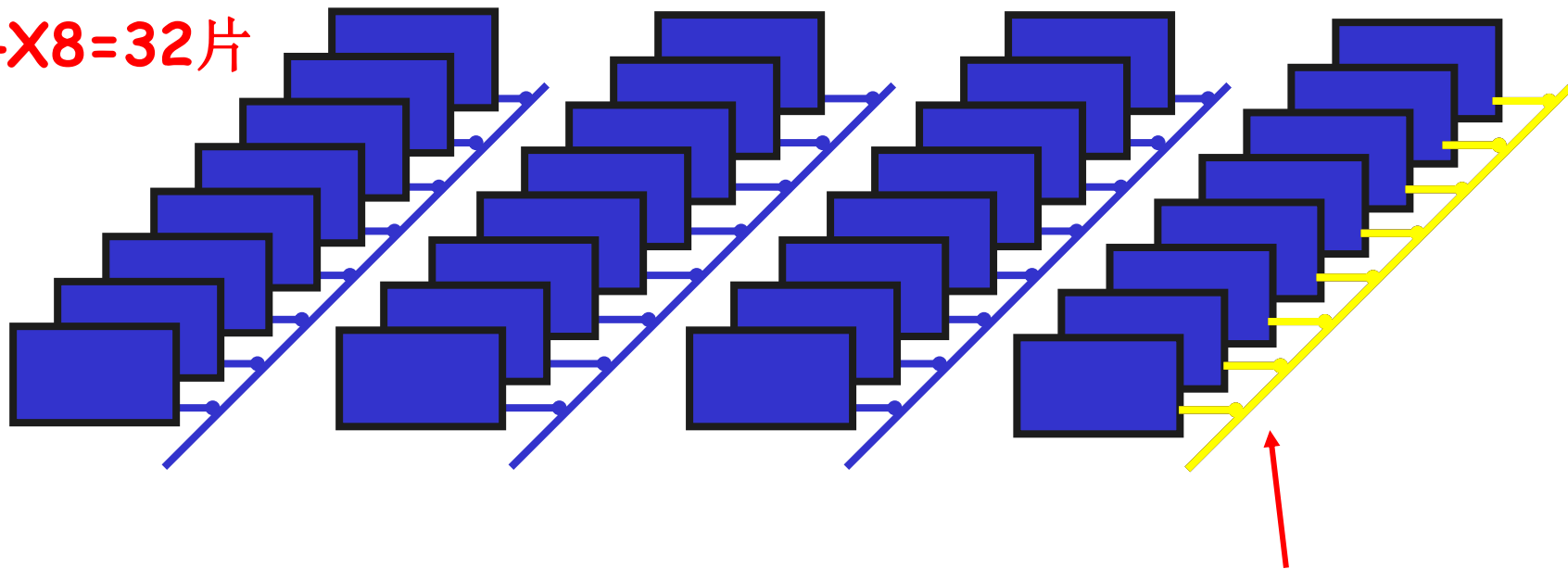
存储芯片片选线的作用

用 $16\text{K} \times 1$ 位的存储芯片组成 $64\text{K} \times 8$ 位的存储器

32片

$4 \times 8 = 32$ 片

8片 $16\text{K} \times 1$ 位 8片 $16\text{K} \times 1$ 位 8片 $16\text{K} \times 1$ 位 8片 $16\text{K} \times 1$ 位



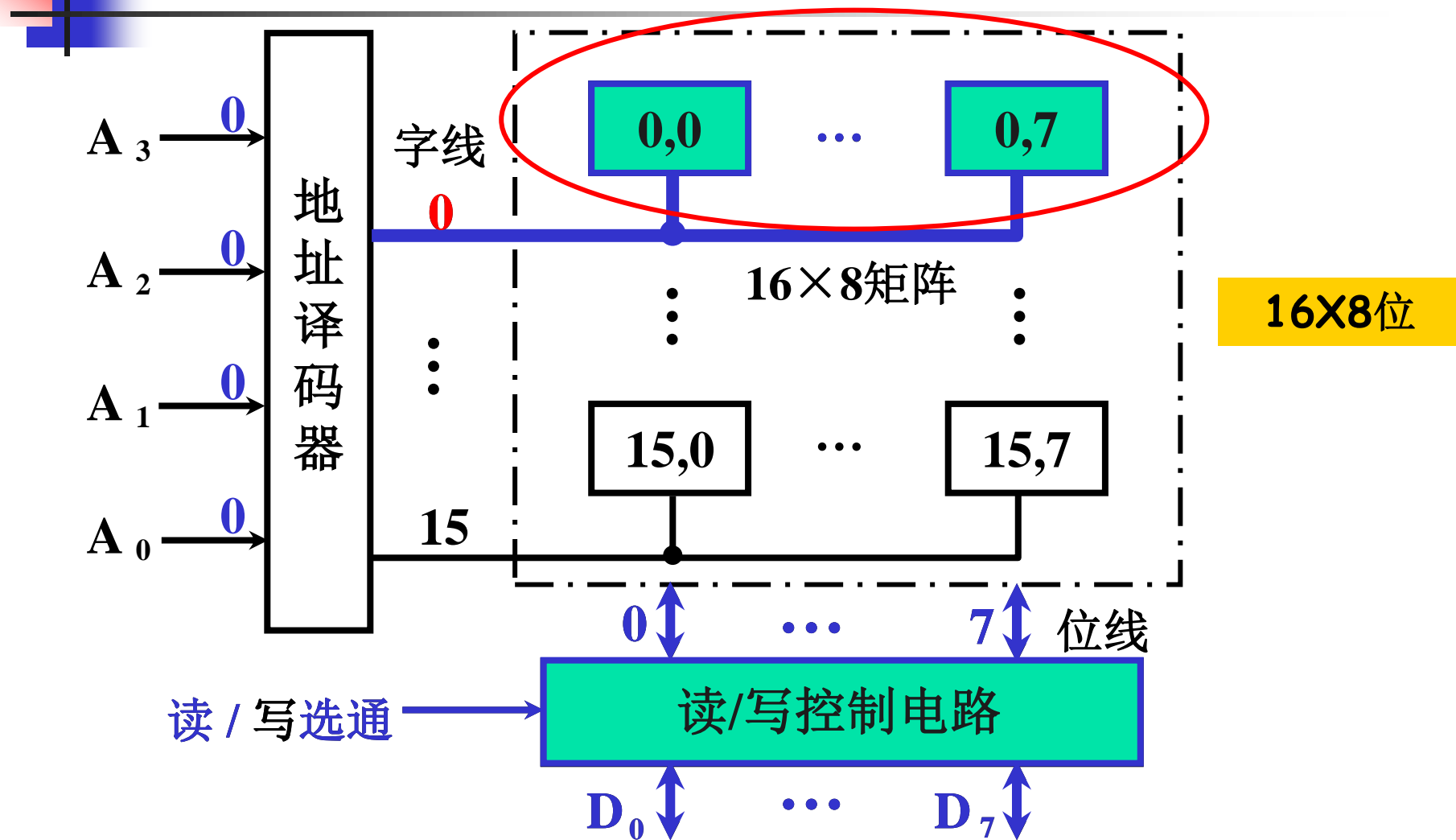
当地址为 65,535 时，此 8 片的片选有效

2. 半导体存储芯片的译码驱动方式

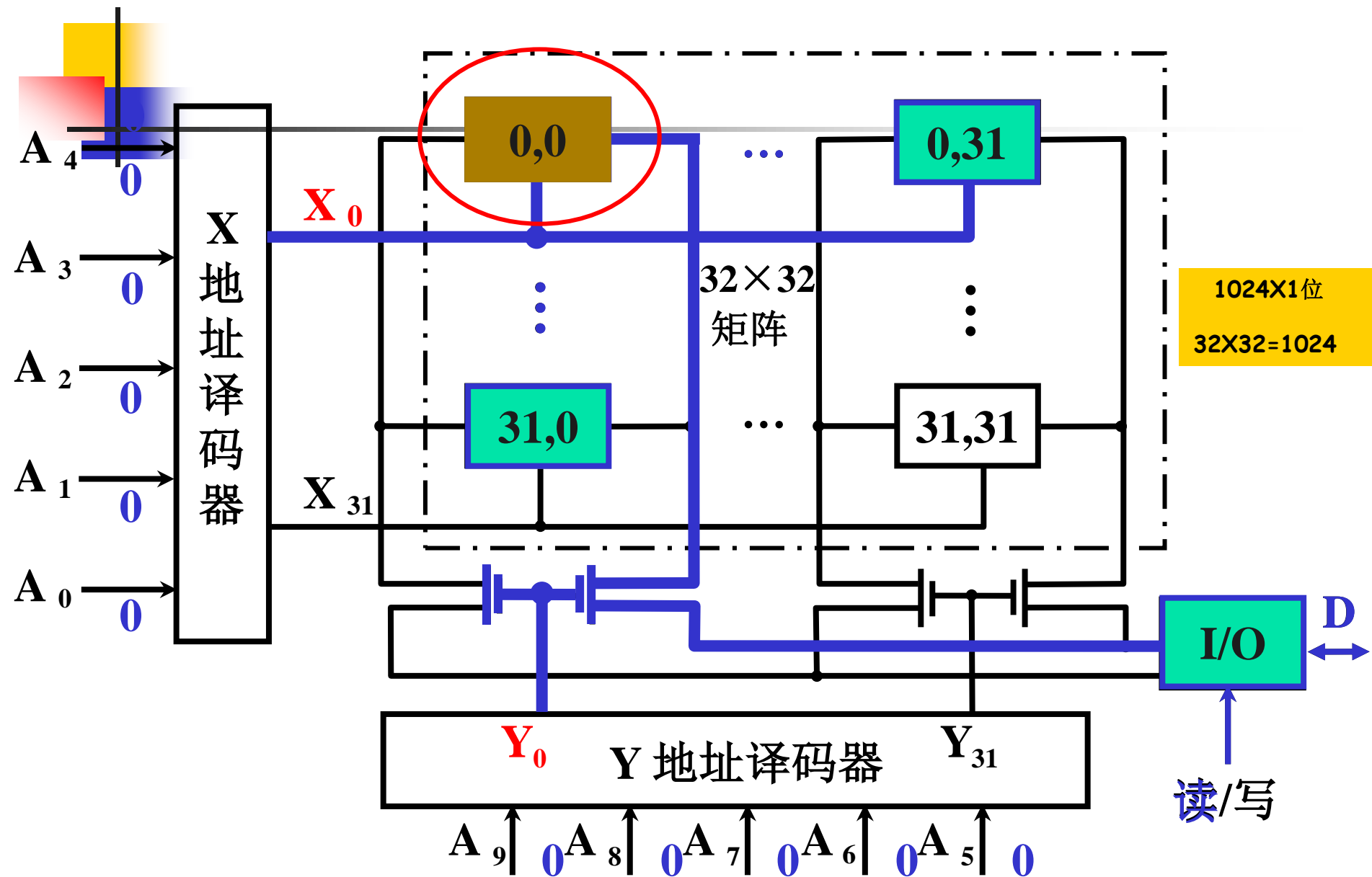
(1) 线选法

(2) 重合法

(1) 线选法



(2) 重合法



三、随机存取存储器 (RAM)

MOS管 T_1 - T_4 构成触发器基本电路

Static RAM

1. 静态 RAM (SRAM)

(1) 静态 RAM 基本电路

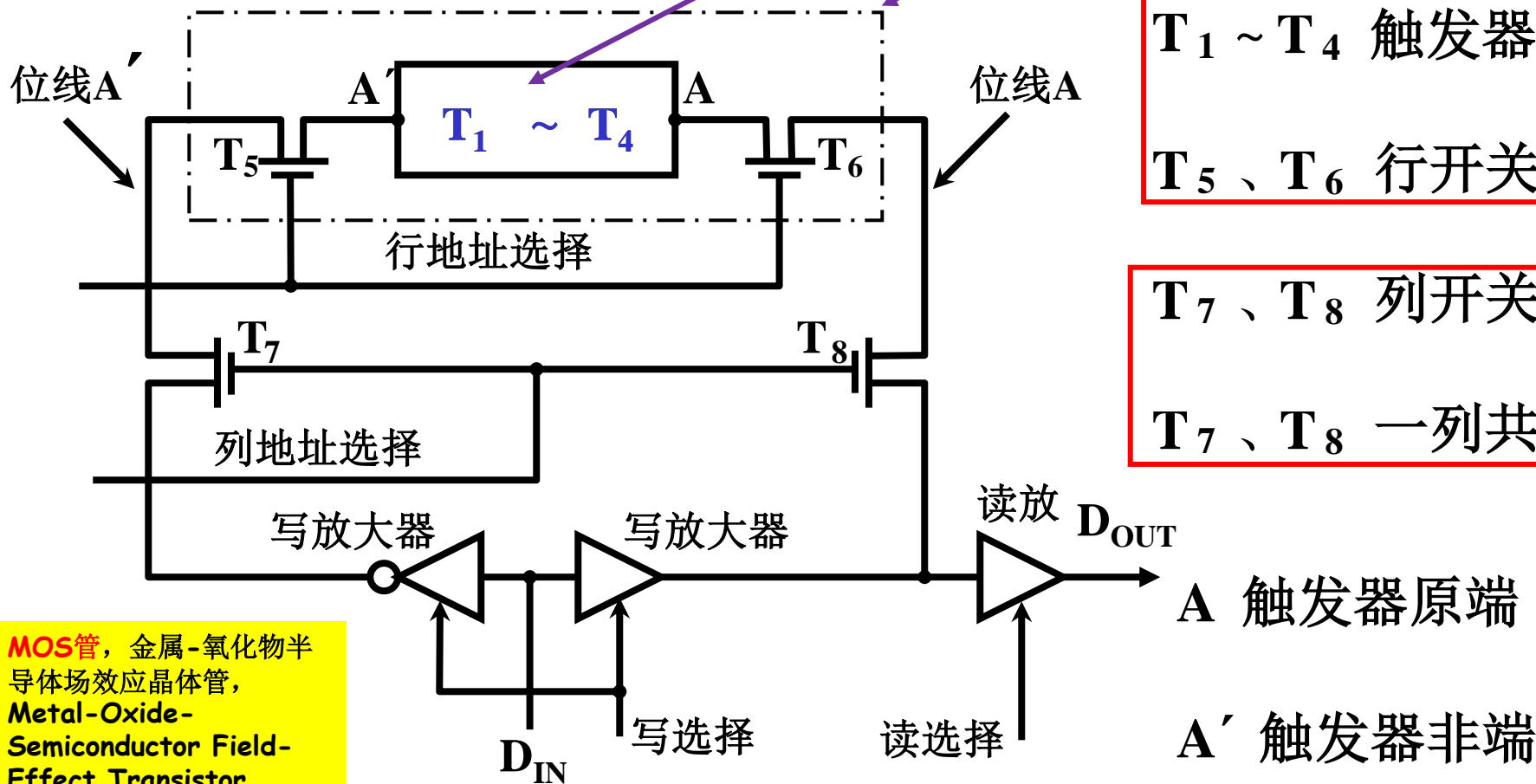
6个MOS管 (T_1 - T_6) 组成存放1位二进制数 (0或1) 的基本电路 (虚线框内)

$T_1 \sim T_4$ 触发器

T_5 、 T_6 行开关

T_7 、 T_8 列开关

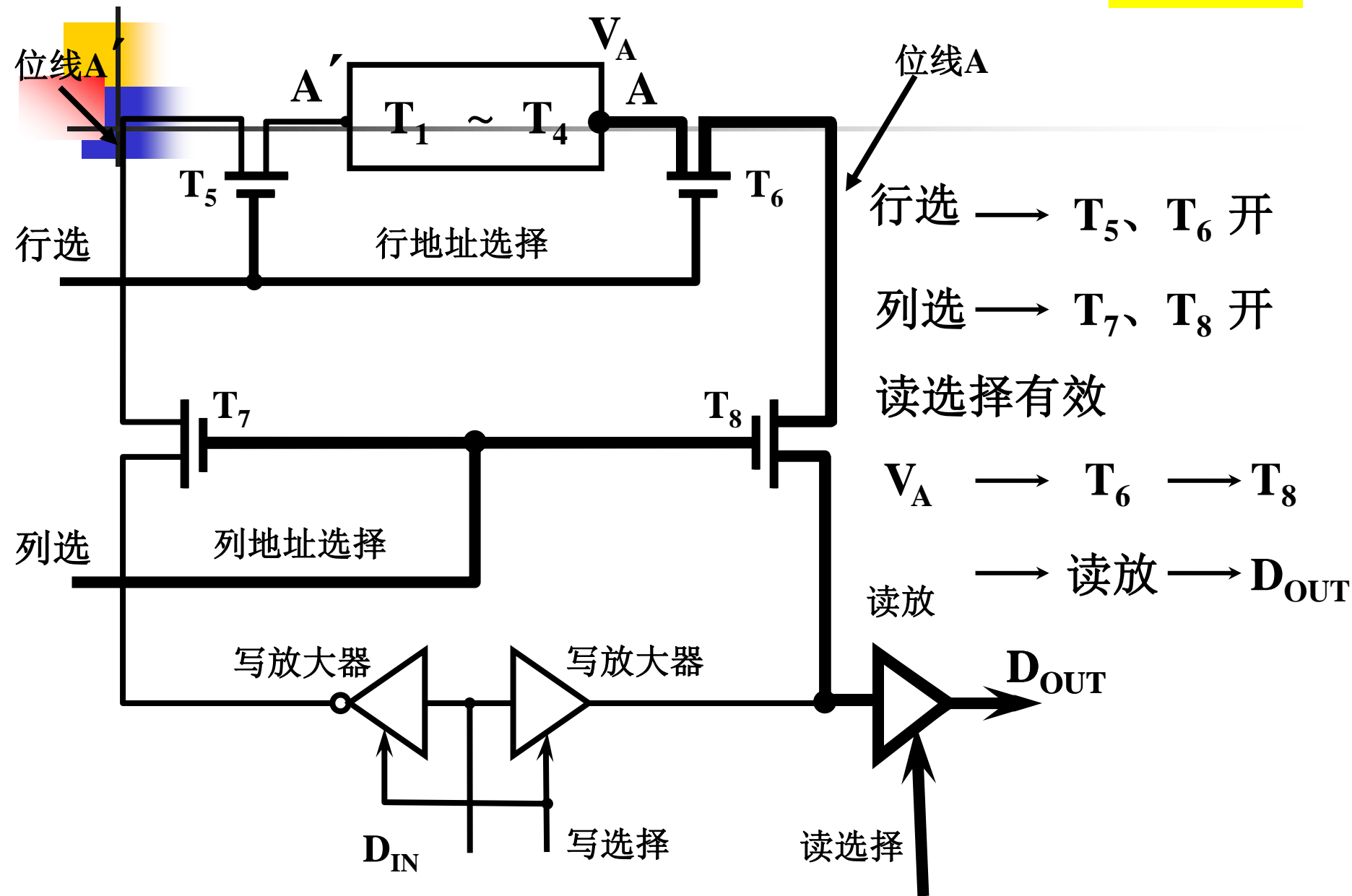
T_7 、 T_8 一列共用



MOS管, 金属-氧化物半导体场效应晶体管,
Metal-Oxide-Semiconductor Field-Effect Transistor

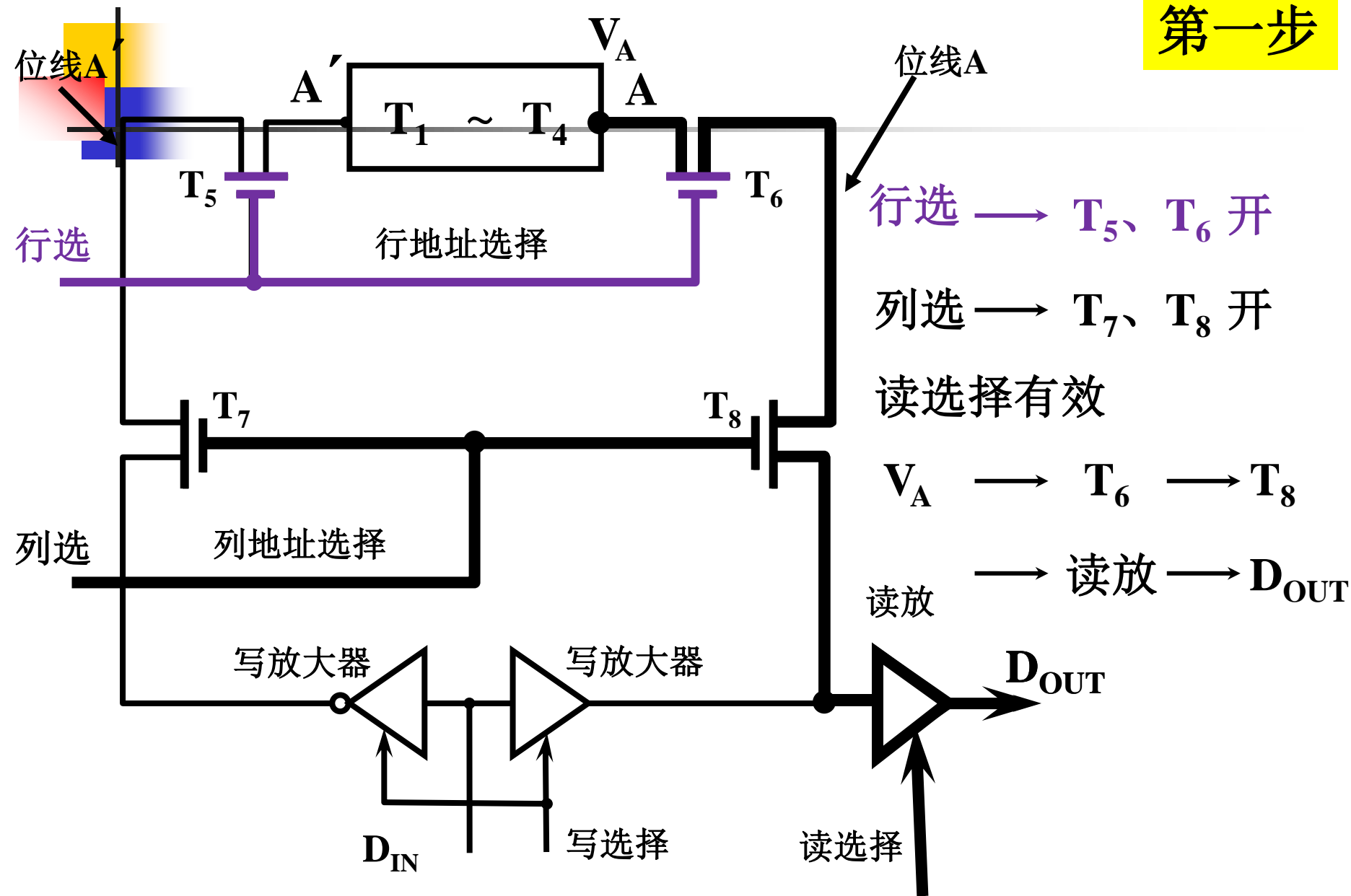
① 静态 RAM 基本电路的 读 操作

共四步



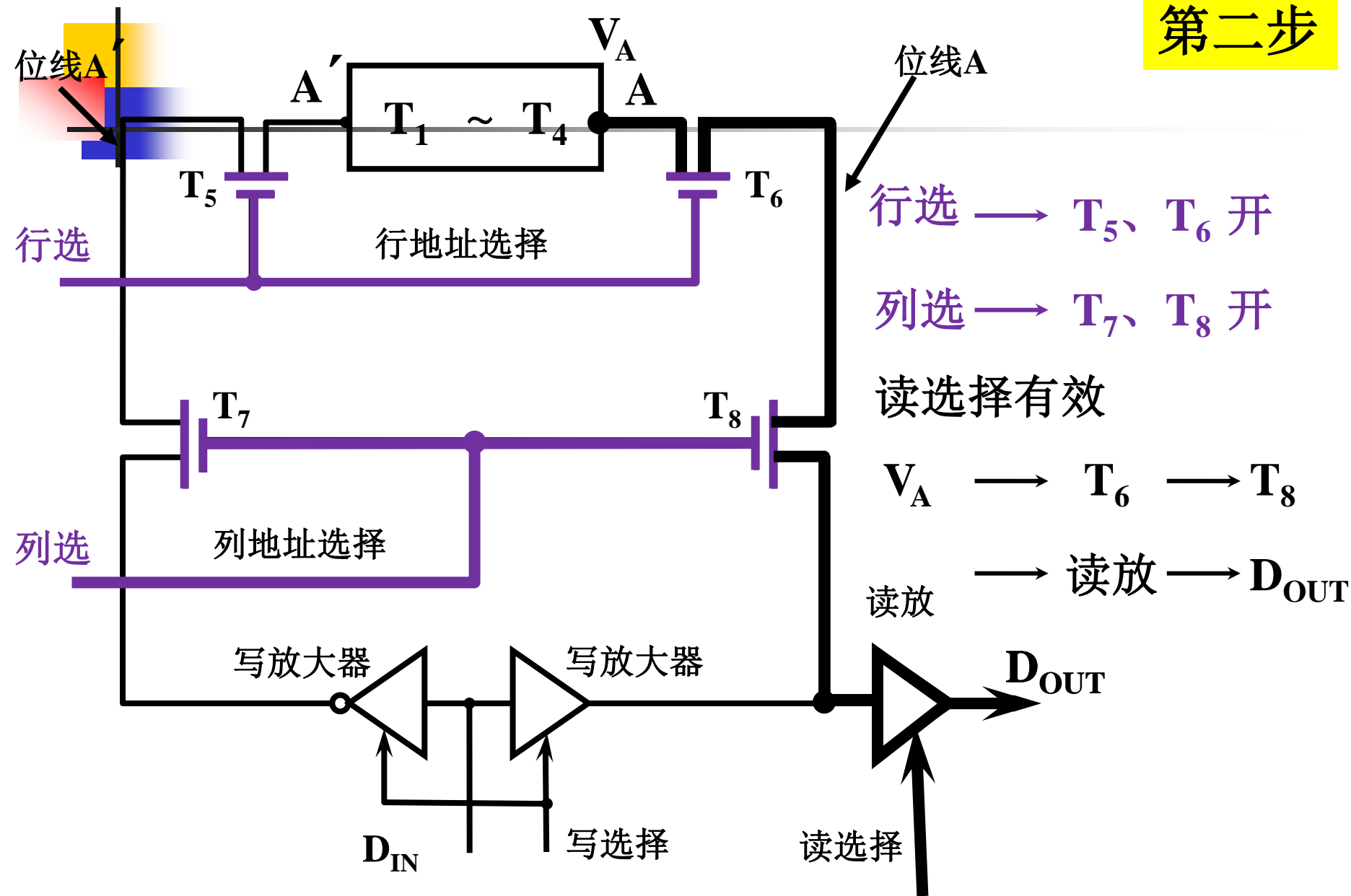
① 静态 RAM 基本电路的 读 操作

第一步



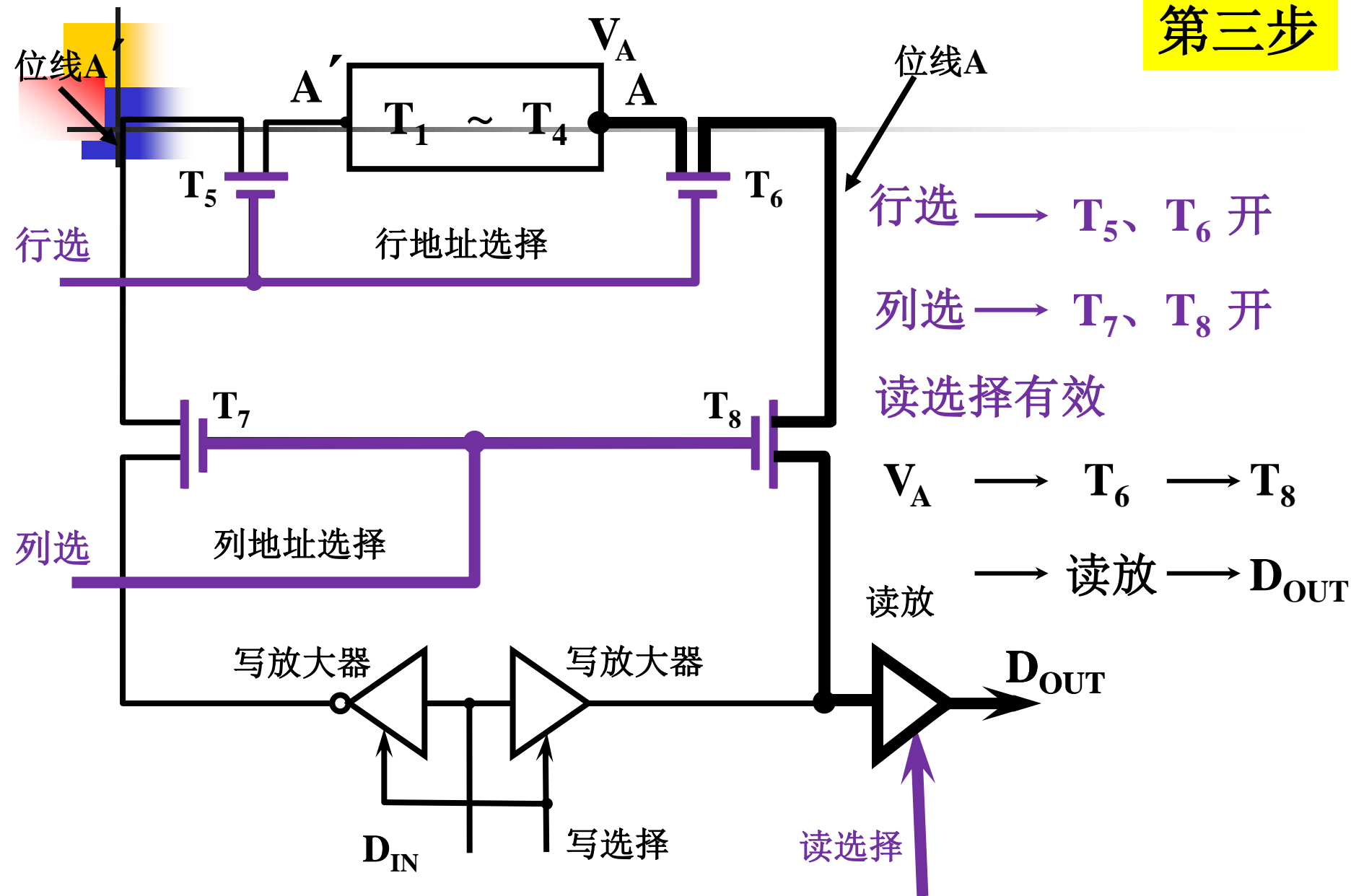
① 静态 RAM 基本电路的 读 操作

第二步



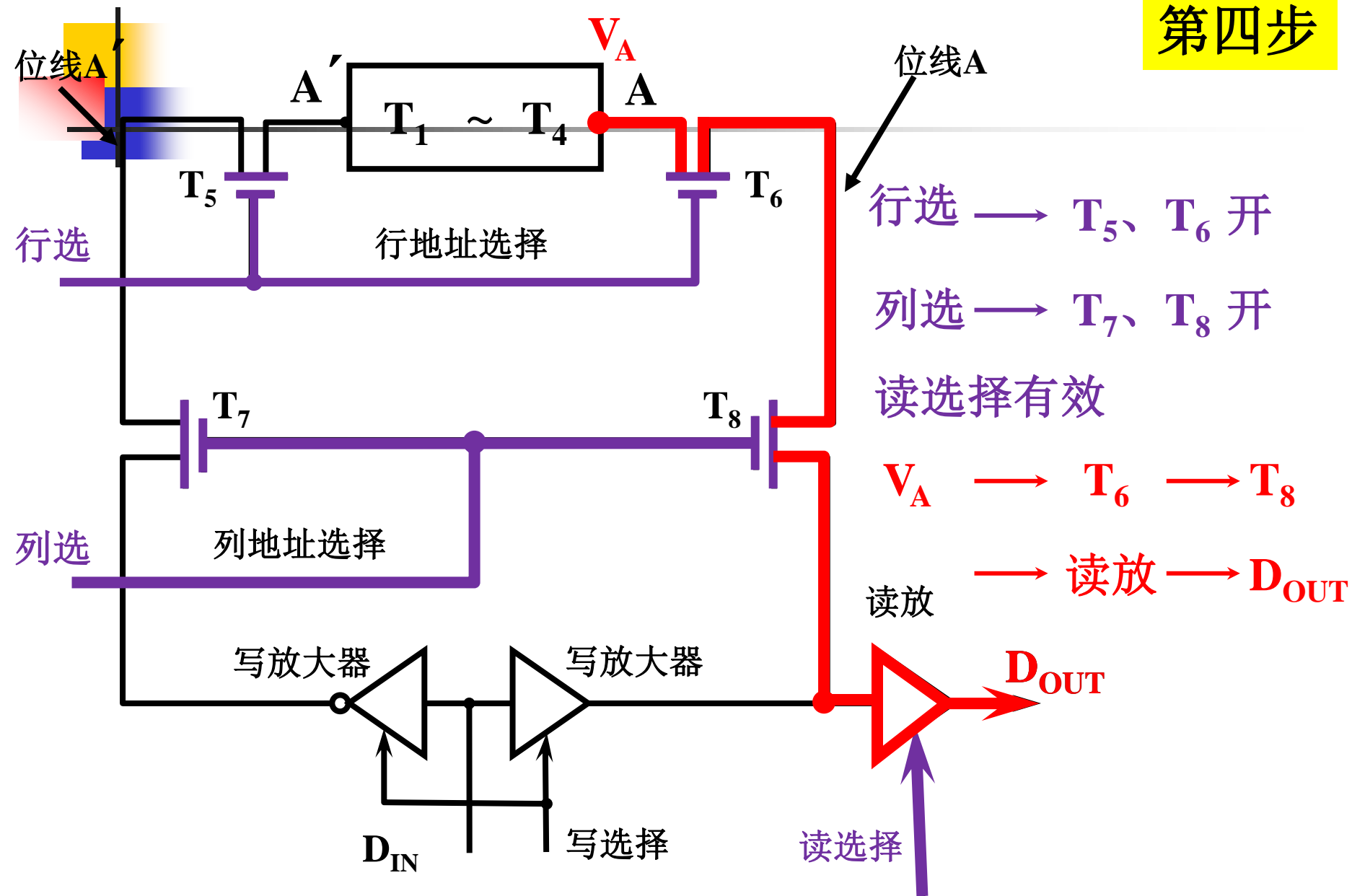
① 静态 RAM 基本电路的 读 操作

第三步



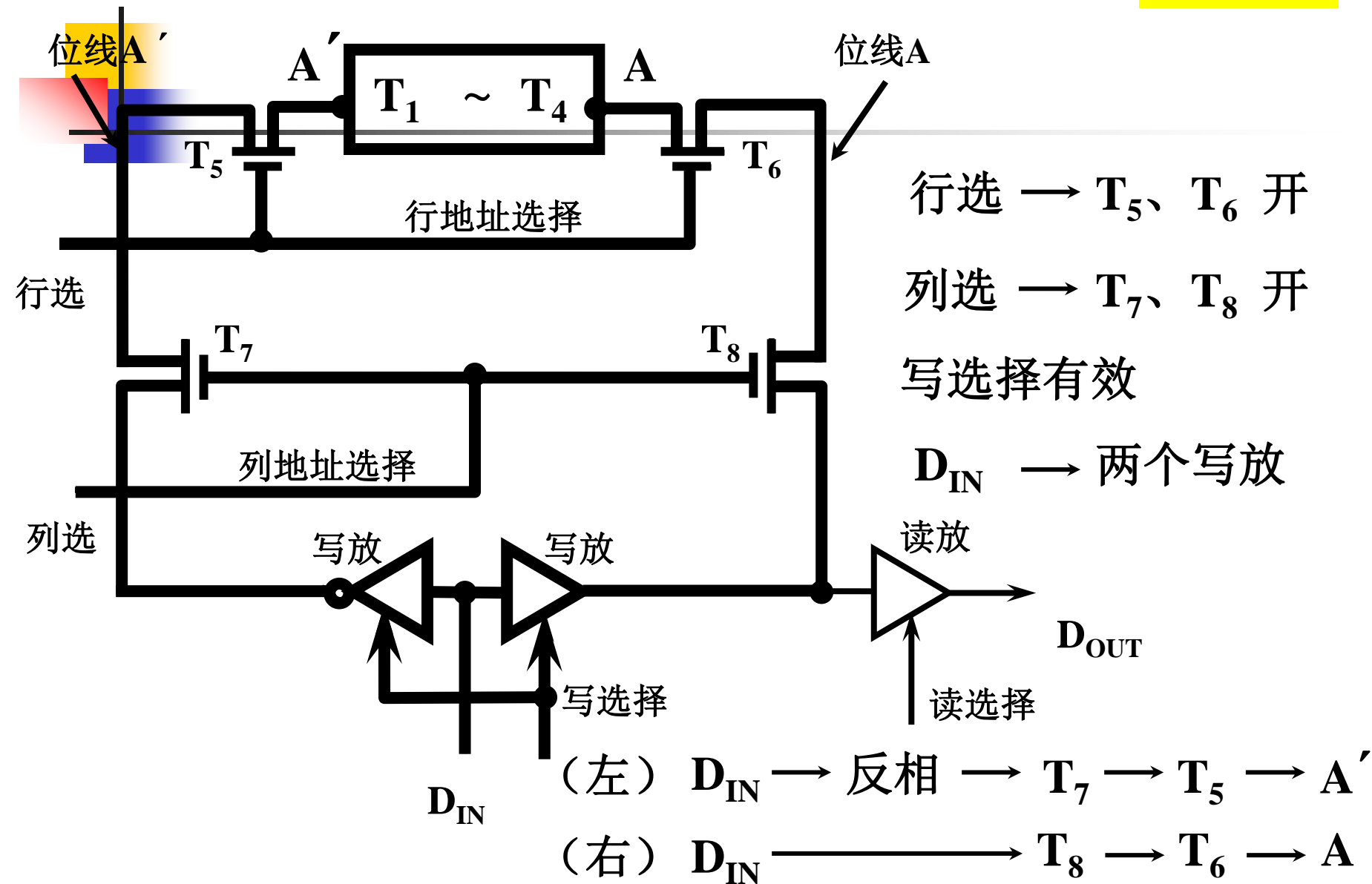
① 静态 RAM 基本电路的 读 操作

第四步



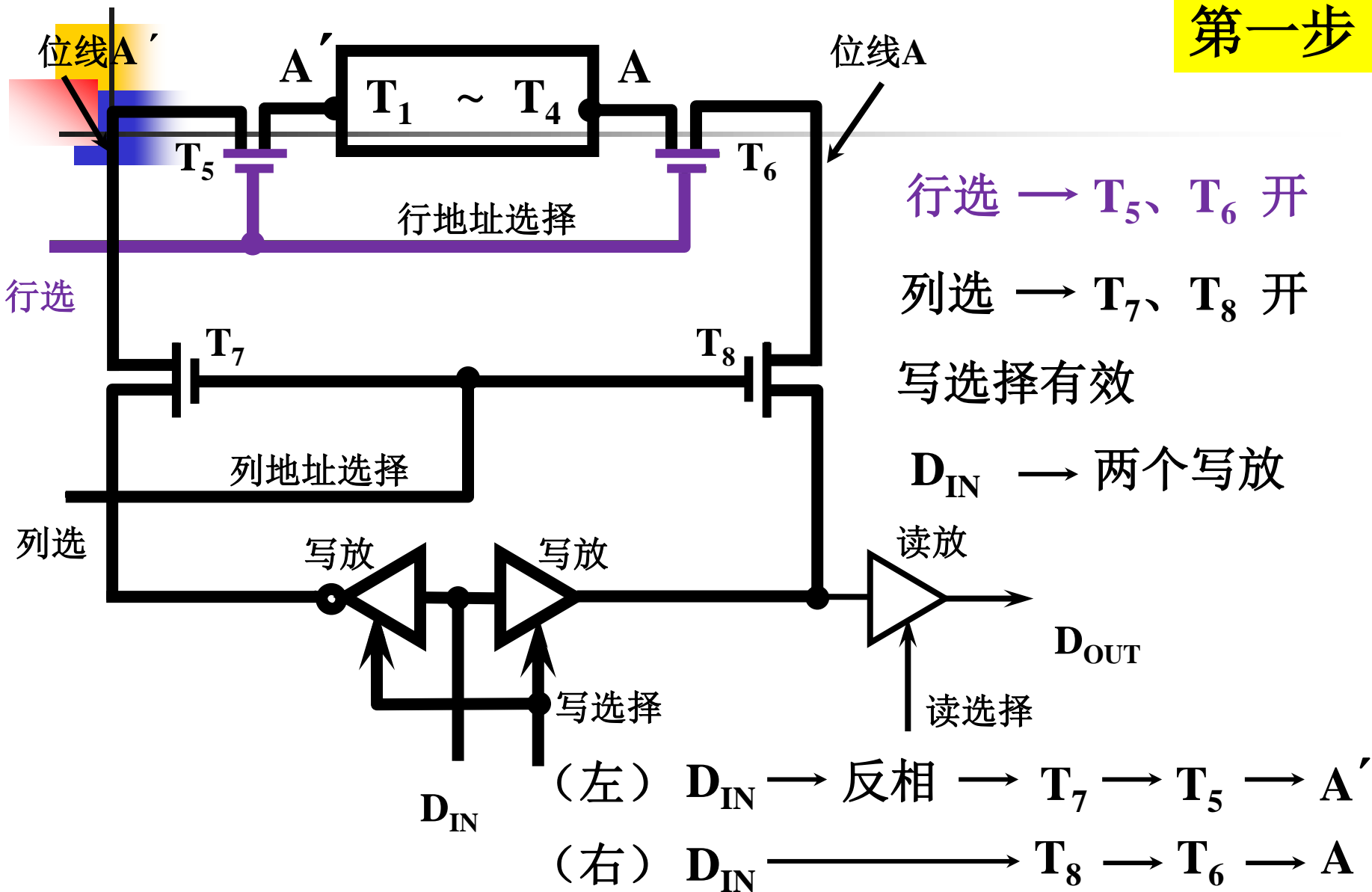
② 静态 RAM 基本电路的 写 操作

共四步



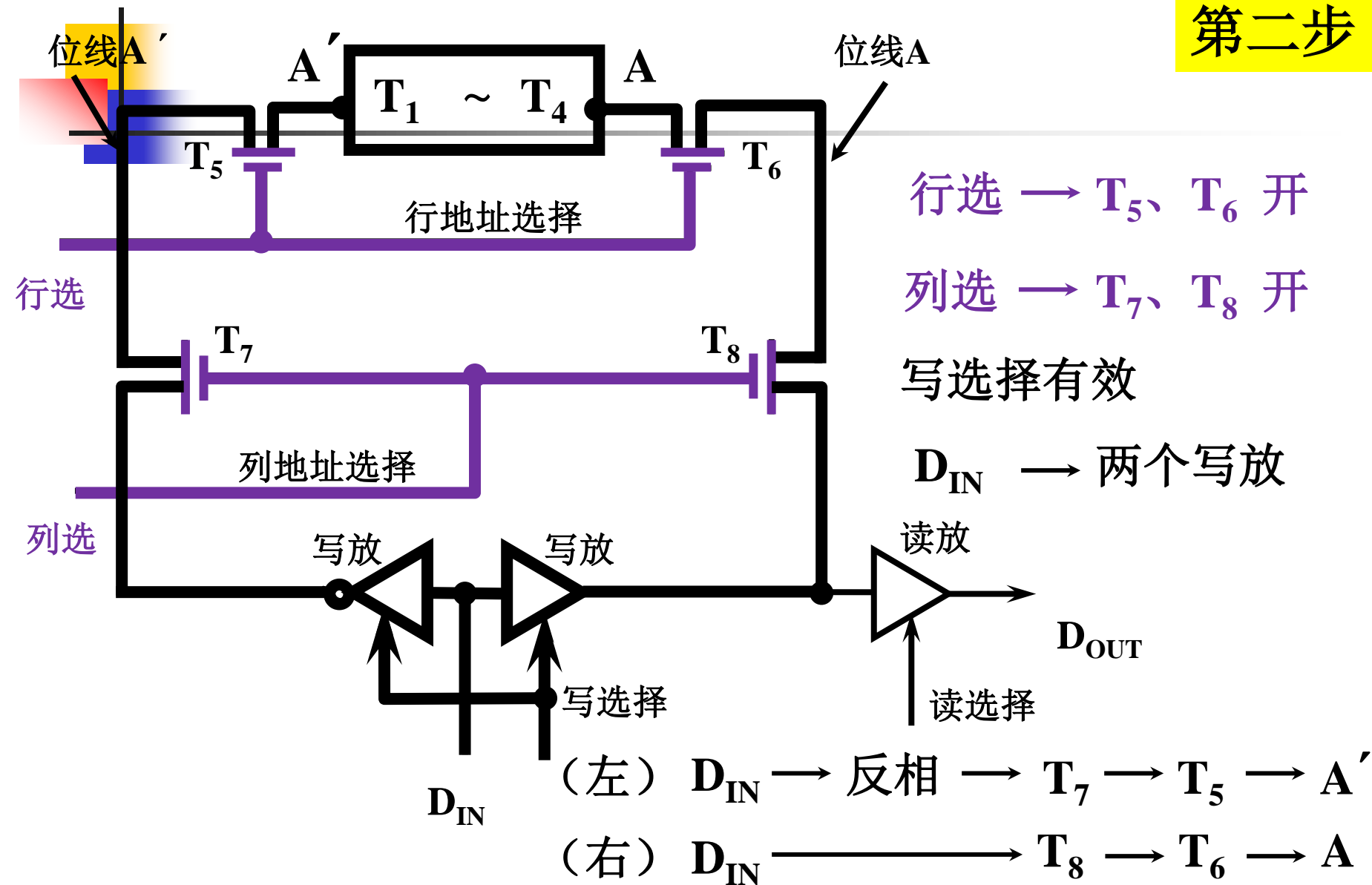
② 静态 RAM 基本电路的 写 操作

第一步



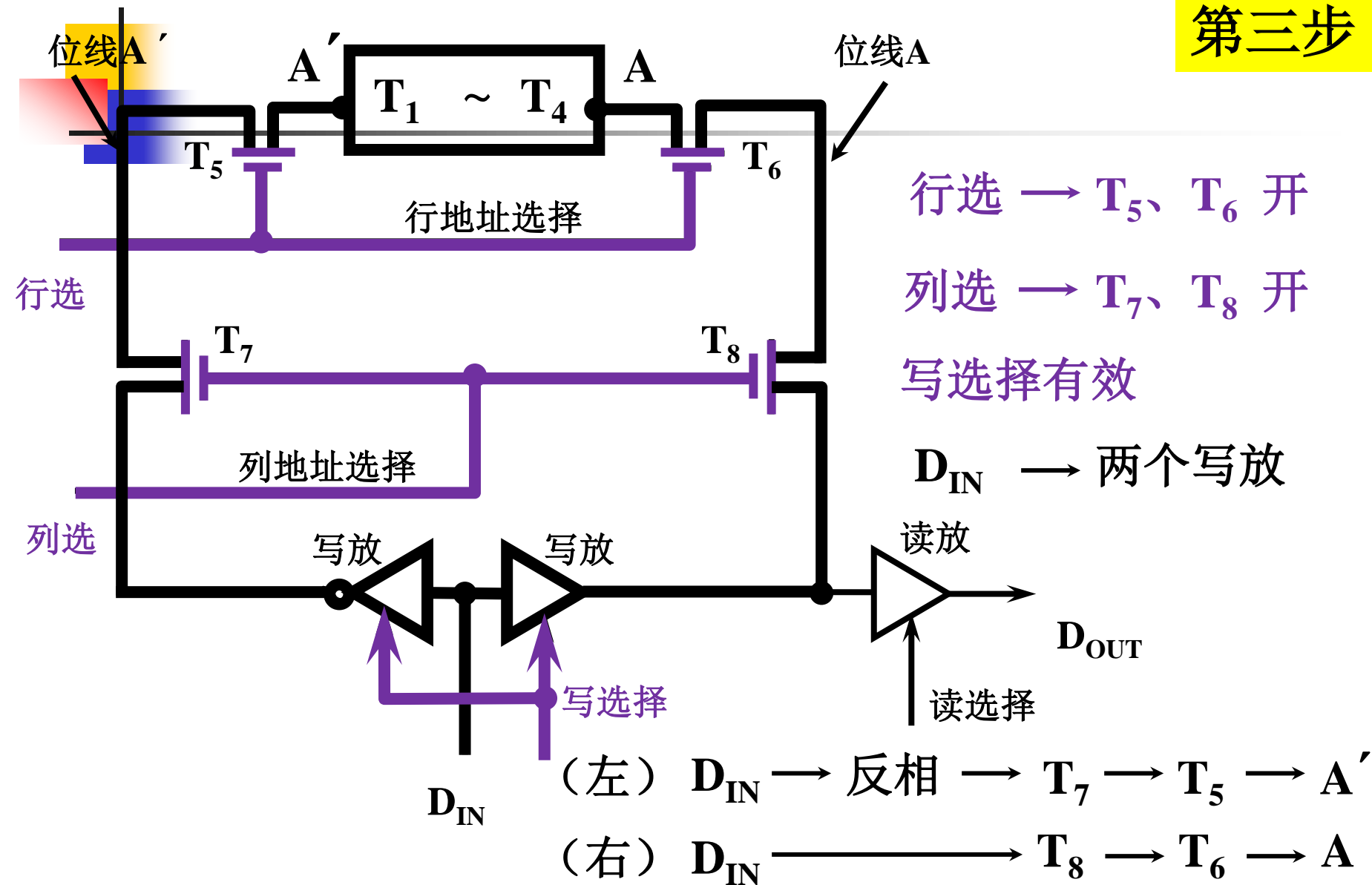
② 静态 RAM 基本电路的 写 操作

第二步



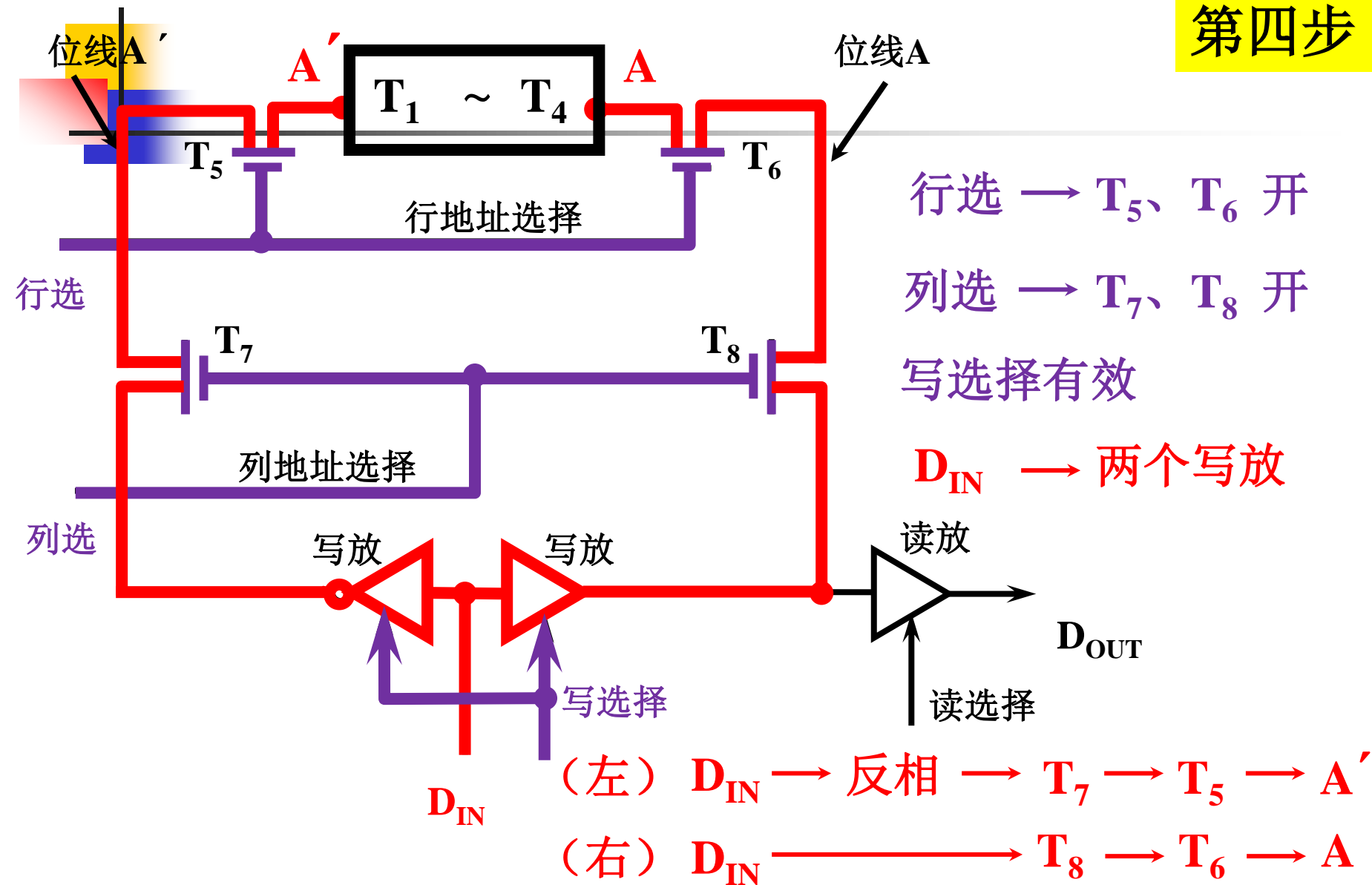
② 静态 RAM 基本电路的 写 操作

第三步



② 静态 RAM 基本电路的 写 操作

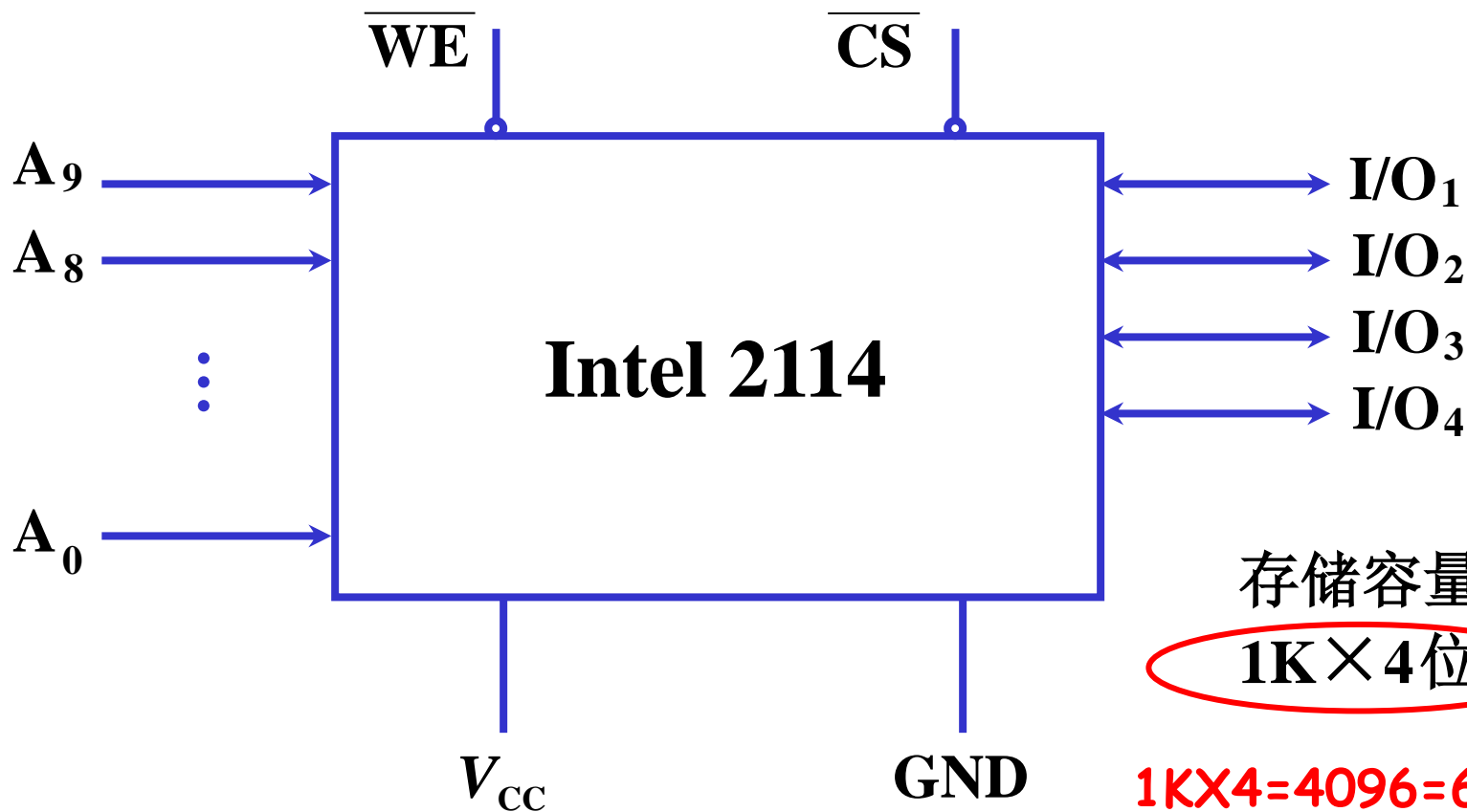
第四步



(2) 静态 RAM 芯片举例

① Intel 2114 外特性

1K × 4 位 SRAM

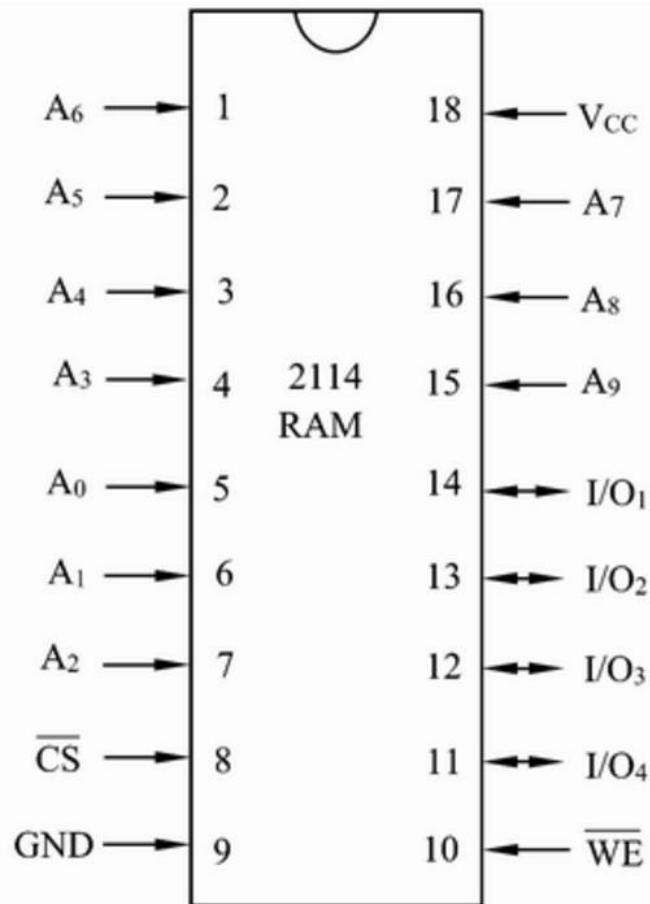


存储容量

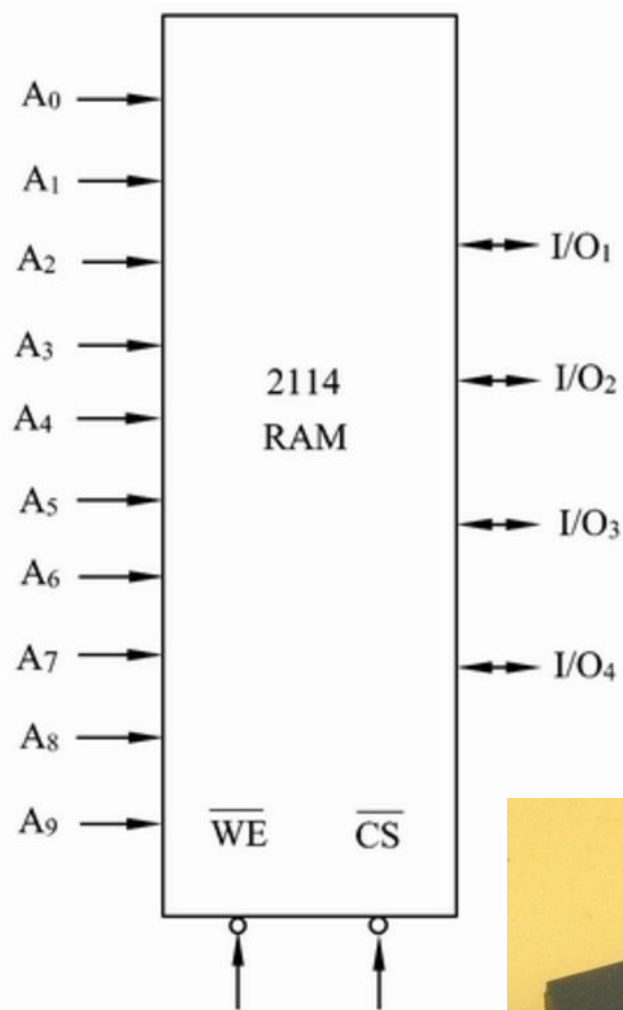
1K × 4 位

$1K \times 4 = 4096 = 64 \times 64$

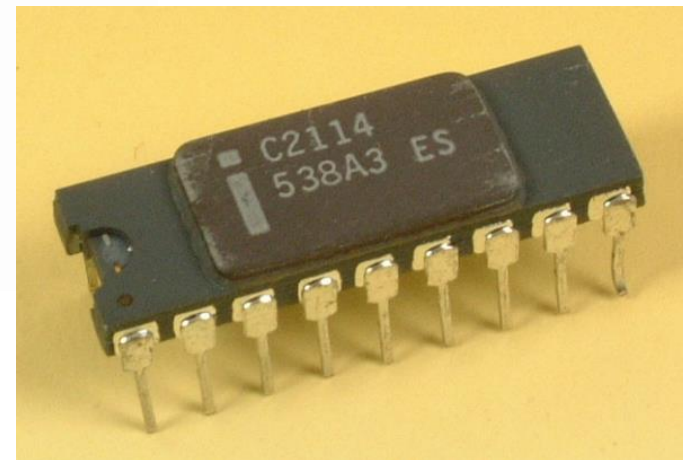
Intel 2114



(a)



(b)



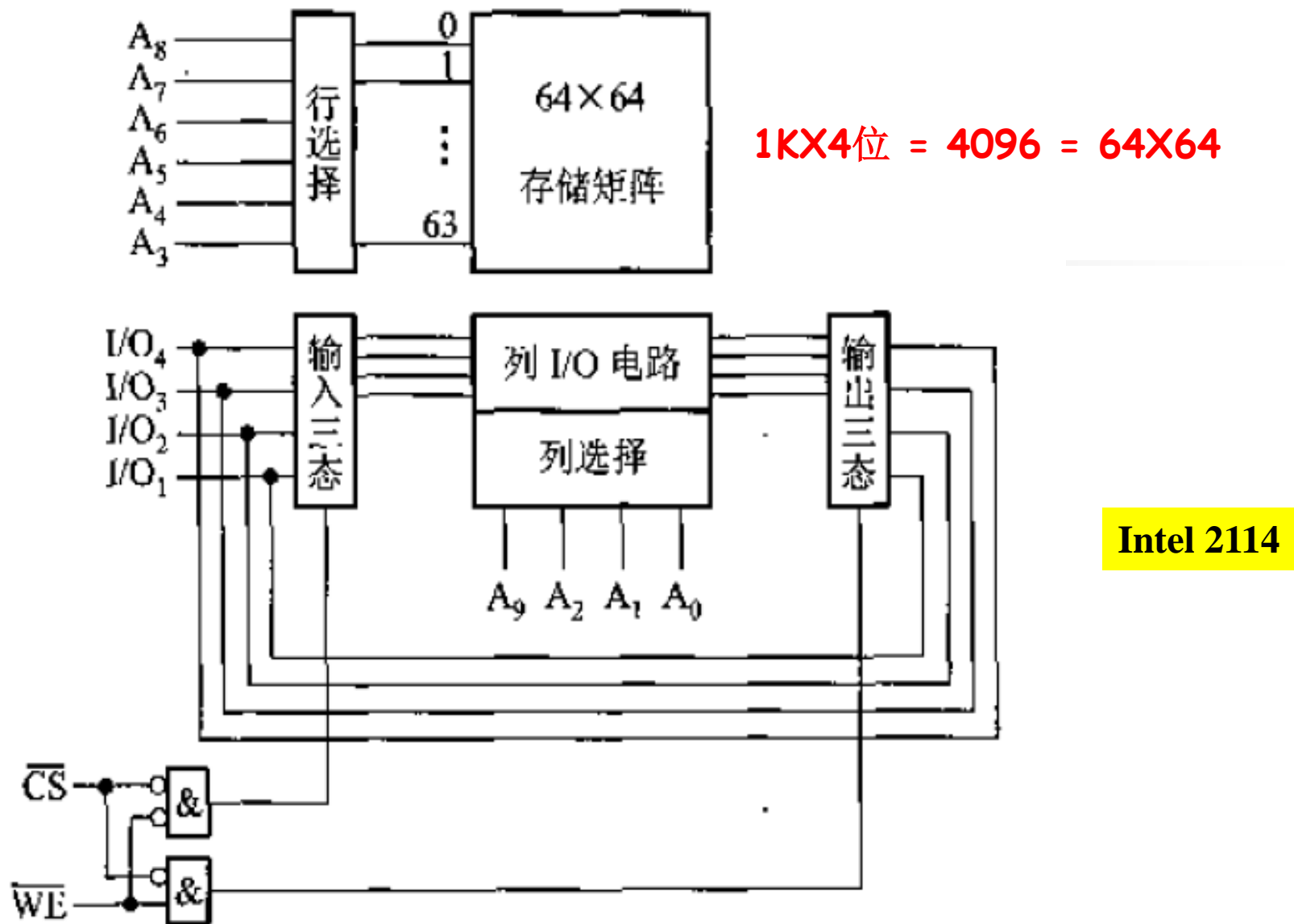
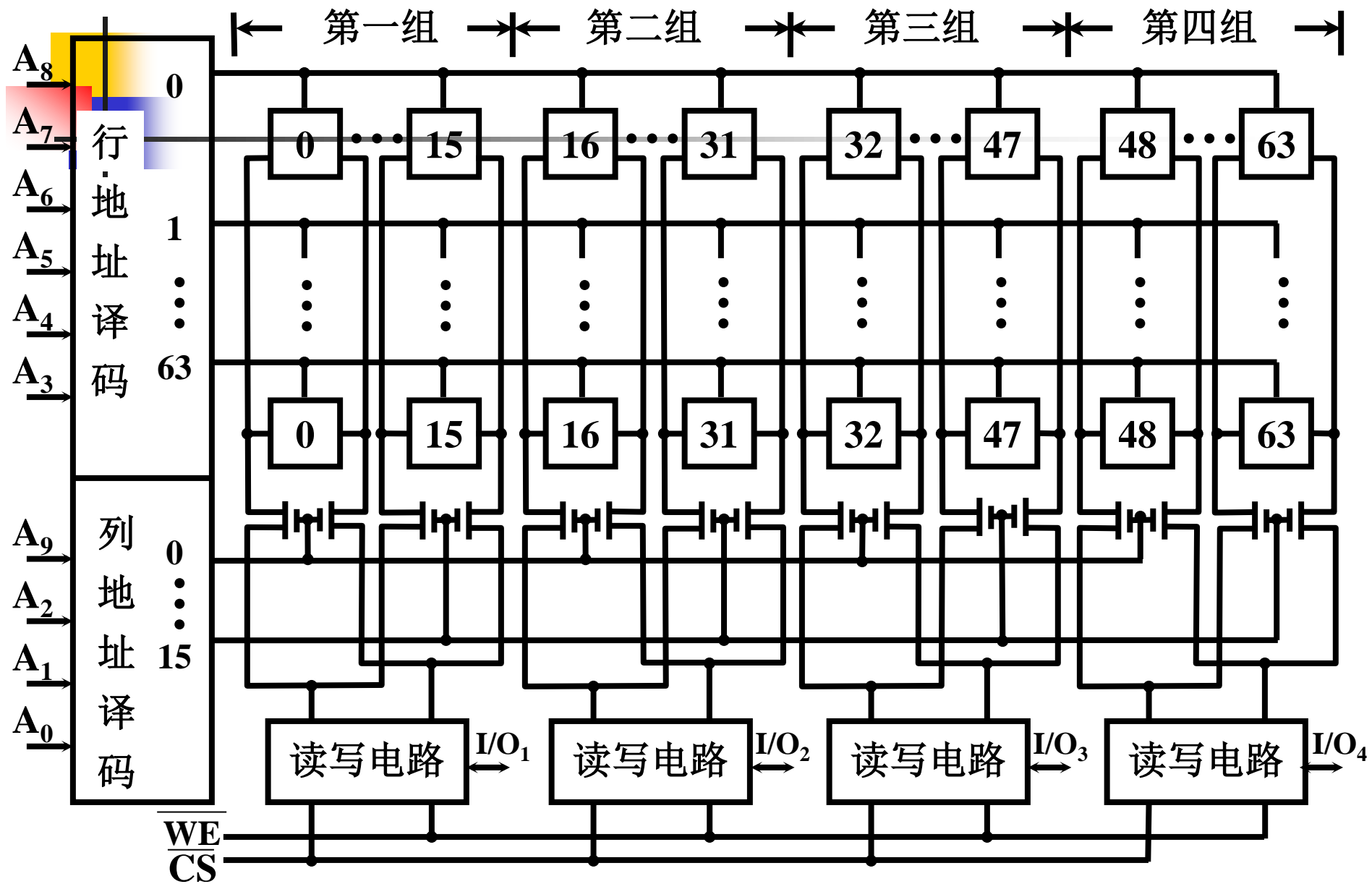


图 4.13 2114 RAM 芯片结构示意图

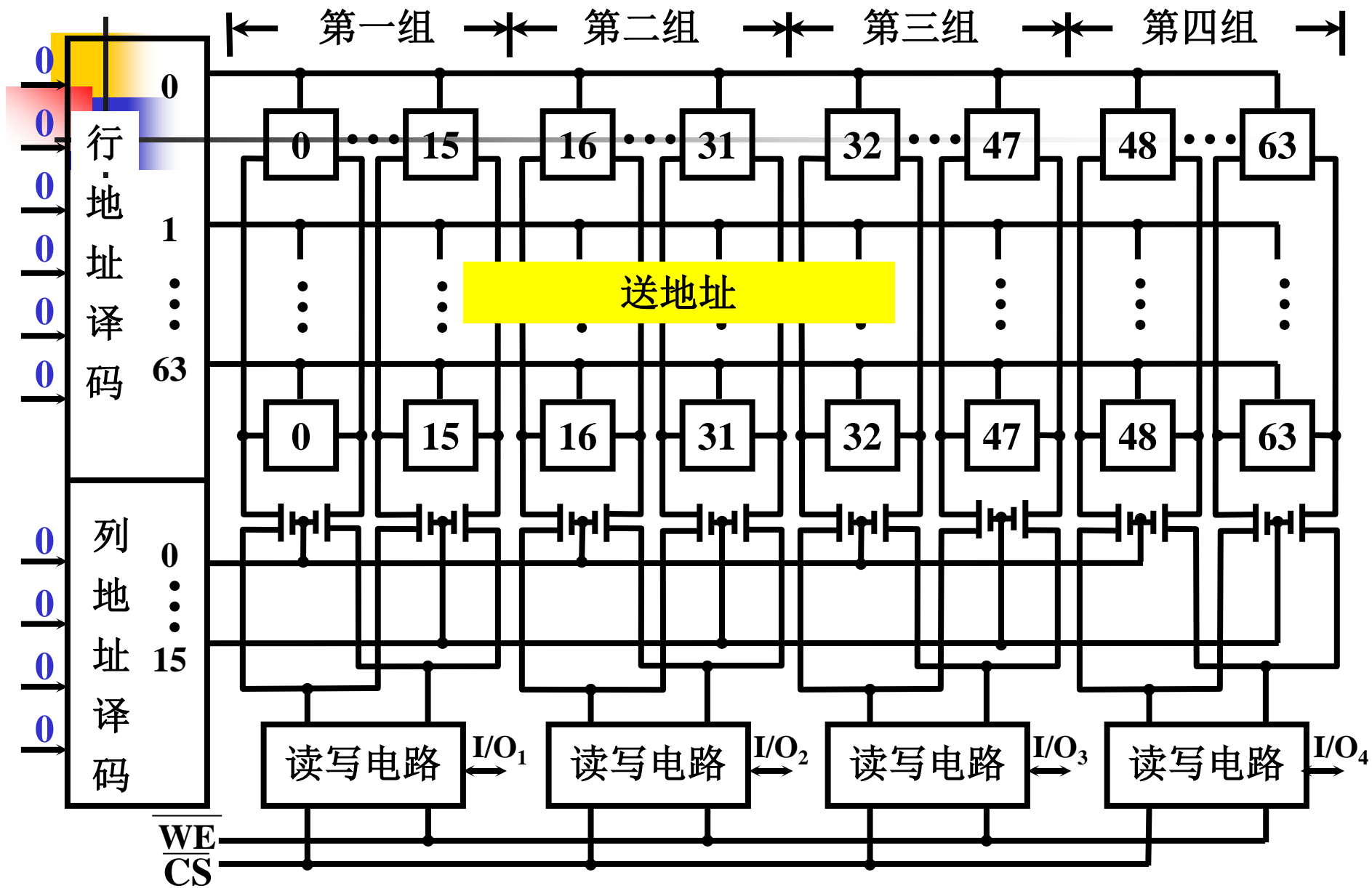
② Intel 2114 RAM 矩阵 (64×64) 读

共五步



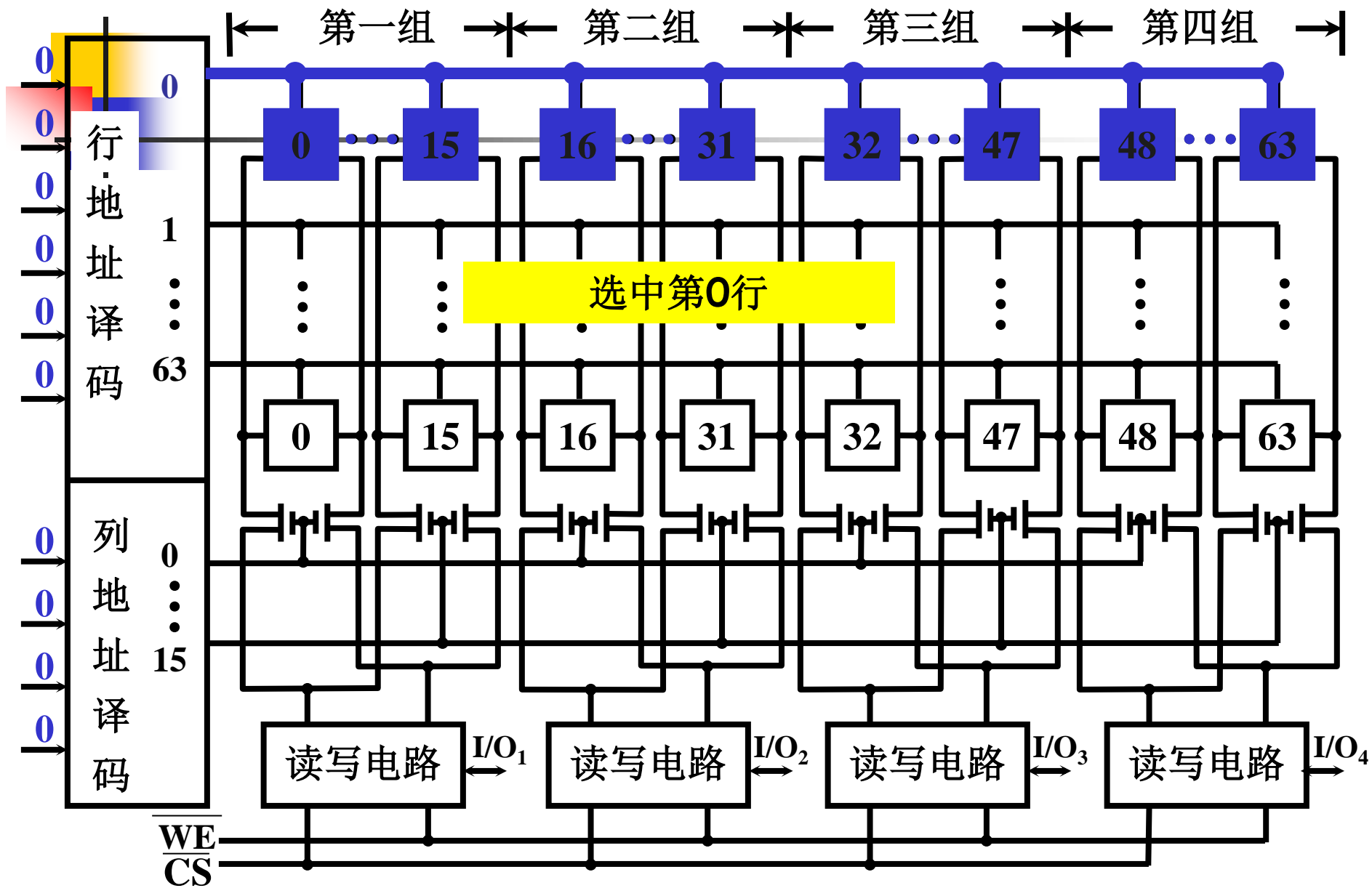
② Intel 2114 RAM 矩阵 (64×64) 读

第一步



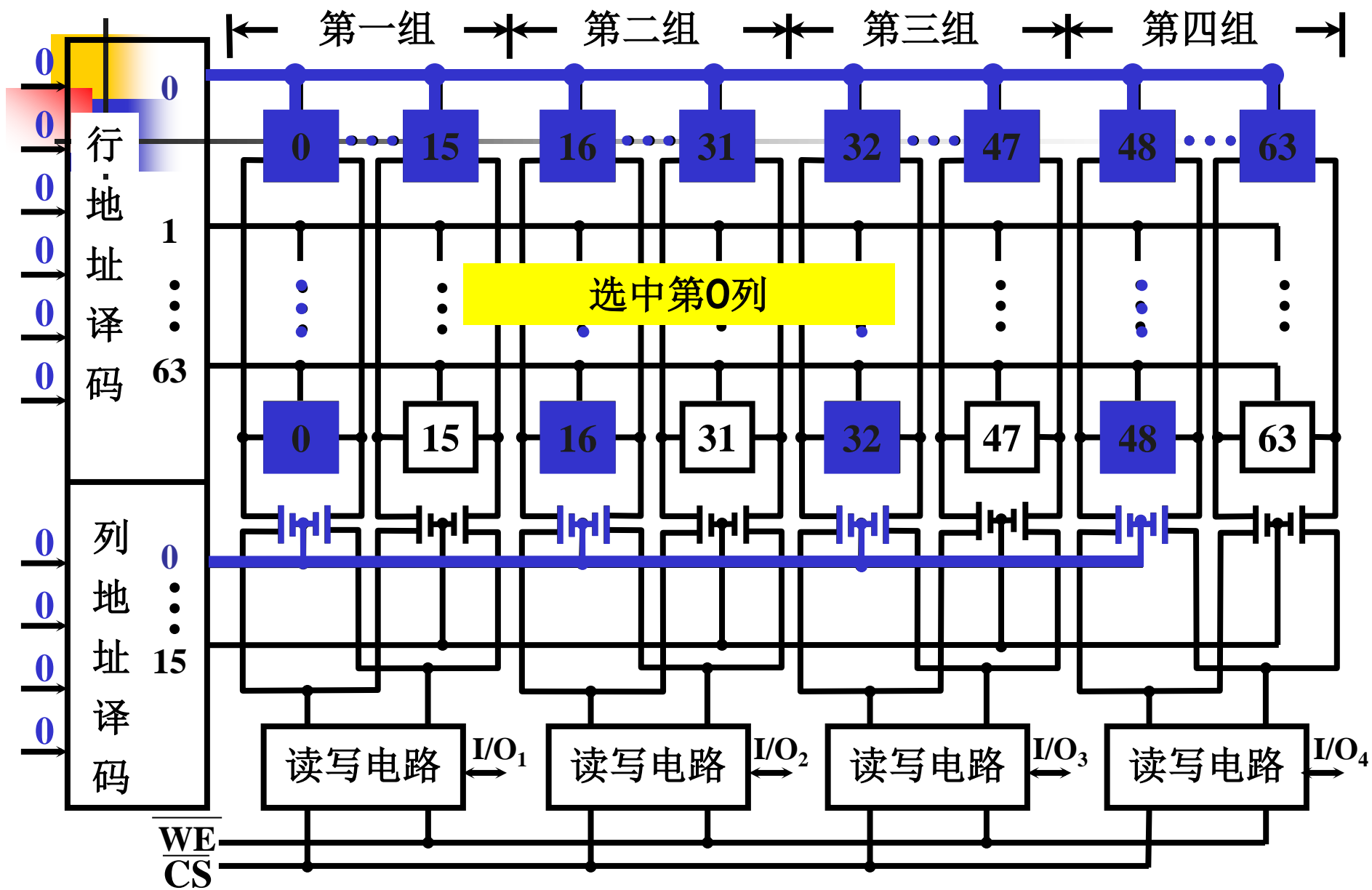
② Intel 2114 RAM 矩阵 (64×64) 读

第二步



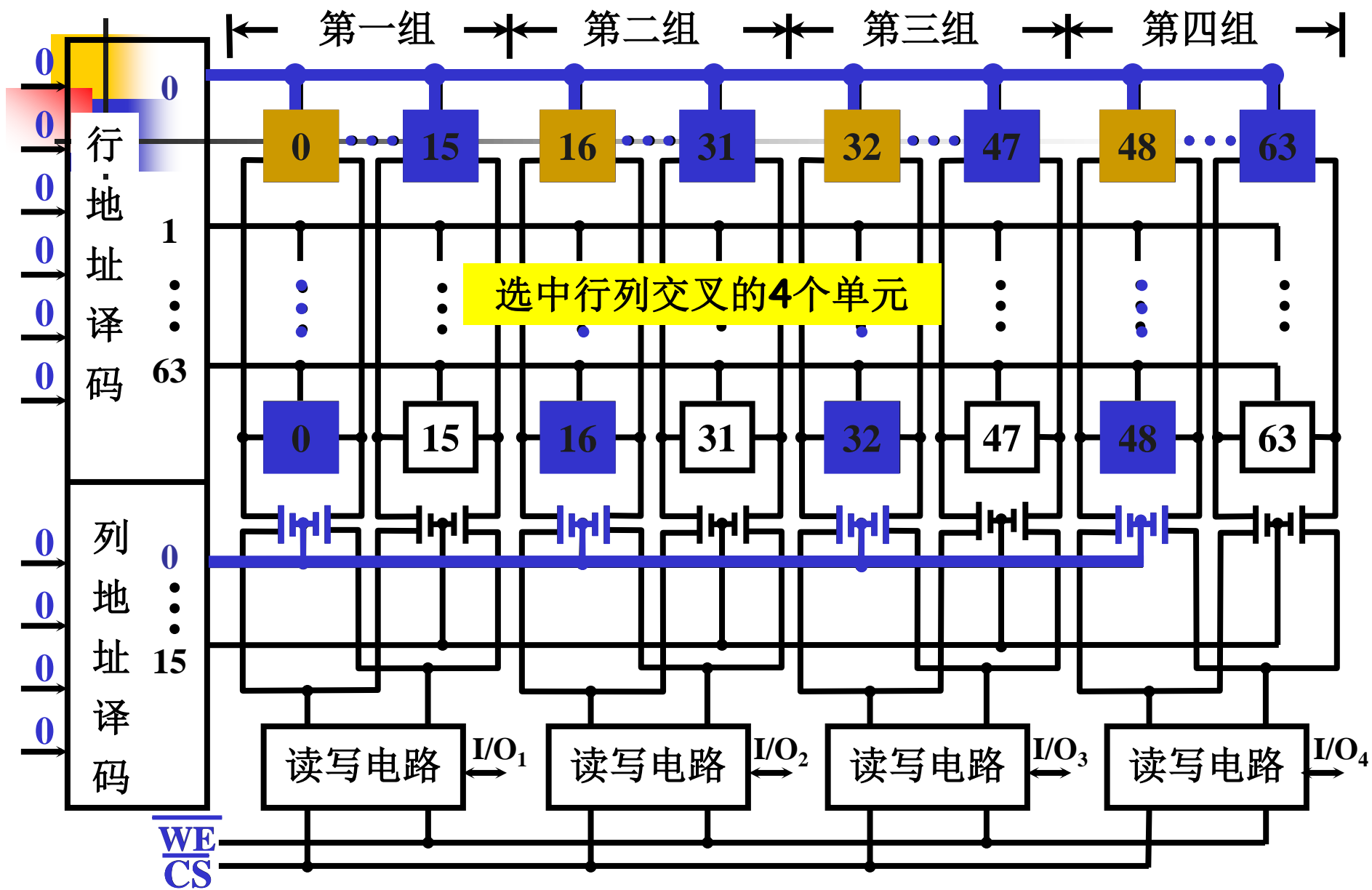
② Intel 2114 RAM 矩阵 (64 × 64) 读

第三步



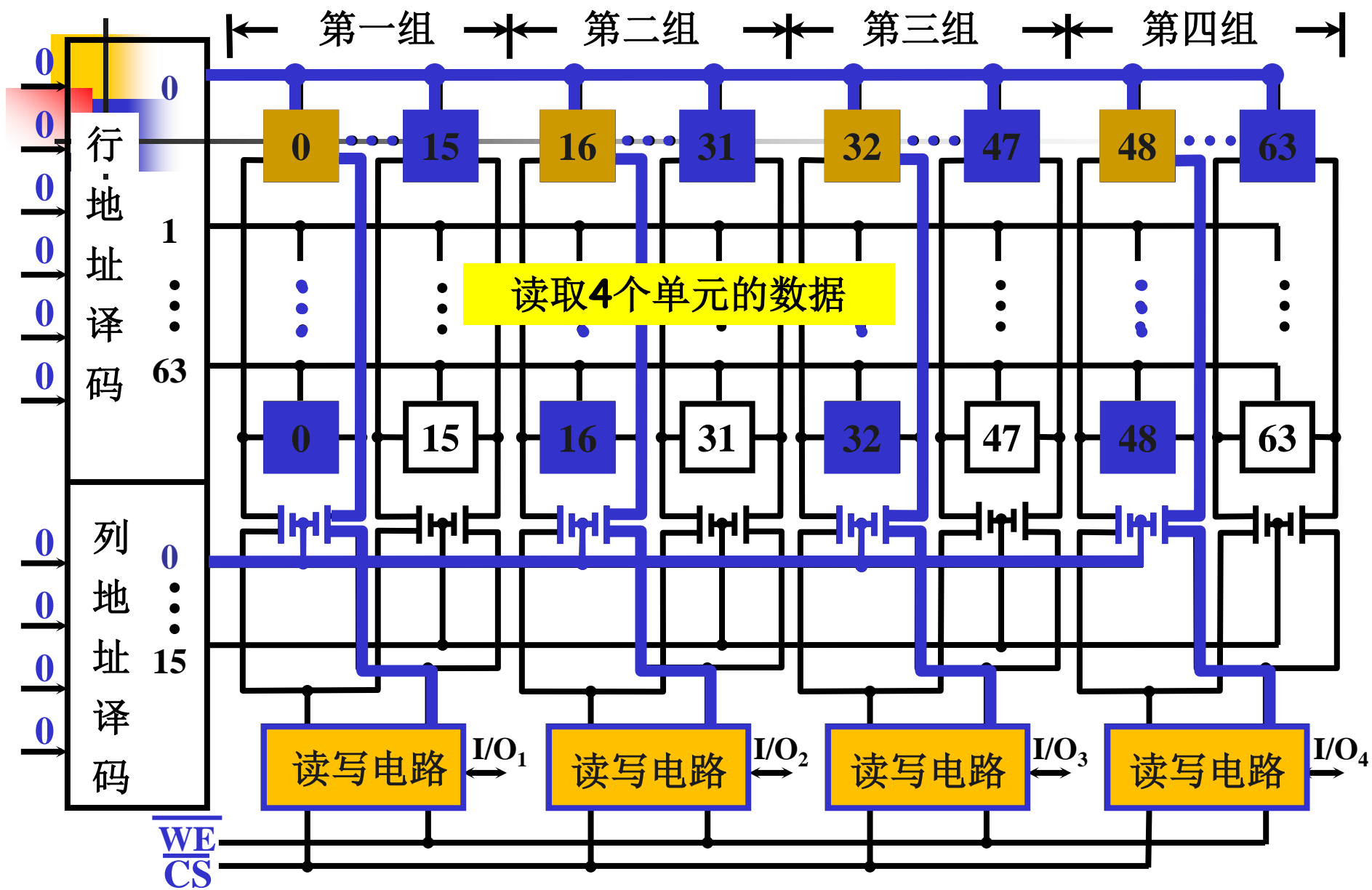
② Intel 2114 RAM 矩阵 (64×64) 读

第四步



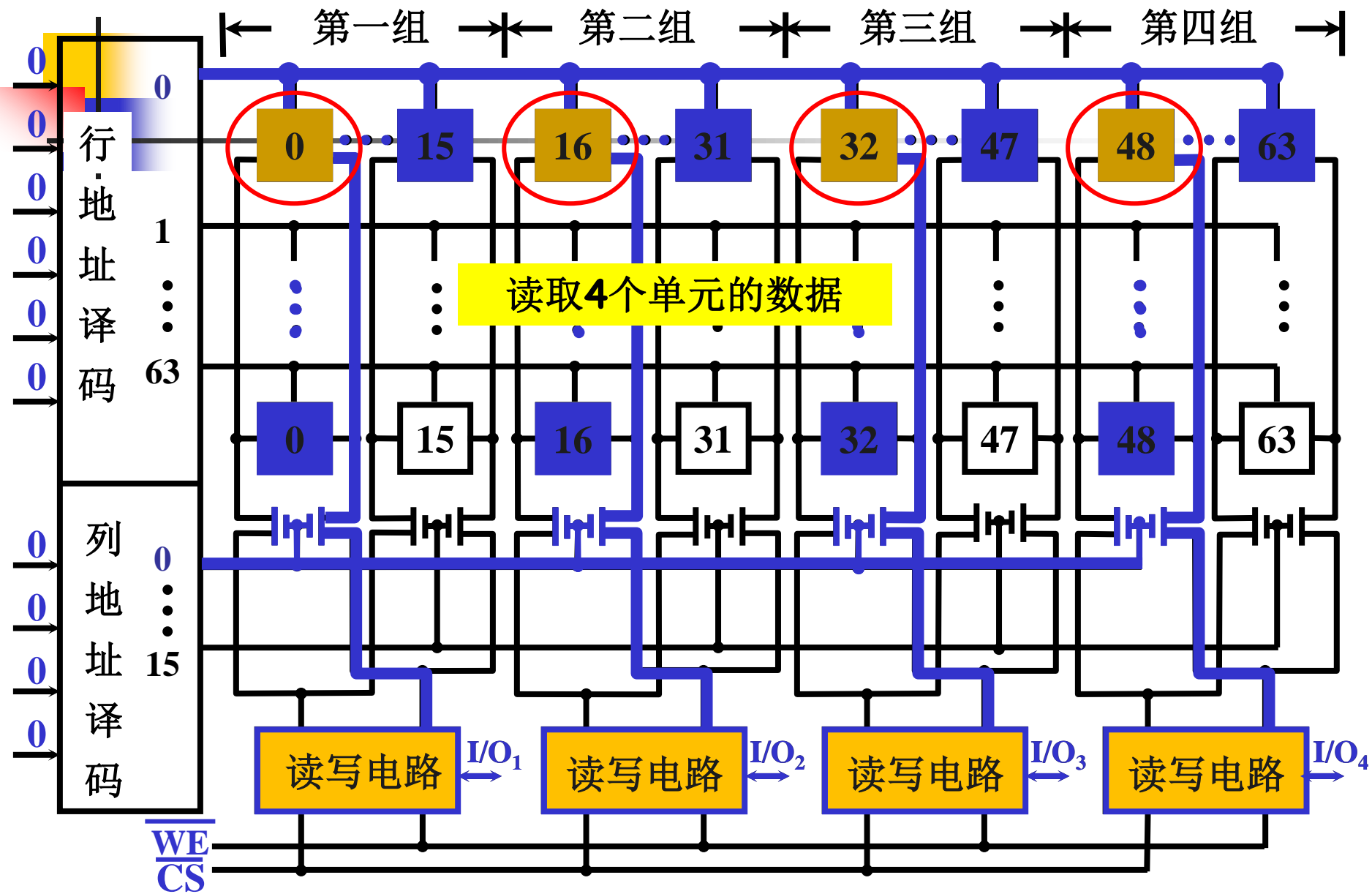
② Intel 2114 RAM 矩阵 (64 × 64) 读

第五步



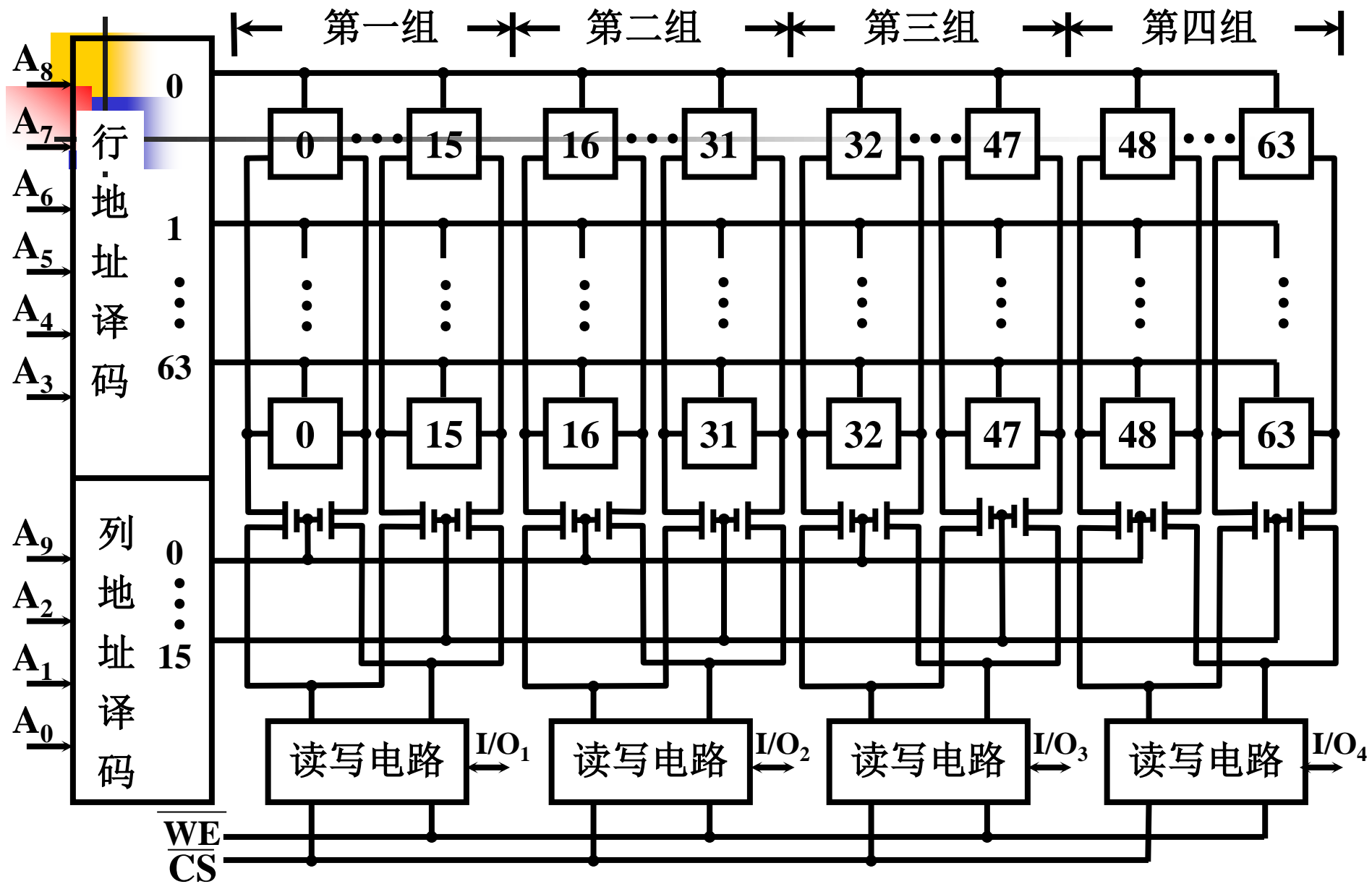
② Intel 2114 RAM 矩阵 (64 × 64) 读

第五步



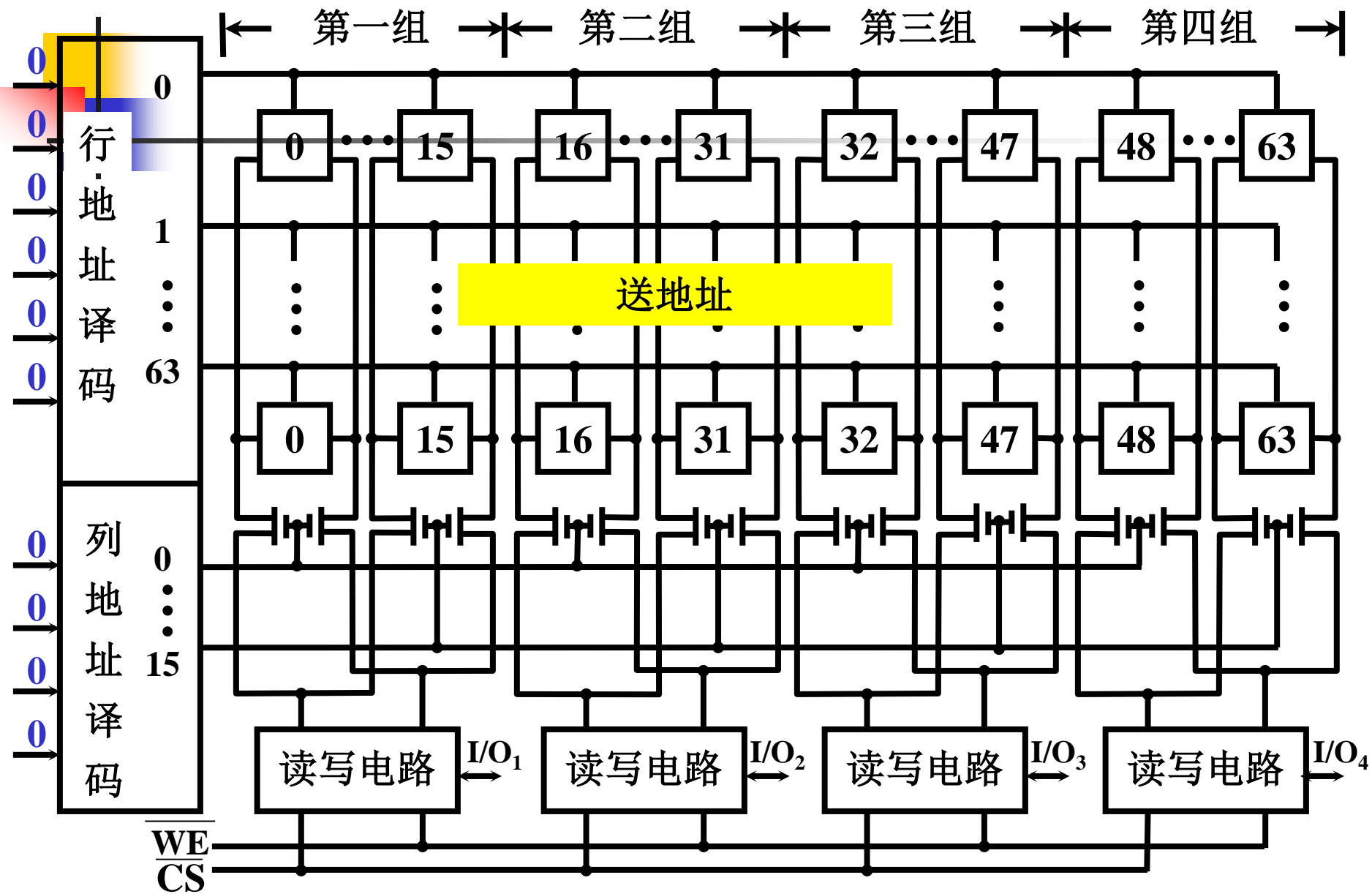
③ Intel 2114 RAM 矩阵 (64×64) 写

共五步



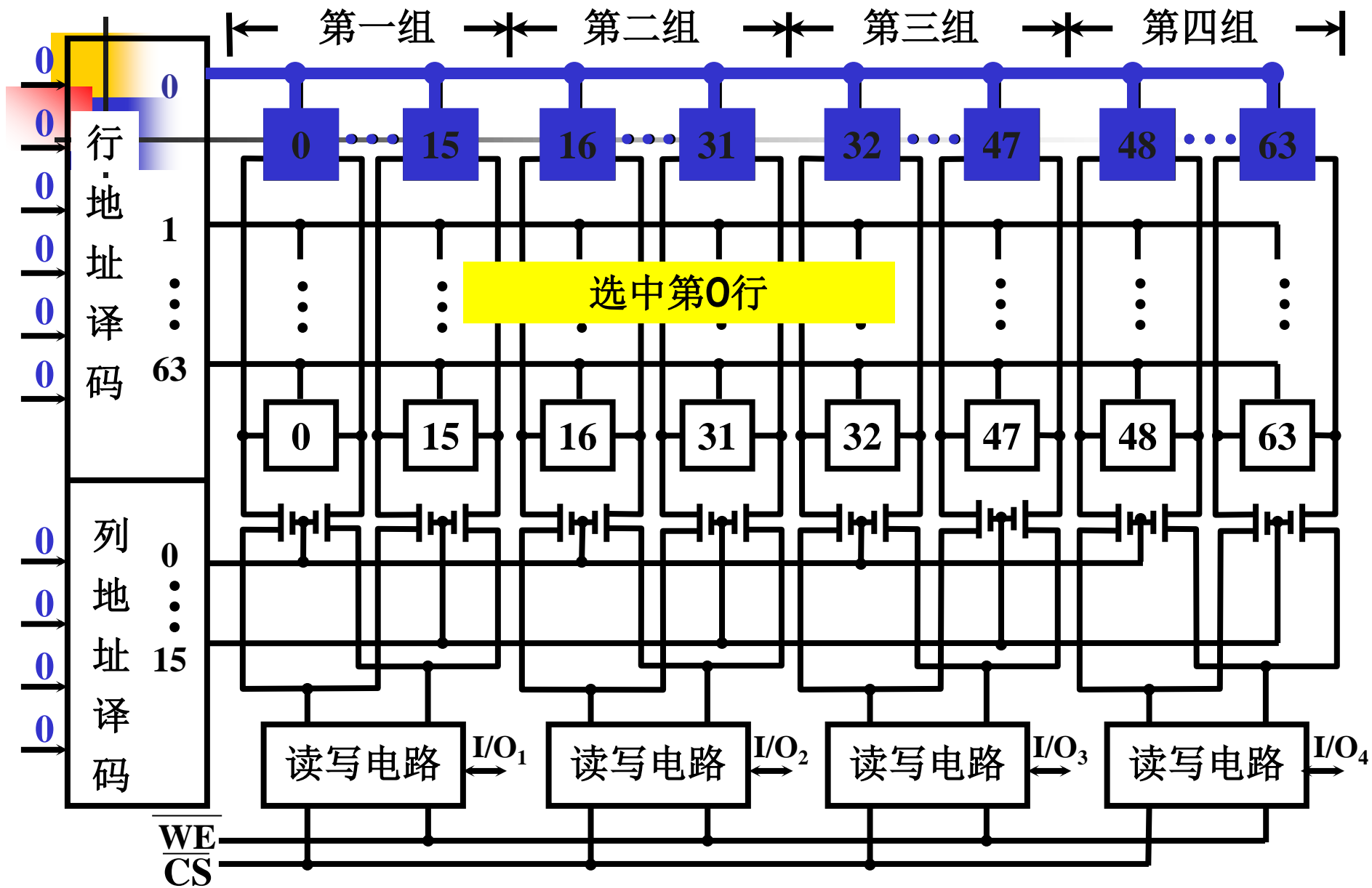
③ Intel 2114 RAM 矩阵 (64×64) 写

第一步



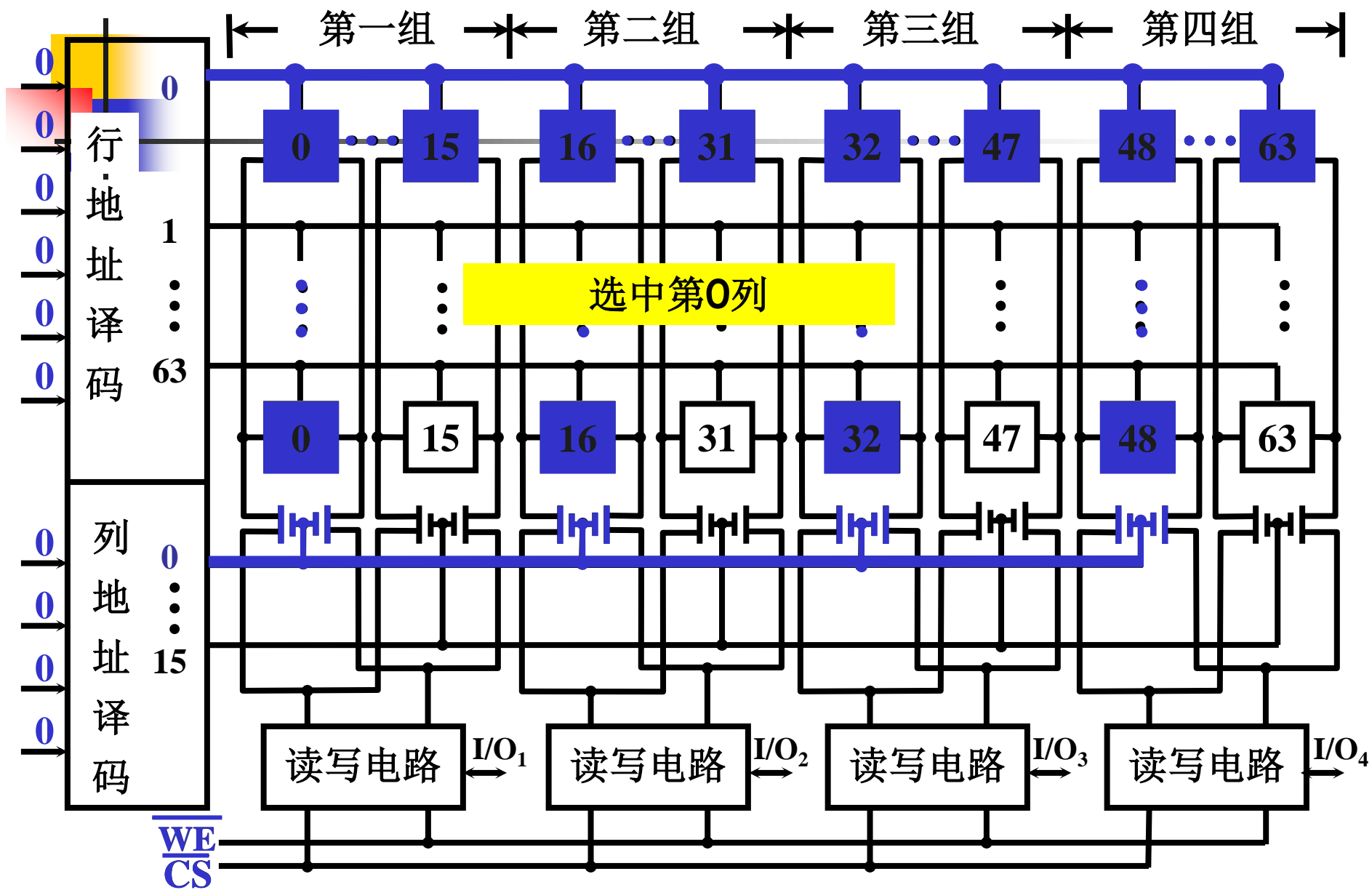
③ Intel 2114 RAM 矩阵 (64 × 64) 写

第二步

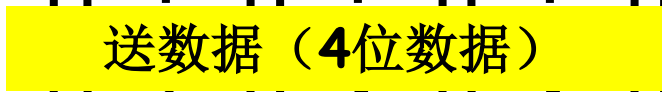


③ Intel 2114 RAM 矩阵 (64×64) 写

第三步

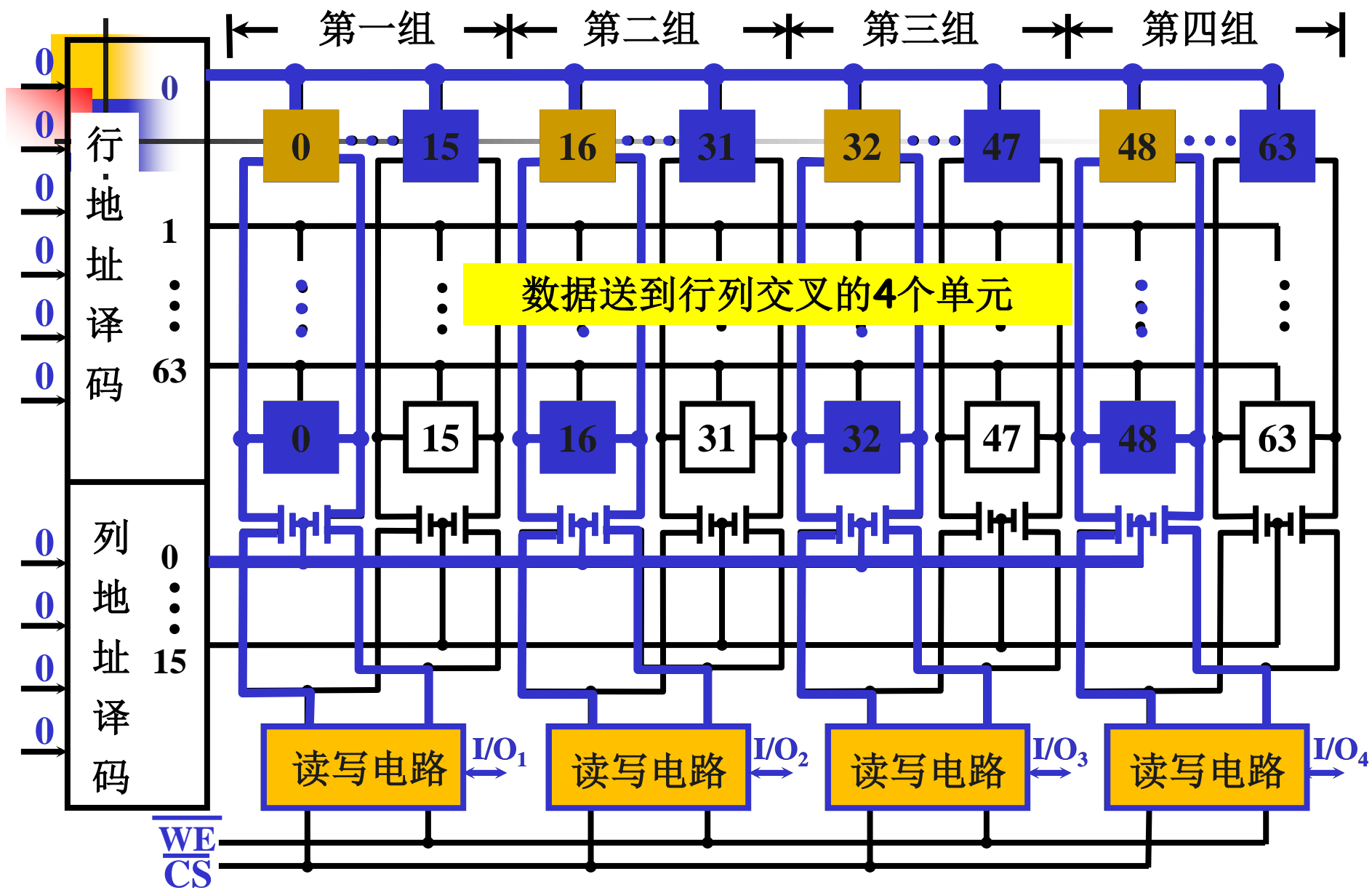


第四步



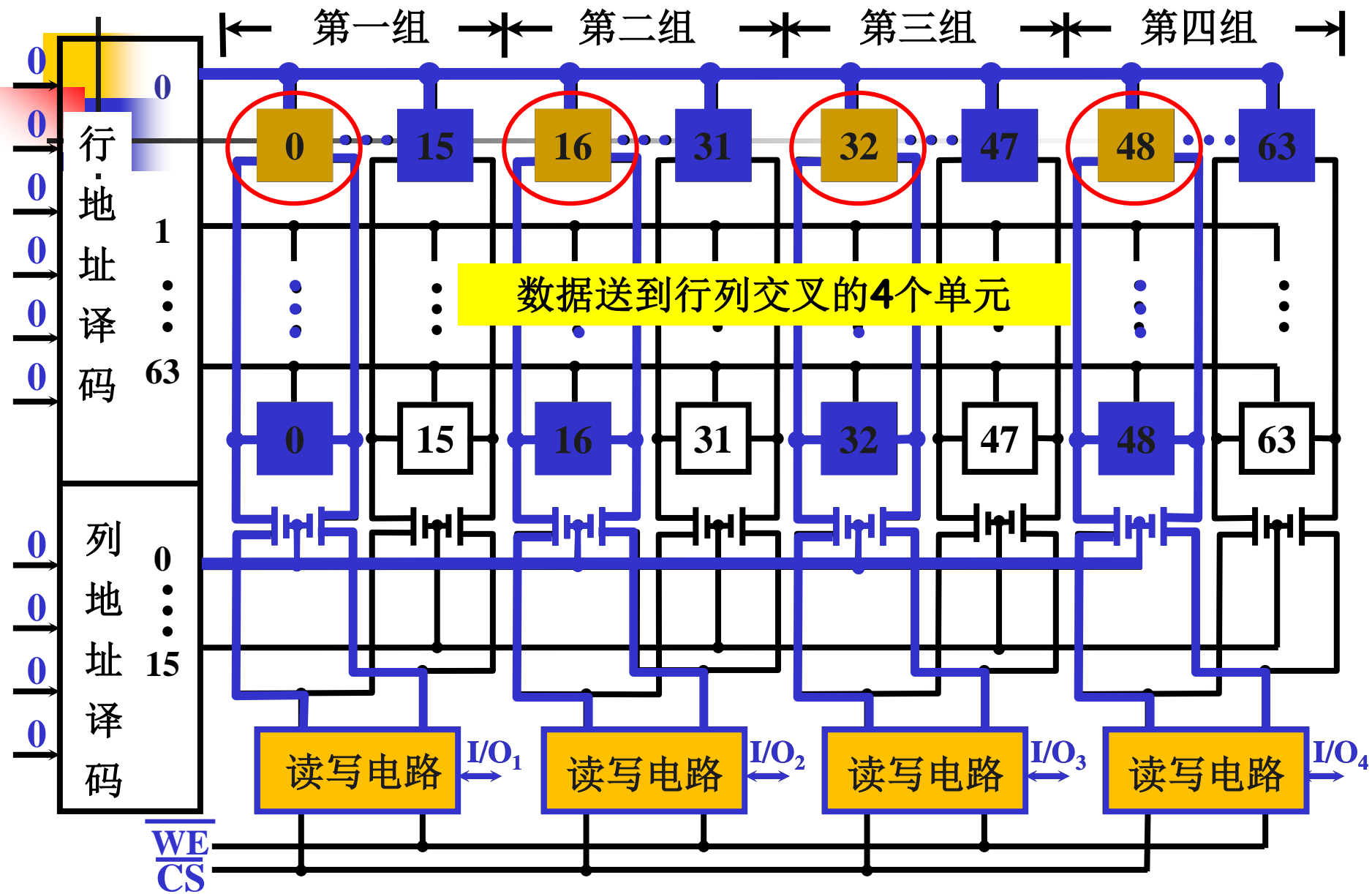
③ Intel 2114 RAM 矩阵 (64 × 64) 写

第五步



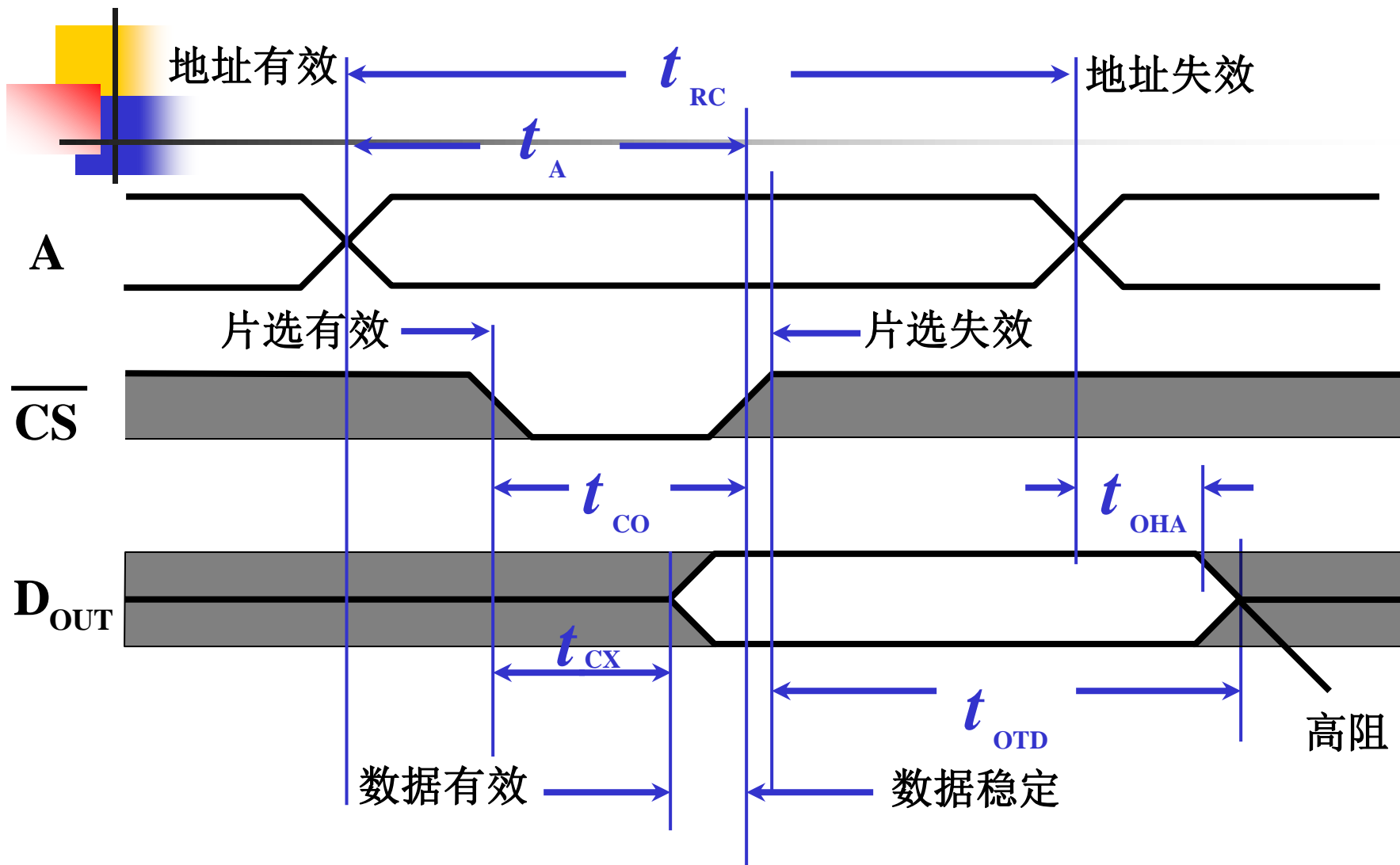
③ Intel 2114 RAM 矩阵 (64 × 64) 写

第五步



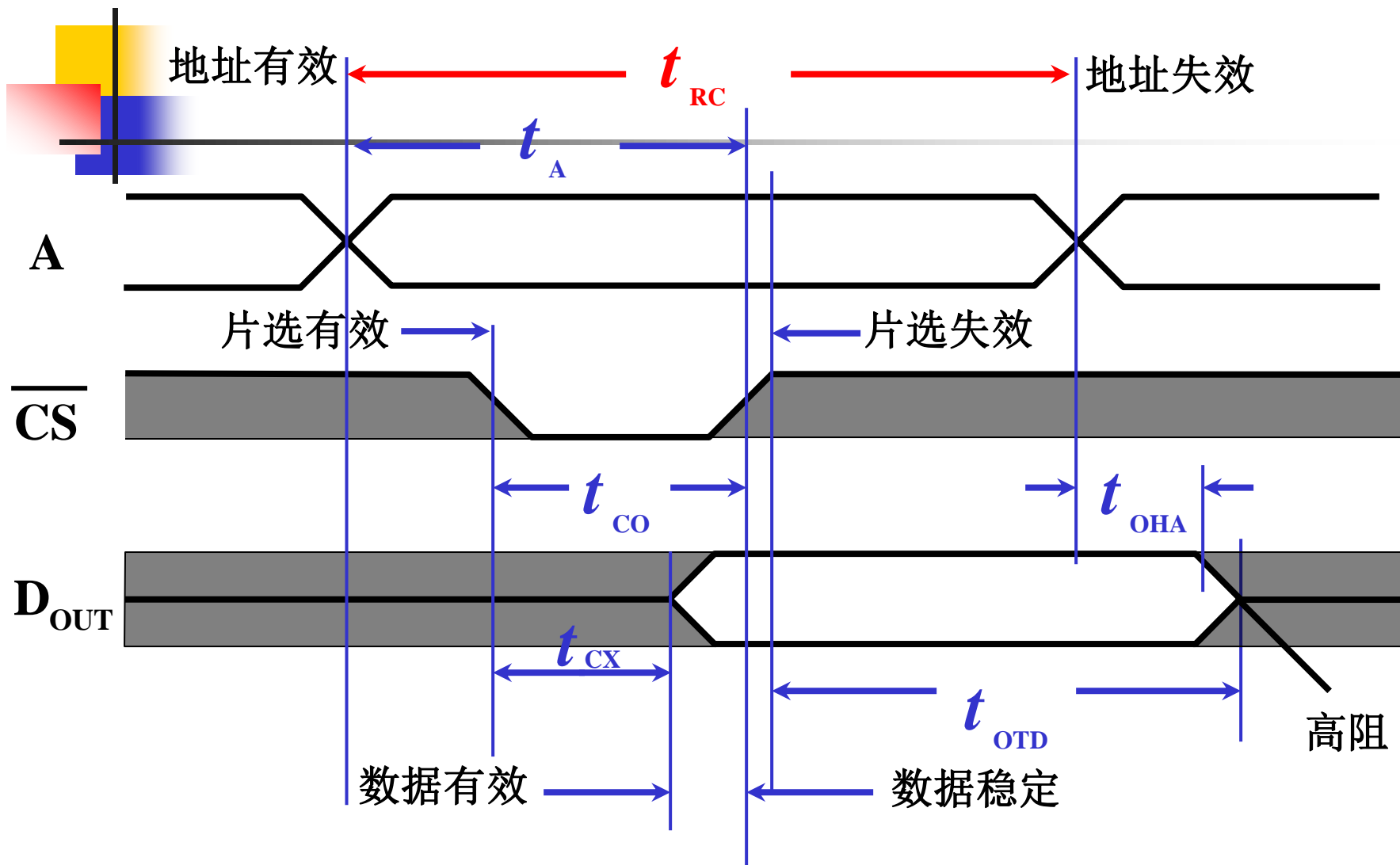
(3) 静态 RAM (2114)读 时序

时序图



(3) 静态 RAM (2114) 读 时序

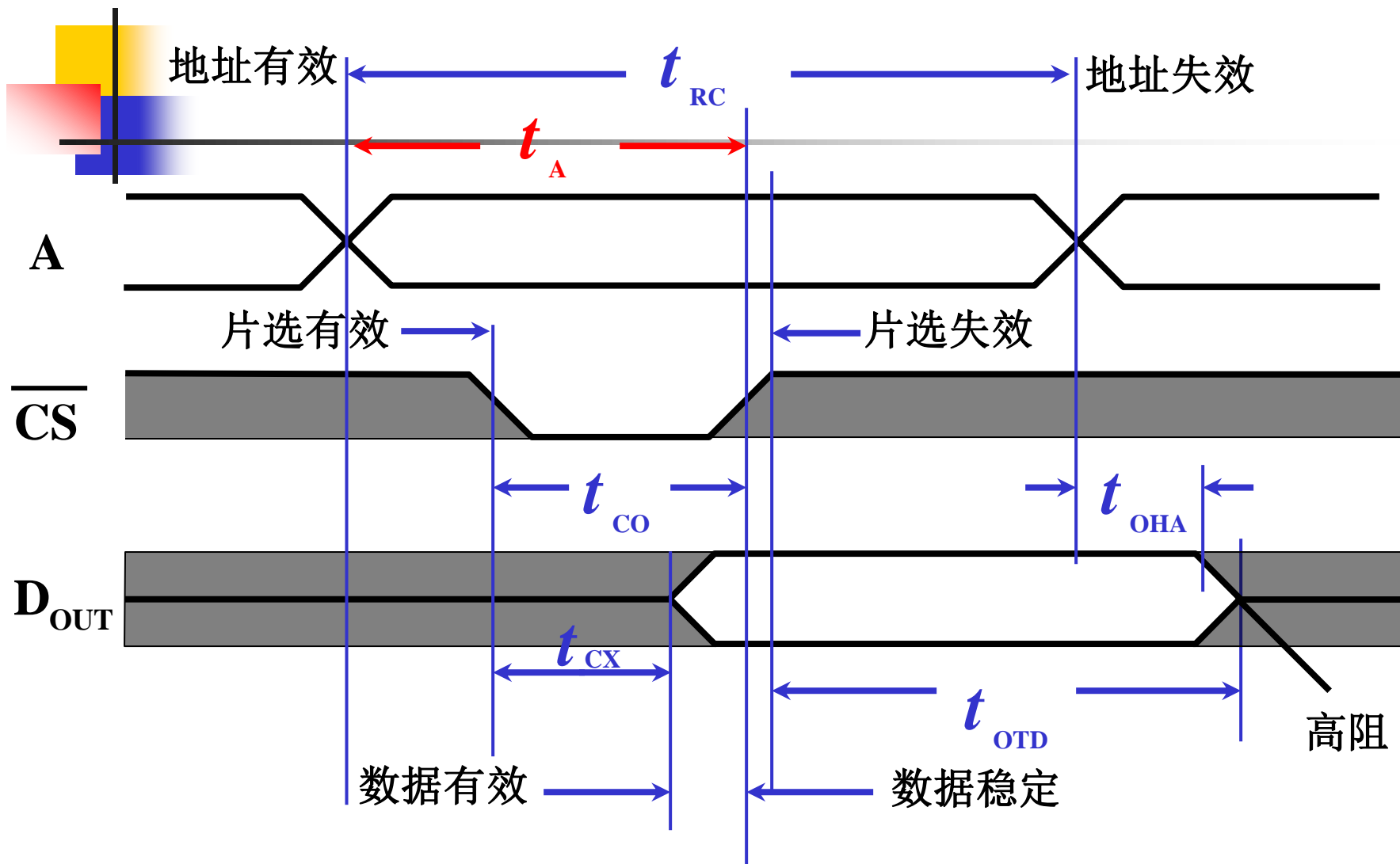
时序图



(1) t_{RC} : 读周期 (从地址有效到地址失效的时间)

(3) 静态 RAM (2114) 读 时序

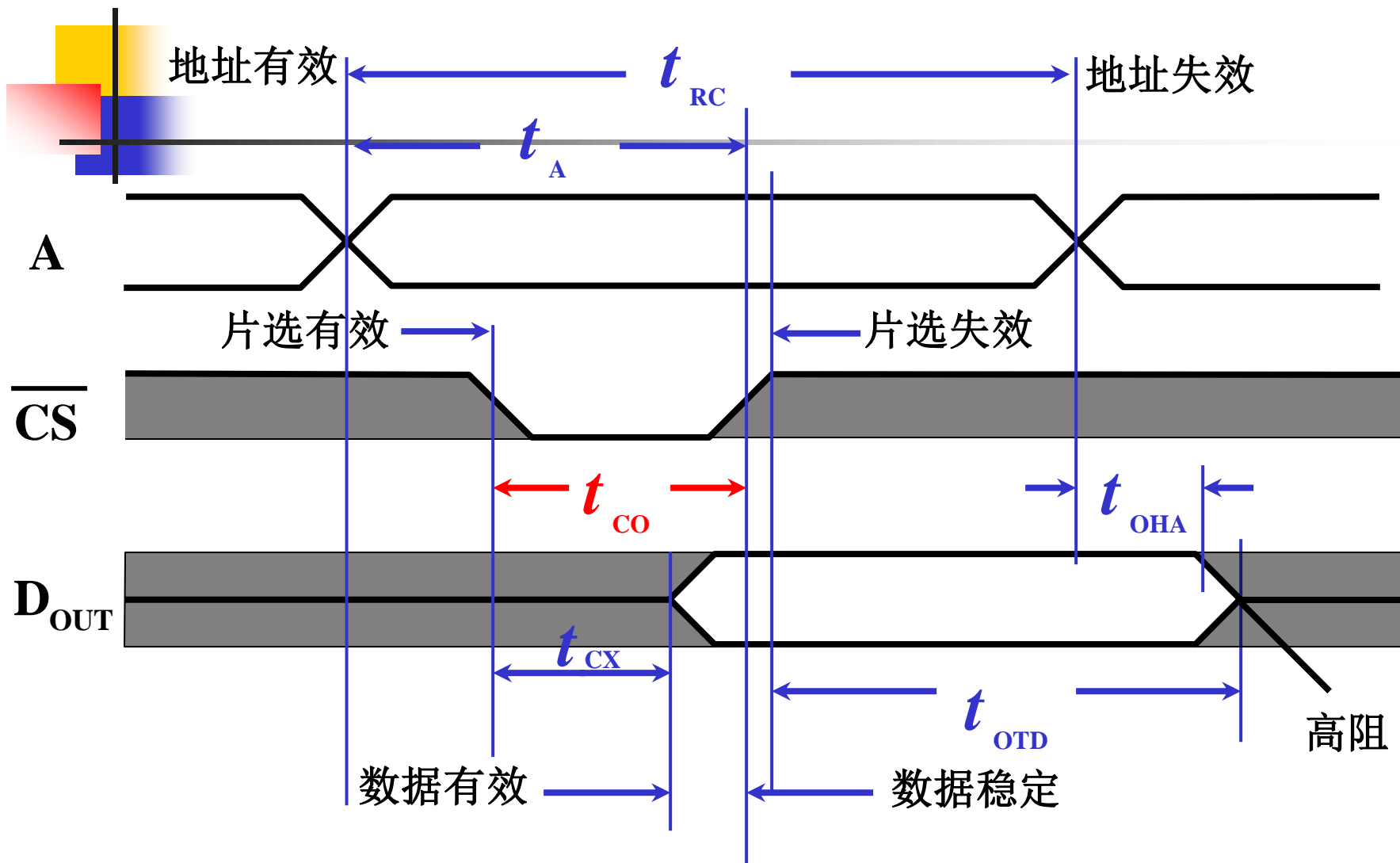
时序图



(2) t_A : 读时间 (从地址有效到数据稳定的时间)

(3) 静态 RAM (2114)读 时序

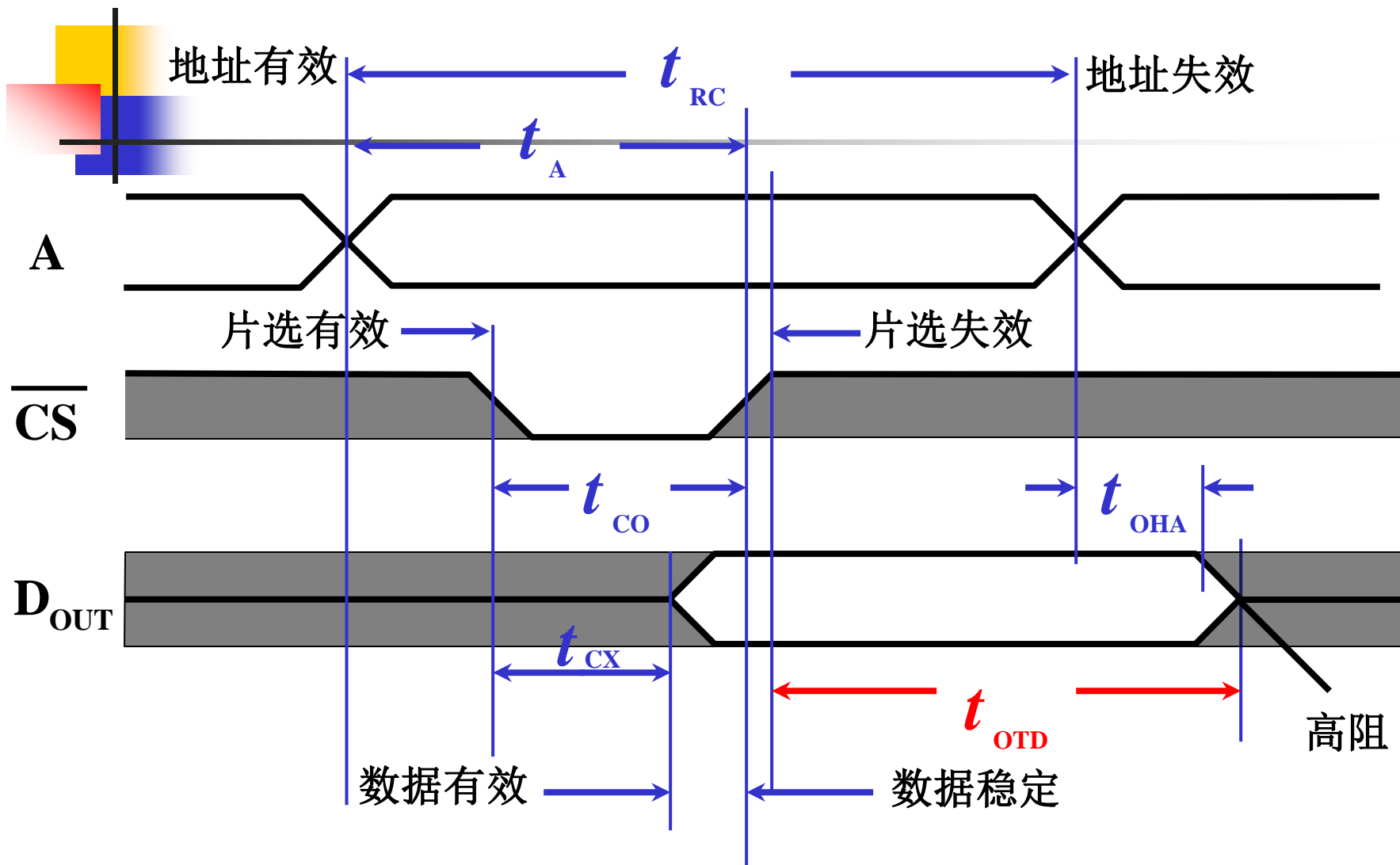
时序图



(3) t_{CO} : 从片选有效到数据稳定的时间

(3) 静态 RAM (2114) 读 时序

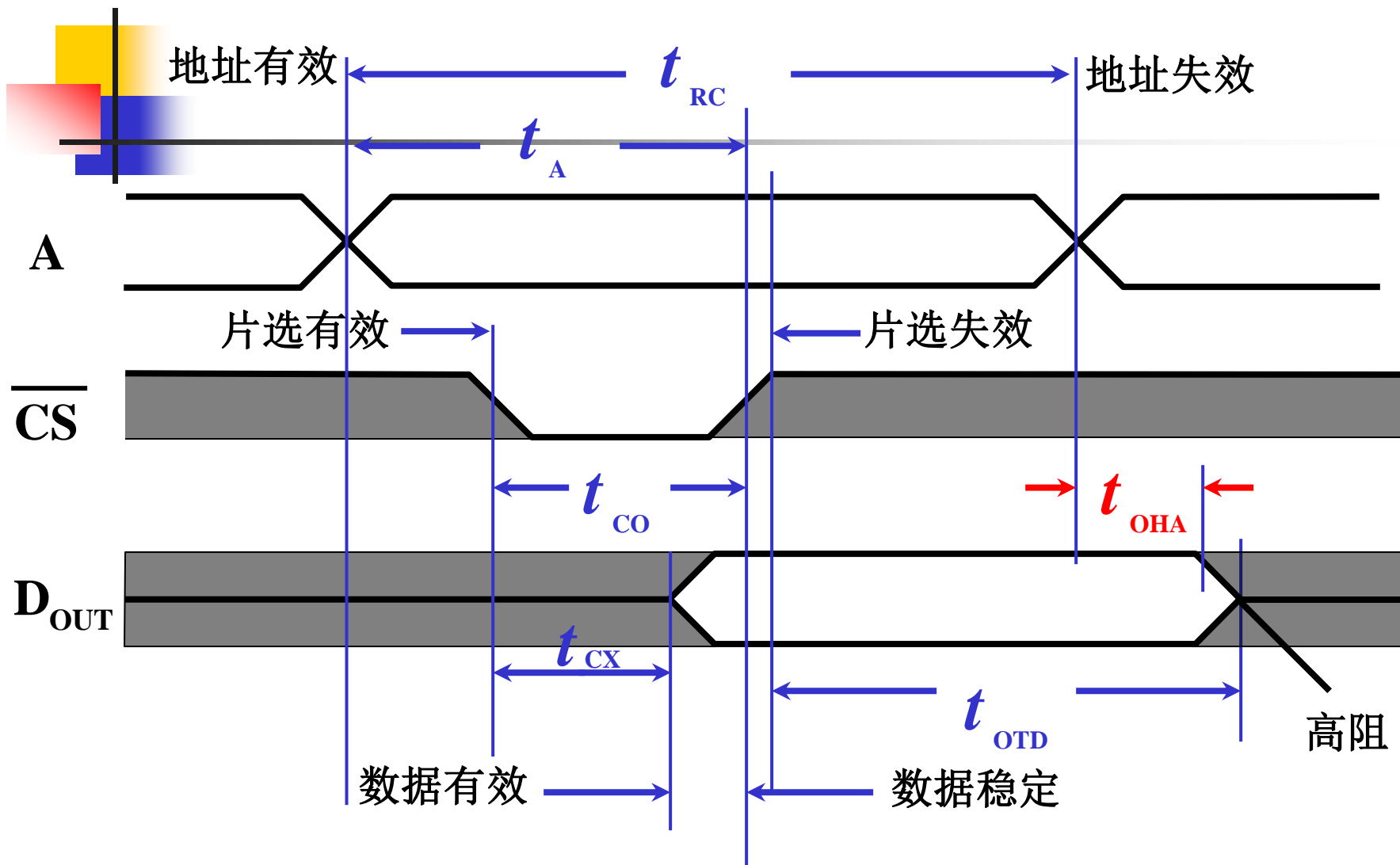
时序图



(4) t_{OTD} : 从片选失效到输出高阻的时间

(3) 静态 RAM (2114)读 时序

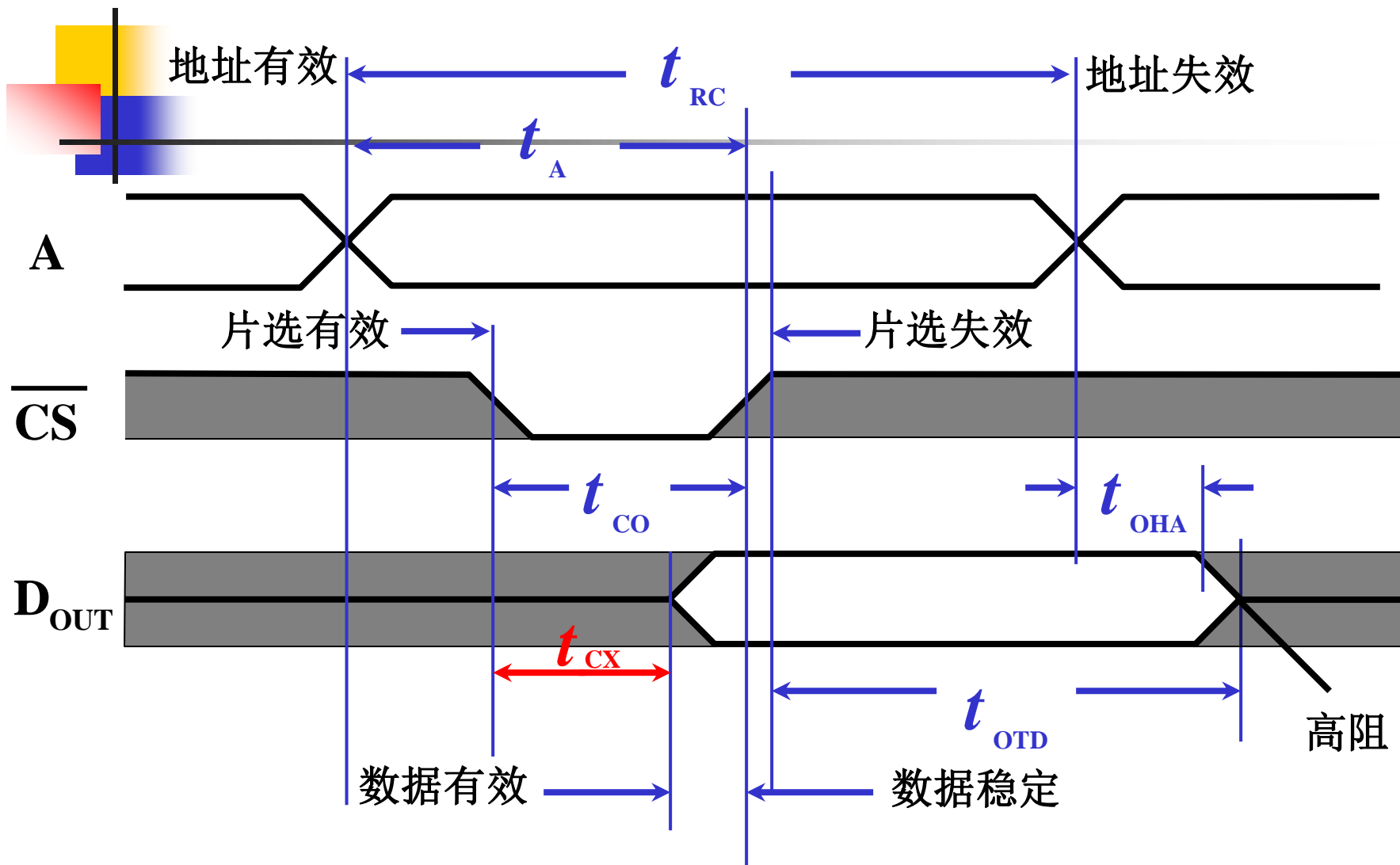
时序图



(5) t_{OHA} : 地址失效后的数据维持时间

(3) 静态 RAM (2114) 读 时序

时序图



(6) t_{CX} : 从片选有效到数据有效的时间



读时序

t_{RC} : 读周期（从地址有效到地址失效的时间）

存取周期(MCT)

读周期 > 读时间

t_A : 读时间（从地址有效到数据稳定的时间）

存取时间(MAT)

t_{CO} : 从片选有效到数据稳定的时间

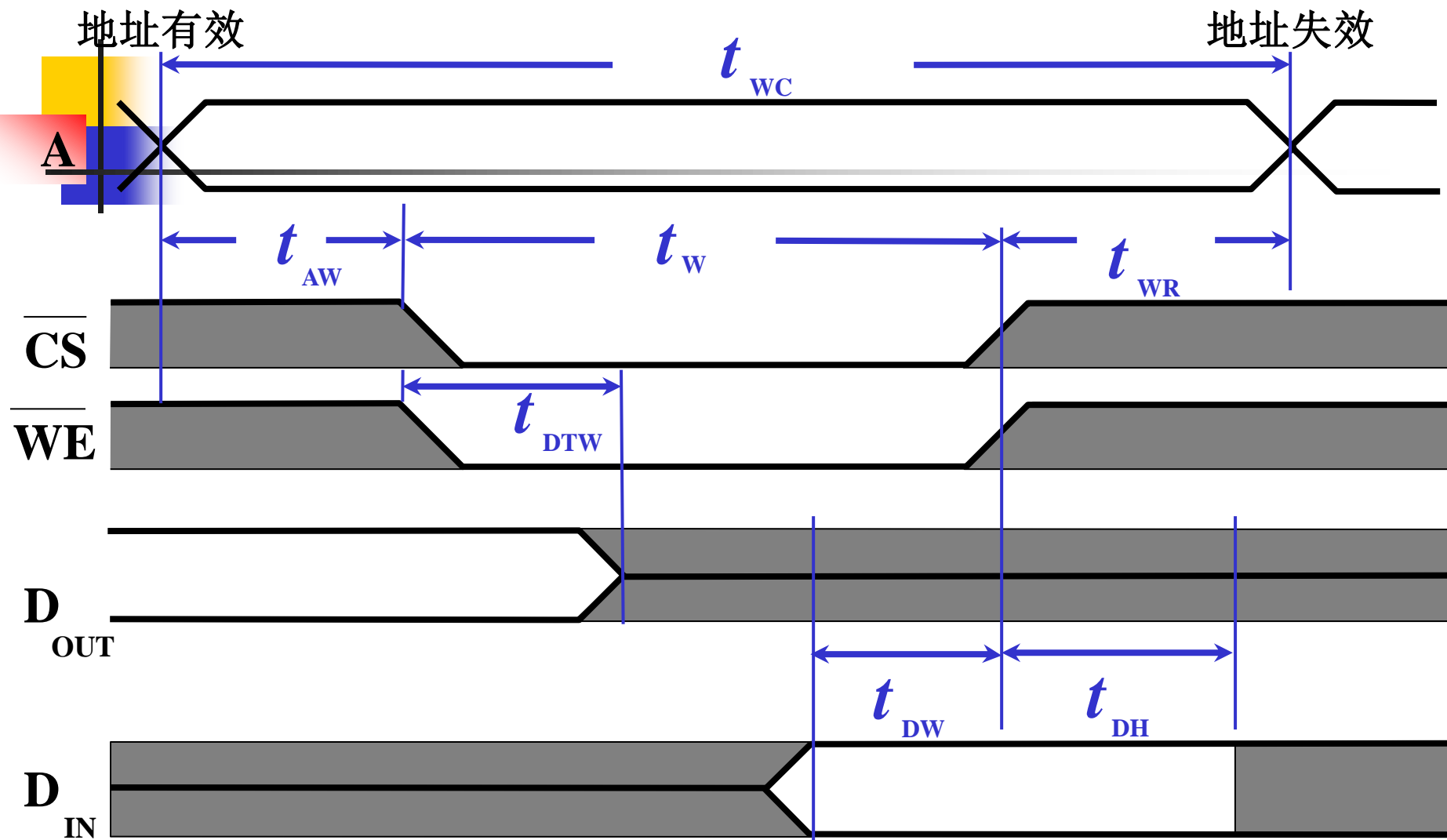
t_{OTD} : 从片选失效到输出高阻的时间

t_{OHA} : 地址失效后的数据维持时间

t_{CX} : 从片选有效到数据有效的时间

(4) 静态 RAM (2114) 写时序

时序图

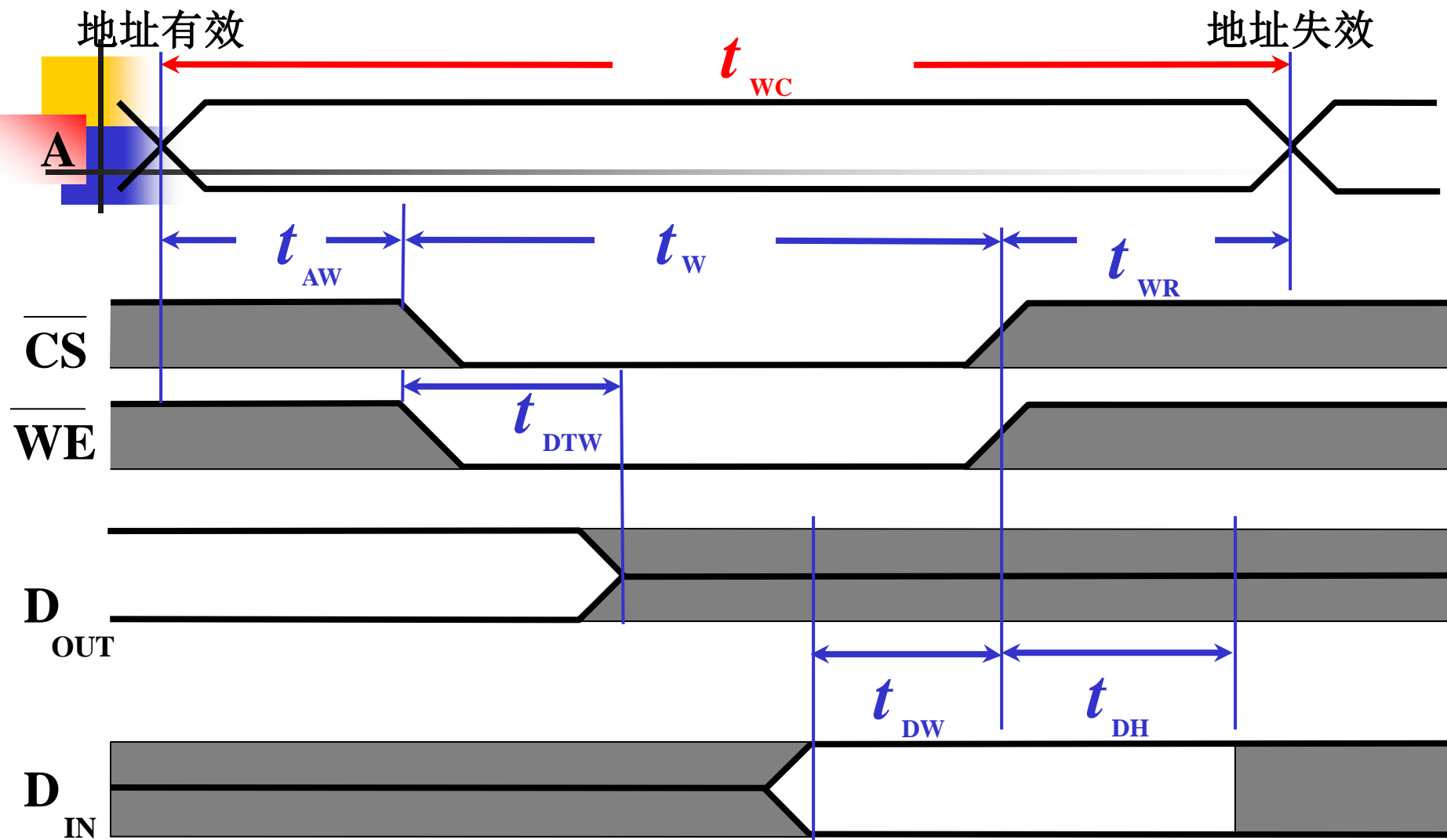


D_{IN}要写入的数据

D_{OUT}写入操作之前数据总线上的数据

(4) 静态 RAM (2114) 写时序

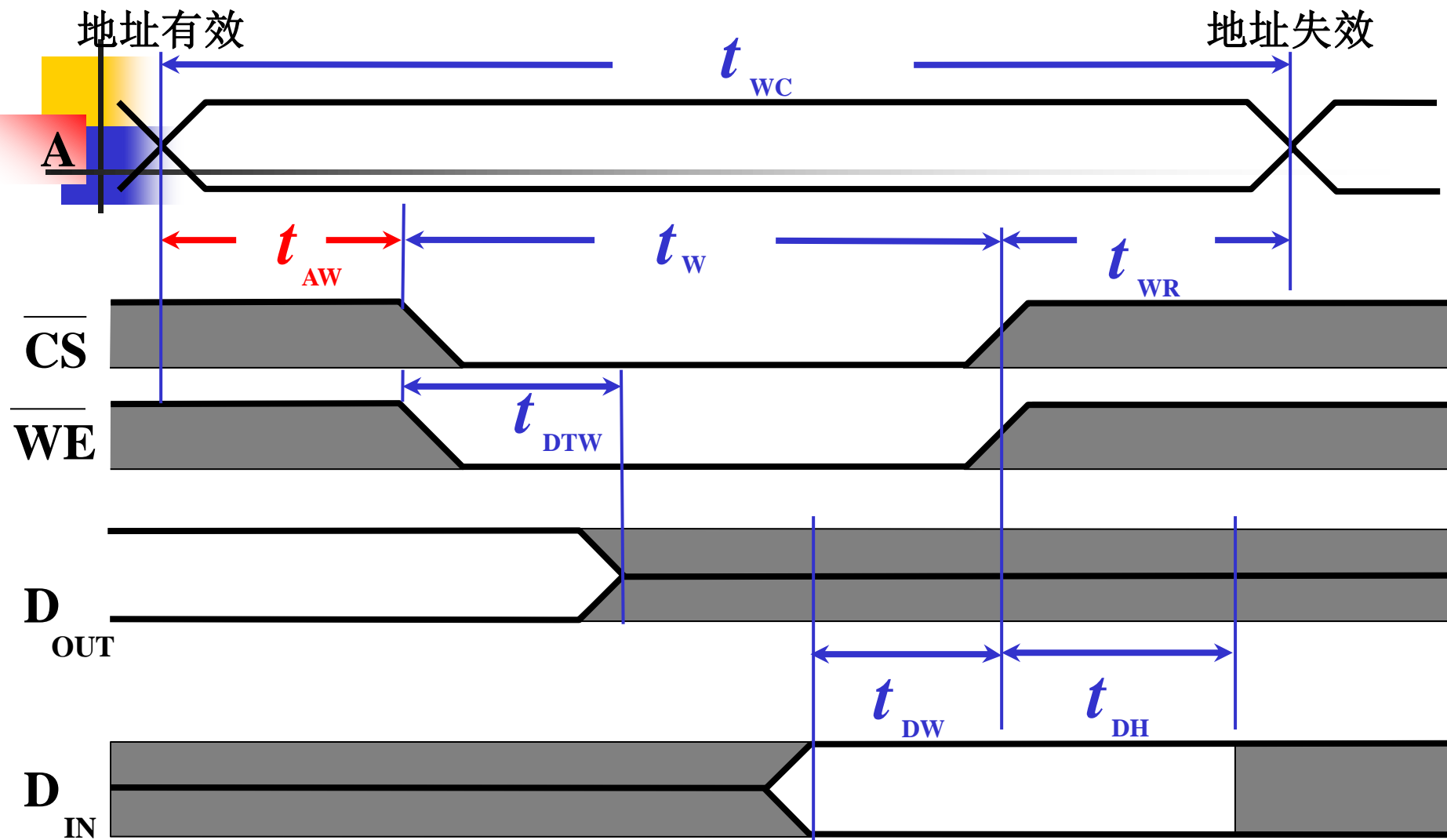
时序图



(1) t_{WC} : 写周期 (从地址有效到地址失效的时间)

(4) 静态 RAM (2114) 写时序

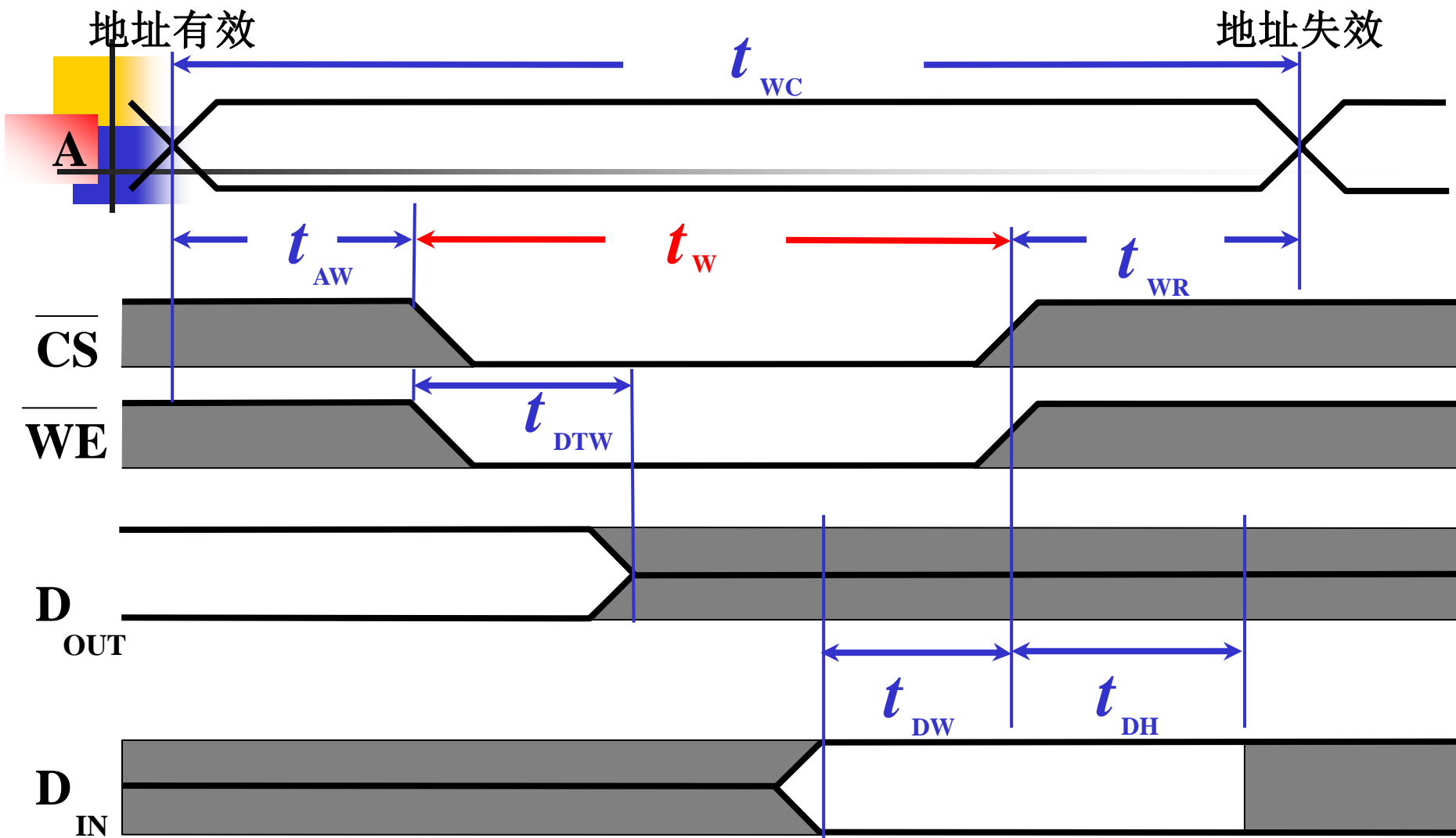
时序图



(2) t_{AW} : 滞后时间 (从地址有效到 $\overline{\text{CS}}$ 和 $\overline{\text{WE}}$ 有效的时间)

(4) 静态 RAM (2114) 写时序

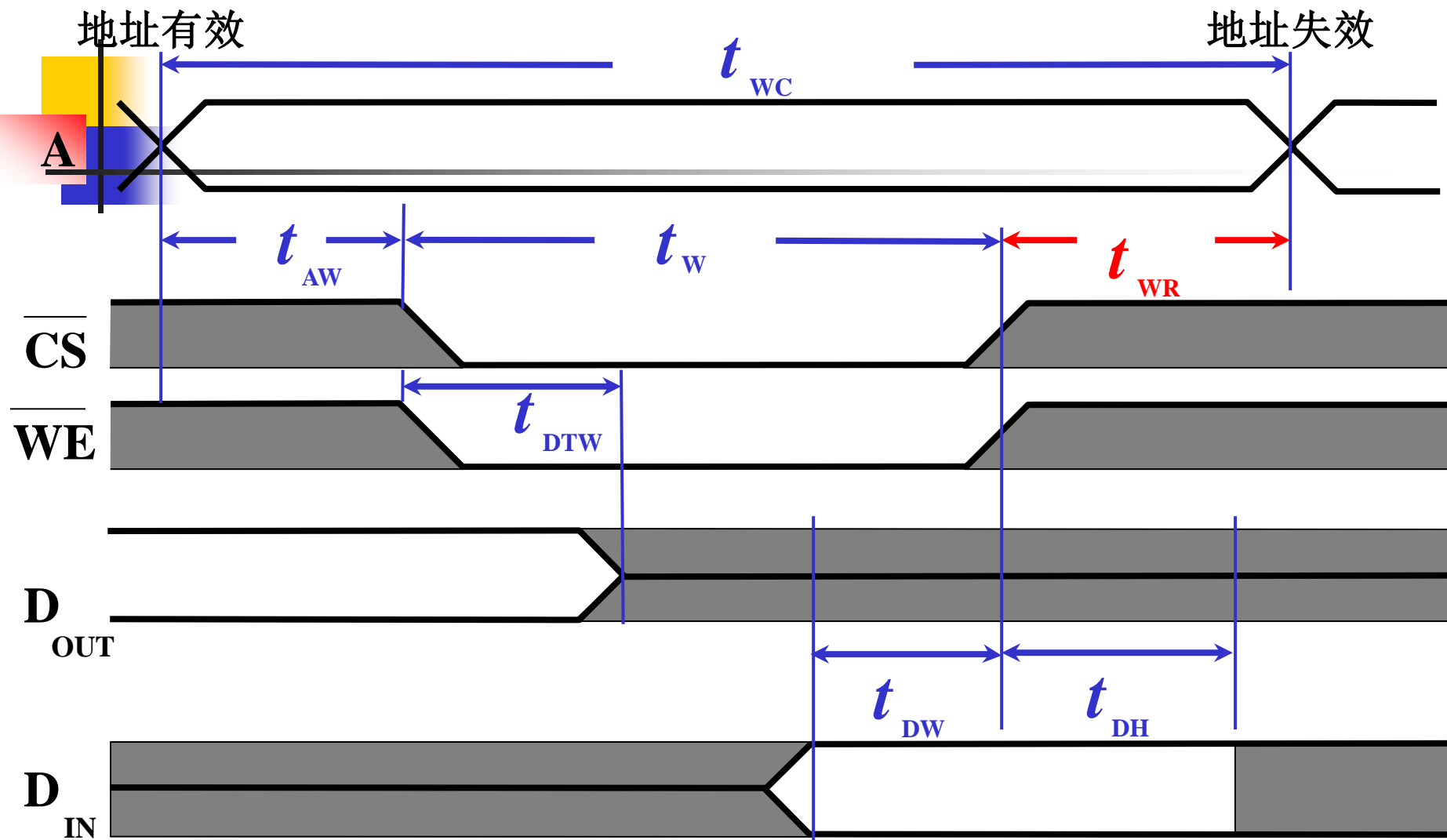
时序图



(3) t_w : 写入时间 (从CS和 \overline{WE} 有效到 \overline{CS} 和 \overline{WE} 失效的时间)

(4) 静态 RAM (2114) 写时序

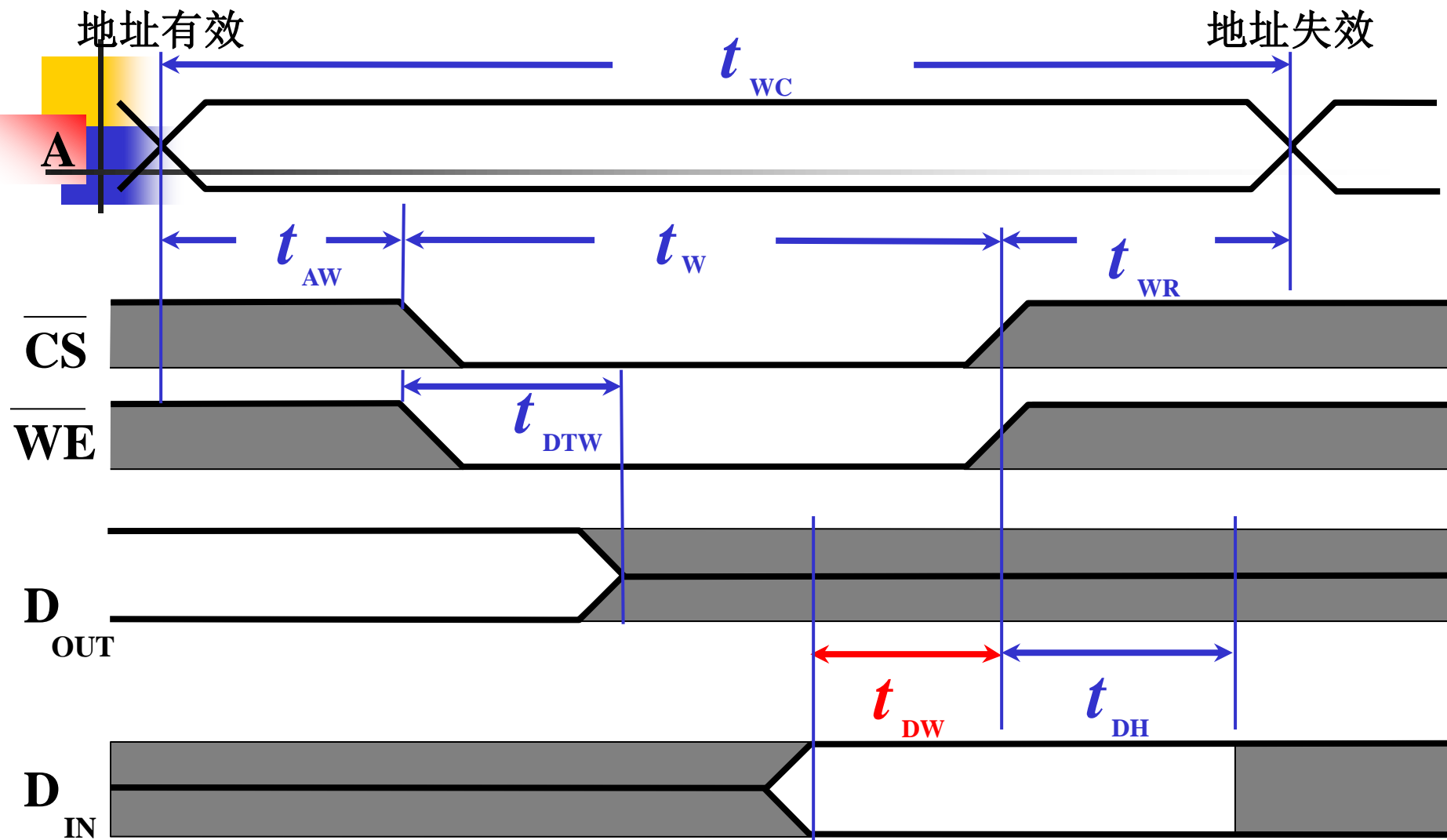
时序图



(4) t_{WR} : 写恢复时间 (从 \overline{CS} 和 \overline{WE} 失效到地址失效的时间)

(4) 静态 RAM (2114) 写时序

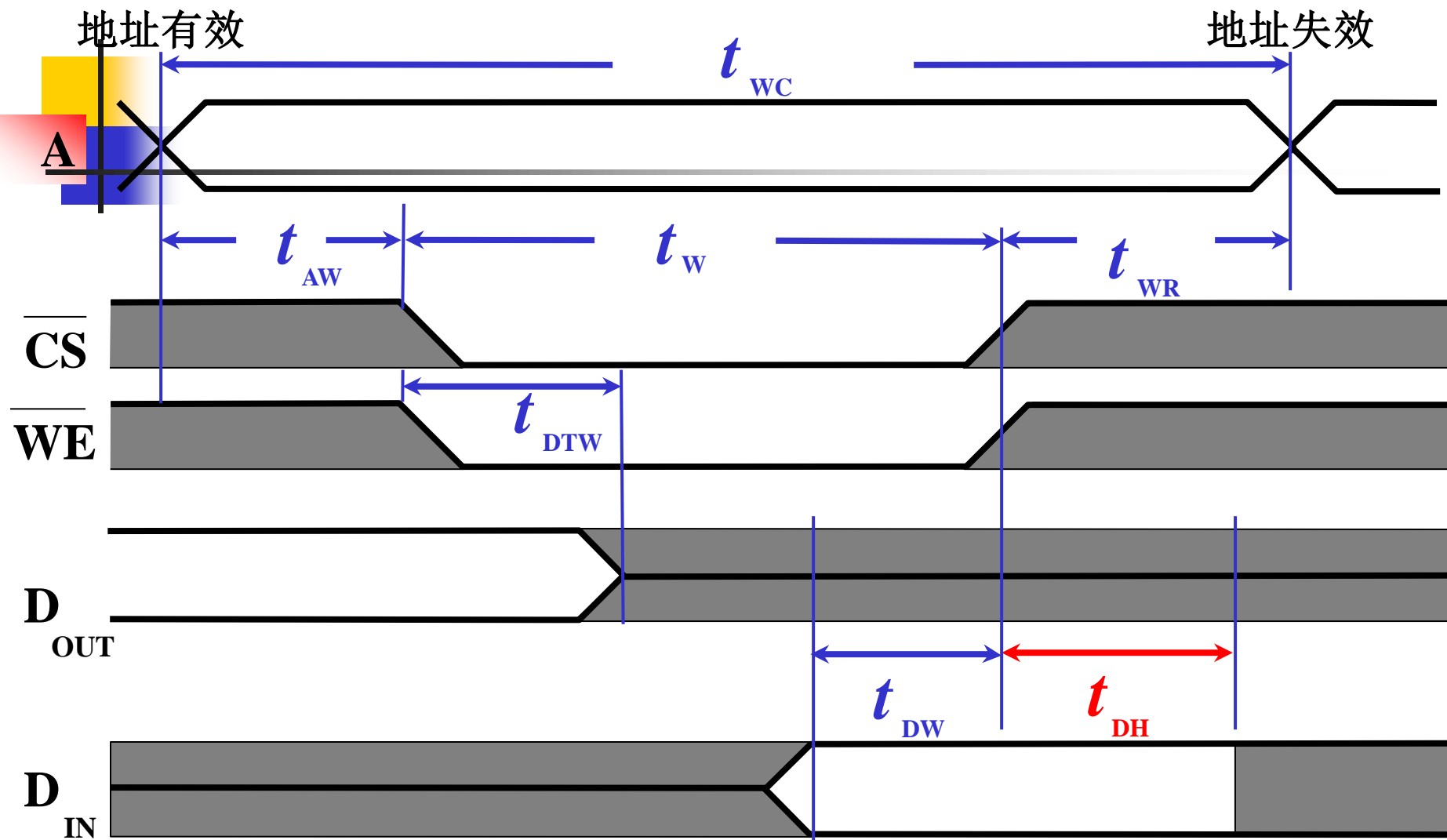
时序图



(5) t_{DW} : \overline{CS} 和 \overline{WE} 失效前的写入数据维持时间

(4) 静态 RAM (2114) 写时序

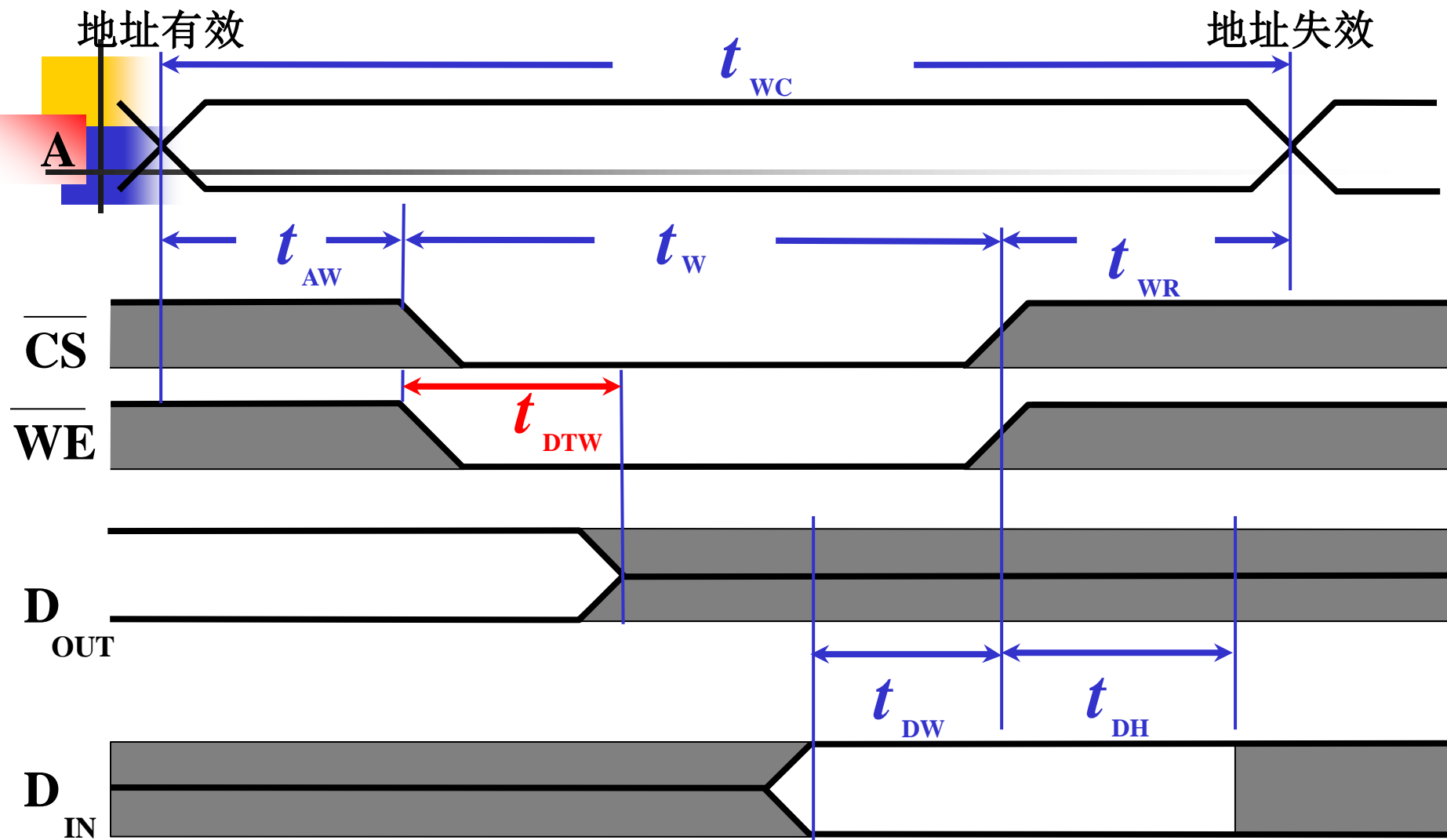
时序图



(6) t_{DH} : $\overline{\text{CS}}$ 和 $\overline{\text{WE}}$ 失效后的写入数据维持时间

(4) 静态 RAM (2114) 写时序

时序图



(7) t_{DTW} : 从 $\overline{\text{CS}}$ 和 $\overline{\text{WE}}$ 有效到 D_{OUT} 无效的时间

写时序

写周期 > 写入时间

t_{WC} : 写周期（从地址有效到地址失效的时间） 存取周期(MCT)

t_{AW} : 滞后时间（从地址有效到/ \overline{CS} 和/ \overline{WE} 有效的时间）

存取时间(MAT)

t_W : 写入时间（从/ \overline{CS} 和/ \overline{WE} 有效到/ \overline{CS} 和/ \overline{WE} 失效的时间）

t_{WR} : 写恢复时间（从/ \overline{CS} 和/ \overline{WE} 失效到地址失效的时间）

t_{DW} : / \overline{CS} 和/ \overline{WE} 失效前的数据维持时间

t_{DH} : / \overline{CS} 和/ \overline{WE} 失效后的数据维持时间

t_{DTW} : 从/ \overline{CS} 和/ \overline{WE} 有效到 D_{OUT} 无效的时间

2. 动态 RAM (DRAM)

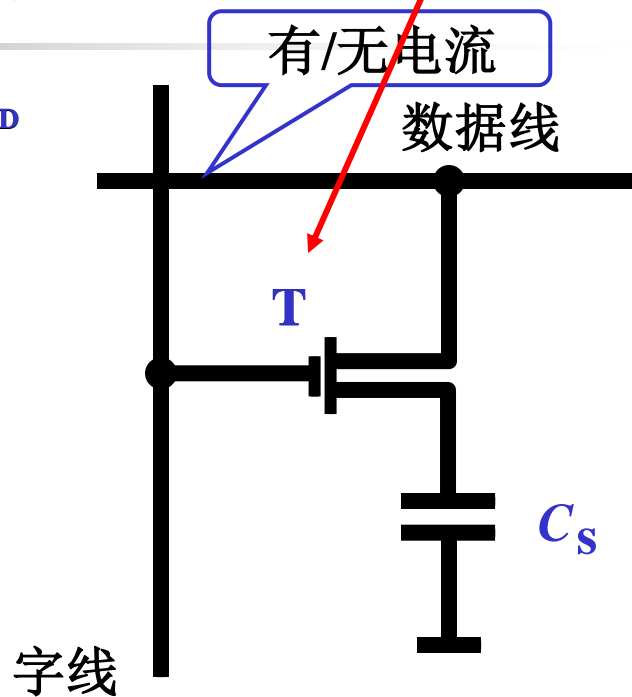
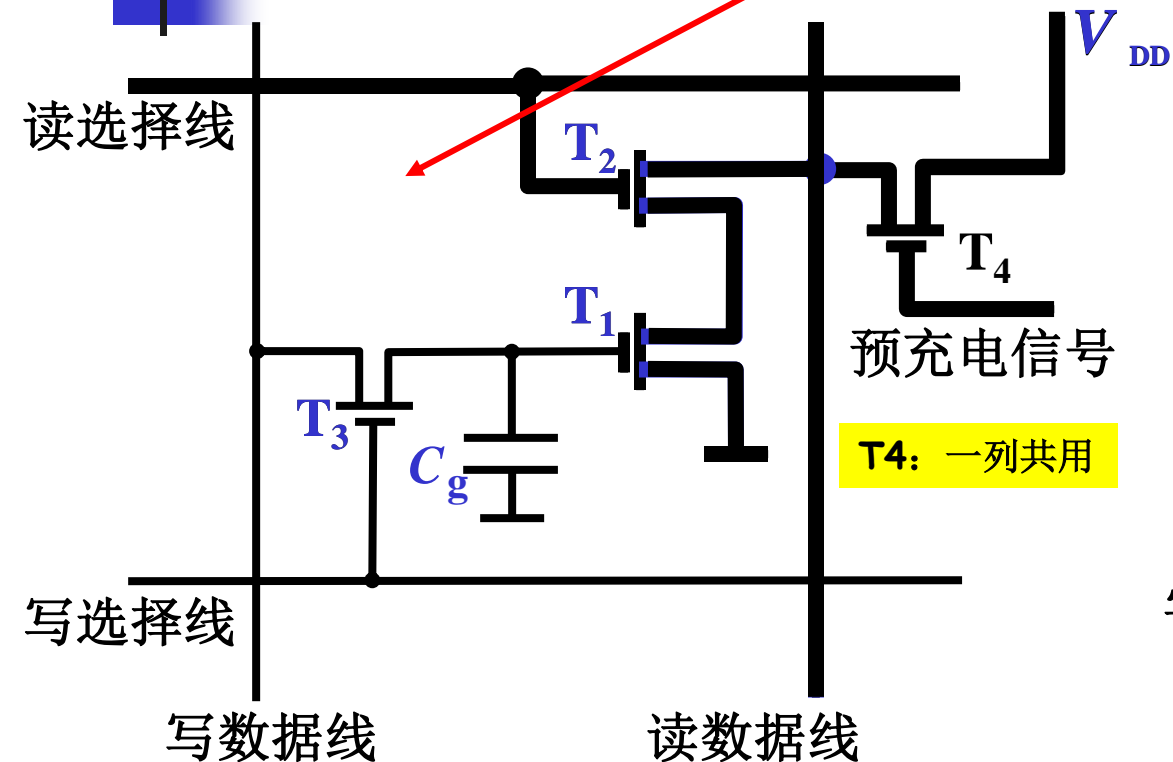
(1) 动态 RAM 基本单元电路

三管式

T1、T2、 T3

单管式

T



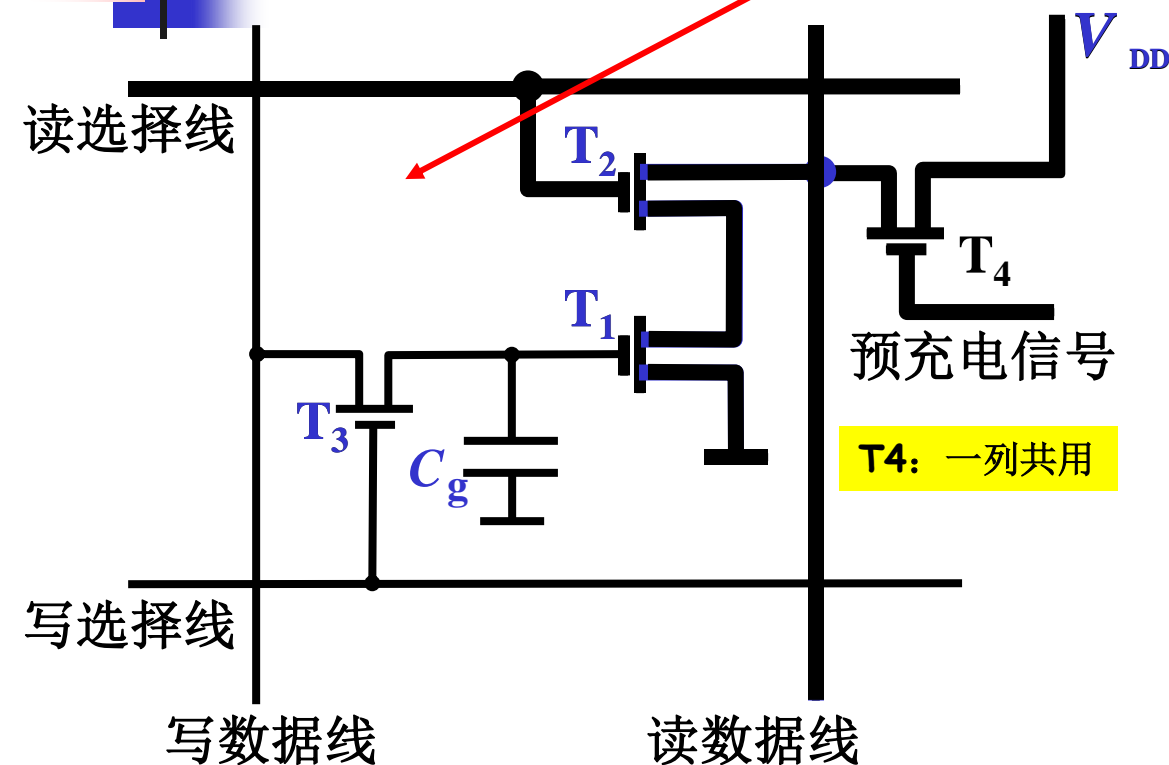
2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

三管式 读出

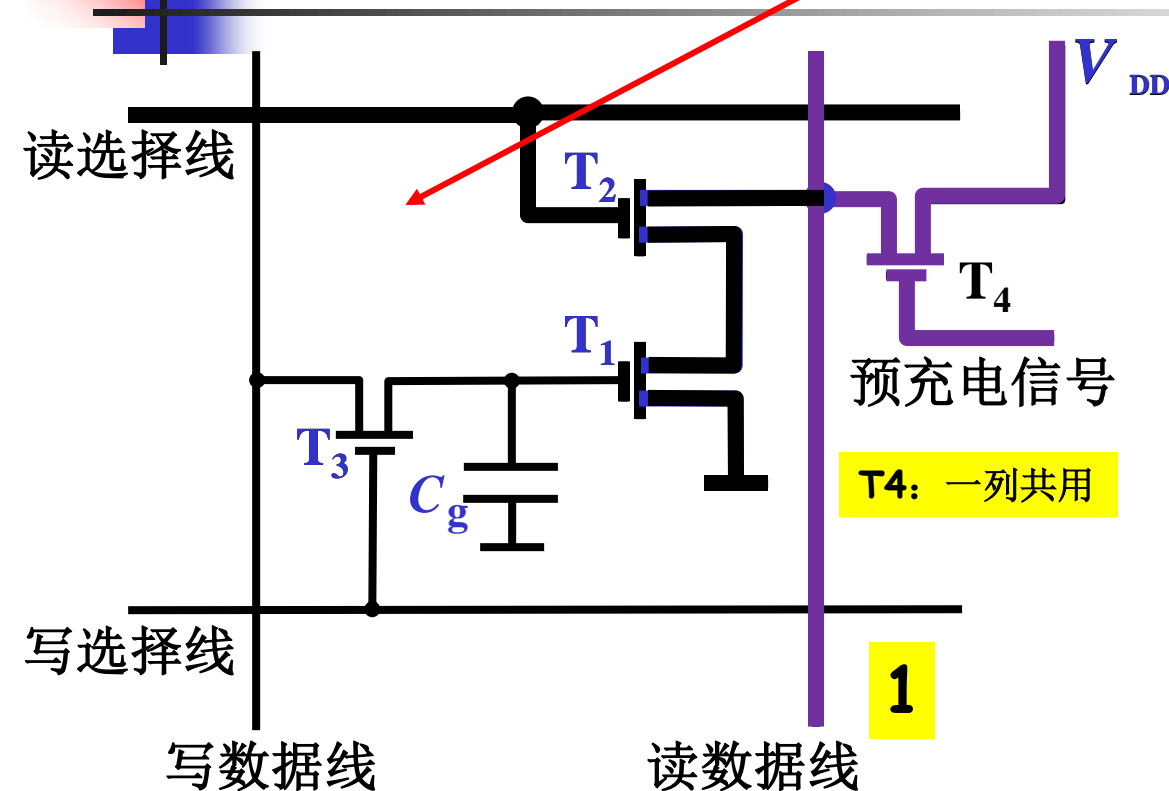
共三步

T4: 一列共用



2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



三管式 读出

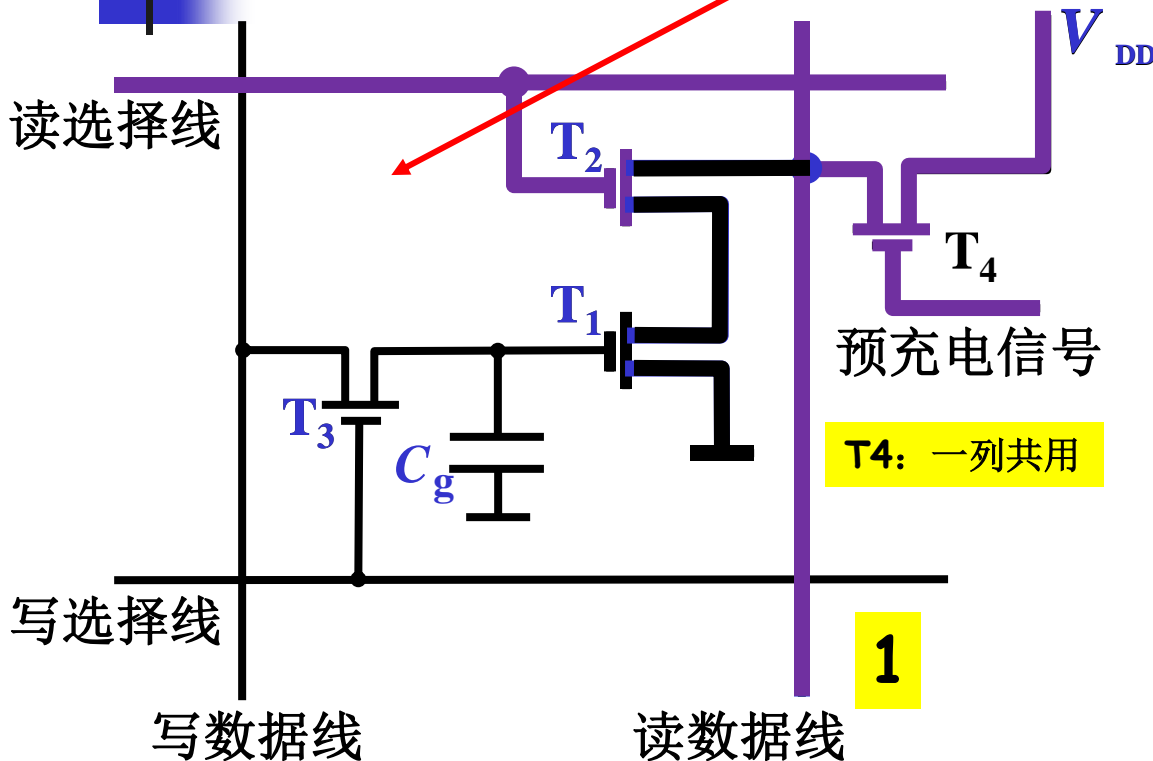
第一步

1、预充电信号打开 T_4 ，使读数据线达高电平(1)；

T_4 : 一列共用

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



三管式 读出

第二步

1、预充电信号打开 T_4 ，使读数据线达高电平(1)；

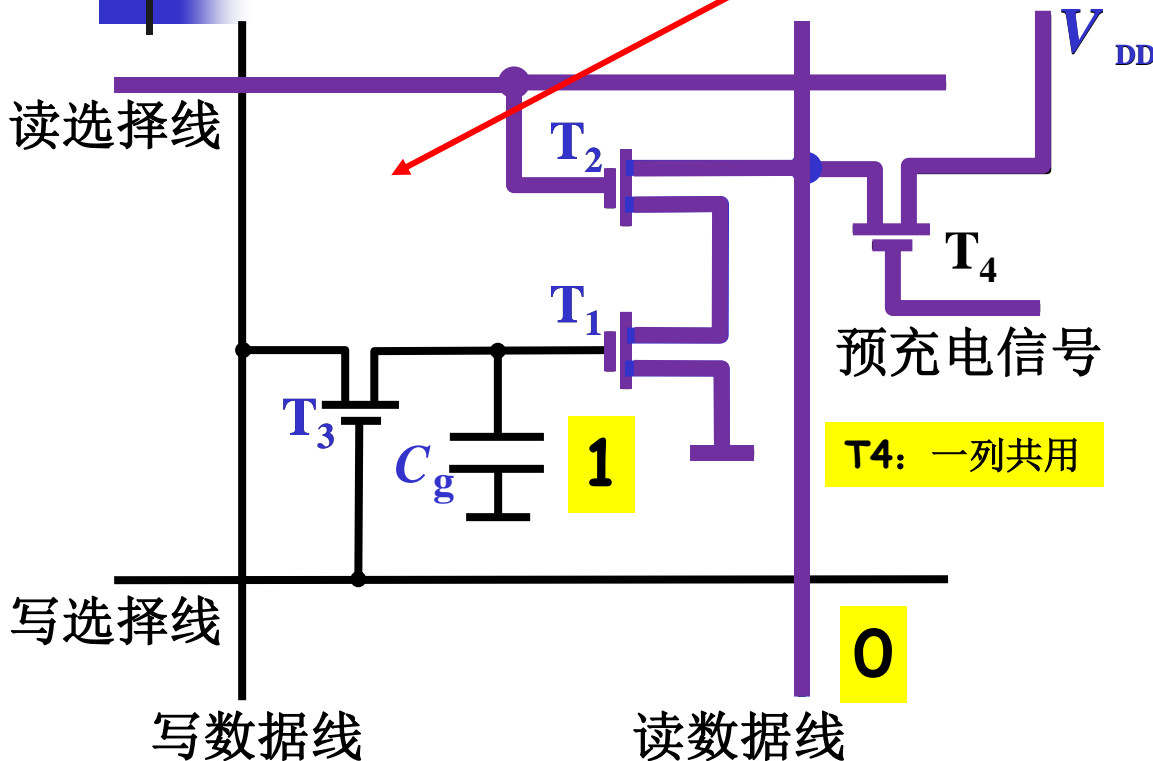
2、读选择线打开 T_2 ；

T4: 一列共用

1

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



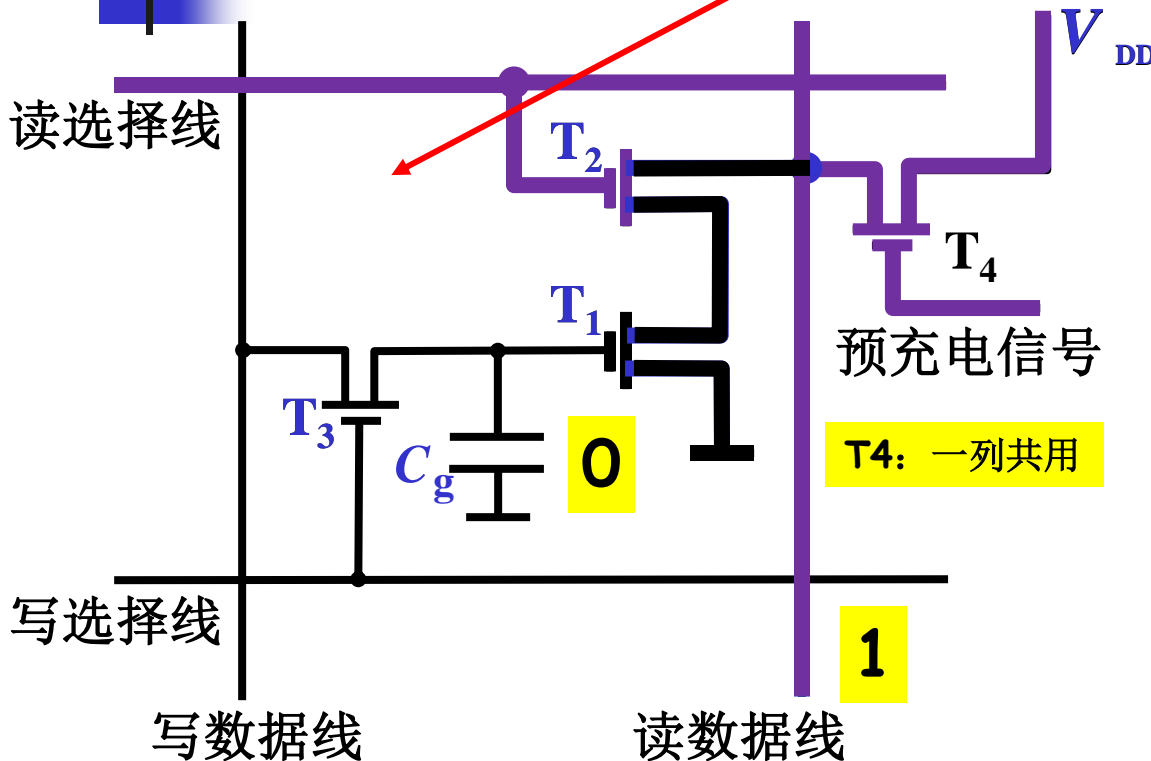
三管式 读出

第三步(1)

- 1、预充电信号打开 T_4 ，使读数据线达高电平(1)；
- 2、读选择线打开 T_2 ；
- 3、若 C_g 存入高电平(1)， T_1 打开，读数据线为低电平(0)；

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



读出与原存信息相反

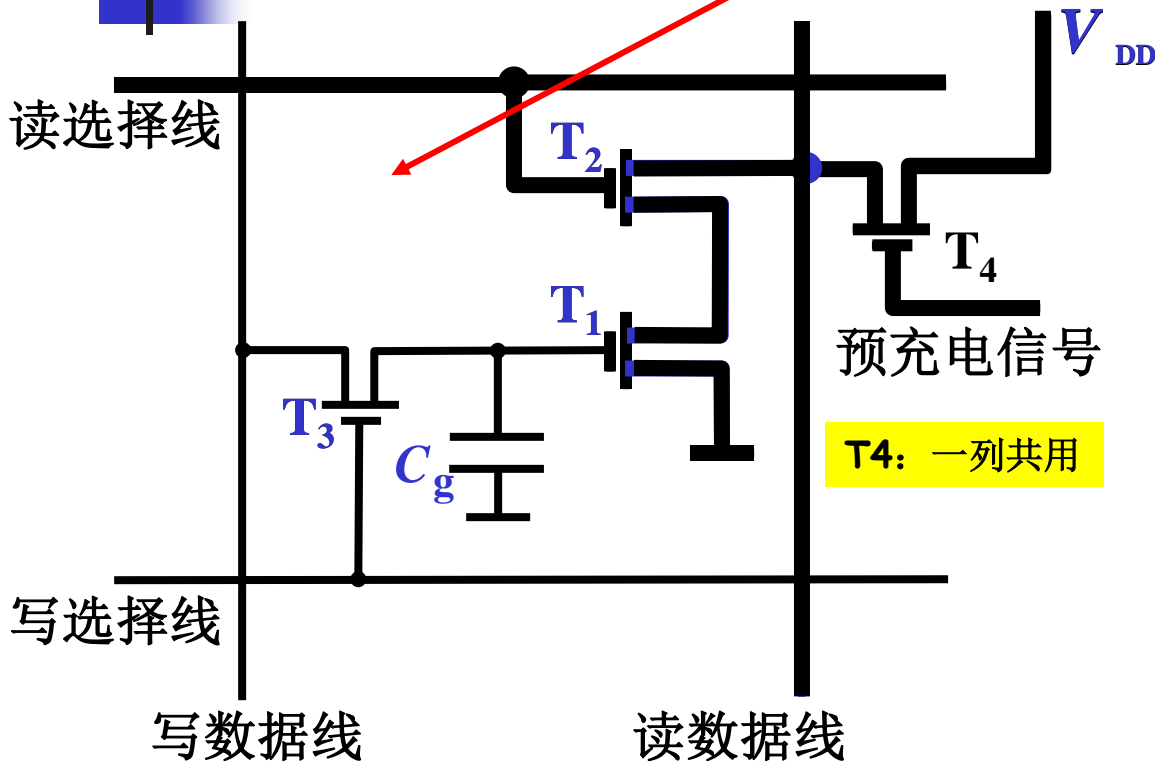
三管式 读出

第三步(2)

- 1、预充电信号打开 T_4 ，使读数据线达高电平(1)；
- 2、读选择线打开 T_2 ；
- 3、若 C_g 存入高电平(1)， T_1 打开，读数据线为低电平(0)；若 C_g 存入低电平(0)， T_1 截至，读数据线为高电平(1)。

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



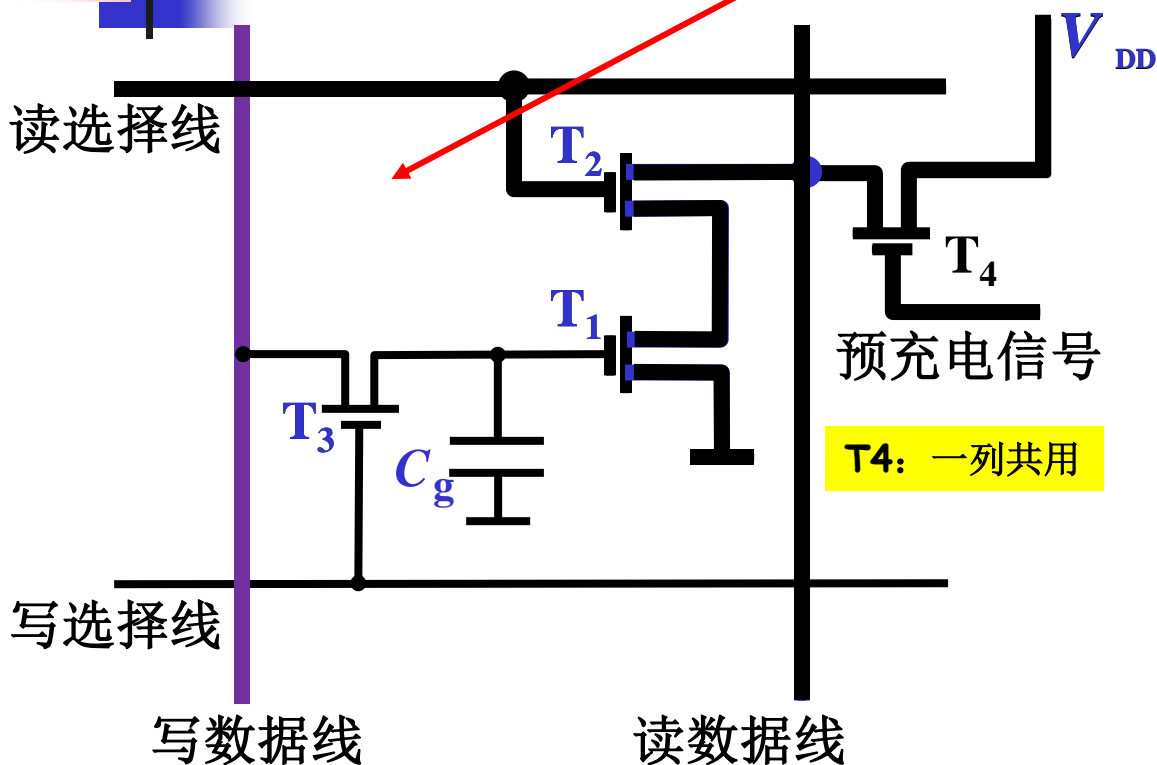
T4: 一列共用

三管式 写入

共三步

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



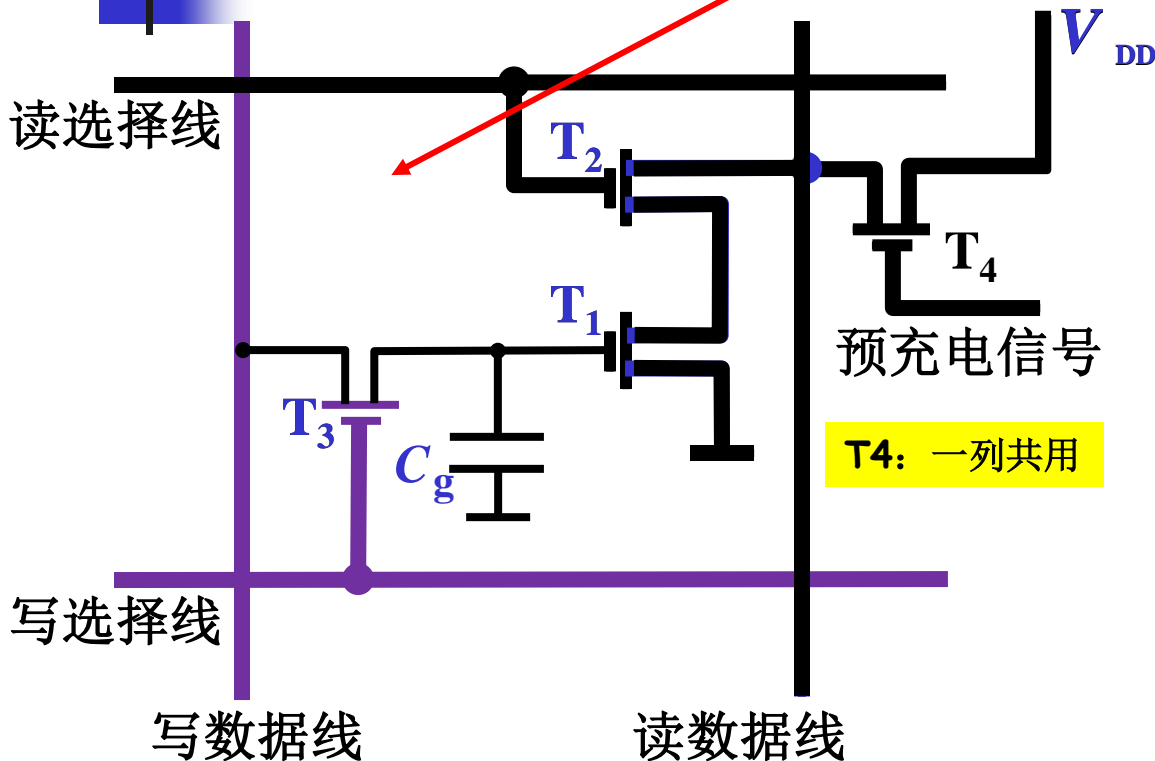
三管式 写入

第一步

1、写入信号加到写数据线上;

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



三管式 写入

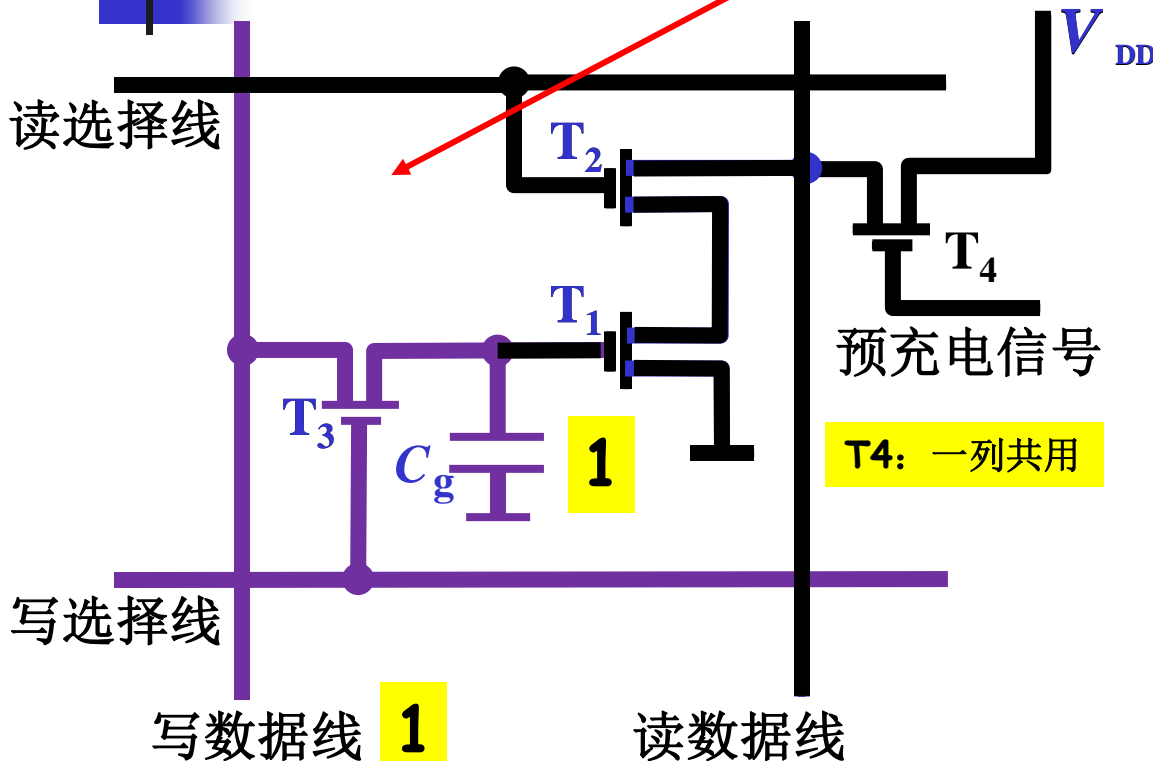
第二步

1、写入信号加到写数据线上;

2、写选择线打开 T_3 ;

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



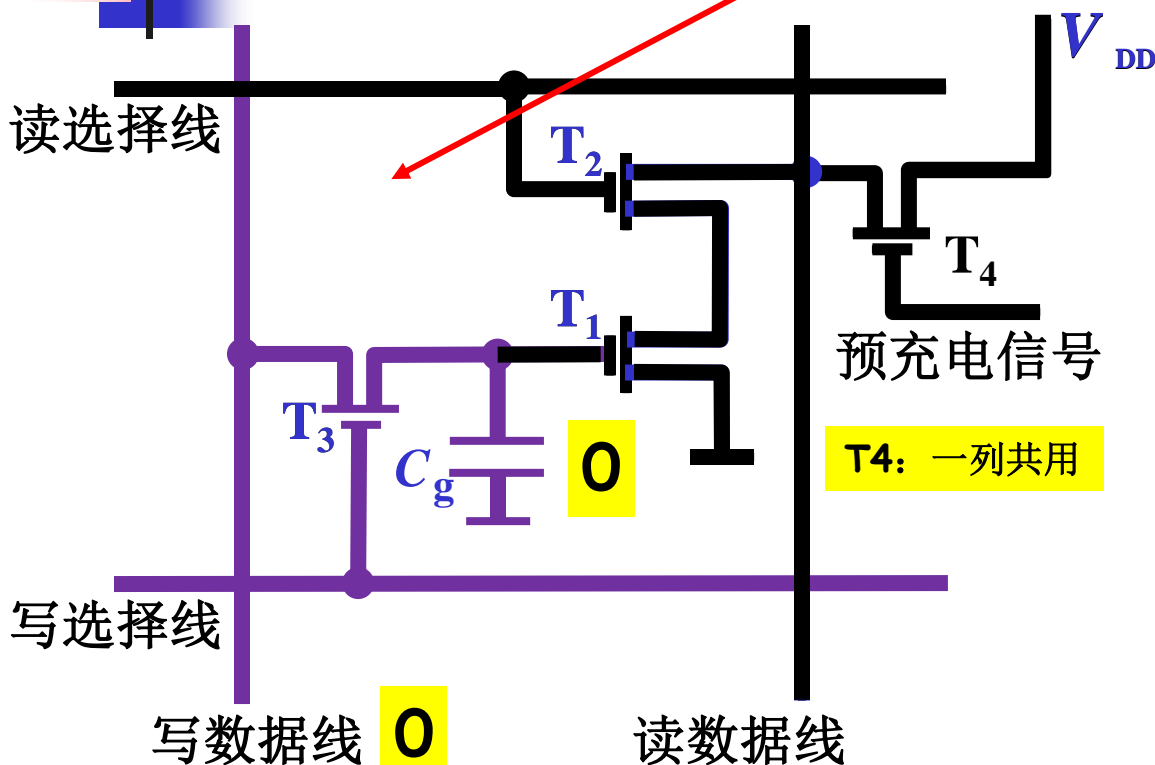
三管式 写入

第三步(1)

- 1、写入信号加到写数据线上；
- 2、写选择线打开 T_3 ；
- 3、若写入信号为高电平(1)，则 C_g 充电(使 $C_g=1$)；

2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路



写入与输入信息相同

三管式

T1、T2、T3

三管式 写入

第三步(2)

1、写入信号加到写数据线上；

2、写选择线打开 T_3 ;

3、若写入信号为高电平(1)，则 C_g 充电(使 $C_g=1$)；若写入信号为低电平(0)，则 C_g 放电(使 $C_g=0$)。

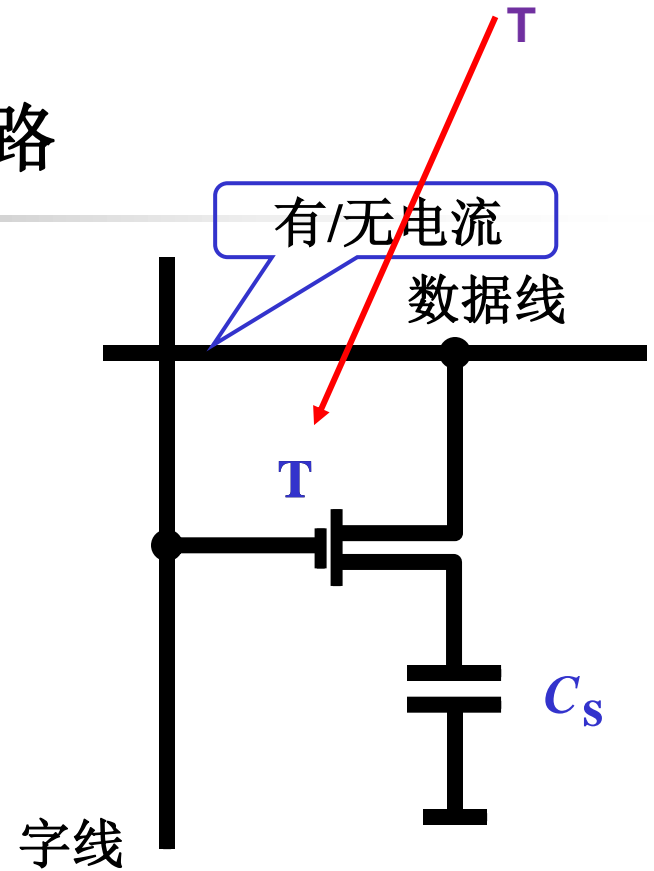
2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

单管式

单管式 读出

共二步



2. 动态 RAM (DRAM)

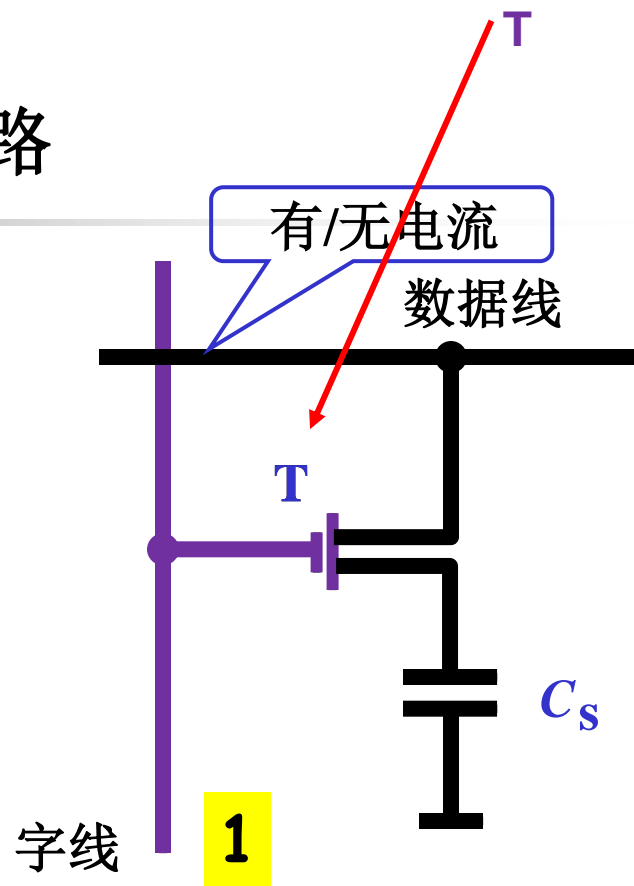
(1) 动态 RAM 基本单元电路

单管式

单管式 读出

第一步

1、字线上的高电平(1)使T管导通；



2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

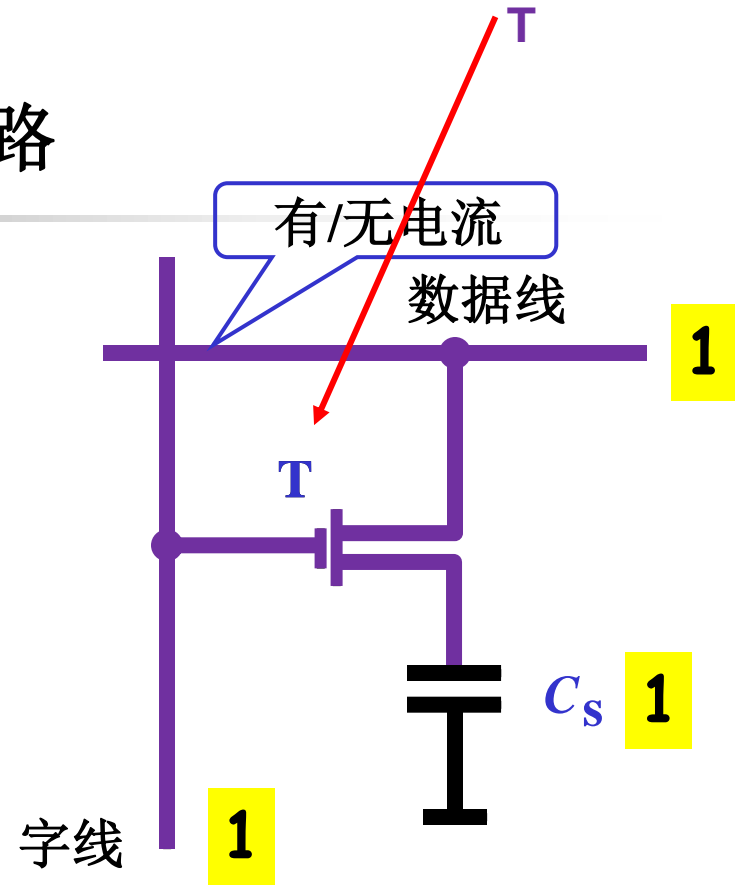
单管式

单管式 读出

第二步(1)

1、字线上的高电平(1)使T管导通；

2、若 C_s 有电荷(1)，经T管在数据线上产生电流，可视为读出“1”；



2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

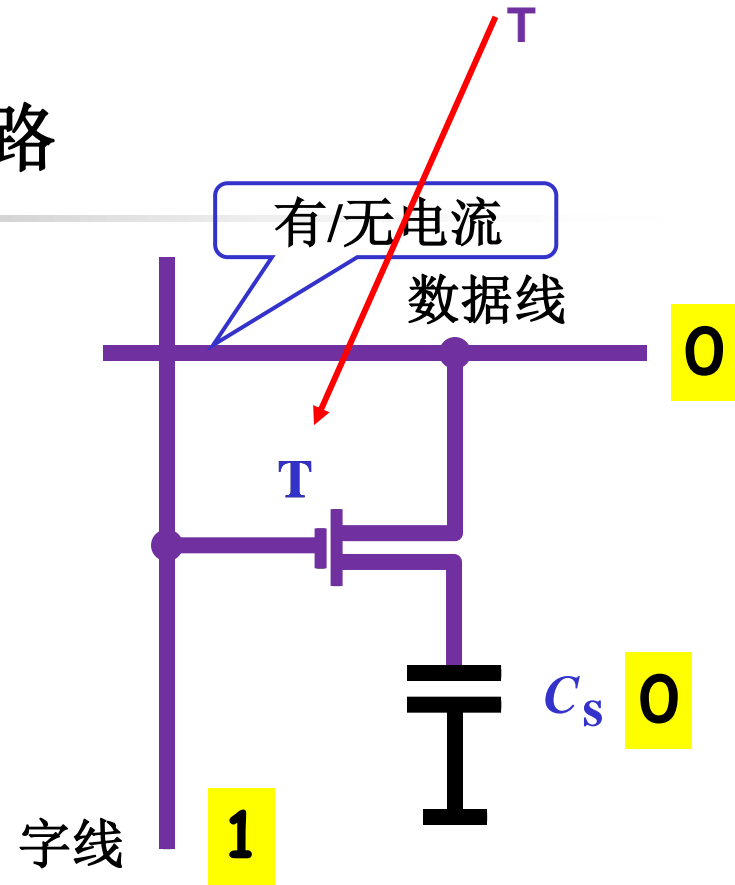
单管式

单管式 读出

第二步(2)

1、字线上的高电平(1)使T管导通；

2、若 C_s 有电荷(1)，经T管在数据线上产生电流，可视为读出“1”；若 C_s 无电荷(0)，则数据线上无电流，可视为读出“0”。



读出时数据线有电流 为“1”，无电流为“0”

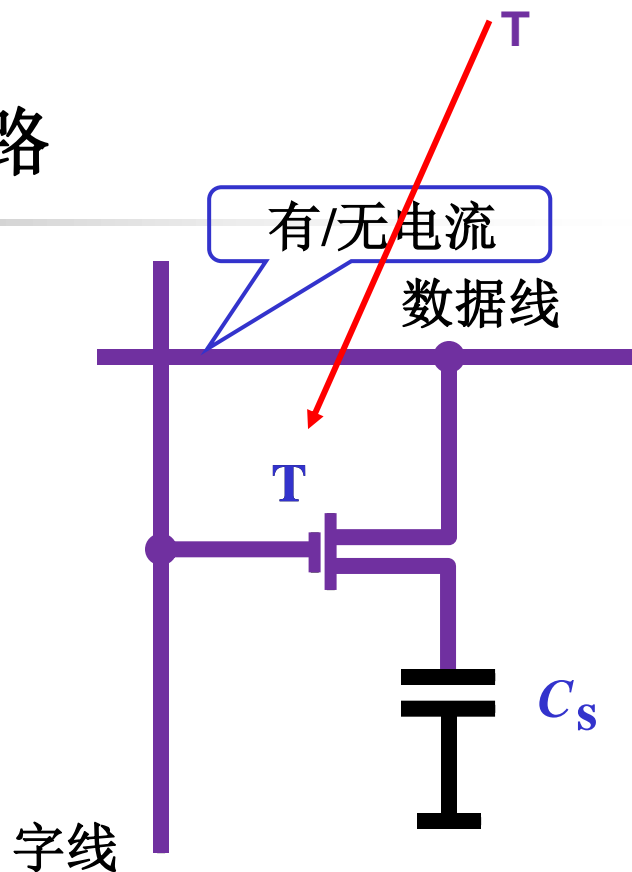
2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

单管式

单管式 读出

读操作结束时， C_s 的电荷已释放完毕，故是破坏性读出，必须再生。

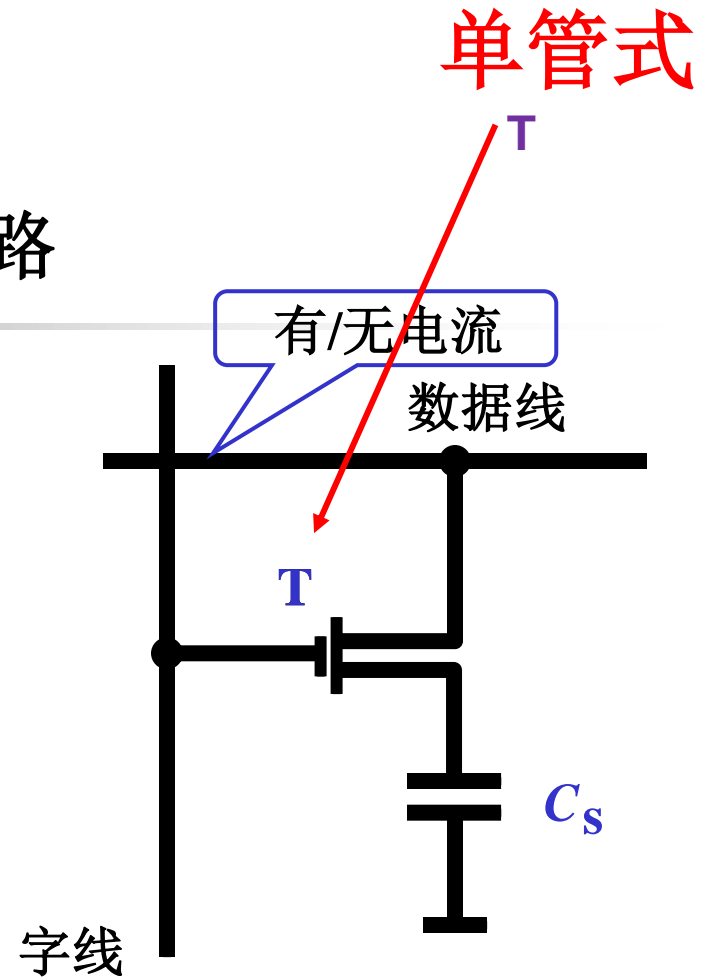


2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

单管式 写入

共二步



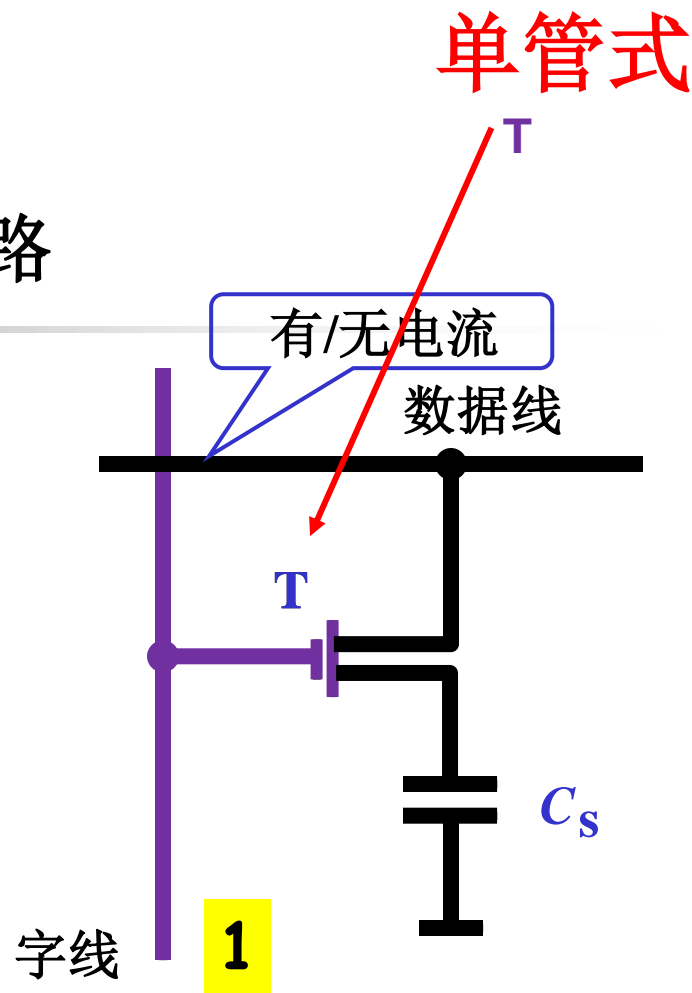
2. 动态 RAM (DRAM)

(1) 动态 RAM 基本单元电路

单管式 写入

第一步

1、字线上的高电平(1)使T管导通；



2. 动态 RAM (DRAM)

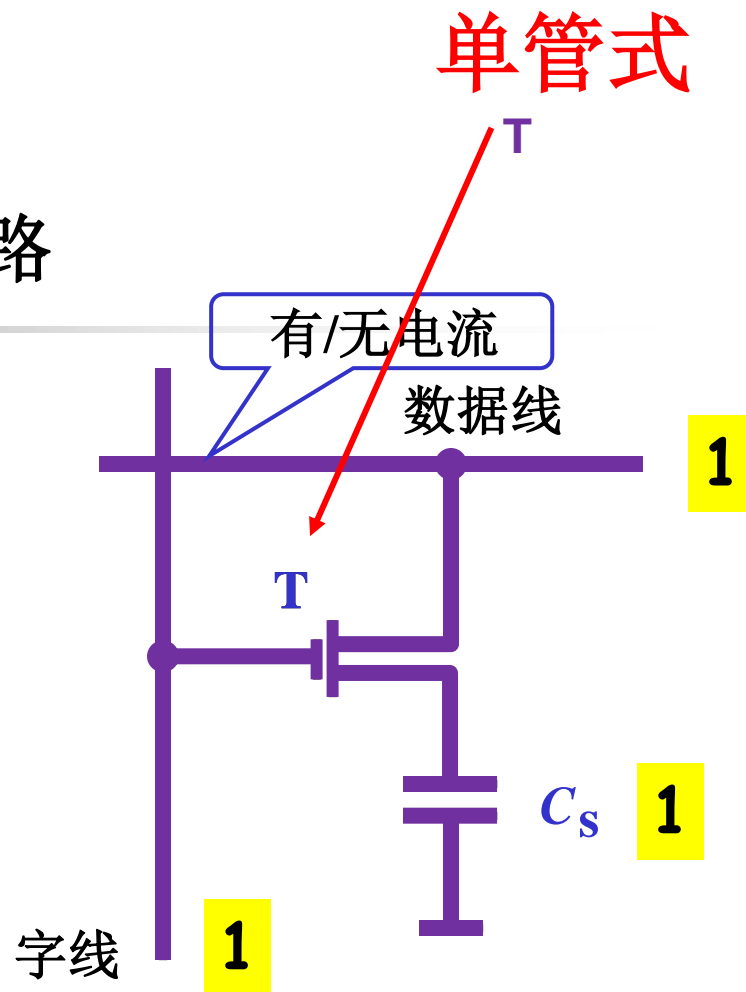
(1) 动态 RAM 基本单元电路

单管式 写入

第二步(1)

1、字线上的高电平(1)使T管导通；

2、若数据线为高电平(1)，则通过T管，向 C_s 充电，使其为“1”；



2. 动态 RAM (DRAM)

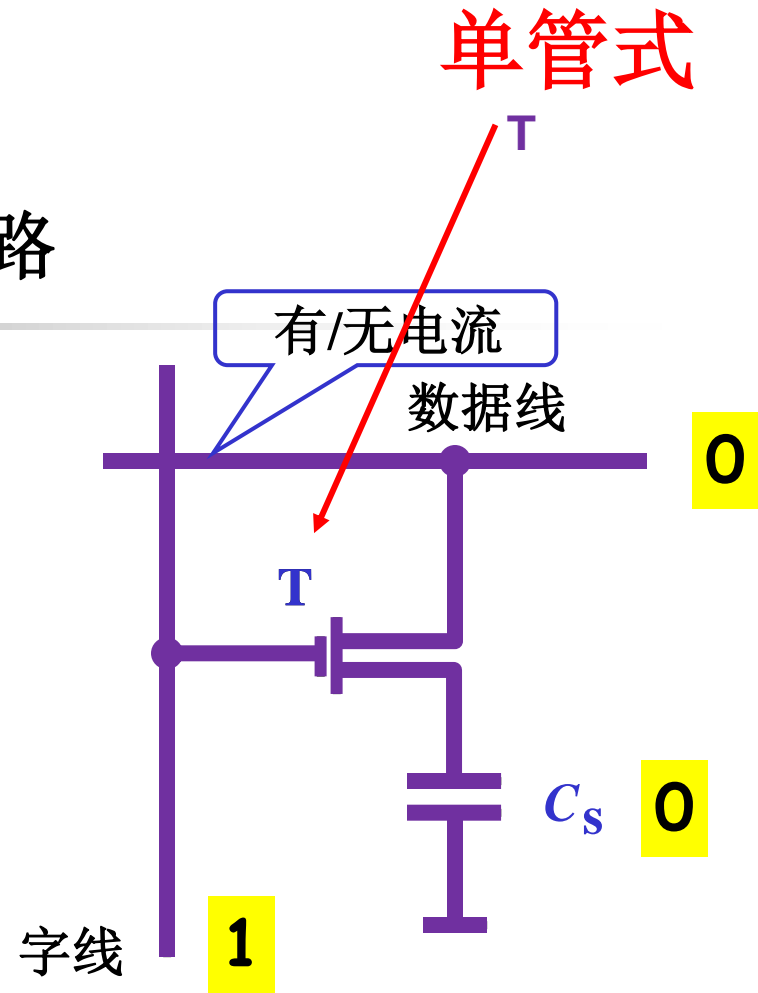
(1) 动态 RAM 基本单元电路

单管式 写入

第二步(2)

1、字线上的高电平(1)使T管导通；

2、若数据线为高电平(1)，则通过T管，向 C_s 充电，使其为“1”；若数据线为低电平(0)，则通过T管，使 C_s 放电，使其为“0”。



写入时 C_s 充电为“1”，放电为“0”



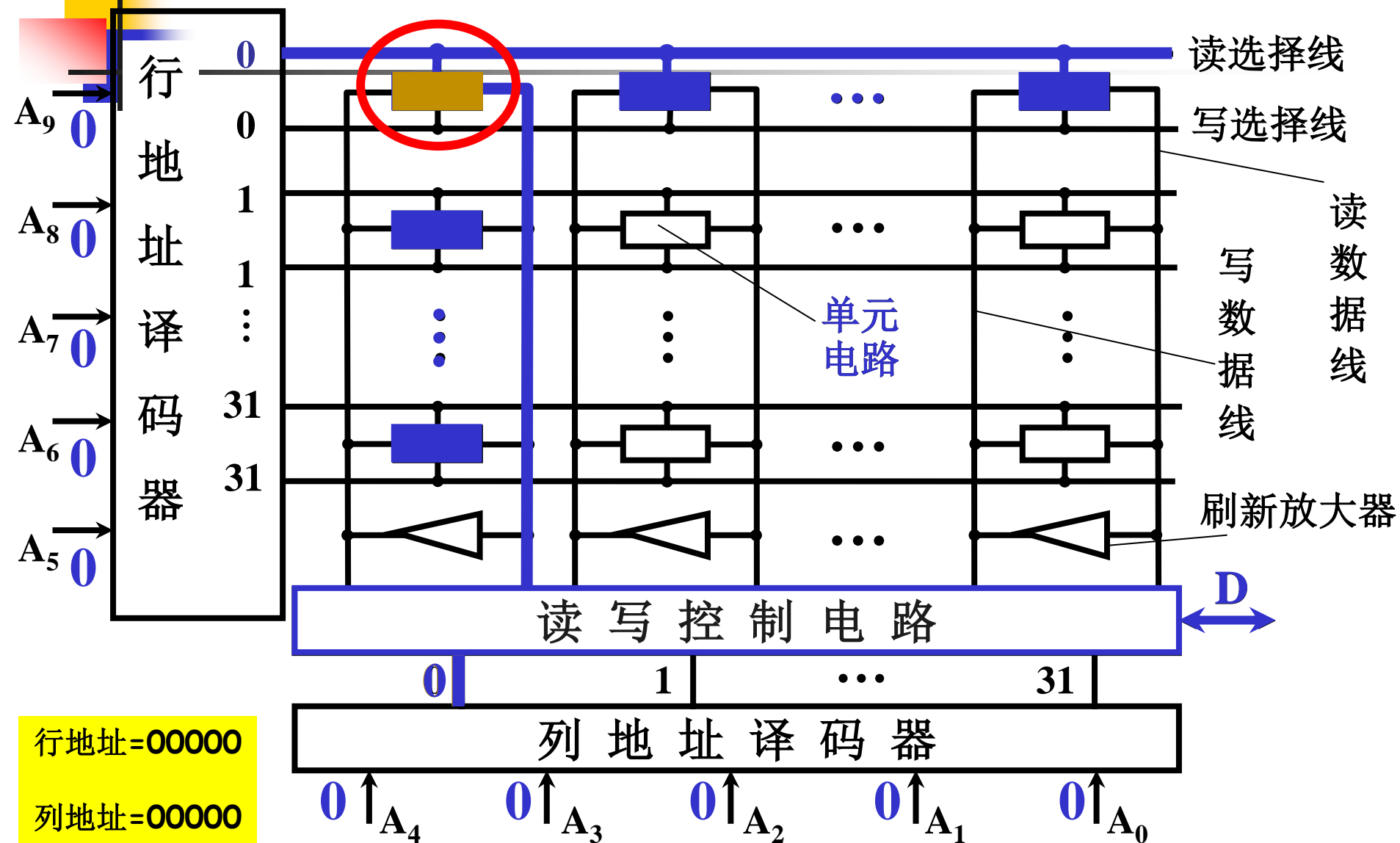
DRAM的再生或刷新

- 由于**DRAM**采用**电容**存储电荷的原理来寄存信息，即电容上存有足够多的电荷表示存“**1**”，电容上无电荷表示存“**0**”，而电容上的电荷一般只能维持**1-2ms**，因此即使电源不掉电，信息也会自动消失。
- 必须在**2ms**内对其所有存储单元恢复一次原状态，这个过程称为**再生或刷新**。

(2) 动态 RAM 芯片举例

① 三管动态 RAM 芯片 (Intel 1103) 读

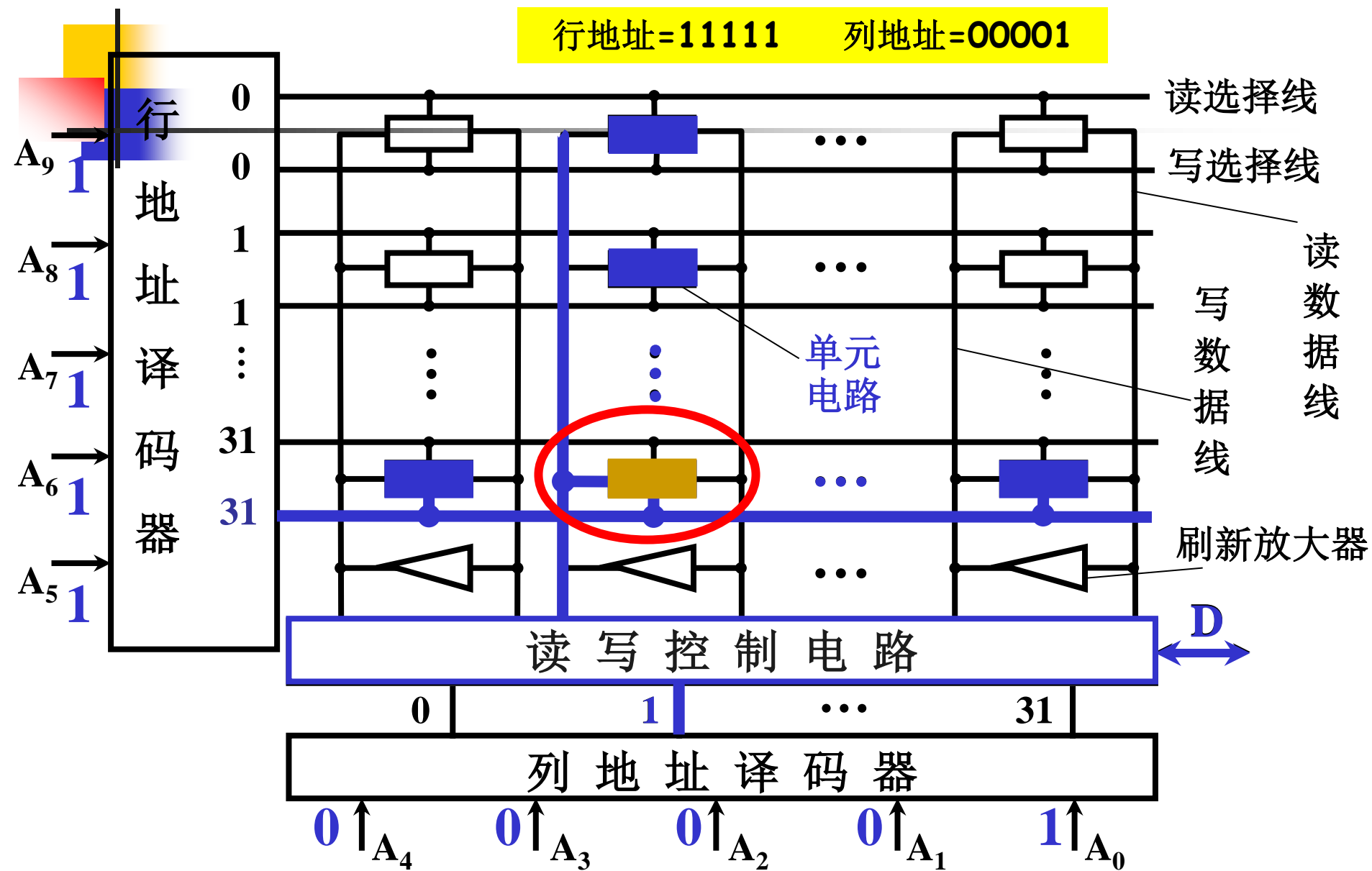
1KX1位 = 32X32



Intel 1103

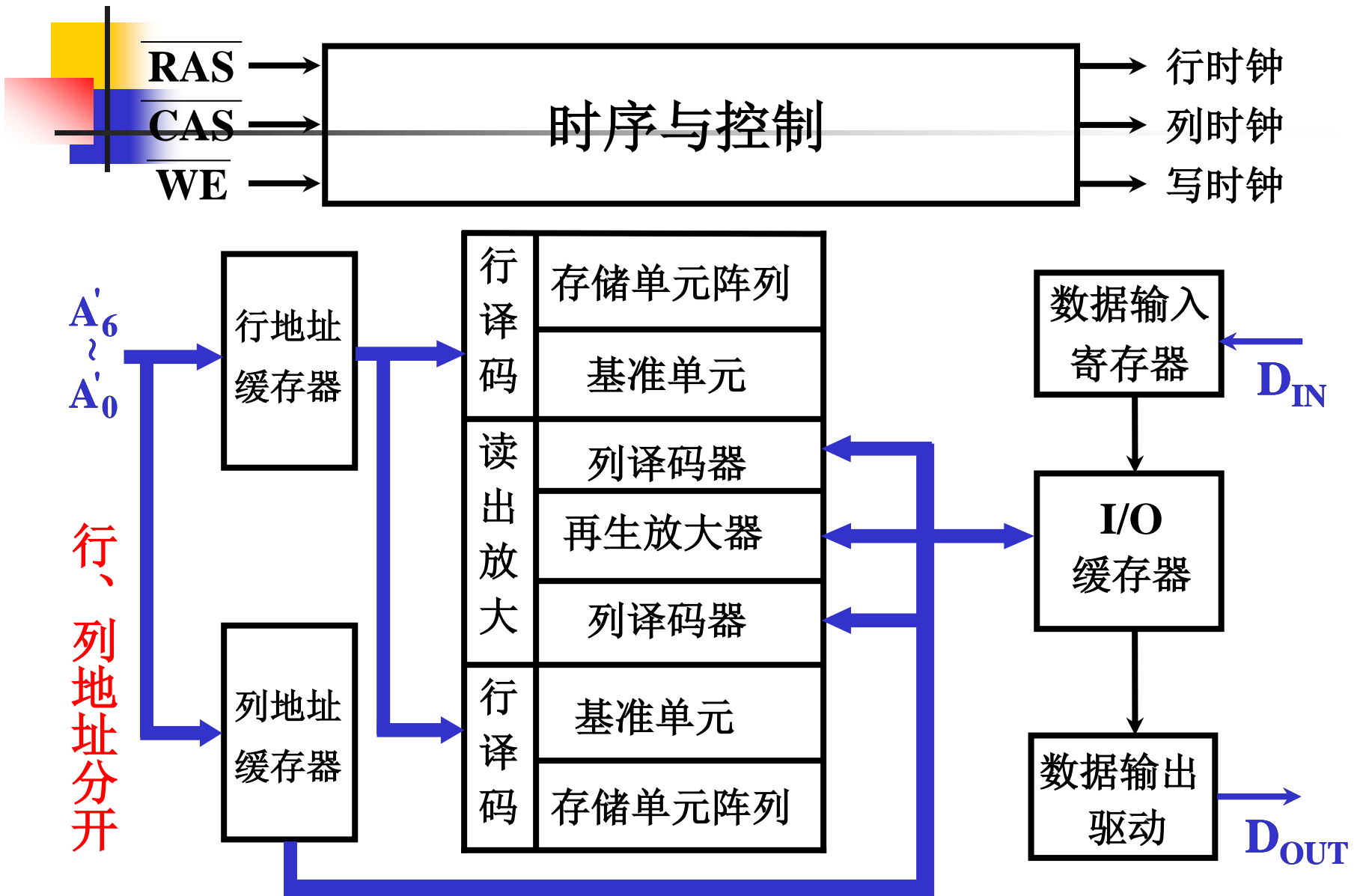


② 三管动态 RAM 芯片 (Intel 1103) 写

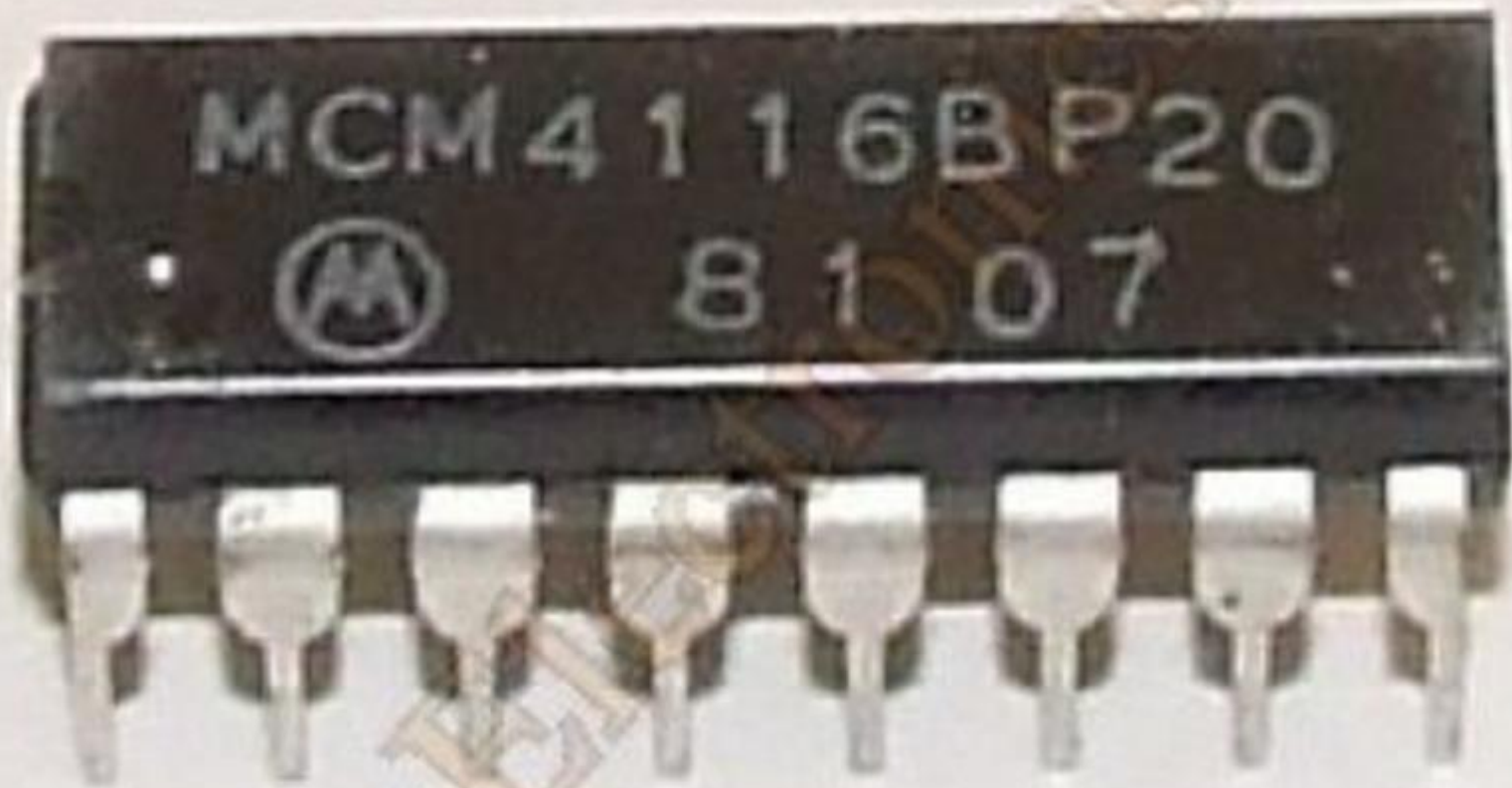


③ 单管动态 RAM 4116 (16K × 1位) 外特性

$2^{14} \times 1$ 位

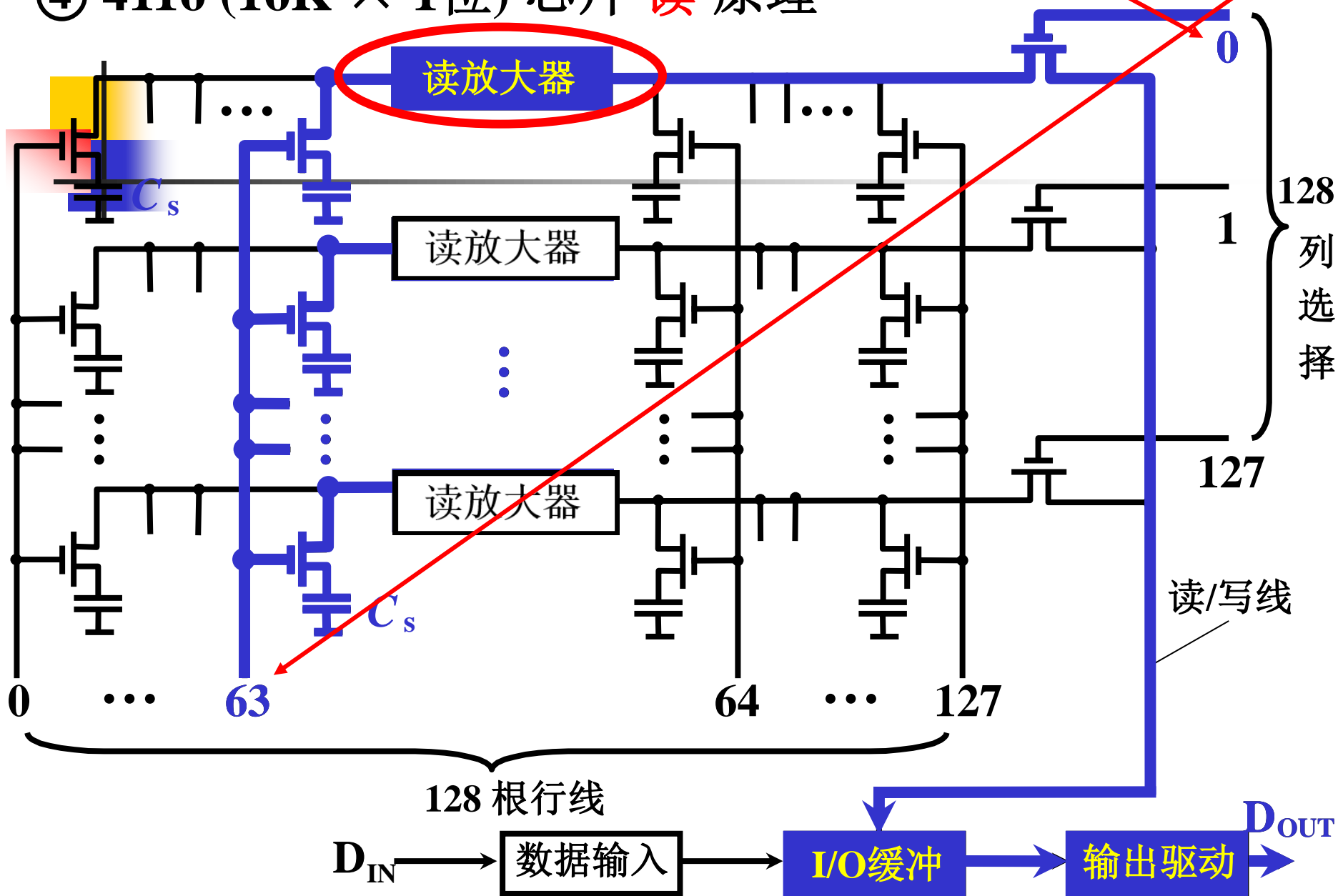


RAM 4116



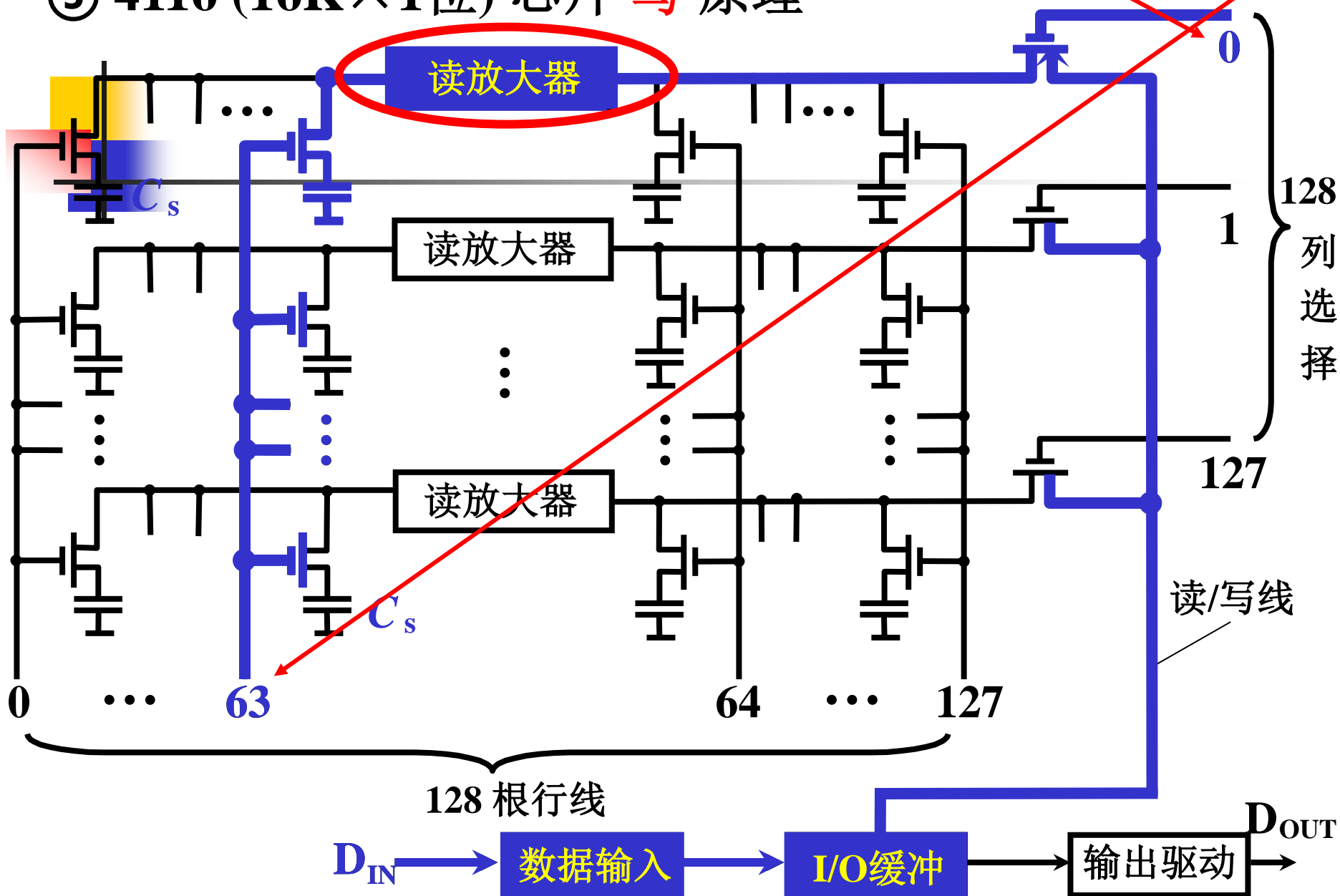
单管动态 RAM

④ 4116 (16K × 1位) 芯片 读 原理



单管动态 RAM

⑤ 4116 (16K×1位) 芯片 写 原理



(3) 动态 RAM 时序

行、列地址分开传送

读时序

行地址 $\overline{\text{RAS}}$ 有效

写允许 $\overline{\text{WE}}$ 有效(高)

列地址 $\overline{\text{CAS}}$ 有效

数据 D_{OUT} 有效

写时序

行地址 $\overline{\text{RAS}}$ 有效

写允许 $\overline{\text{WE}}$ 有效(低)

数据 D_{IN} 有效

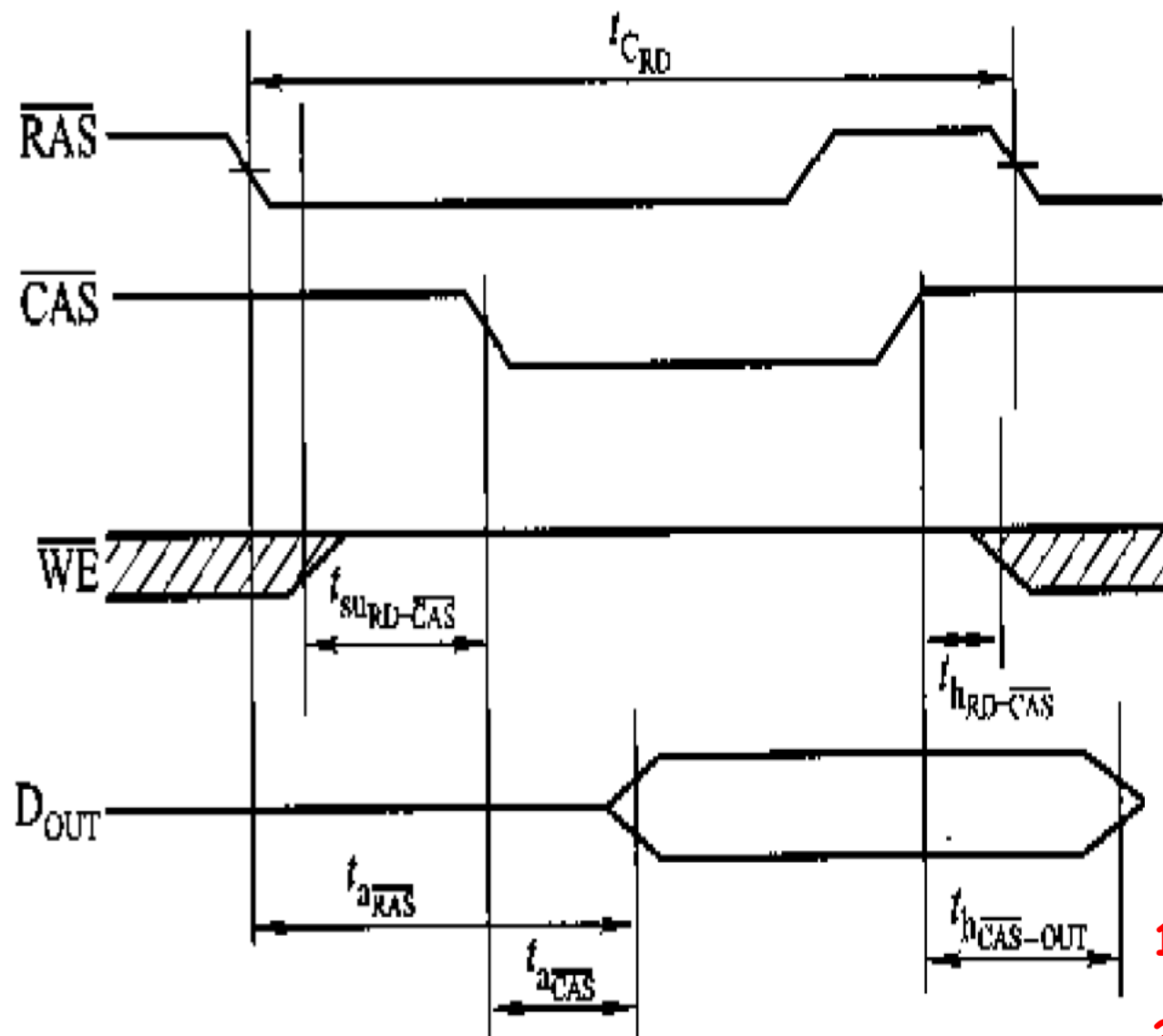
列地址 $\overline{\text{CAS}}$ 有效

RAS(Row Address Strobe, 行地址选通脉冲)

CAS (Column Address Strobe, 列地址选通脉冲)

动态 RAM

读时序



读时序

1. 行地址 \overline{RAS} 有效
2. 写允许 \overline{WE} 有效(高)
3. 列地址 \overline{CAS} 有效
4. 数据 D_{OUT} 有效

图 4.22 动态 RAM 读工作方式时序图

动态 RAM

读时序

t_{CRD} : 读工作周期 存取周期(MCT)

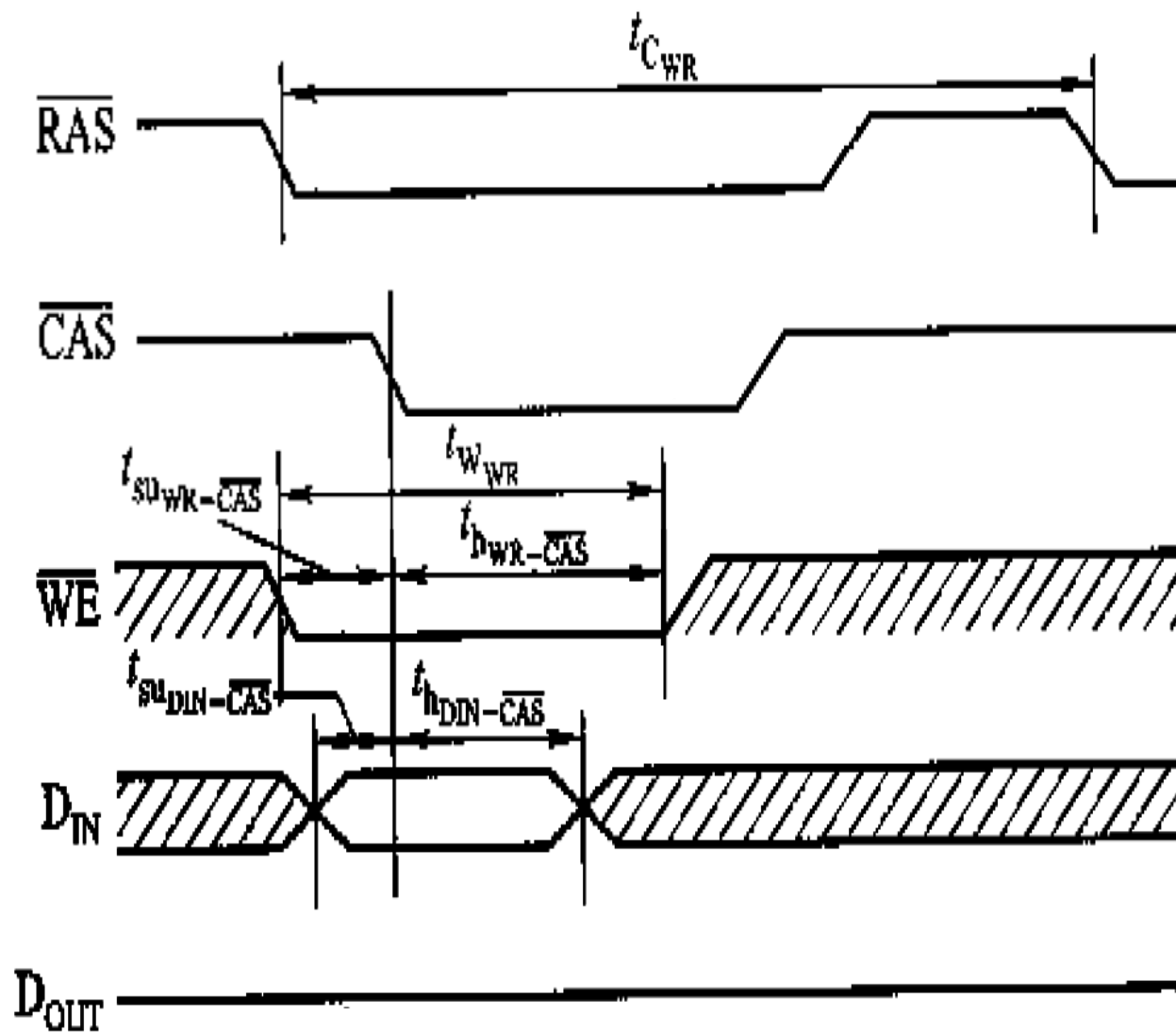
$t_{\text{suRD-CAS}}$: 写允许/**WE**有效（高）到列地址/**CAS**有效的的时间

$t_{\text{hRD-CAS}}$: 列地址/**CAS**无效到写允许/**WE**无效的时间

t_{aRAS} : 行地址/**RAS**有效到数据 D_{OUT} 有效的的时间

t_{aCAS} : 列地址有效到数据 D_{OUT} 有效的的时间

$t_{\text{hCAS-OUT}}$: 列地址/**CAS**无效到数据 D_{OUT} 无效的时间



动态 RAM

写时序

写时序

1. 行地址 \overline{RAS} 有效
2. 写允许 \overline{WE} 有效(低)
3. 数据 D_{IN} 有效
4. 列地址 \overline{CAS} 有效

图 4.23 动态 RAM 写工作方式时序图

动态 RAM

写时序

t_{CWR} : 写工作周期 存取周期(MCT)

t_{WWR} : 写允许/**WE**有效（低）的持续时间

$t_{\text{suWR-CAS}}$: 写允许/**WE**有效（低）到列地址/**CAS**有效的时间

$t_{\text{hWR-CAS}}$: 列地址/**RAS**有效到写允许/**WE**无效的时间

$t_{\text{suDIN-CAS}}$: 数据 D_{IN} 有效到列地址/**CAS**有效的时间

$t_{\text{hDIN-CAS}}$: 列地址/**CAS**有效到数据 D_{IN} 无效的时间



(4) 动态 RAM 刷新（再生）

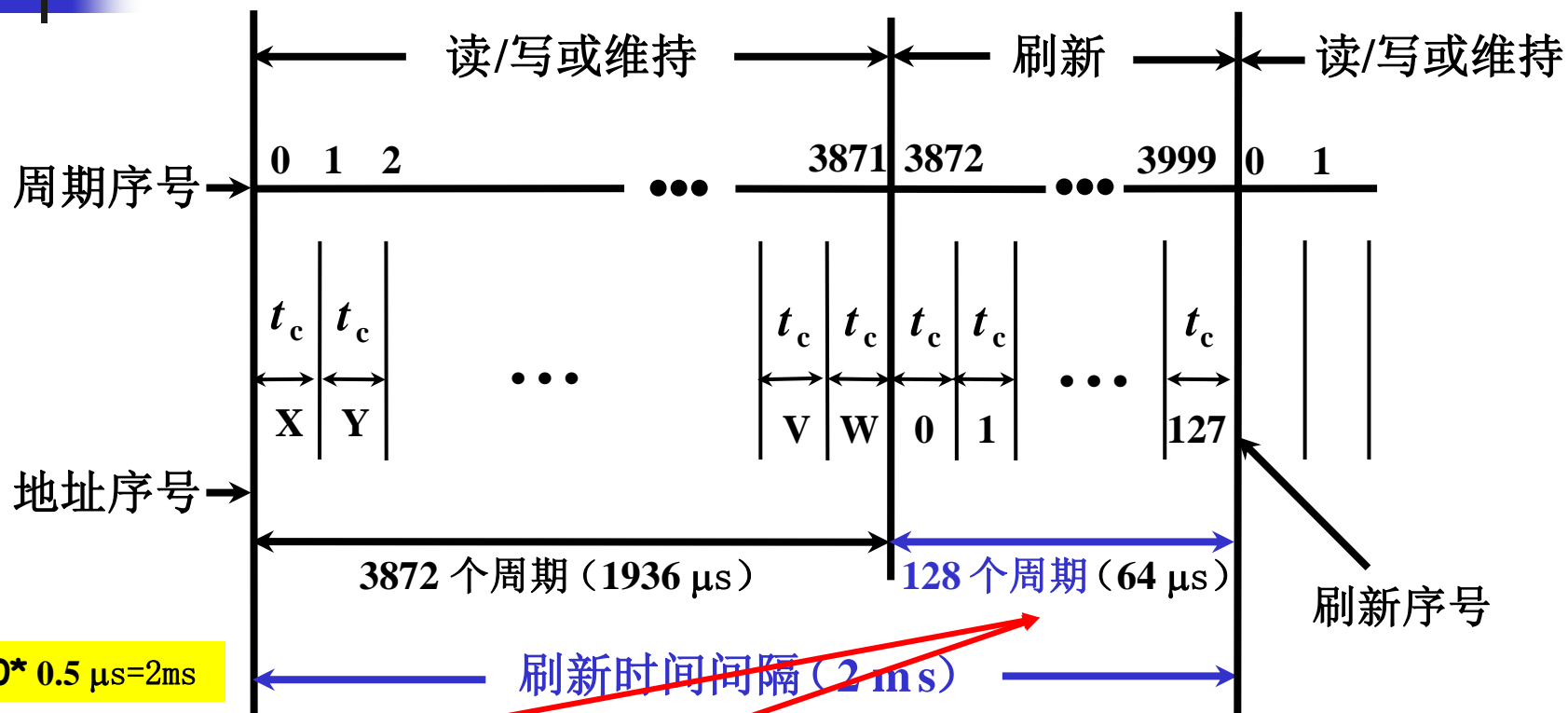
- 刷新的过程实质上是先将原存储信息**读出**，再由刷新放大器形成原信息**并重新写入**的**再生**过程
- 图4.19中的刷新放大器、图4.20的读出-再生放大器、图4.21的读放大器均起刷新作用
- 定时刷新：在**2ms**（刷新周期、再生周期）内对**DRAM**的全部基本单元电路进行一次刷新
- 刷新是一行行进行的（**逐行刷新**）
- **集中刷新、分散刷新、异步刷新**

(4) 动态 RAM 刷新 (续)

刷新与行地址有关

逐行进行刷新

① 集中刷新 (存取周期为 $0.5 \mu\text{s}$) 以 128×128 矩阵为例



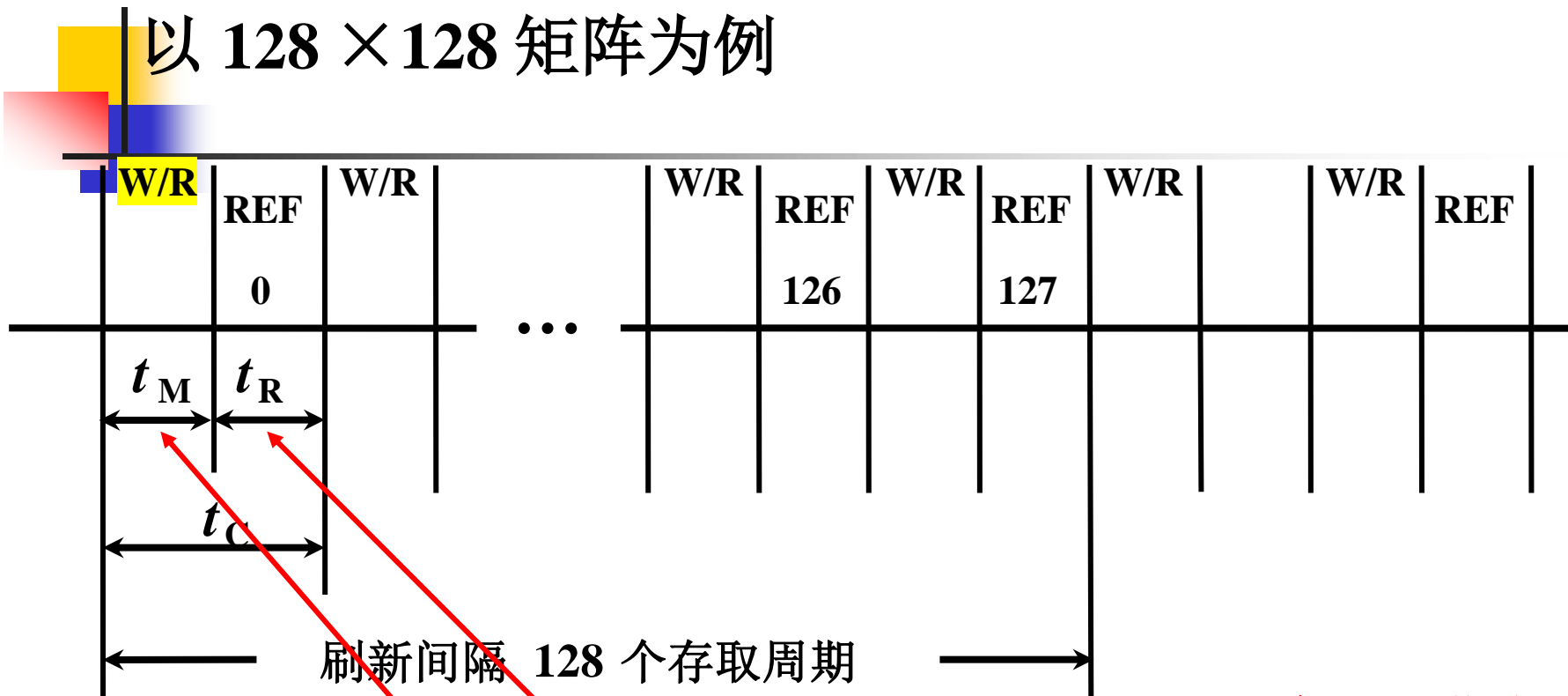
$$4000 \times 0.5 \mu\text{s} = 2 \text{ ms}$$

“死区” 为 $0.5 \mu\text{s} \times 128 = 64 \mu\text{s}$

“死时间率” 为 $128/4000 \times 100\% = 3.2\%$

② 分散刷新 (存取周期为 $1\ \mu\text{s}$)

以 128×128 矩阵为例



$$t_C = t_M + t_R$$

↓ ↓
读写 刷新

无“死区”

但是存取周期长了，使整个系统速度降低了

(存取周期为 $0.5\ \mu\text{s} + 0.5\ \mu\text{s}$)

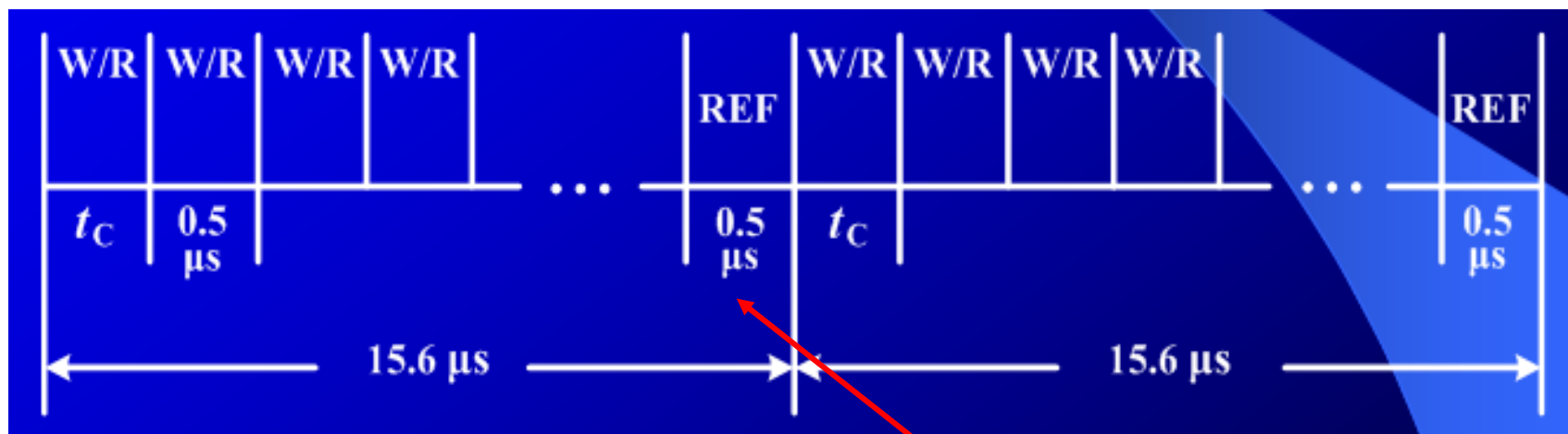
集中刷新：存取周期 $t_C = 0.5\ \mu\text{s}$

③ 异步刷新（分散刷新与集中刷新相结合）

对于 128×128 的存储芯片（存取周期为 $0.5 \mu\text{s}$ ）

每隔 $15.6 \mu\text{s}$ ($=2\text{ms}/128$) 刷新一行

每行每隔 2 ms 刷新一次



“死区” 为 $0.5 \mu\text{s}$

如将刷新安排在指令译码阶段，不会出现“死区”

三种刷新方式的比较

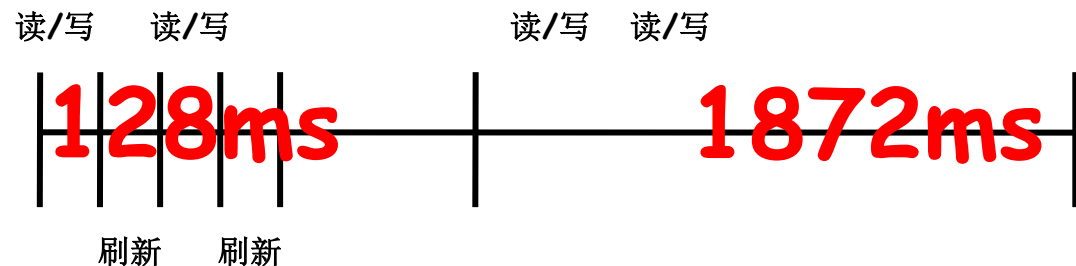
集中刷新

有死区 死区率=3.2%



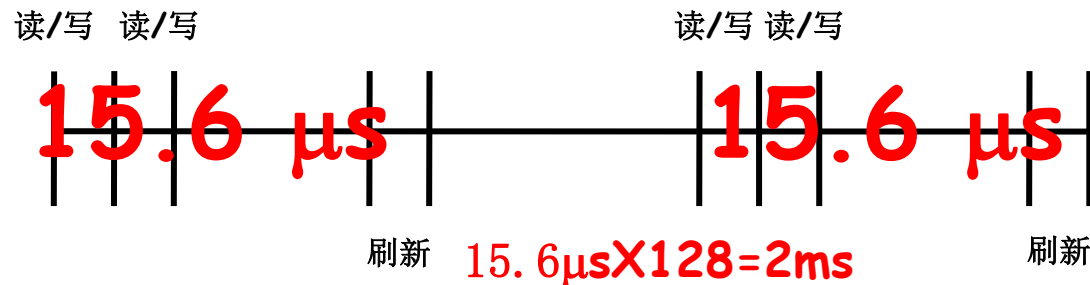
分散刷新

无死区，但是刷新期间，存储周期增加了（存储周期=1 μs ）



异步刷新

既没有增加存储周期，又可以做到无死区（刷新放在指令译码阶段）



3. 动态 RAM 和静态 RAM 的比较



	主存  DRAM	SRAM  缓存
存储原理	电容	触发器
集成度	高	低
芯片引脚	少	多
功耗	小	大
价格	低	高
速度	慢	快
刷新	有	无



DRAM与SRAM的比较

- **DRAM(单管/三管)集成度远高于SRAM(6管)**
- **DRAM的行列地址是按先后顺序输送(地址线少), SRAM的地址是一次输送(地址线多)**
- **DRAM功耗比SRAM小**
- **DRAM价格比SRAM便宜(1/8 - 1/16的价格)**
- **DRAM容量比SRAM大(4-8倍)**
- **DRAM的存取周期比SRAM慢(慢8-16倍), DRAM速度比SRAM慢**
- **DRAM需要再生(刷新), 需配置再生(刷新)电路, 电路复杂**



第4次作业——习题(P150-153)

- 4.1
- 4.3
- 4.5
- 4.6
- 4.8
- 4.9
- 4.11



关于作业的提交

- **1周内**必须提交（上传到学院的**FTP**服务器上），否则认为是迟交作业；如果期末仍然没有提交，则认为是未提交作业
 - 作业完成情况成绩=第**1**次作业提交情况*第**1**次作业评分+第**2**次作业提交情况*第**2**次作业评分+.....+第**N**次作业提交情况*第**N**次作业评分
 - 作业评分：**A**（好）、**B**（中）、**C**（差）三挡
 - 作业提交情况：按时提交（**1.0**）、迟交（**0.5**）、未提交（**0.0**）
- 请采用电子版的格式（**Word**文档）上传到**FTP**服务器上，文件名取“学号+姓名+第**X**次作业.doc”
 - 例如：**11920192203642+袁佳哲+第1次作业.doc**
- 第**4**次作业提交的截止日期为：**2021年3月17日晚上24点**



The End

Thanks