

# 数字 IC 第 14 次课堂总结

## 一、存储器的分类

存储器（Memory）是系统集成芯片中的重要功能部件之一，用于存储数据或控制指令，本章节最关键的内容是存储器的分类、寻址方法、SRAM 和 ROM。之前章节中介绍的 Latch 和 DFF 其实也能够存储信息，不过其中 DFF 一般用作寄存器。寄存器一般整合在 CPU 内，其读写速度跟 CPU 的运行速度基本匹配，性能优越，但成本很高。存储器在 CPU 外，容量一般比较大，但读写速度相对较慢。不同种类的存储器，具有不同的特点和应用领域，图 1 按照功能、存取方式和电路结构对存储器进行了分类。

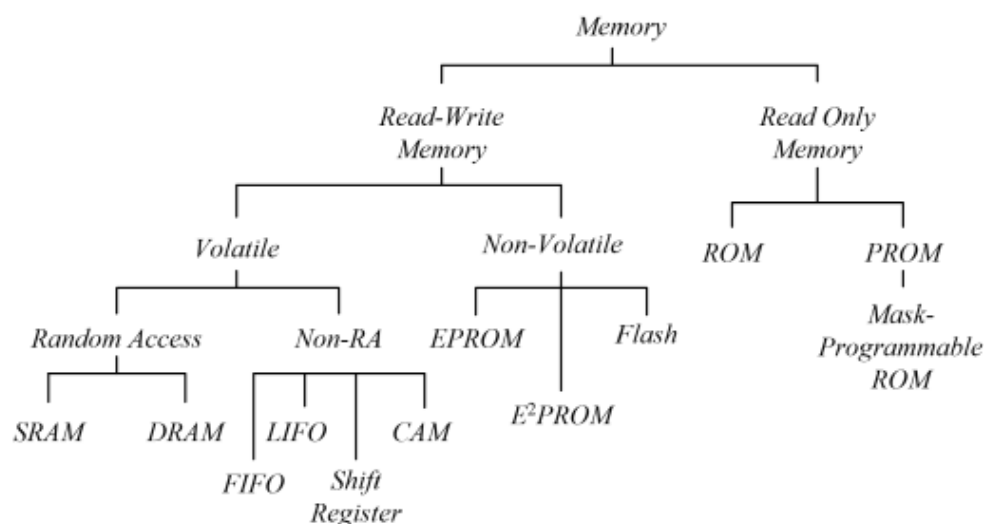


图 1 存储器的分类

存储器有两种操作，读出（Read）和写入（Write），据此可以将存储器分为两大类：只读存储器（Read-Only Memory，ROM）和读写存储器（Read-Write Memory，RWM）。ROM 的数据在芯片制造时就已固定下来，用户使用时只能读出，某一存储单元是否存在晶体管决定了这一单元是 0 还是 1。ROM 属于非挥发性/非易失性存储器（Non-Volatile Memory，NVM），切断电源并不会造成数据丢失。而 RWM 的优点是具备读和写两种功能，相比之下更加灵活。

### (1) ROM 与 PROM

可编程 ROM（Programmable ROM，PROM）只能写入一次，因此和掩膜 ROM（Mask ROM，MROM）一起被归入了 ROM。掩膜 ROM 是厂家根据客户要存储的信息，设计专用的掩模板进行生产的，生产出成品后，ROM 中的信息即可被读出使用，但不能改变。PROM 的制造工艺和掩膜 ROM 基本一致，不过 PROM 的基本单元中存在熔丝，用户可以烧断这些熔

丝，从而写入信息。写入之后信息就会固定下来，只可读出，不能被写入。

根据切断电源后数据是否还会存在，可以把 RWM 分为挥发性的（Volatile）和非挥发性的（Non-Volatile）。存储器中，数据的存在方式是存储单元状态的有序排列，如果重新上电后存储单元的状态被打乱了，那么就会丢失数据。可擦除可编程 ROM（Erasable Programmable ROM, EPROM）、电擦除可编程 ROM（Electrically Erasable Programmable ROM, E<sup>2</sup>PROM）和快闪 ROM（Flash）都属于非挥发性 RWM。

## (2) EPROM

EPROM 中的内容可以反复擦写，用电信号编程、用紫外线擦除。EPROM 的机理是热载流子注入浮栅，从而改变了 MOS 器件的开启电压，控制 MOS 管的导通与截止。以 NMOS 为例，编程时，栅极施加高电压，同时源漏之间也施加高电压，形成强电场，强电场作用下发生雪崩倍增，产生大量动能较高的热载流子（NMOS 中的热载流子是电子）。电子克服栅介质（即 SiO<sub>2</sub>）的势垒作用，注入到浮栅中，浮栅因此带有负电荷，屏蔽了栅极的作用，MOS 管始终无法开启。要擦除数据，则需要将浮栅中的电子泄放掉。紫外线或 X 射线照射可一次性全部擦除原有数据。在紫外线照射下，电子能量增加，所以能够从浮栅全部释放。

## (3) E<sup>2</sup>PROM

E<sup>2</sup>PROM 也采用基本的浮栅结构，两者的不同在于电子注入浮栅的机理以及数据擦写的方式。E<sup>2</sup>PROM 的浮栅是薄栅结构，栅氧化层的最小厚度仅为 10nm，因此编程所需的高压条件相对 EPROM 大幅降低。数据编程或擦除时，电子因为隧穿效应注入或离开浮栅。

## (4) Flash

上世纪 80 年代中期，人们发现把热载流子编程和隧穿擦除结合在一起是一种实现一个单管 EPROM 单元的方法，这种新技术被称为 Flash。Flash 结合了 EPROM 的编程能力和 E<sup>2</sup>PROM 的擦除能力，读写速度都很快。

## (5) SRAM 与 DRAM

挥发性 RWM 可以分为随机存取存储器（Random Access Memory, RAM）和非随机存取存储器。RAM 不仅可以存储大量信息，而且在操作过程中能任意读取或写入某个字的信息，相对的是 FIFO（先进先出）、LIFO（后进先出）等等需要按顺序进行读取或写入的存储器类型。RAM 又可分为静态随机存取存储器（Static Random-Access Memory, SRAM）和动态随机存取存储器（Dynamic Random Access Memory, DRAM）。SRAM 只要保持通电，里面储存的数据就可以一直保持。DRAM 利用电容内是否存储电荷来代表一个 bit 是 1 还是 0。由于晶

体管会有漏电电流，导致电容上的电荷随时间减少，因此 DRAM 需要周期性地刷新。与 SRAM 相比，DRAM 的优势在于结构简单——每一个 bit 的数据都只需一个电容和一个晶体管来处理。

## 二、SRAM

存储器中，1 bit 是基本单元，用于存储 0 或 1，通常 8 bit 的存储单元构成一个字节(Byte)， $2^n$  个字节可以构成一个字 (Word)，一个 word 通常有 16 bit、32 bit 或 64 bit。逻辑信号处理中，Word 是数字信号处理的基本单元。字长为 16 bit 时，Word、Byte 和 bit 的关系如下图所示。

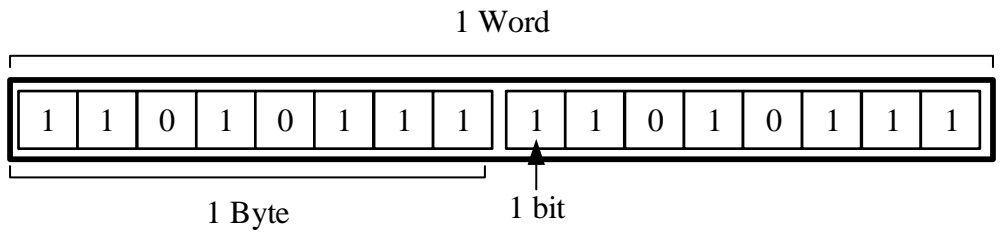


图 2 关系示意图

不同类型的存储器具有不同结构的存储单元，表现为存储单元中所用晶体管元件数量的明显不同。由于 RAM Cell 单元在存储阵列中大量使用，因此单元内的晶体管数对系统的集成规模和成本将产生重要影响。

SRAM 中经常采用的是图 3 所示的经典 6T 结构，其中两个倒相器构成静态锁存器结构，可存储数据 0 或 1，而对该存储单元数据的存储和读取，需要通过字线 (Word line) 信号控制的两个读/写开关传输门。该结构中字线控制 MOS 管开启或夹断，MOS 管夹断时，数据不变；MOS 管开启时，既可以写入数据又可以读出数据。两条位线 (Bit line) 则用于读出或写入。为了便于数据写入，要求由反相器构成的 Latch 不能有太强的锁存能力，否则写入时可能难以改变 Latch 的状态。但是也不能太弱，否则轻微的干扰就可能改变存储的数据，因此该 Latch 应当是一个 Weak loop，从而兼顾读操作和写操作。

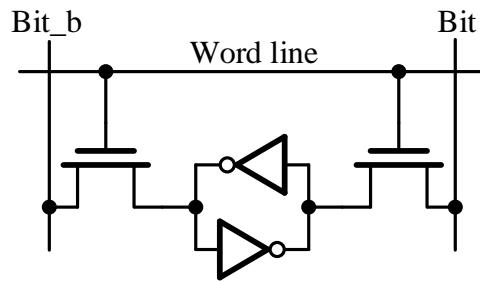


图 3 6T 单元

其他的 SRAM 基本单元如图 4 所示，它们的都是对 6T 单元的改进

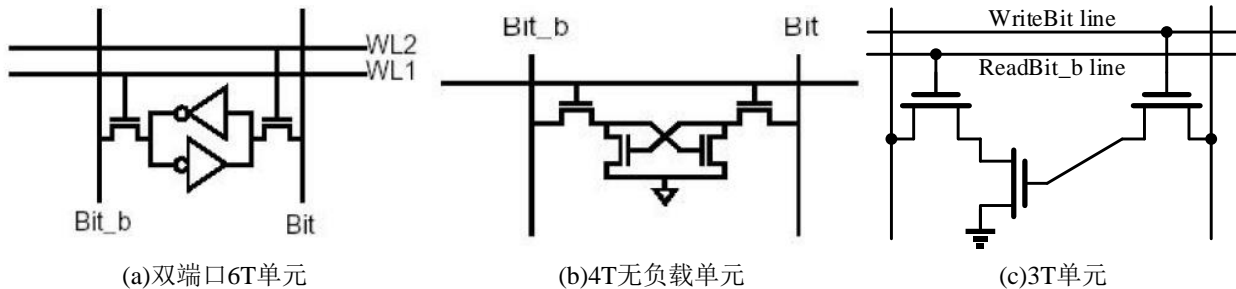


图 4 其他 SRAM 基本单元

### 三、存储阵列的寻址

只有将同类型的存储单元组成阵列结构，才能提高存储容量，规则的阵列结构也有利于布局布线和单元寻址。在逻辑信号处理中，读写操作的基本单元是 Word，并对其中的所有 bit 进行读写操作。寻址就是找出要操作的 Word 的过程。寻址方式一般可分为直接寻址和译码寻址，其中译码寻址也可称为间接寻址。

#### (1) 一维与二维阵列结构

直接寻址是一种基本的寻址方法，其特点是：在指令格式的地址字段中直接给出实际的地址，而不需要经过某种变换，所以称这种寻址方式为直接寻址方式。对大容量数据字 Word 的寻址，如采用这种一一对应的寻址方式，信号控制线即地址线将与 Word 的数量一样多，这显然没有实现的可能。译码寻址技术成为唯一有效降低输入寻址信号规模的技术手段，如译码寻址可将  $m$  个存储单元经编码后降低到仅包含  $n=\log_2(m)$  个地址信号。然而，图 5 右图所示的一维译码寻址方式，虽然有效减小了信号线数量，但译码器的扇入数过大、难以实现，而且存储器版图为细长条，长宽失配，也容易造成面积浪费。一般最佳的芯片版图是长和宽较为接近的矩形，这样易于在晶圆上拼接各个版图，同时也不会造成太多面积浪费，从而减少芯片制造成本。

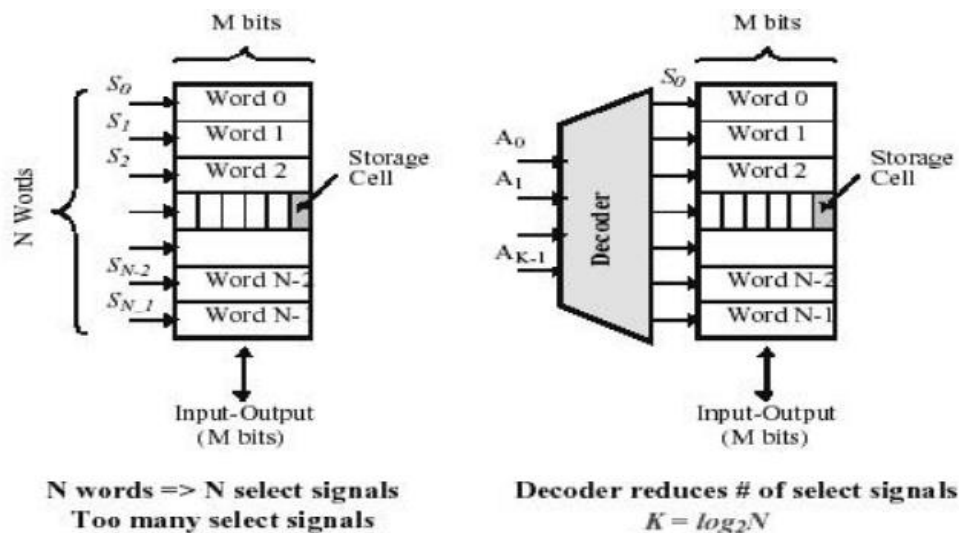


图 5 直接寻址和译码寻址（一维）

解决版图为细长条的方法是将一维译码寻址方式变为二维译码寻址方式，即采用行列译码，如下图所示。

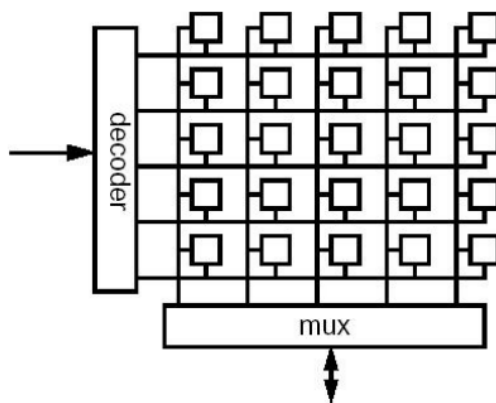


图 6 二维译码寻址

存储器容量相同时，行和列的地址线总量不变，但版图的布局可以更加接近正方形。另外，由于一个 Word 中包括许多存储单元 bit，且这些存储单元排布成了长条状，即每个 Word 的版图宽度会大于版图高度。因此为了使版图尺寸比例接近正方形，二维译码寻址方式中行数一般会大于列数。

## (2) 分块与层次结构

对于大容量的 RAM，即使已采用二维阵列地址译码技术，字线和位线的负载仍然过大，读写速度受到严重限制，实现相同的读写速度需要消耗更大的功耗。以 128k-Byte SRAM 为例，数据线总位数为 17 bit，如果行地址为 10 bit，即对应 1k 的字线，则列地址为 7 bit，对应 128 的位线，总计形成 128k Byte 的数据容量。此时，每一个位线有 1k（1024）个 MOS 管的 S/D 电容负载，而每一个字线有 128 个 MOS CGS 栅电容负载。可见，存储阵列的容量越大，

则位线和字线的负载电容越重，数据的读写速度也越慢。

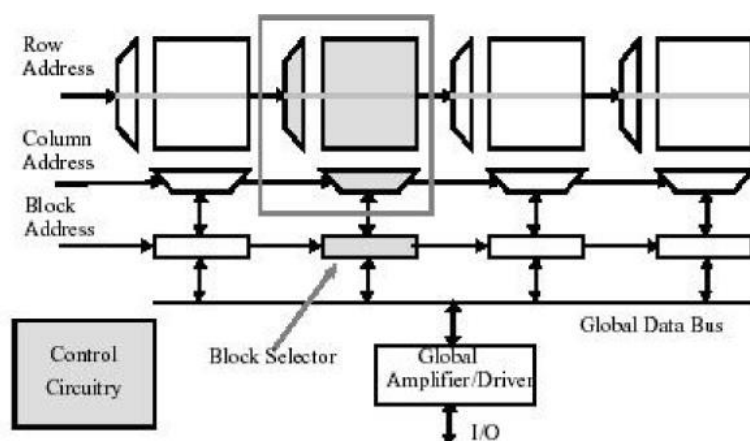


图 7 分层分页寻址结构示意图

采用分块（页）技术可部分缓解该问题。分块技术将整个存储阵列分为  $m$  个相同的子阵列，每次只能选中  $m$  个子模块中的某一个，由于阵列规模显著减小，由此有效降低了子阵列中位线和字线的长度，负载减小、功耗降低。工作时，首先决定选中哪一个模块，然后才是对模块中存储单元进行二维寻址。因此，分块技术本质上是一个三维以上的多维寻址技术，为此采用两级译码电路结构，第一级为模块选择，第二级为模块中具体的存储单元的选择。依次类推，采用多级译码结构可实现多维寻址。

### (3) 静态预充电技术

对于巨大规模的存储阵列，即使采用分层多级或分步的译码结构，字线和位线的负载电容仍然较大。为提高被选中存储单元的读写速度，在电路设计层面可采用位线的静态预充电技术。由 PMOS 上拉有源负载或 NMOS 下拉有源负载电路组成，负载控制信号可采用固定偏置或时钟驱动。

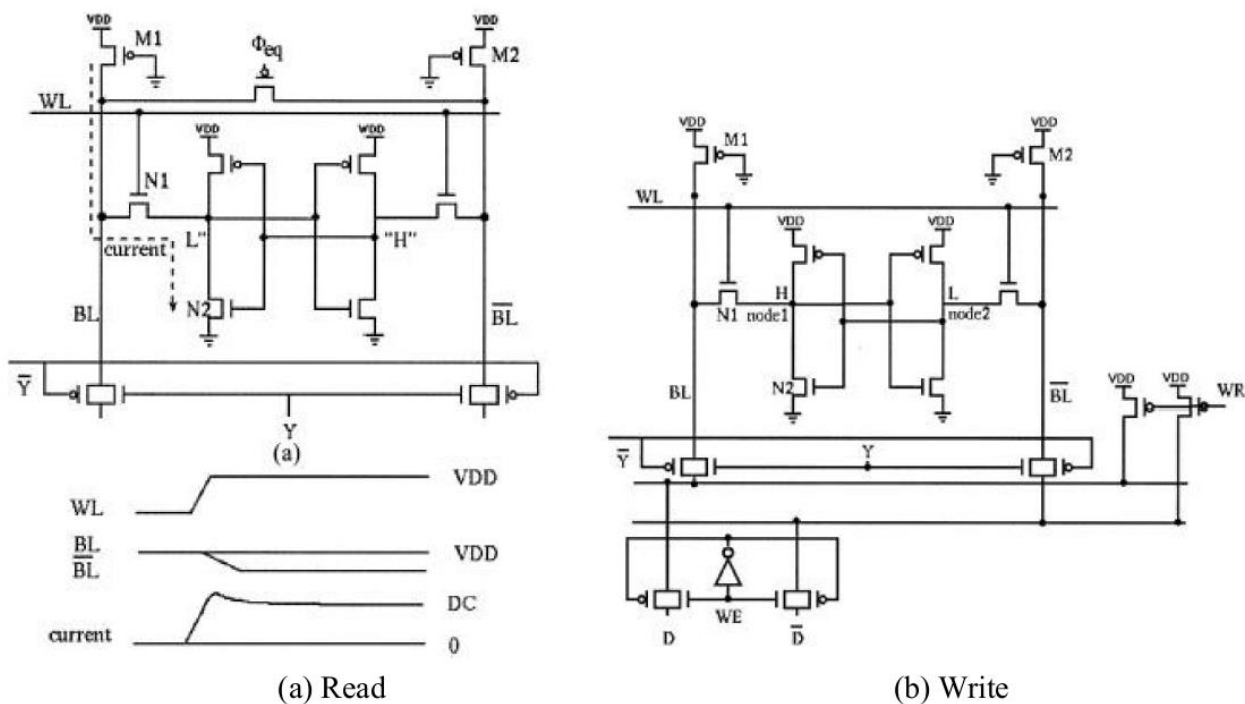


图 8 读写电路结构

信号读出模式下，经列译码使  $Y=1$ ，则某位线被选中。而在该位线未选中之前，负载  $M1$  和  $M2$  将  $BL$  和  $BLb$  均预充电到高电平  $VDD$ 。当行译码选中某字线  $WL$  后，且该字线所对应的某位线也被选中，设该单元中对应  $BL$  的位线存储的是低电平，则  $BL$  一侧开始放电，使该位线被拉低，与  $BL$  互补的  $BLb$  则保持  $VDD$  高电平。

数据写入模式下， $WR=0$ ，PMOS 负载会上拉外部  $BL$  和  $BLb$  的电位并预充电到高电平。当单元被选中后，该单元的字线和位线使该单元可以被写入。一旦写信号使能， $WE=1$ ，外部数据就会经传输门输入位线，即  $D$  和  $Db$ ，送入  $BL$  和  $BLb$ 。该数据再经过列地址选择的  $BL$  和  $BLb$  输入某内部位线。由于  $M1$  和  $M2$  的  $W/L$  小，导通电阻大、驱动弱，与外部数据相比， $BL$  和  $BLb$  的状态由外部数据状态决定，因此选通的单元中的  $BL$  或  $BLb$  能够被放电至低电平。若  $D=0$ ，则  $BL=0$ ，若  $Db=0$ ，则  $BLb=0$ ，如此外部数据被写入某存储单元中。

#### (4) 地址译码器

地址译码电路可采用基本的 NAND/NOR 逻辑实现，电路结构上可采用具有低功耗特点的静态 CMOS 逻辑，或具有高速特点的动态 NOR/OR 逻辑、钟控 AND 逻辑、或时序输出的源耦合逻辑等。异步时序存储器中没有外部时钟输入，因此控制时序需要内部产生，当新的地址产生时，表明一个新的存储单元的寻址动作开始。地址瞬态检测电路（Address Transition Detect Circuits, ATD）在存储器地址发生变化时会产生一个正脉冲信号，用以启动相应的时序控制电路。

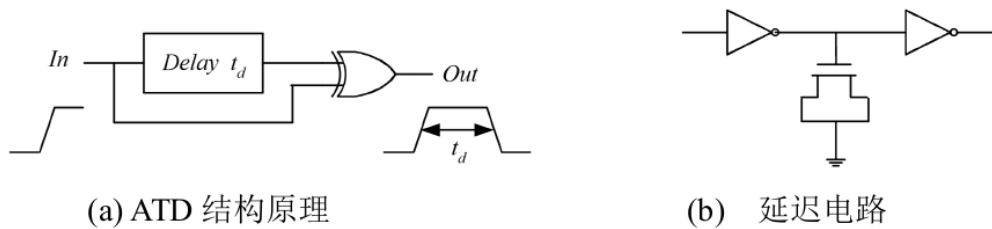


图 9 ATD 结构原理和延迟电路

ATD 控制过程将分两步完成：首先是在存储器地址发生变化时，能够检测到这种地址的变化，其次，当检测到地址发生改变后，使 ATD 开始输出高电平，并经过一段延迟后，ATD 的输出端再恢复到低电平，即在新地址到来后，ATD 完成会输出一个窄脉冲。异或电路可以检测输入信号的变化，结合延迟电路，就可以实现完整的 ATD 功能。

在图 10 用于检测地址数据变化的 ATD 电路中， $C_s$  为 ATD 电路边沿检测信号， $C_s=0$  时，会使  $ATBR=0$ 、 $Resetb=1$ ， $Q_1$  和  $Q_2$  的反馈回路被断开，此时 ATD 处于固定的输出状态，不响应输入地址的变化。当  $C_s=1$  时，ATD 正常工作。图中  $Q_1$  为上拉预充电管， $Q_2$  与反相器、与非门构成反馈回路，形成 ATD 中的下降沿输出。

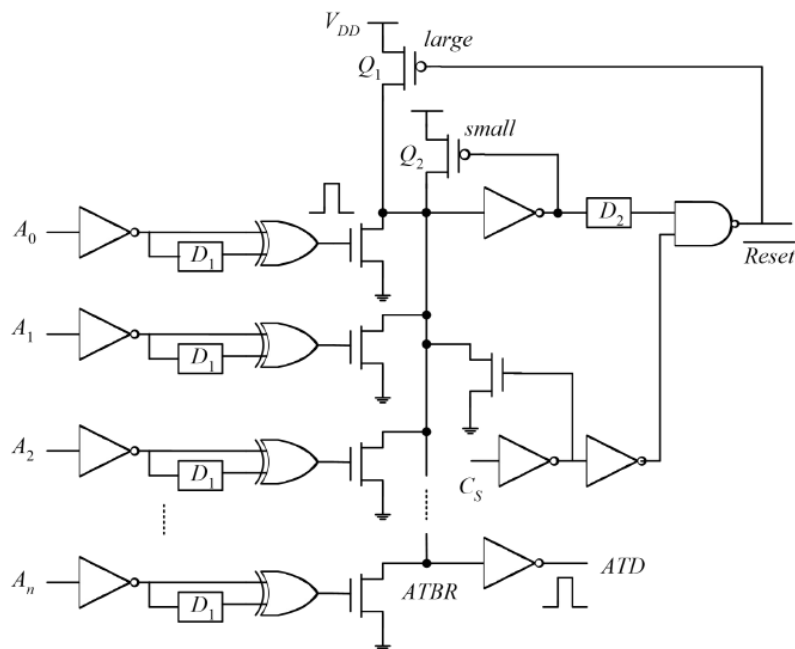


图 10 用于检测地址数据变化的 ATD 电路

## 四、ROM

一般 RAM 中，掉电后电容或锁存器不工作，因此数据会丢失。而常见的 ROM 中，是用字线与位线交叉位置处是否存在晶体管来区分 0 和 1 的，所以掉电后数据也不会丢失。ROM 中存在两种类型的单元阵列结构：并联的或门 NOR 阵列结构，串联的与门 NAND 阵列结构。



NOR 阵列中每个 MOS 管的漏端都需要和 BL 连接，因此在同样的容量下 NOR 阵列占用的面积大，但由于串联电阻小，因此速度较快。NAND 与门阵列特点正好相反，占用面积小，但速度相对较慢。

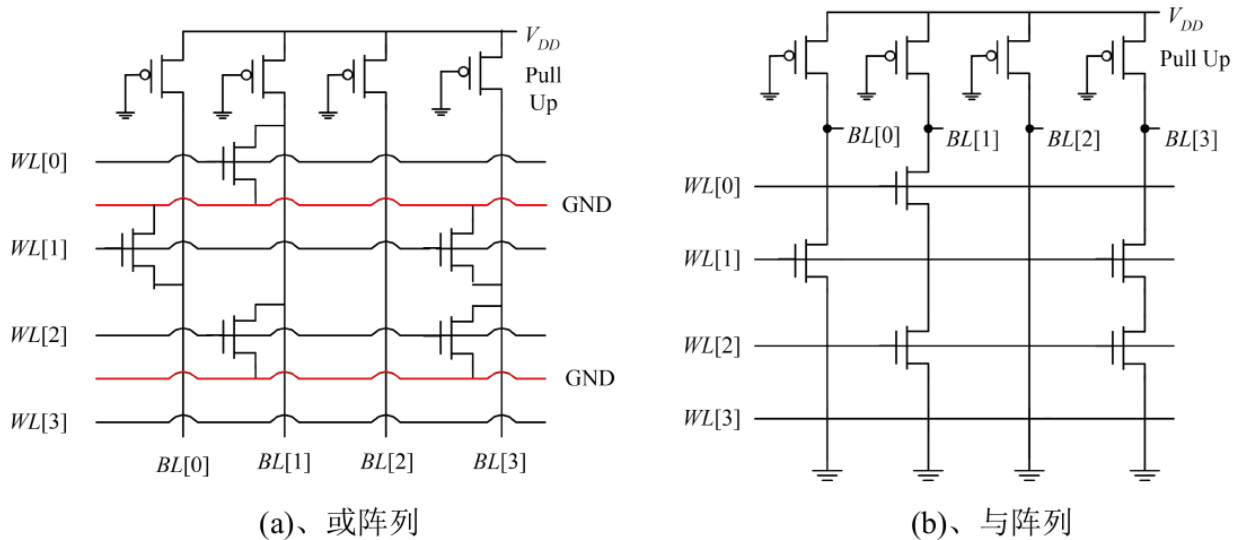


图 11 ROM 的 2 种阵列结构

图 11(a)是一种采用或阵列 ROM 电路，不同于 SRAM 中的互补位线输出，其中每个 ROM 存储单元仅有一条位线输出。该阵列中未被选中的 WL 输出低电平，因此没有任何一个 Word 被选中时，位线的默认输出是 1（即 VDD）。当某一 Word 被选中，即 WL=1 时，没有 MOS 管的 WL、BL 交叉点仍然是高电平，因此该存储单元存放的数据是 1；当交叉点处存在 MOS 管时，交叉点处的 NMOS 管会导通，而且其导通电阻远小于 PMOS 管的导通电阻，从而使位线 BL 的电位接近 GND，这表示该存储单元存放的是 0。

与阵列的分析过程恰好相反，未被选中的字线 WL 是高电平，因此没有任何一个 Word 被选中时所有 MOS 管均导通，而且导通电阻远小于 PMOS 的导通电阻，应此位线的默认输出是 0（即 GND）。当某一个 Word 被选中时，WL=0，导致这一 Word 中的 MOS 管不开启，从而使存在 MOS 管的位线输出 1，不存在 MOS 管的位线输出 0。