

第4章 存储逻辑

[4.1 特殊存储部件](#)

[4.2 随机读写存储器RAM](#)

[4.3 只读存储器ROM](#)

[4.4 FLASH存储器](#)

[4.5 存储器容量的扩充](#)



[返回目录](#)

4.1 特殊存储部件

- [4.1.1 寄存器堆](#)
- [4.1.2 寄存器队列](#)
- [4.1.3 寄存器堆栈](#)

导入

存储逻辑是时序逻辑和组合逻辑相结合的产物。

能够存储 $m \times n$ 个二进制比特数的逻辑电路，我们叫做**存储器**。

特殊存储部件：寄存器堆、寄存器队列、寄存器堆栈，它们是由[寄存器](#)组成。

特点：

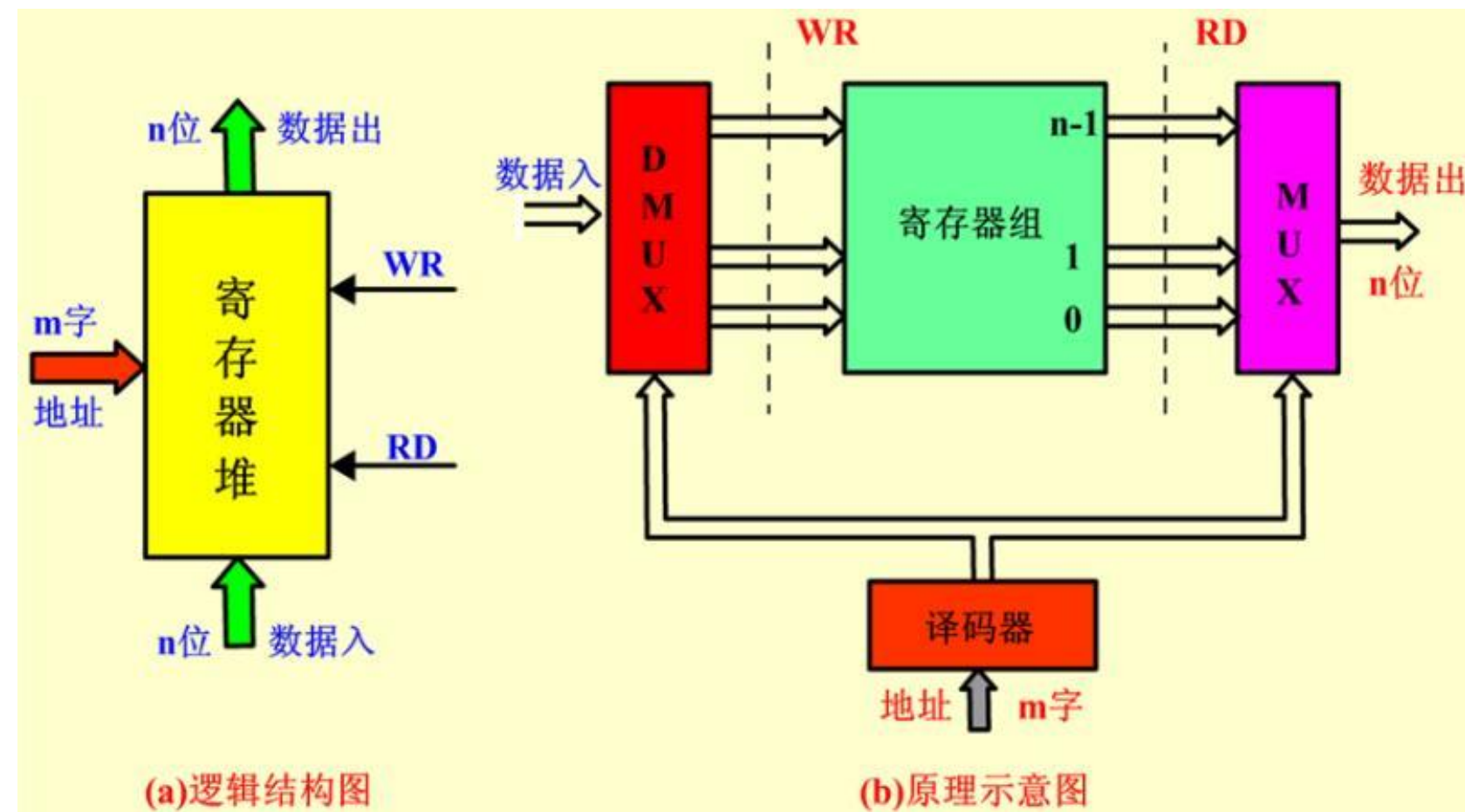
存储容量小，逻辑结构简单，工作速度快。

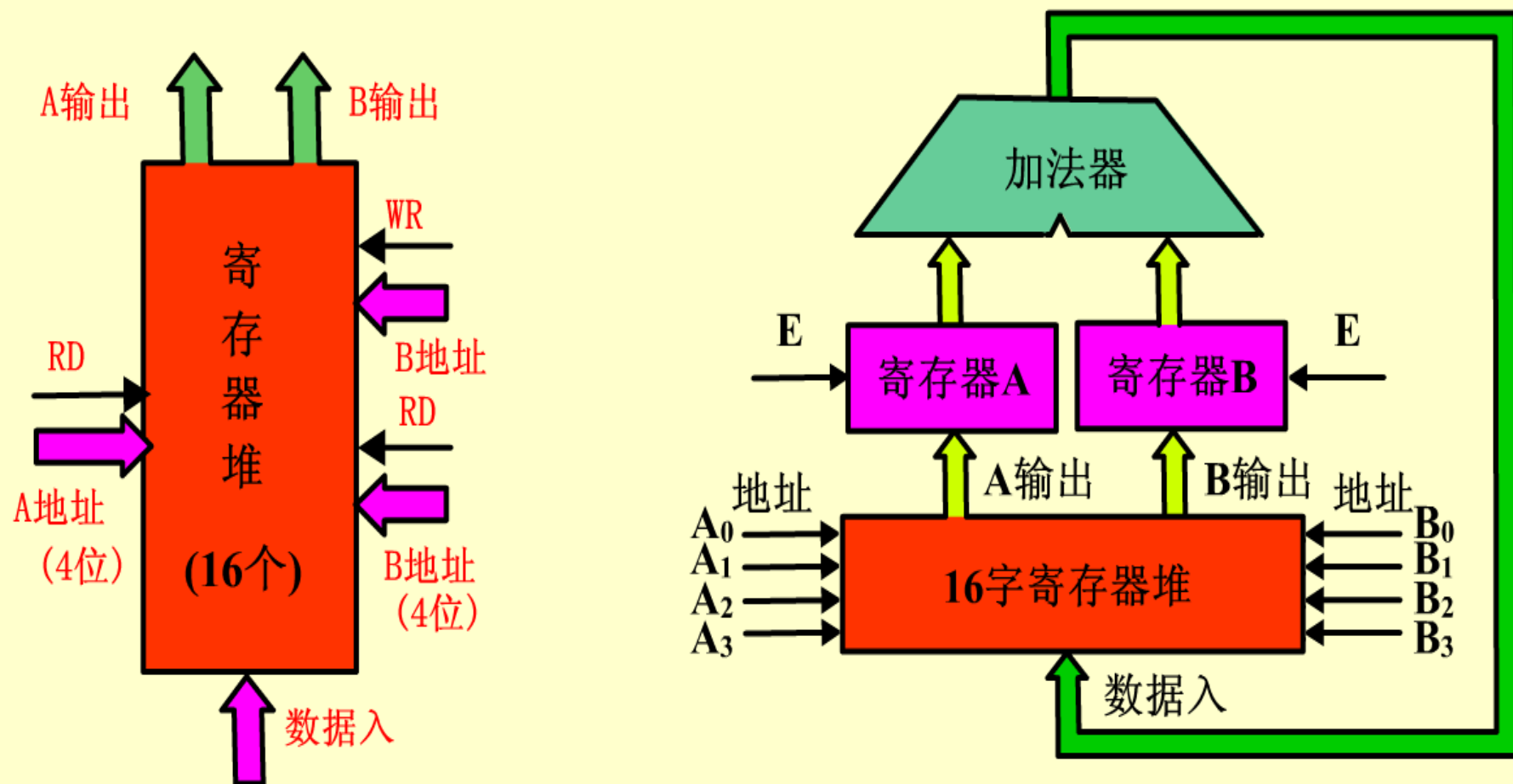
- **寄存器：**寄存器就是一种常用的时序逻辑电路，但这种时序逻辑电路只包含存储电路。暂存指令、数据和位址。
 - 1) 内部用来CPU存放数据的一些小型存储区域，用来暂时存放参与运算的数据和运算结果。
 - 2) 外部外部寄存器是计算机中其它一些部件上用于暂存数据的寄存器，它与CPU之间通过“端口”交换数据，外部寄存器具有寄存器和内存储器双重特点。有些时候我们常把外部寄存器就称为“端口”
- **内存：**
- **外存（辅助）：**

4.1.1 寄存器堆

一个寄存器是由 n 个触发器或锁存器按并行方式输入且并行方式输出构成。它以字（ n 位）为单位存储。

当要存储更多的字时，需要使用集中的寄存器组逻辑结构：
寄存器堆。它实际上是一个容量极小的存储器。





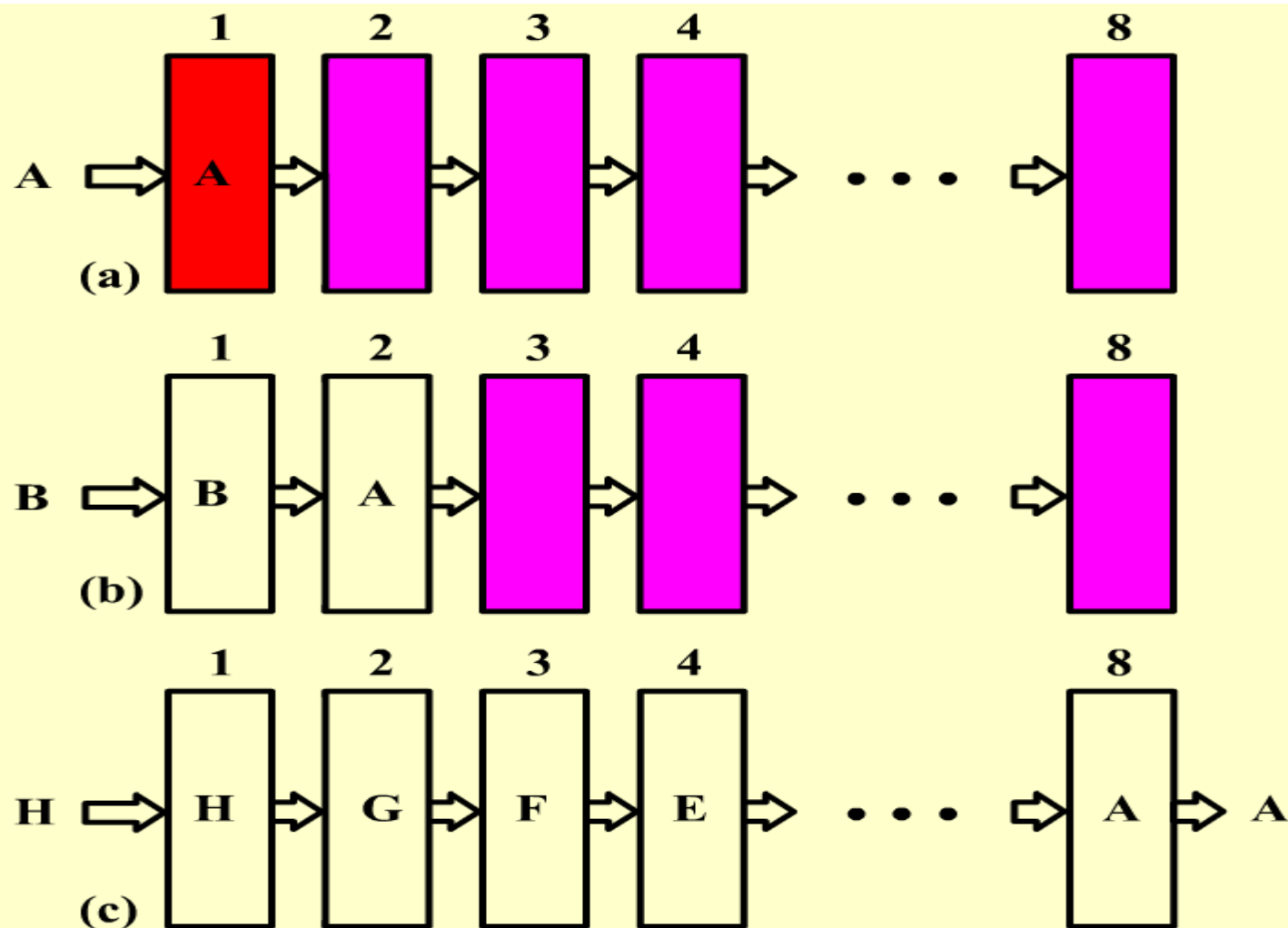
(a) 双端口输出寄存器堆逻辑结构

(b) 简单运算器通路

图4.2 双端口寄存器堆及其应用

4.1.2 寄存器队列

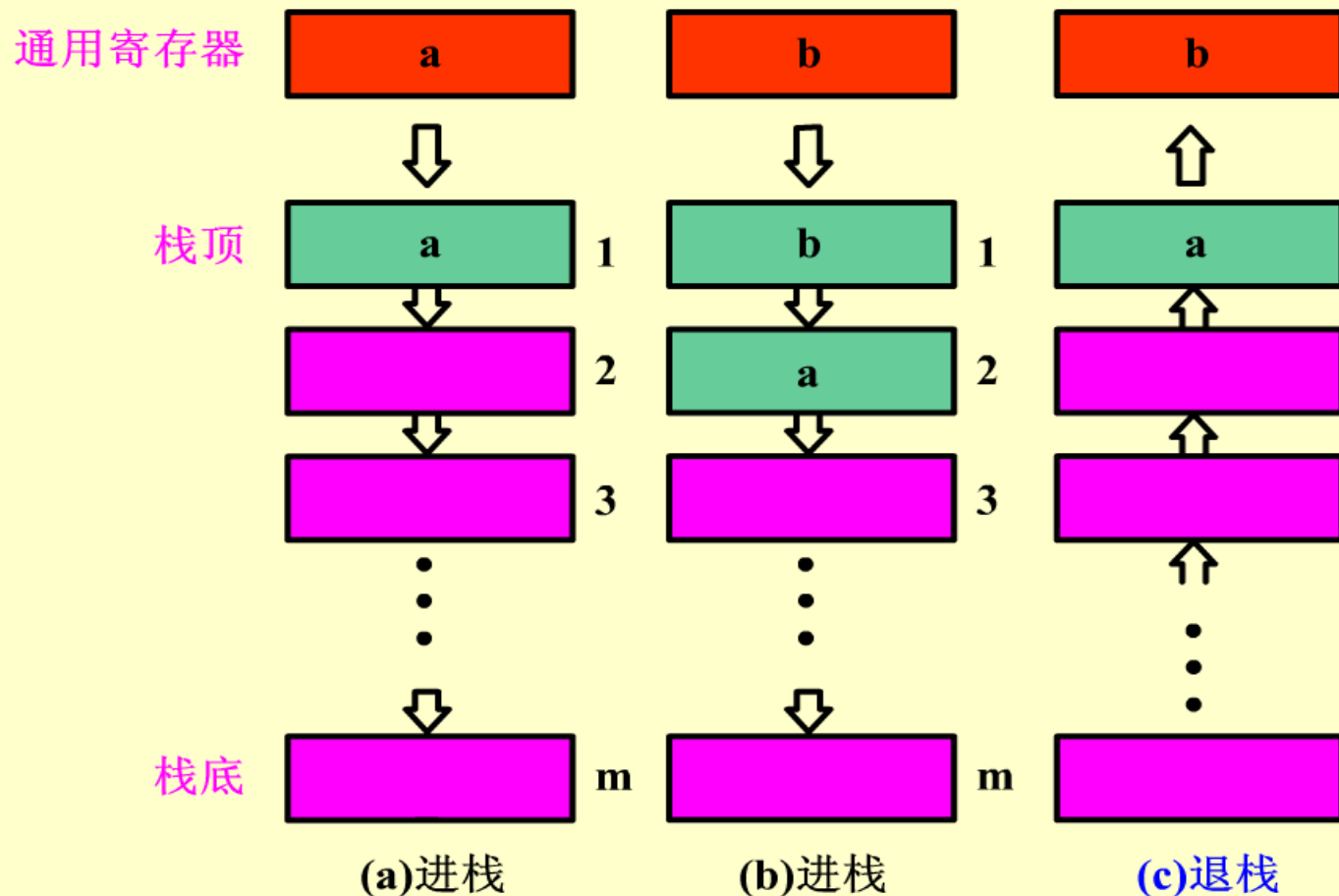
寄存器队列是以FIFO（先进先出）方式用若干个寄存器构建的小型存储部件。



4.3 寄存器队列逻辑结构示意图

4.1.3 寄存器堆栈

寄存器堆栈是以LIFO（后进先出）方式用若干个寄存器构建的小型存储部件。功能上，寄存器队列正好相反。



4.4 寄存器堆栈逻辑结构示意图

4.2 随机读写存储器RAM

- [4.2.1 RAM的逻辑结构](#)
- [4.2.2 地址译码方法](#)
- [4.2.3 SRAM存储器](#)
- [4.2.4 DRAM存储器](#)

导入

寄存器堆只存放有限的几个数据，本节所述半导体随机读写存储器（简称RAM），可存放大量的数据。

从工艺上，RAM分为双极型和MOS型两类。

从机理上，RAM分为[SRAM存储器](#)和[DRAM存储器](#)两类。

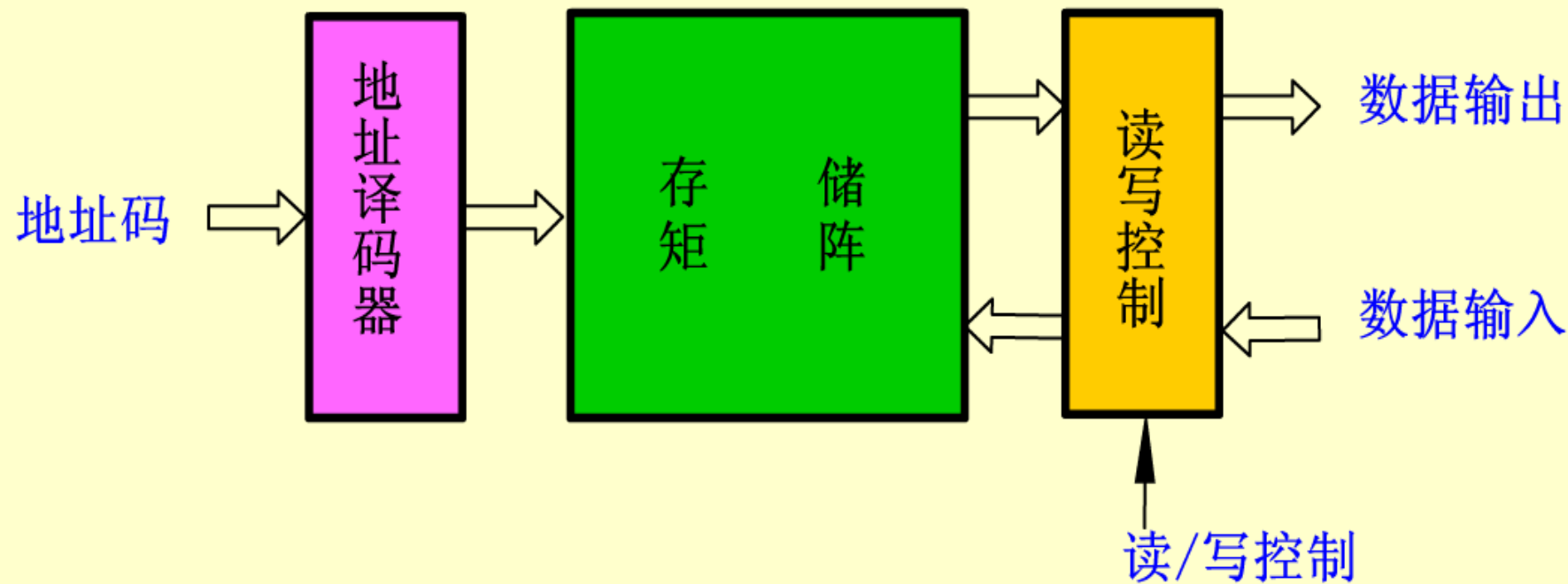
RAM属于[易失性存储器](#)（断电后信息会丢失）。

4.2.1 RAM的逻辑结构

RAM的构成：

主体是存储矩阵，另有地址译码器和读写控制电路两大部分。

存储矩阵：若干排成阵列形式的存储元（每个存储元能存储一个二进制数比特 bit）。

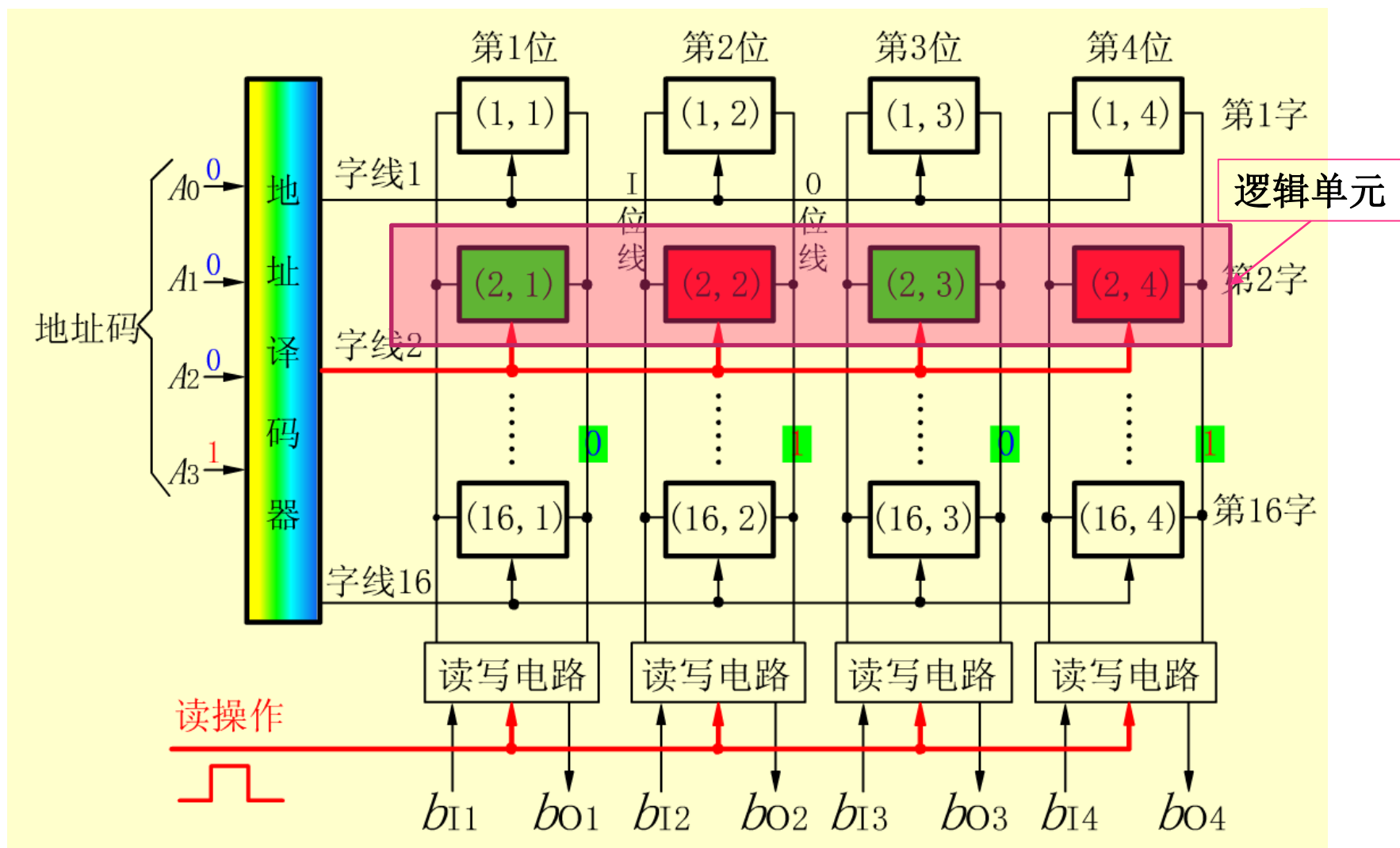


4.5 RAM的逻辑结构图

4.2.2 地址译码方法

存储器按存储矩阵组织方式不同，可分为：单译码结构和双译码结构。

1. 单译码结构

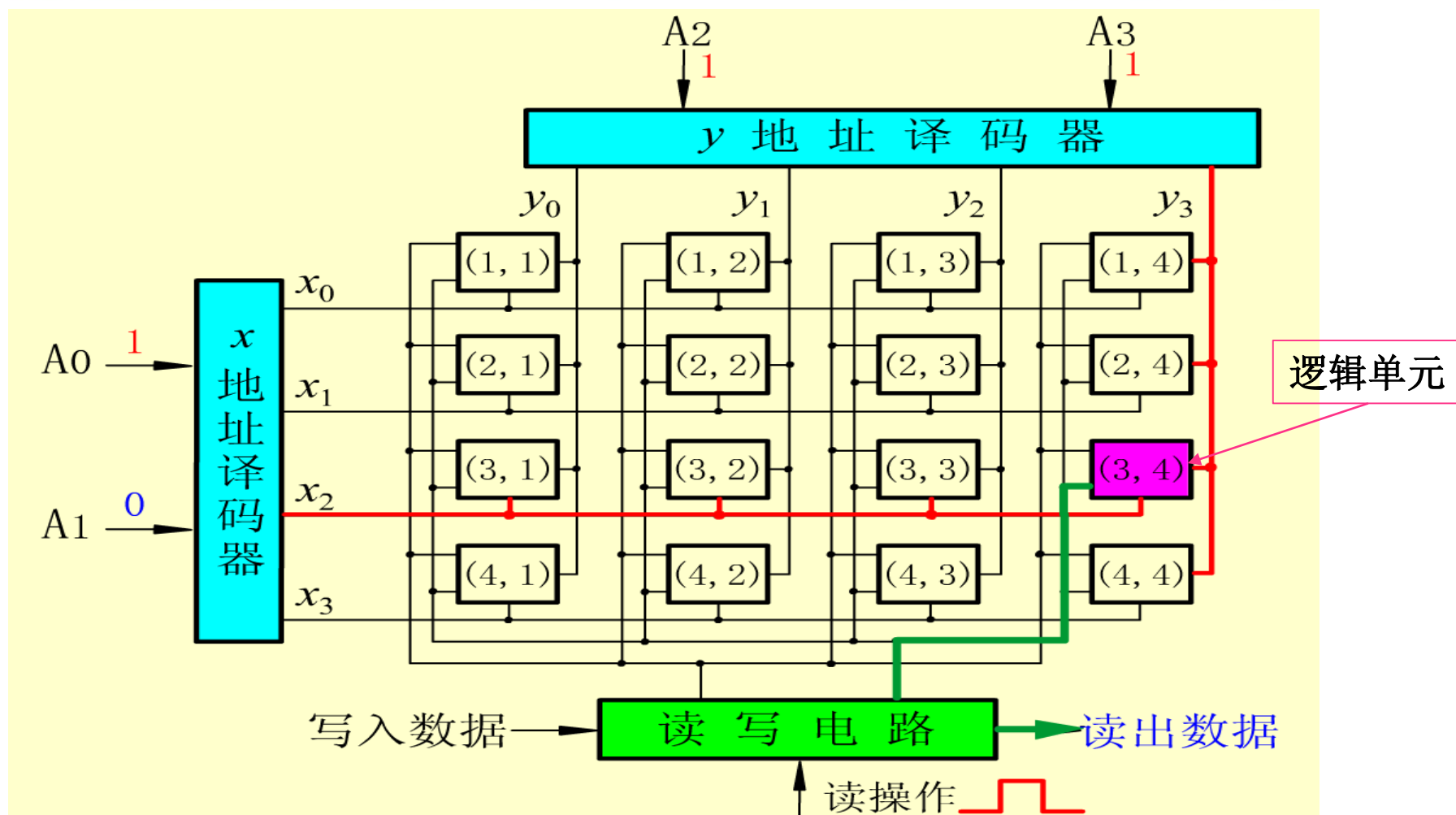


2. 双译码结构

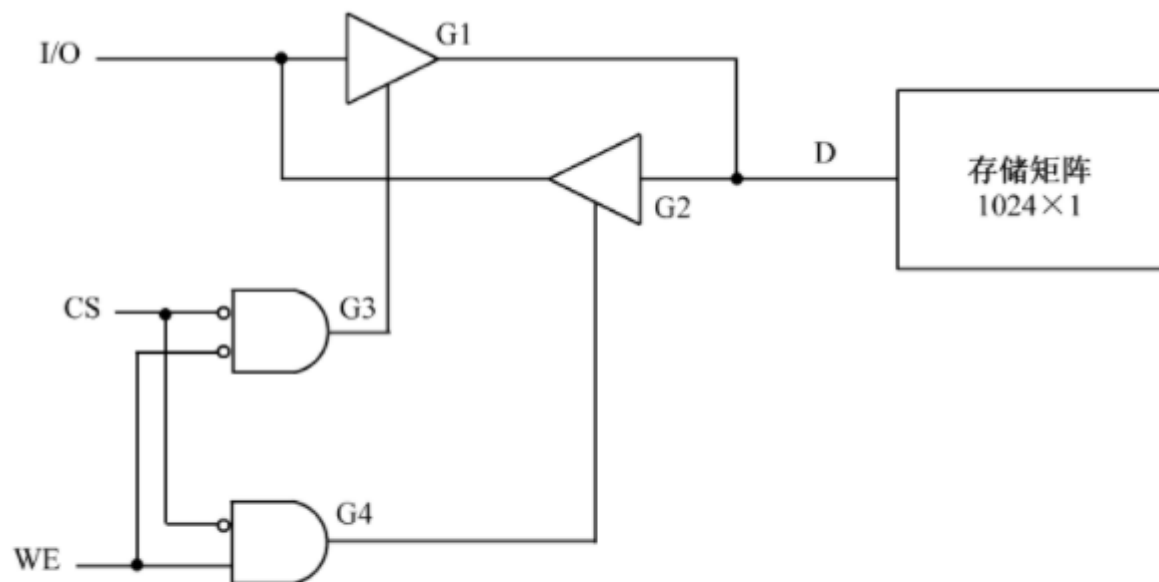
每个存储元有两条选择线

被选中的存储元：行选线X和列选线Y有效时的交叉点存储元

双译码结构RAM：需要有X（行地址）和Y（列地址）两个地址译码器



读出和写入的数据线是公用，为控制电路中数据的流向，设立了专门的读写控制电路。



读写控制电路

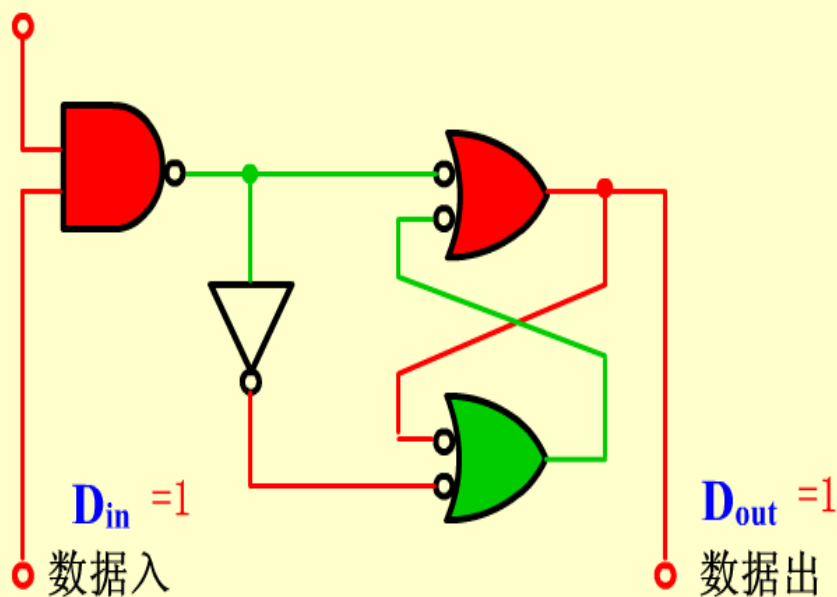
4.2.3 SRAM存储器

1. SRAM存储元

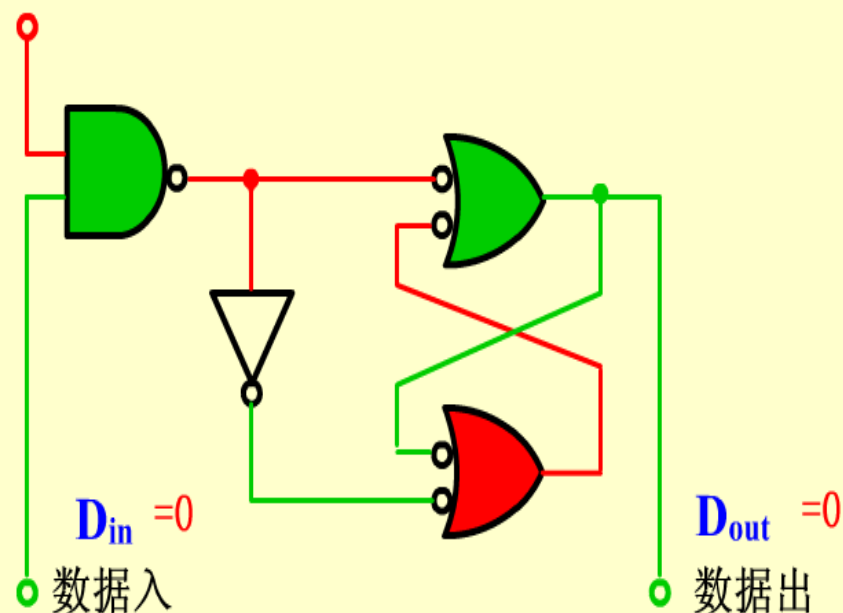
SRAM存储器：静态随机读写存储器

SRAM存储元，用一个锁存器构成。

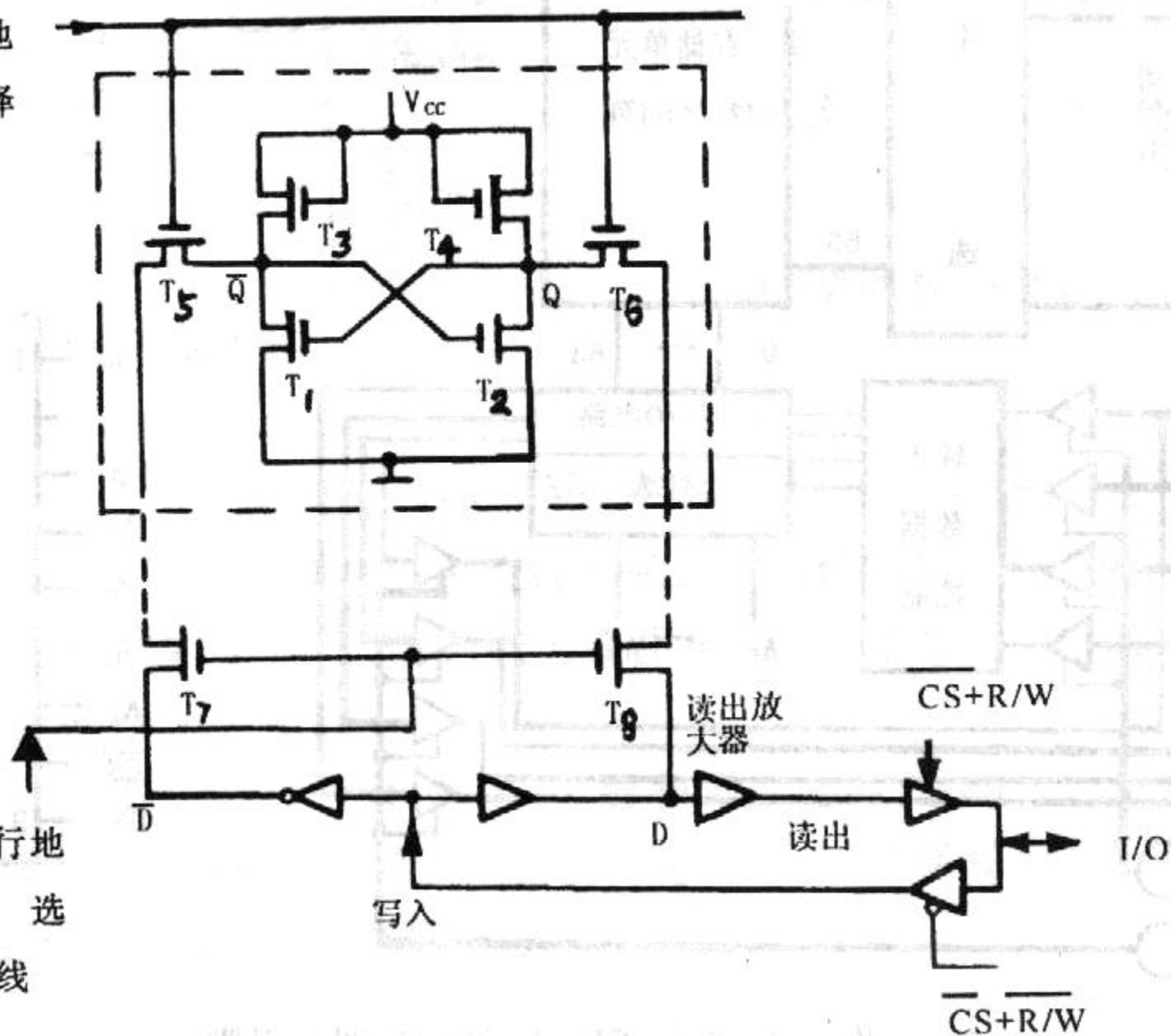
行选择线 高电平



行选择线 高电平



X 行地
址选择



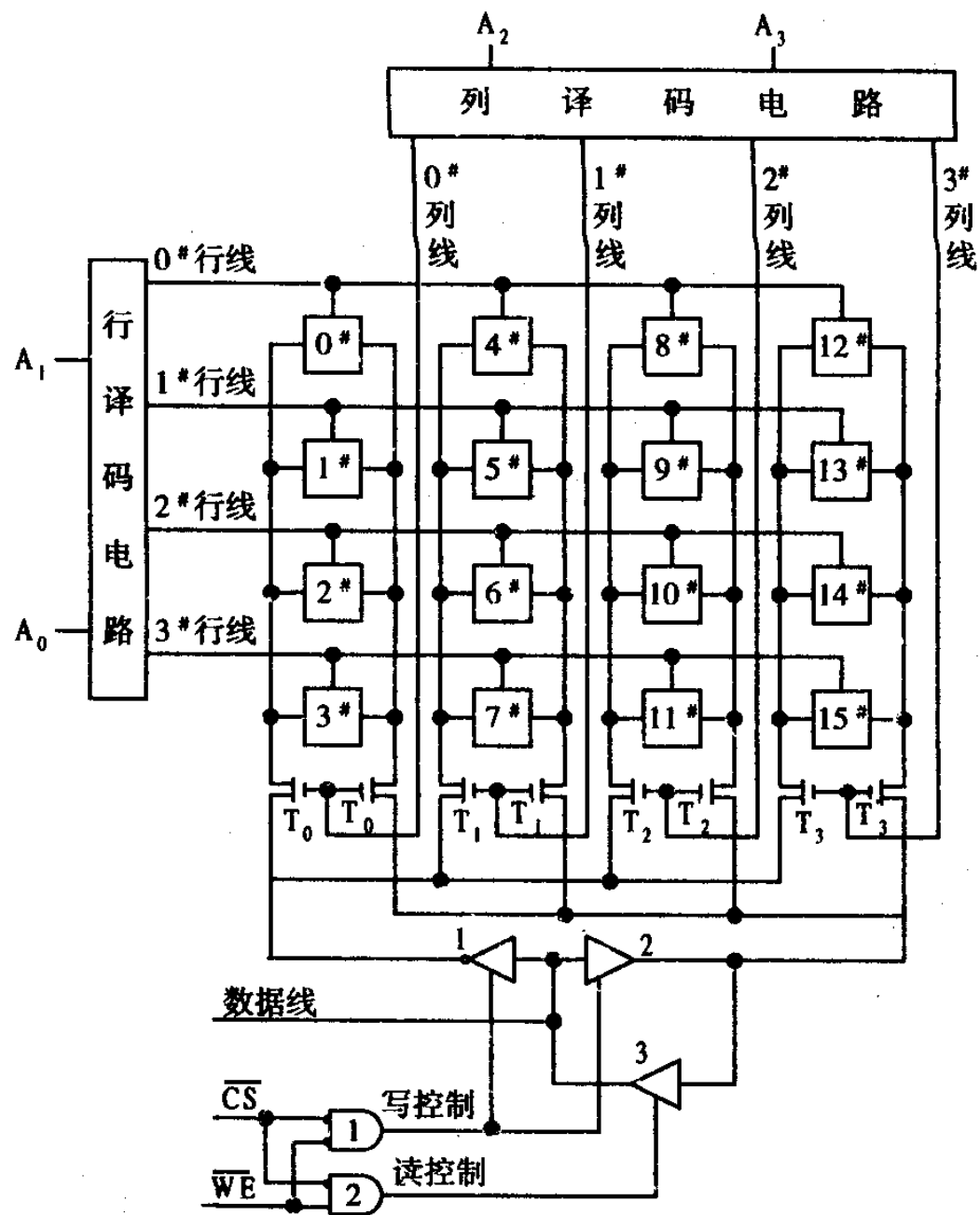
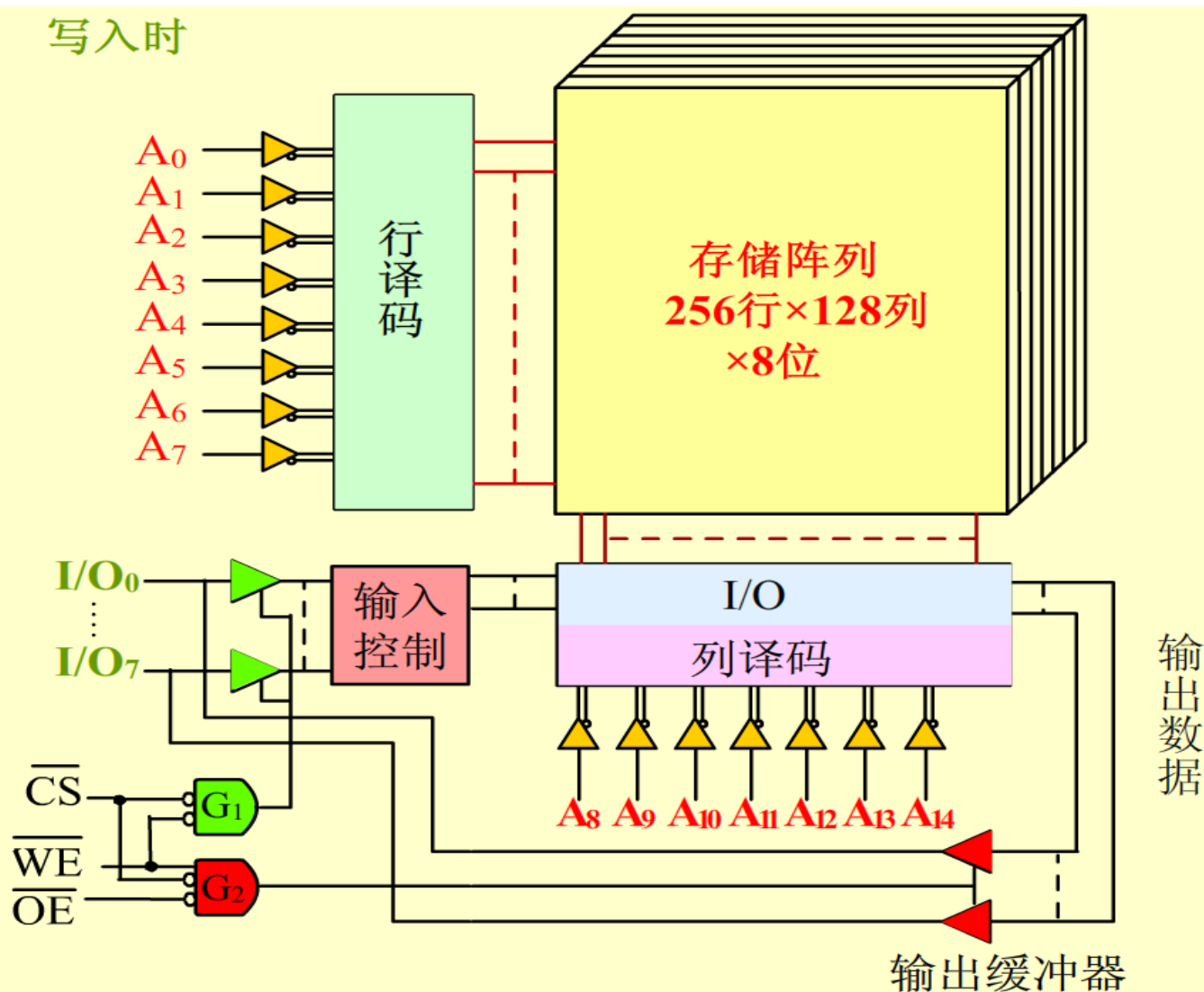


图 7.3 16×1 静态 RAM 原理图

2. SRAM存储器结构

芯片的位数：字长1位、4位、8位、16位、32位、64位等

写入时



32K×8位SRAM芯片逻辑图与内部结构图

4.2.4 DRAM存储器

1. DRAM存储元

DRAM存储器：动态随机读写存储器

DRAM存储器的存储元不使用锁存器，而是用 1 个小电容器。

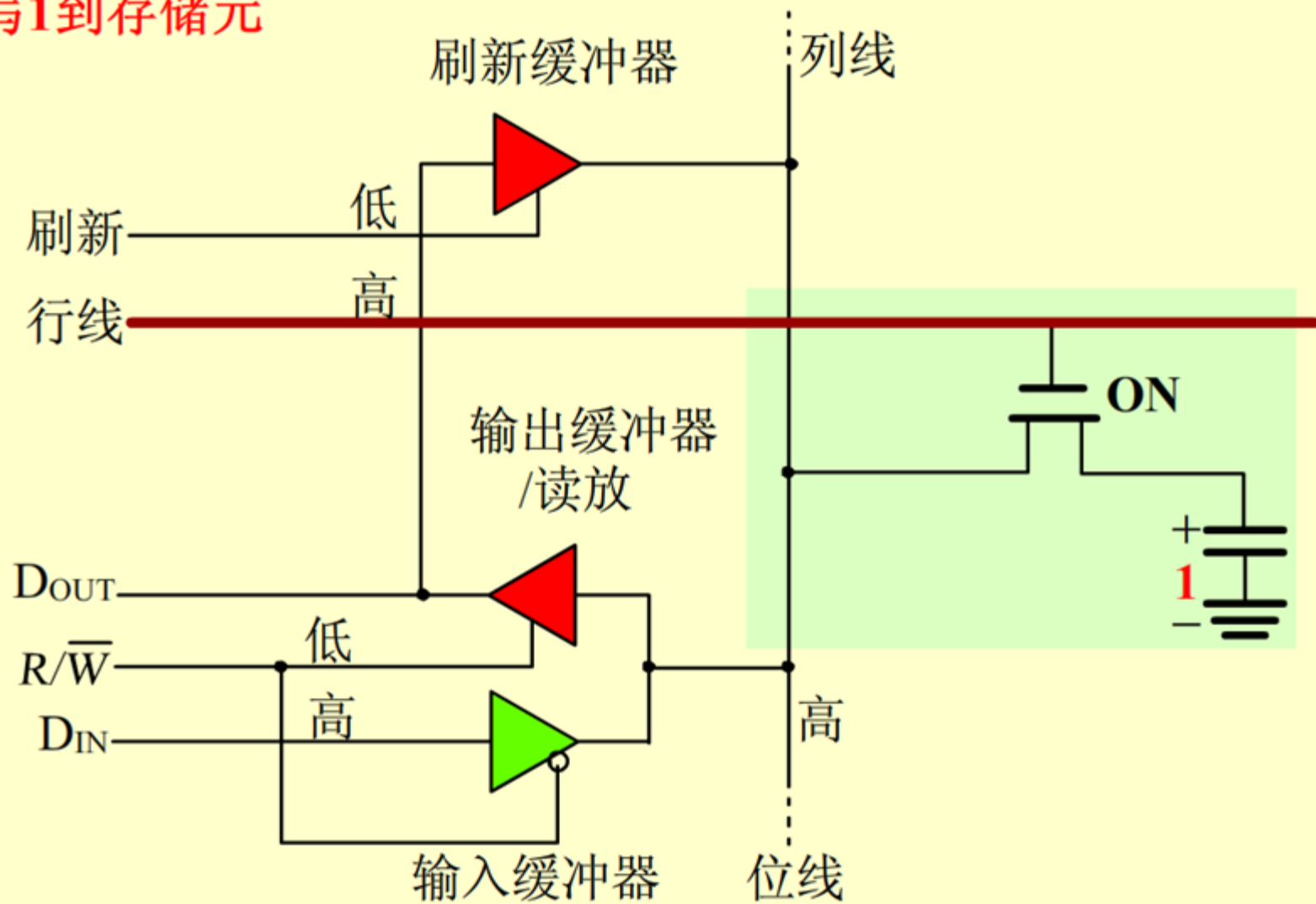
存储机理：靠电容器上的充放电荷记忆二进制信息。

优点：非常简单，集成度高，成本较低。

缺点：超过一定周期，电容电荷泄漏而可能丢失所存信息。

措施：必须及时补充电荷，这种过程叫做**刷新或再生**

写1到存储元

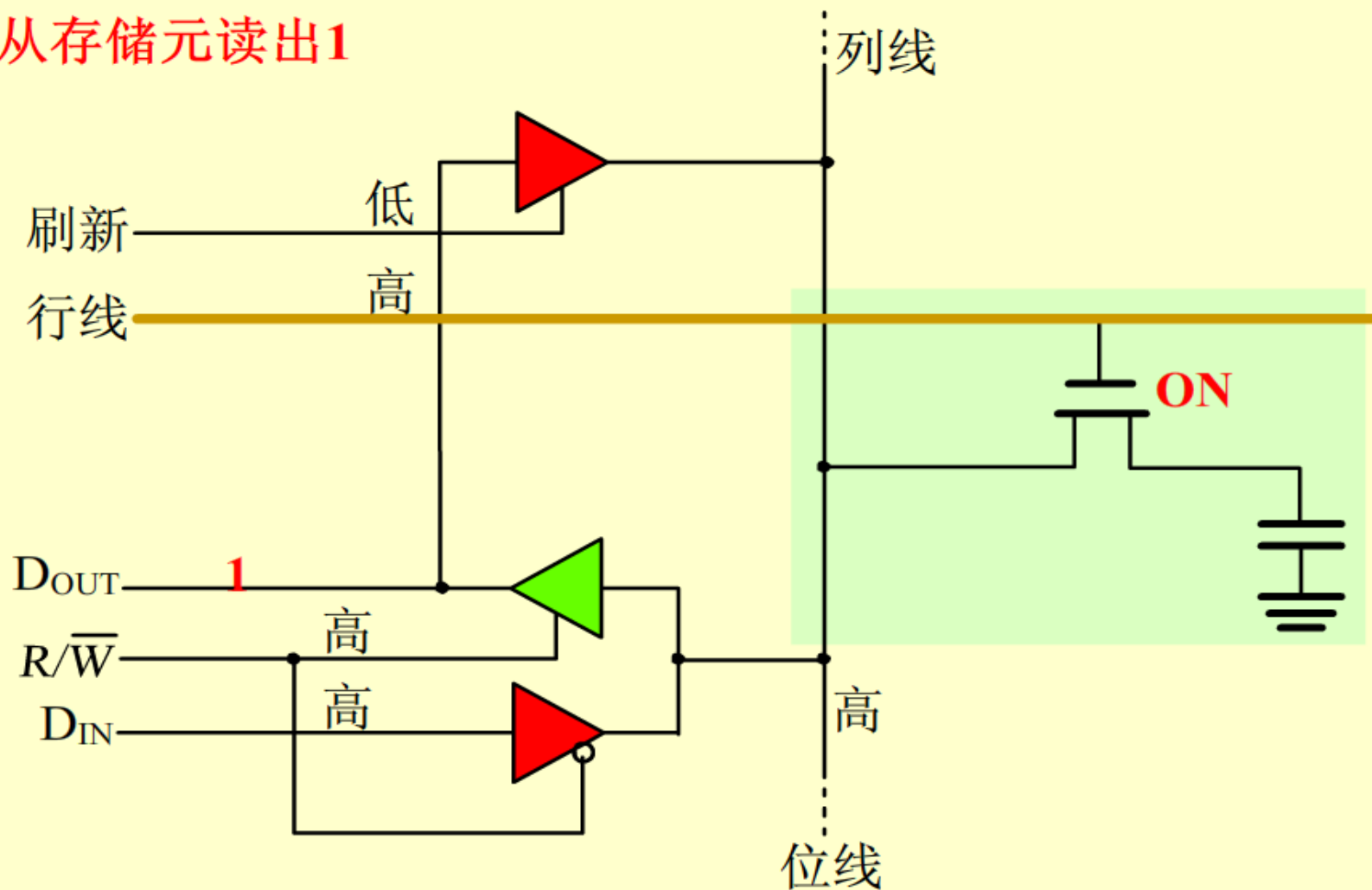


▲ 打开

▲ 关闭

DRAM存储元的基本操作

从存储元读出1

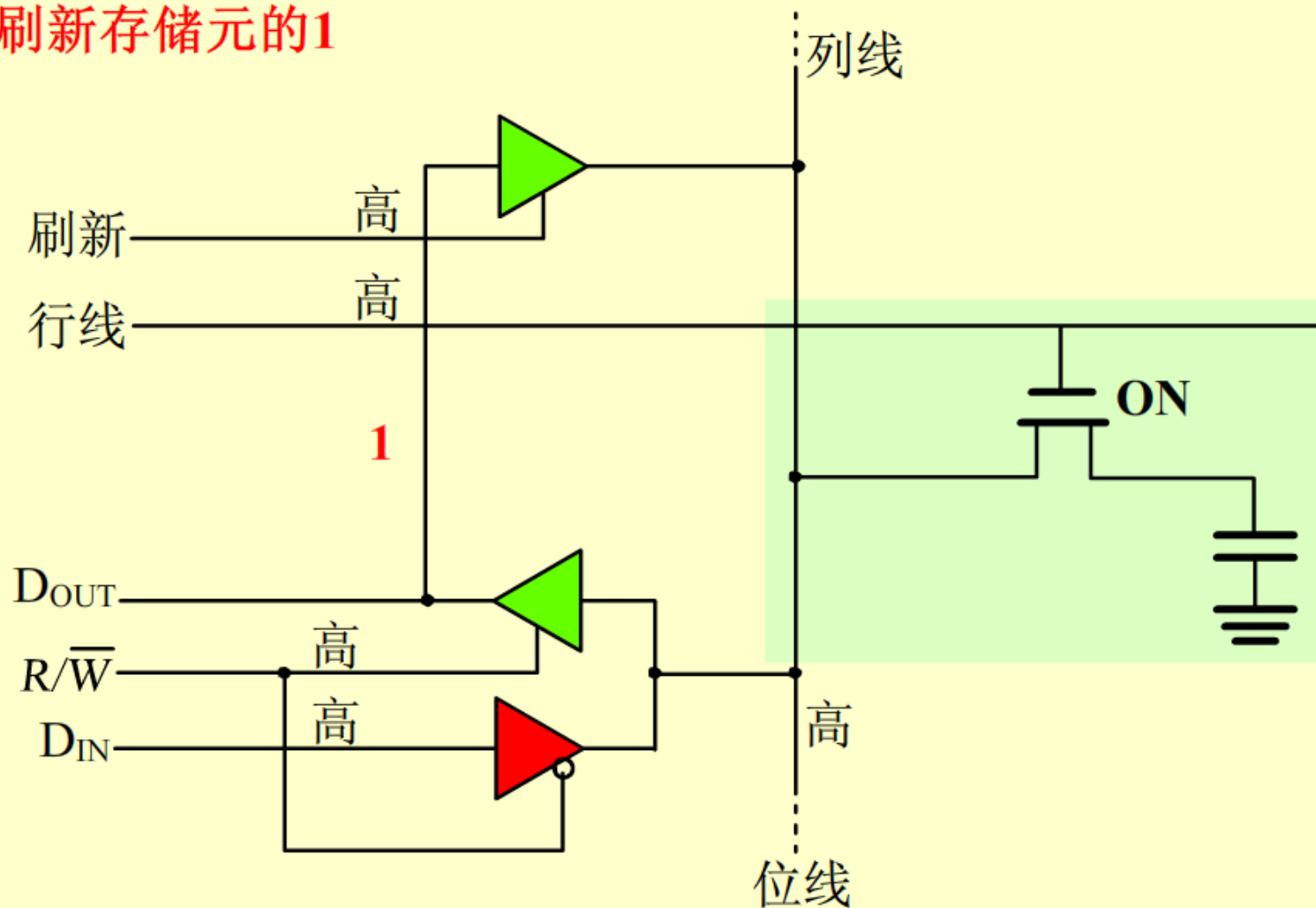


▲ 打开

▲ 关闭

DRAM存储元的基本操作

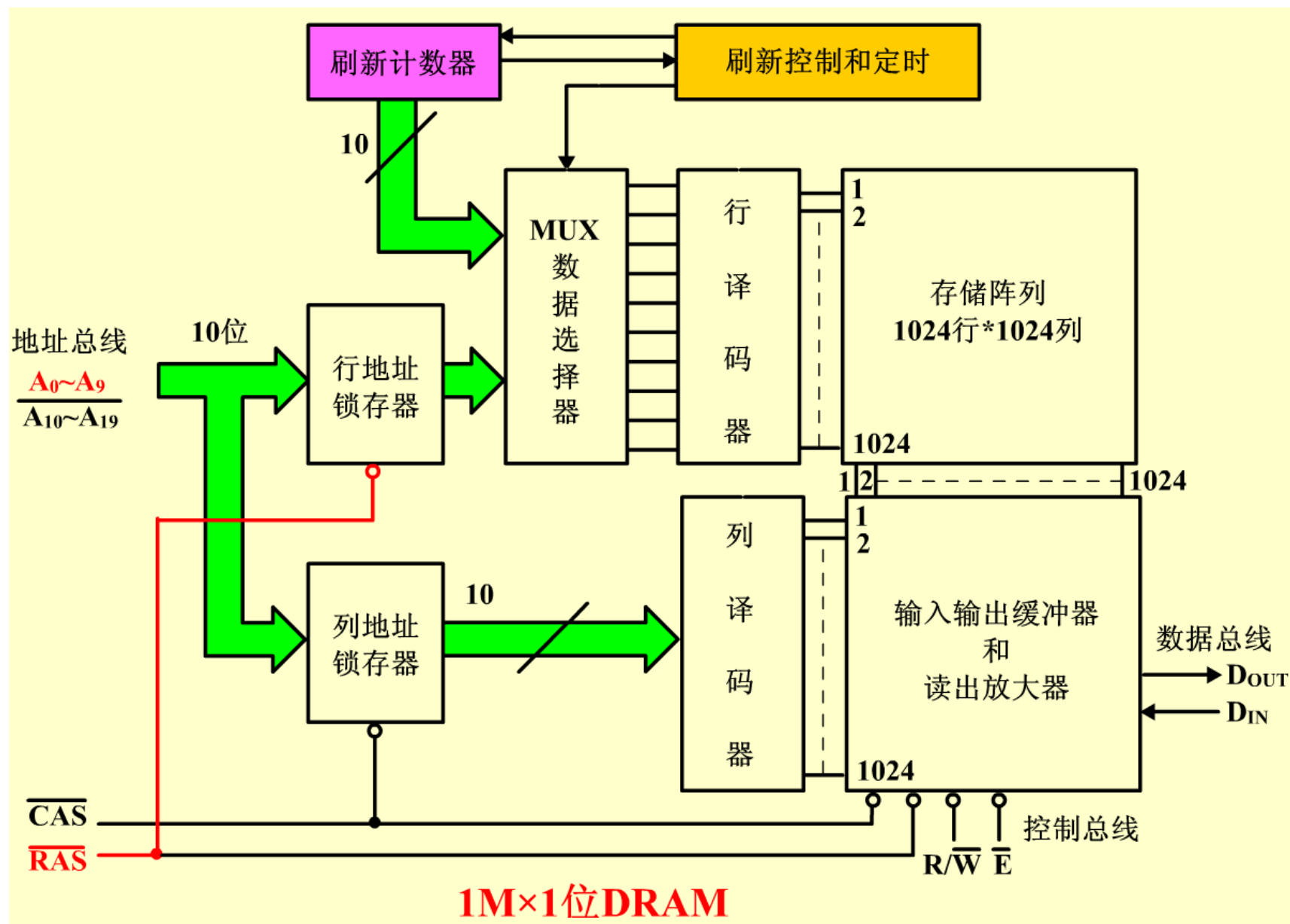
刷新存储元的1



▲ 打开
▲ 关闭

DRAM存储元的基本操作

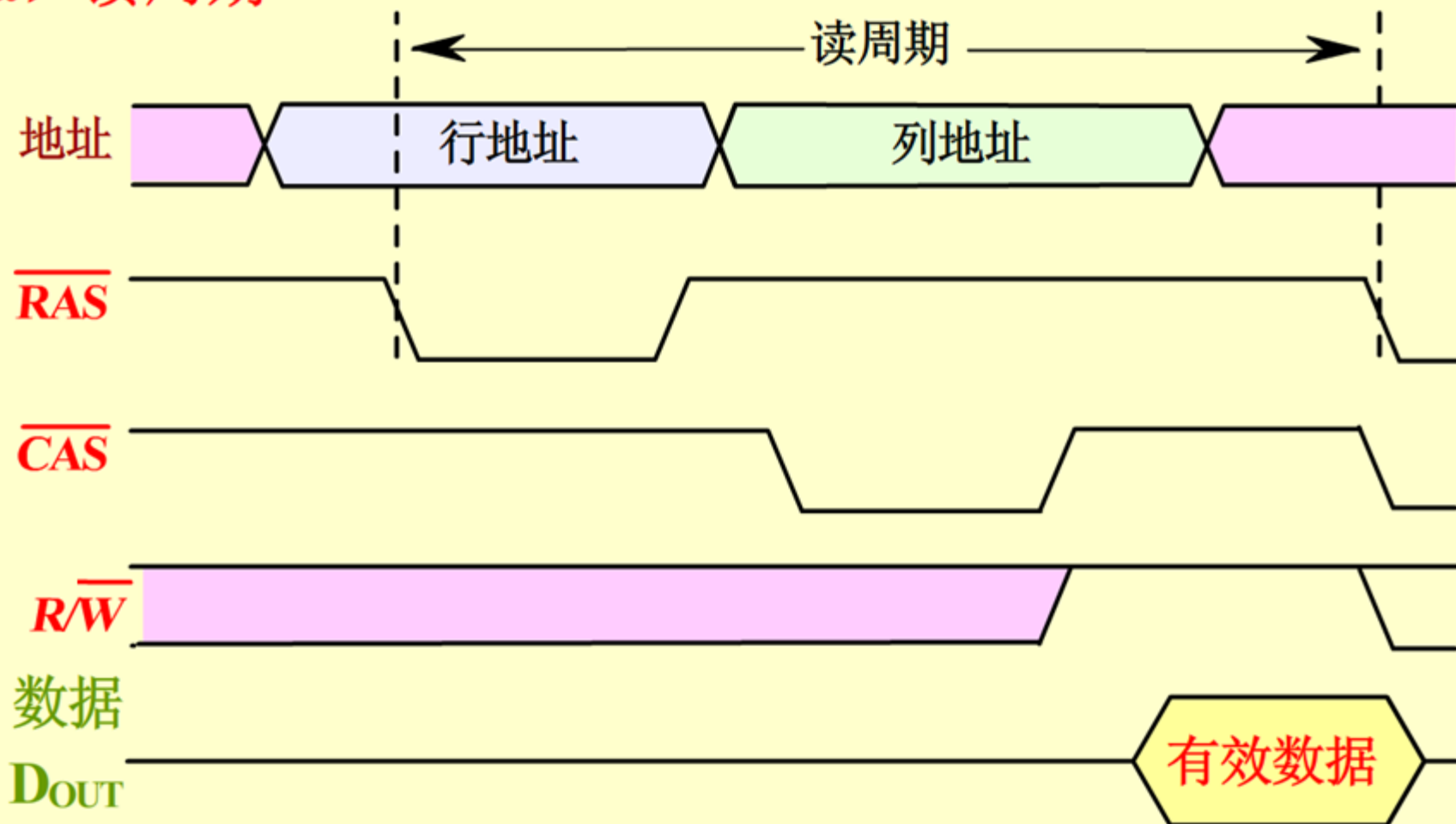
2. DRAM基本结构



2. DRAM基本结构



(a) 读周期

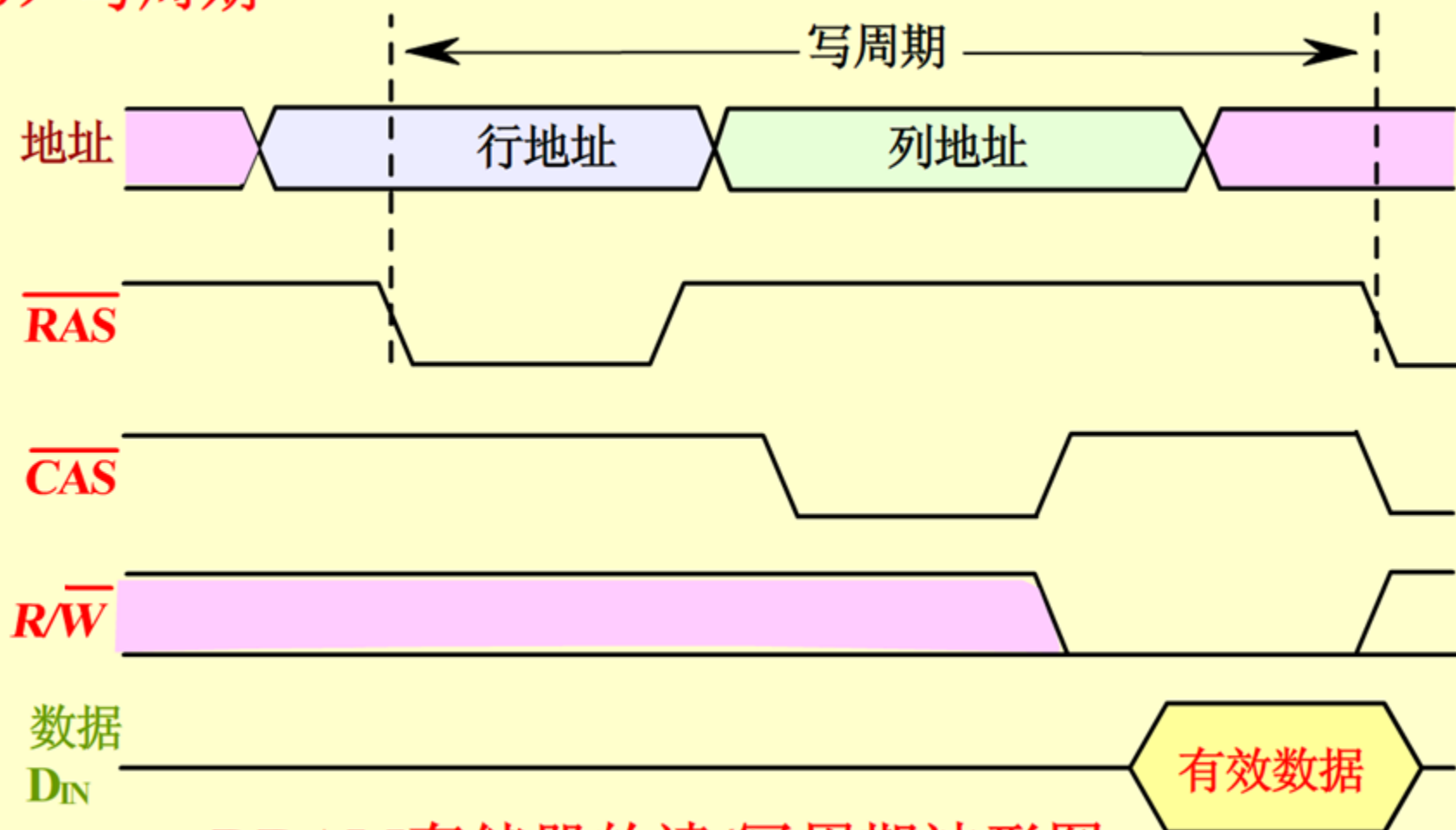


DRAM存储器的读/写周期波形图

2. DRAM基本结构



(b) 写周期



DRAM存储器的读/写周期波形图

4.3 只读存储器ROM

- 4.3.1 掩模ROM
- 4.3.2 可编程ROM

导入

只读存储器简称ROM,它只能读出不能写入。ROM的最大优点是具有不易失性,即使电源断电,ROM中存储的数据不会丢失,因而在计算机系统中得到了广泛的应用。

ROM分为:掩模ROM和可编程ROM两类,

◆ 掩模式只读存储器 (ROM)

这类ROM所存的数据,在芯片制造过程中就确定了,使用时只能读出,不能改变。优点是可靠性高,集成度高。缺点是不能改写。这种器件只能专用,用户可向厂家定做。

可编程ROM又可以分为两类:

◆ 一次编程只读存储器 (PROM)

PROM在产品出厂时,所有存储元均置成全0或全1,用户根据需要可自行将某些存储元改为1或0。

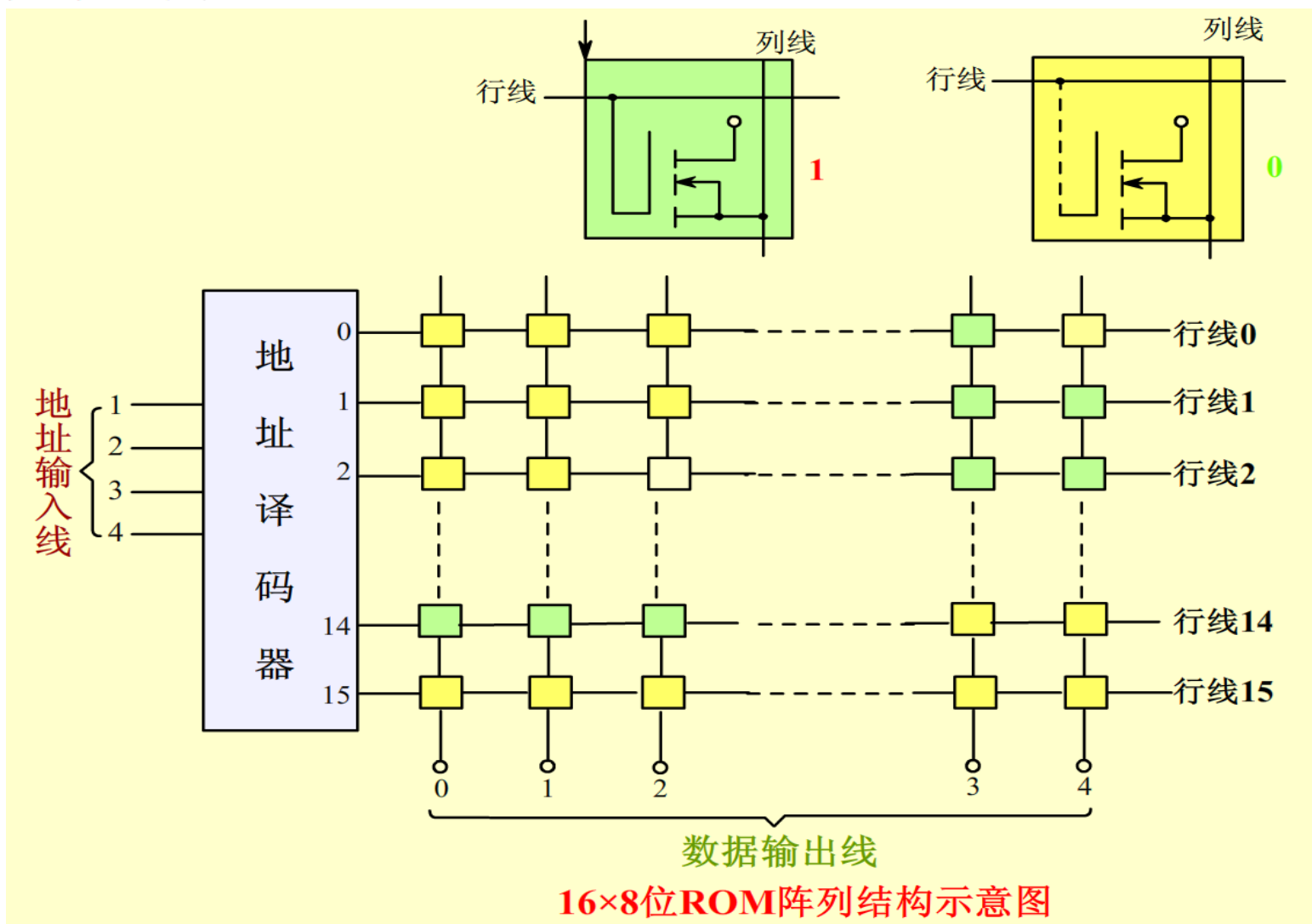
◆ 多次改写编程的只读存储器

这类ROM有EPROM, E²PROM。

4.3.1 掩模ROM

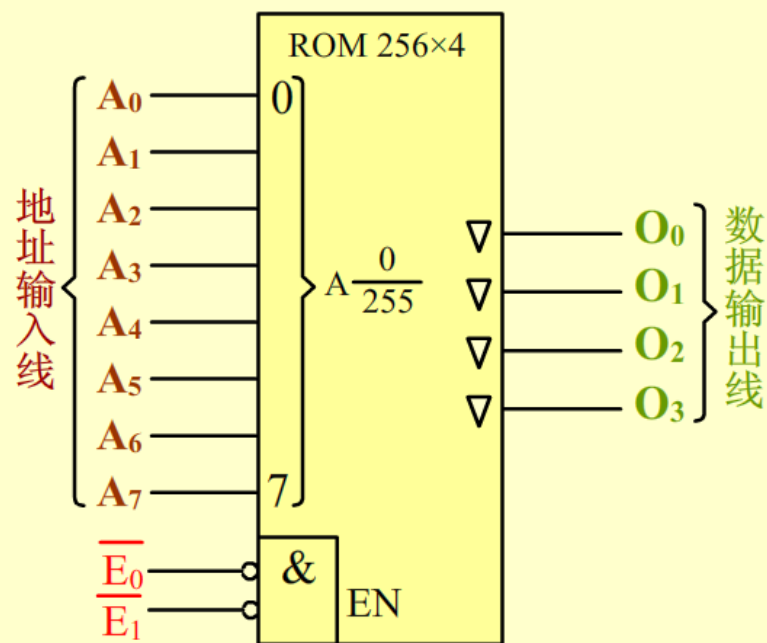
1. 掩模ROM的阵列结构和存储元

大部分ROM芯片利用在行选线和列选线交叉点上的晶体管是导通或截止来表示存 0、1。

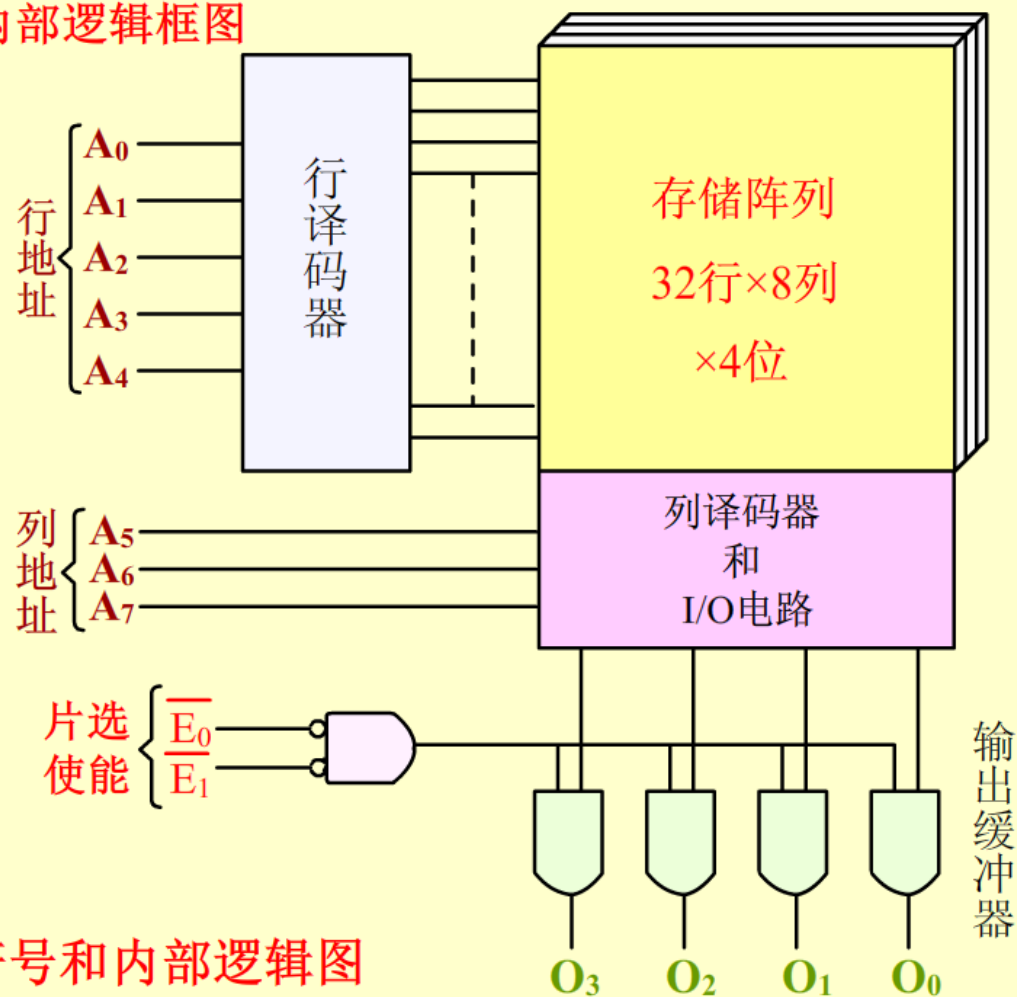


2. 掩模ROM的逻辑符号和内部逻辑框图

(a) 掩模ROM逻辑符号



(b) 内部逻辑框图

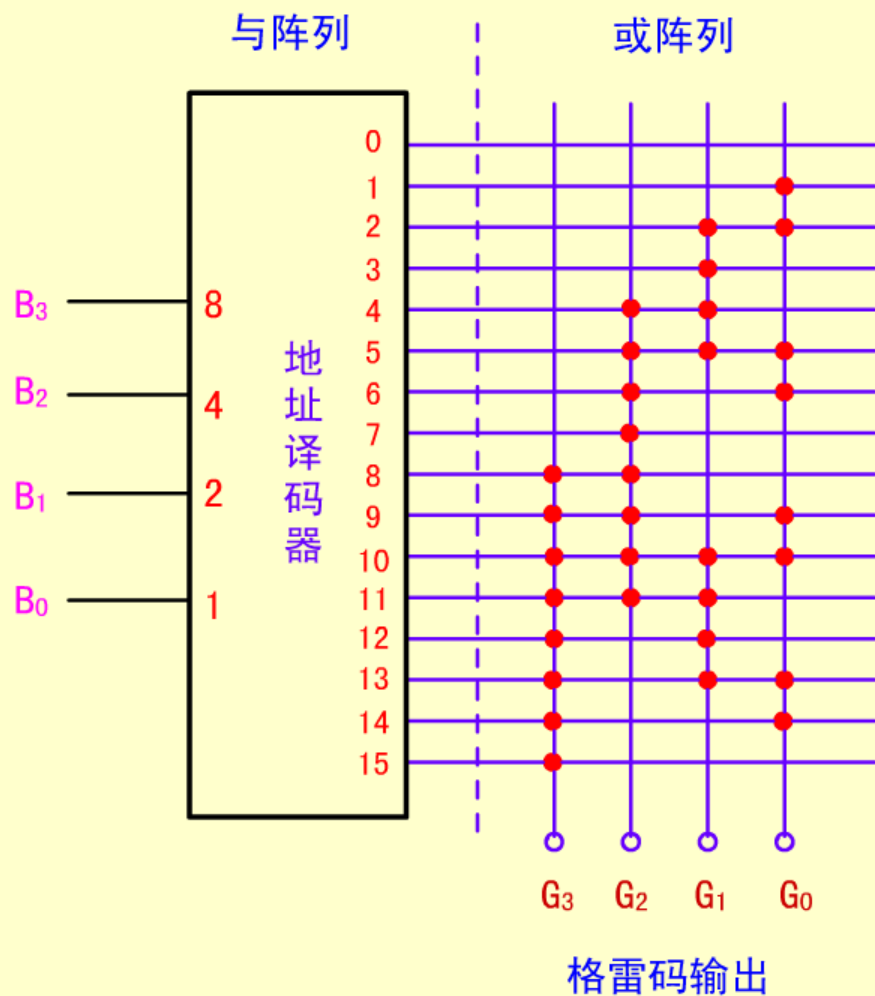


掩模ROM逻辑符号和内部逻辑图

【例1】 用ROM实现 4 位二进制码到格雷码的转换。

二进制码				格雷码			
B ₃	B ₂	B ₁	B ₀	G ₃	G ₂	G ₁	G ₀
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

二进制数由地址线输入



3. ROM结构的点阵图表示法

图4.15画出上例中ROM编程的点阵图。

左面部分是地址译码器, 译码每个输出对应一个[最小项](#)

右面部分是一个或阵列, 交叉点对应真值表项

每列格雷码输出为[最小项](#)的或形式:

真值表中对应项为‘真’的, 交叉点上有黑点, 表示ROM存储元编1 (对应), 反之为0

4.5 存储器容量的扩充*

- 4.5.1 字长位数扩展
- 4.5.2 字存储容量扩展

4.5.1 字长位数扩展

给定的芯片字长位数较短，不满足设计要求的存储器字长，此时需要用多片给定芯片扩展字长位数。

信号线：

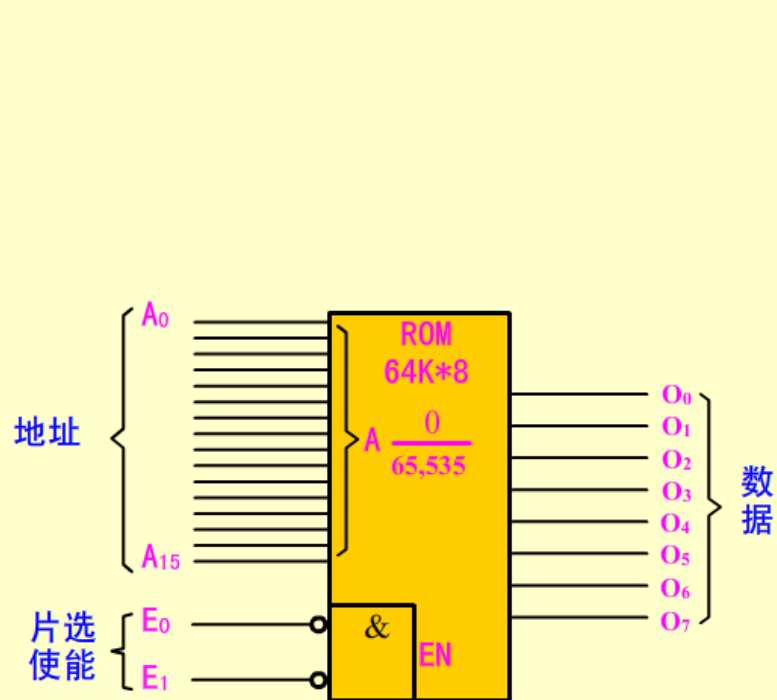
地址线和控制线公用，而数据线单独分开连接。

所需芯片数：设计要求存储容量除以已知芯片存储容量

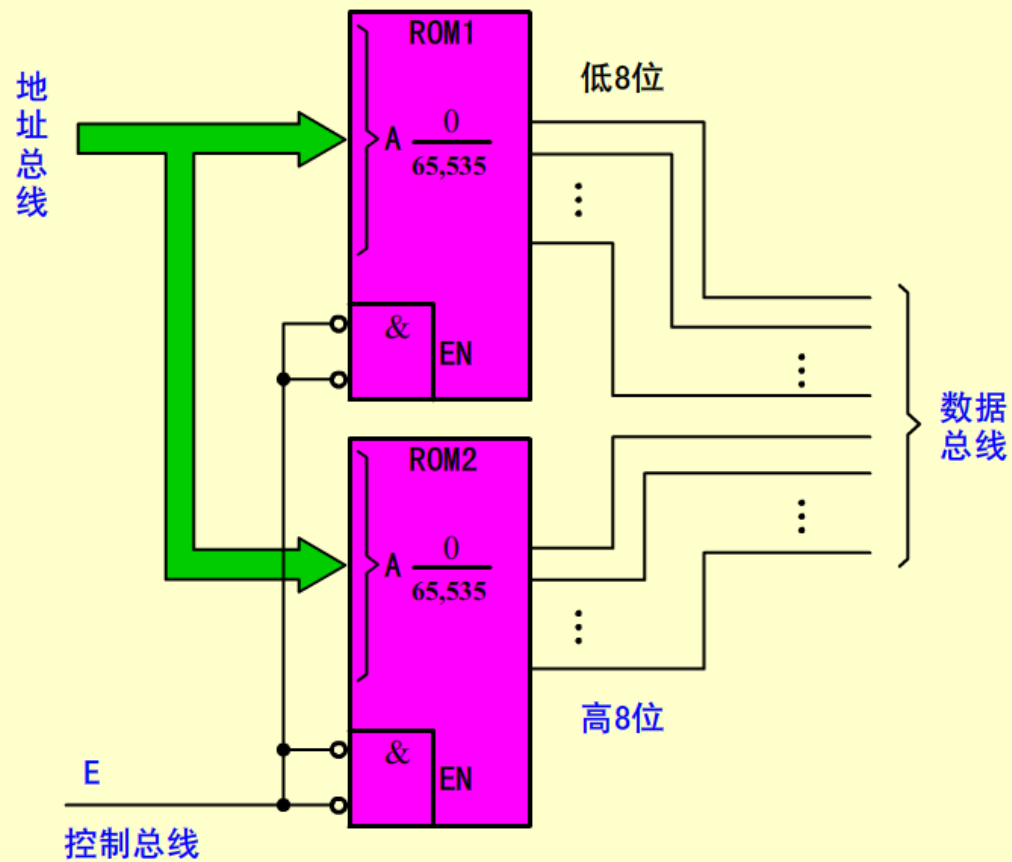
【例2】 利用 $64\text{K} \times 8$ 位ROM芯片，设计一个 $64\text{K} \times 16$ 位的ROM。

解：

设计的存储器字长为16位，存储容量不变，因此连接图如图4.21所示。其中两个芯片的地址总线公用，控制总线也公用，而数据线分成高8位和低8位。



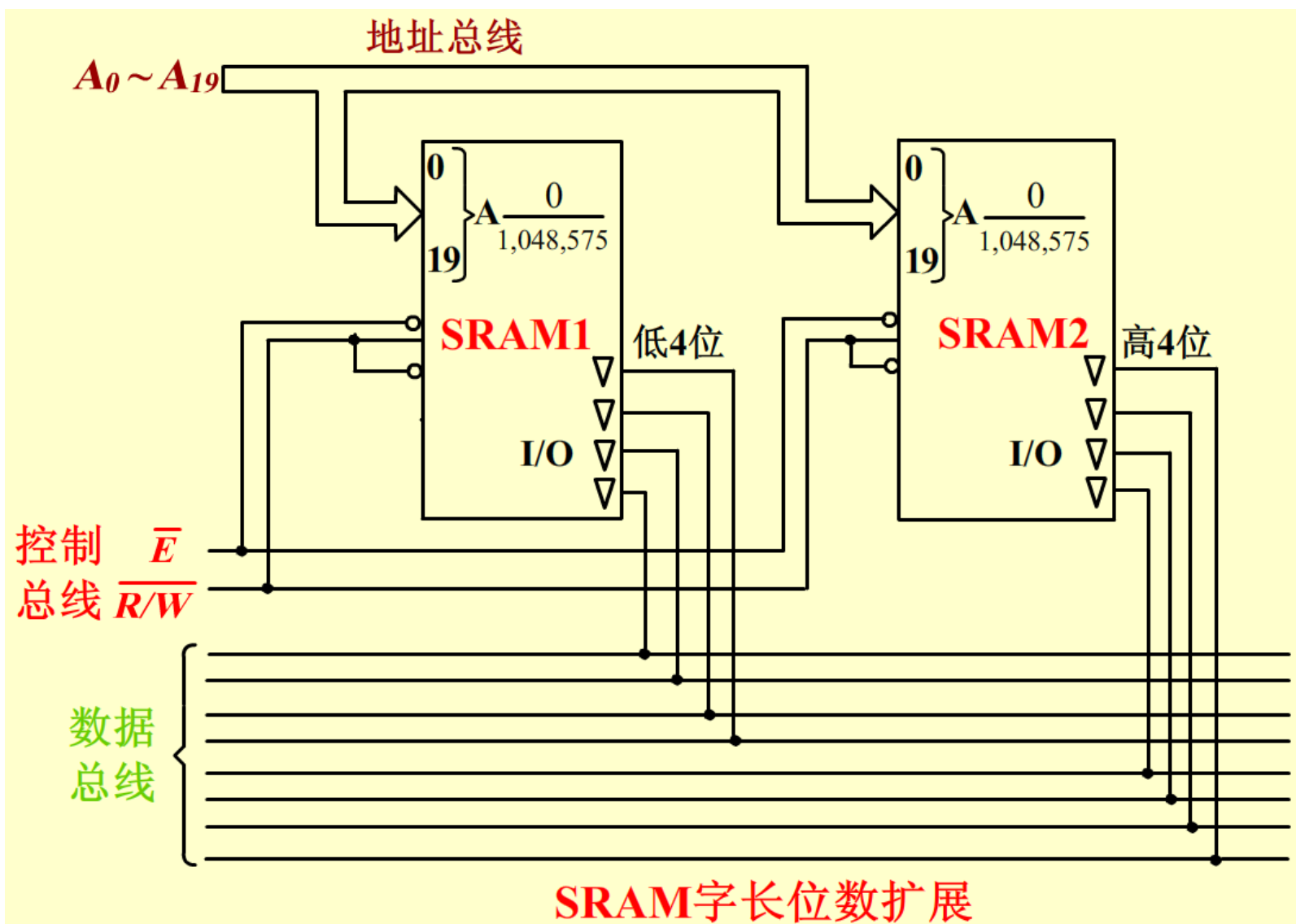
(a) 逻辑图



(b) 64k*16位ROM连接图

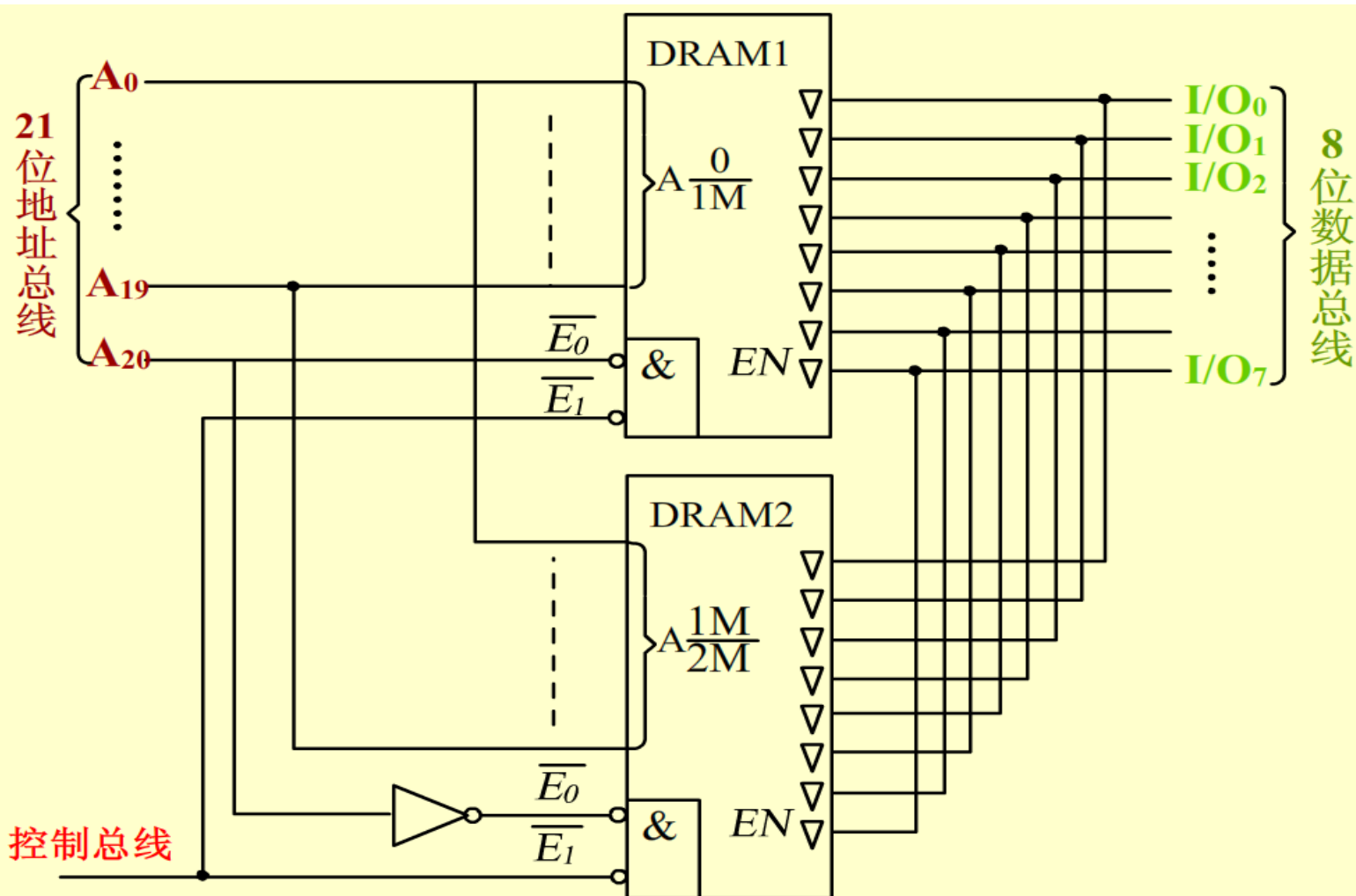
64K*16位ROM设计

【例3】 利用 $1\text{M} \times 4$ 位RAM芯片，设计一个 $1\text{M} \times 8$ 位的SRAM存储器。



4.5.2 字存储容量扩展

给定的芯片存储容量较小（字数少），不满足设计要求的总存储容量，此时需要用多片给定芯片来扩展字数。



DRAM高存储容量扩展