

微 型 机 原 理

(下 册)

北 京 工 业 大 学

目 录

第六章 微型计算机的存贮器

- [6.1] 半导体存贮器的分类..... (1)
- [6.2] 读/写存贮器 RAM..... (2)
- [6.3] 只读存贮器 ROM (31)

第七章 可编程序接口芯片

- [7.1] Z80 PIO 并行输入/输出接口芯片 (36)
- [7.2] Z80 CTC 计数器/定时器芯片..... (59)

第八章 TP801 单板微型计算机

- [8.1] TP801 单板微型计算机的主要技术指标和功能..... (77)
- [8.2] TP801 单板微型机的硬件结构 (79)

第九章 TP801 监控程序介绍

典型程序剖析

- [9.1] 概述..... (93)
- [9.2] TP—BUG 介绍 (94)
- [9.3] TP—BUG 的程序流程图主要程序段及其说明 (101)

- 附录一 Z80 机器码指令表..... (138)

- 附录二 TP—BUG 程序清单 (151)

第六章 微型计算机的存储器

微型计算机几乎都采用半导体存储器芯片，几乎不采用磁芯存储器，这是因为半导体存储器与微处理器的配合更理想，尺寸也较小，功耗也低，工作更可靠，扩充又很灵活。在微处理器出现之前，半导体存储器就已经开始应用了，但是当时由于价格昂贵，只能应用于某些存储容量小，而且不考虑成本的场合。由于LSI技术的飞速发展，使半导体存储器的容量迅速提高，价格急剧下跌。70年代微处理器的出现扩大了半导体存储器的市场，更促进半导体存储器的发展，而半导体存储器的发展又促进微型机的发展。可以说，微型机的发展是与半导体存储器的发展紧密相关的。

目前半导体存储器的种类、型号及规格繁多。在设计微型机系统时，如何根据系统要求合理选择半导体存储器芯片，并使之与CPU连接起来，这是微型机系统设计中的一个重要方面。

本章主要介绍适应于单片MOS微处理器的MOS半导体存储器，及其与CPU之间的连接。

[6.1] 半导体存储器的分类

半导体存储器的分类如图6.1所示。

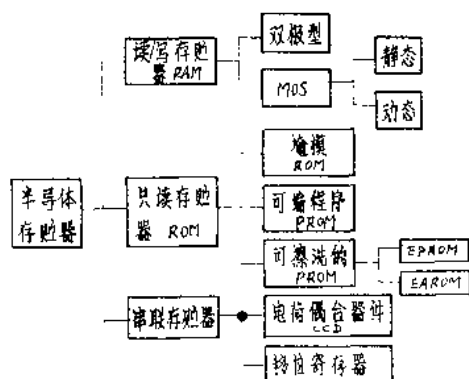


图 6.1

半导体存储器按功能可以分成读/写存储器 RWM (Read Write Memory); 只读存储器 ROM (Read Only Memory); 串联存储器 CCD (Charge Coupled device)。读/写存储器通常又称为随机存储器 RAM (Random Access Memory)。实际上读/写存储器和只读存储器都是随机存储器，因为随机存储器是指所有存储单元都可以在同样时间内进行存取，大部分标准的存储器（包括ROM和PROM）都是随机存取的，移位寄存器则是非随机存取的存储器。所以将读/写存储器称为RAM是一种误称，称为RWM更为确切，但习惯上已将RAM

意指RWM了。因而RWM这个名称却很少使用。

RAM按工艺又可以分为双极型RAM和MOS RAM，而MOS RAM又可以分为静态和动态两种。

双极型RAM的特点是：存取速度高。对于射极耦合逻辑的(ECL)电路可达10ns，对于肖特基TTL逻辑可达25ns。但集成度比MOS的低，功耗比MOS的大，成本也高，它主要用于速度要求高的位片式微型机中。

静态MOS RAM的存贮单元是由MOS双稳态触发器构成的,只要不切断电源,就可以永远保存信息。它也无需时钟。它的集成度高于双极型RAM,却低于动态RAM。而功耗低于双极型、高于动态RAM。

动态RAM是利用MOS路中的栅极电容来保存信息。但是电容器上的电荷总会通过一些漏电阻(虽然MOS电路的漏电阻很大)消失。为了补充信息电荷,动态RAM隔一定时间(1—2ms)要重新写入一次原信息,这种操作称为刷新或再生。它比静态RAM集成度高,功耗低成本也低。不过为了要刷新,要增加刷新电路,而且刷新将会干扰处理器的执行时间。微型机的小系统一般都采用静态RAM。

ROM根据命令只能读出信息(在线使用之前先将信息写入)。它比RAM集成度高,或本低,更重要的是当断开电源时,它原来已写入的信息不会改变。因此,在计算机里ROM常用来存贮固定的程序、常数、系统软件、引导程序和监控程序等。目前有人提出“硅软件可能淘汰软盘”的预言。这是由于它的密度为RAM芯片的四倍,而成本只有RAM芯片的一半;使用硅软件可以腾出主存空间作为他用;它可以大大缩小计算机系统的体积和耗电量;它的工作寿命比软盘长得多,价格也便宜。

ROM可以分为如下几种:

掩模ROM,它是由生产过程中的一道掩模工艺决定了它其中的信息,一旦生产完毕信息就不可能改变了。这种ROM适合于程序成熟,批量生产的场合,一般都在成十万片以上,才较为合算。

PROM是一种在正常运行时不能改变信息,但可以在特殊条件下编程的只读存贮器,制造厂家生产的PROM在出厂时各个单元都处于相同状态。用户根据需要在专用设备上写入自己需要的信息,但只能写入一次。它适合于小批量生产。

可擦洗的ROM,为了适应科研的需要,希望ROM能根据需要写入信息,也希望能把已写入的内容擦去,然后再写。这种可改写的ROM又分为EPROM和EAROM。EPROM擦去信息时要从电路板上取下来,置于紫外线光源下约十分钟,就可以将内容擦除。然后在专用编程器上将新的信息重新写入。EPROM的特点是,即使想改变其中已经写入的一位,也必须把整个内容全部擦去。另一种可重新编程序的ROM是EAROM这是电可改写的ROM,这种ROM的特点是,要擦除其内容,不必从电路板取下来。它是用电来擦洗,可以改变其中个别单元的内容。

EPROM和EAROM都是可以改写的ROM,改写次数可以多次,但它与RAM还是不同,它的写入是在特殊条件下,要花费较长的时间。(相对于读操作)。而在线运行时,它仍然是一种只读存贮器,由于它的可改写性,很适用于调试样机和科研的场合。一般小批量产品的软件,也都使用EPROM,如TP801的监控TP-BUG就是采用EPROM。

[6.2] 读/写存贮器 RAM

6.2-1 RAM 的组成

一个基本存贮单元只能存贮一位二进制信息,目前市场上RAM芯片的容量有1K位

至 64 K 位。故需 1K 至 64K 个存贮单元。这些存贮单元必须有规则地组合起来，一般按矩阵方式排列。为了能对指定的单元进行读出或写入操作，还要有外围电路。这些外围电路通常是：

- 地址译码电路，以便对指定单元进行读写操作。
- 读写控制，它处于数据总线和被选中单元之间，用以控制被选中单元的读和写。
- 片选控制，目前每片 RAM 的存贮容量是有限的，要用若干片才能构成微型机的存贮器。因此不同的地址单元可能处于不同芯片中，不同地址要选中不同芯片。对于每个芯片来讲只有当片选 ($\bar{CS}$) 有效，才能对这一片上的存贮单元进行读写操作。
- 集电极开路或三态输出缓冲器。为了扩展存贮器的字数，常需将几片的数据线并联使用；或与双向的数据总线相接；或存贮器芯片的数据线本身就是双向的（即输入和输出公用一根引脚）。

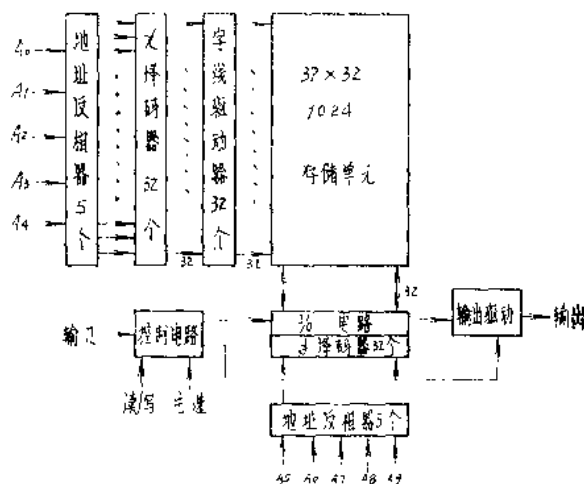


图 6.2 1024 位 MOS 静态 RAM 电路框图

一、静态 RAM 的组成

下面通过两种芯片的结构电路框图，来了解静态 RAM 结构形式。

（图 6.2）给出 1024 位 MOS 静态 RAM 电路框图，框图由两部分组成，一是 $32 \times 32 = 1024$ 的存贮单元阵列。以一位为一个字来说，它组成 1024 个字，每个字的长仅一位。这种存贮单元的组织方式称为 1024×1 位结构；另一是外围电路，包括：地址反相器，地址译码器，字驱动器，I/O 电路，控制电路和输出驱动等。外围电路的作用是保证能对选中的单元进行读或写操作。因为 1024 个单元分别以 32 个行（x 坐标）地址中的一个数值（例如 x_0 ）和 32 个列地址（y 坐标）中的一个数值（例如 y_0 ）来标志他们的地址。译码电路的作用是保证对于一种地址信息 32 个行只有一个为有效电位。同样 32 列中也只有一个为有效电位。因此，对应于一种地址信息只能选中一个单元。1024 字 $\times$ 1 位的 RAM 若用来组成 1024×8 的存贮器就要用 8 片，把 8 片并行起来。8 个 RAM 片对应的同一地址 (x_0y_0) 组成一个字节，每一个芯片对应的 (x_0y_0) 单元中的信息即

为这字节中的一位。

如果存贮量较小，也可以把 RAM 芯片的单元阵列直接组成需要位数的字形式，例如 $32 \text{ 字} \times 8 \text{ 位} = 256 \text{ 位}$ ， $128 \text{ 字} \times 8 \text{ 位} = 1024 \text{ 位}$ 等等。这时 x_i 值代表一个字，而相应每一个 y_i 为这一字中的一位。因此，现在常把阵列中的 X 线称为字线，Y 线称为位线。

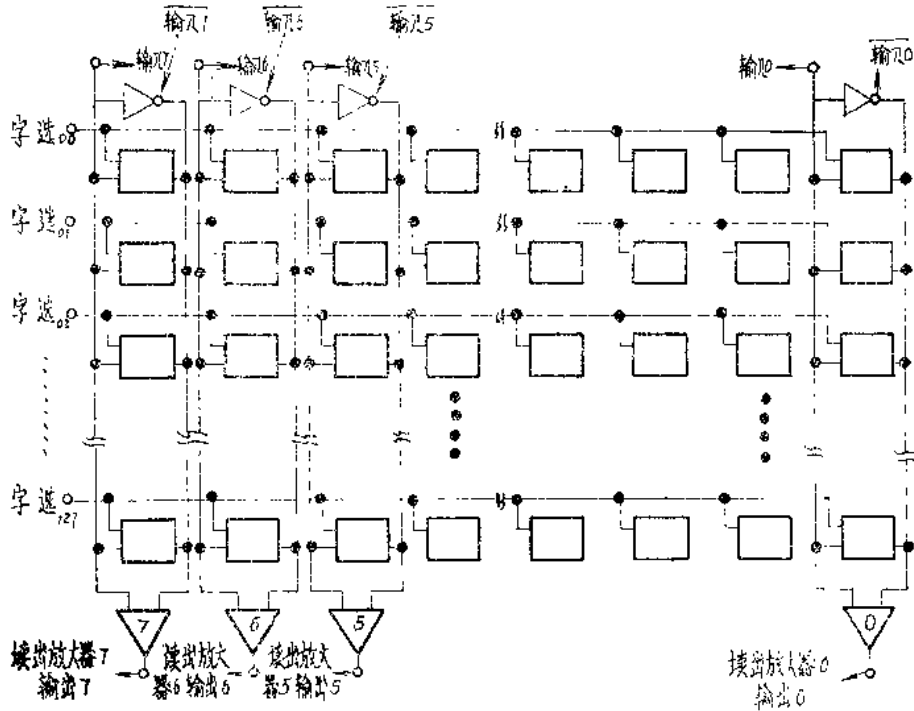


图 6.3 1024 存储单元排列成 128×8 RAM

(图 6.3) 展示了一个由 1024 个单元组成 128×8 RAM。每一个方块代表一个存储单元，能存一位二进制信息。字线画在左边的输入端，只画了 4 个，实际是 128 个，每个代表一个字。字选线 00 连到 8 个存储单元的每一个。这 8 个存储单元可以组成一个字节（8 位二进制）我们称它为 0000H 单元。

输入线在图的顶部，为简化计，8 条线中只画了 4 条。输入线经过反相得“输入非”。图上虽没画出，实际上输入线接三态门，所以不操作时，它们是脱开（“浮空”）的。

输出线画在图的底部，当写操作时，它们被禁止，读操作时，被选中，在每个输出线上均有一个读出放大器。

(图 6.3) 并未将 RAM 画全。仅仅画了存储矩阵及读出放大器。使 RAM 能正常工作尚需要其他附加电路，其中之一就是地址译码器。

存储器阵列是按 128 字节排列的。一个地址译码器能从 128×8 的存储单元中选出任一个单元。因此，要求有 128 选一的译码器，从 CPU 向译码器输入 7 根地址线。我们知道 7 位编码能确定出 128 个不同的地址。

地址译码器由 128 个 7 输入端的与门组成。(图 6.4) 只画了三个与门，第一、二

及最后一个，由图可见当地址为 0000000 时， $\bar{A}_6 \sim \bar{A}_0$ 全高。字线 00 为高，字选 00 被选中。当地为 0000001 时，字线 01 选中。当 $A_6 \sim A_0$ 全为高时，字线 127 被选中。

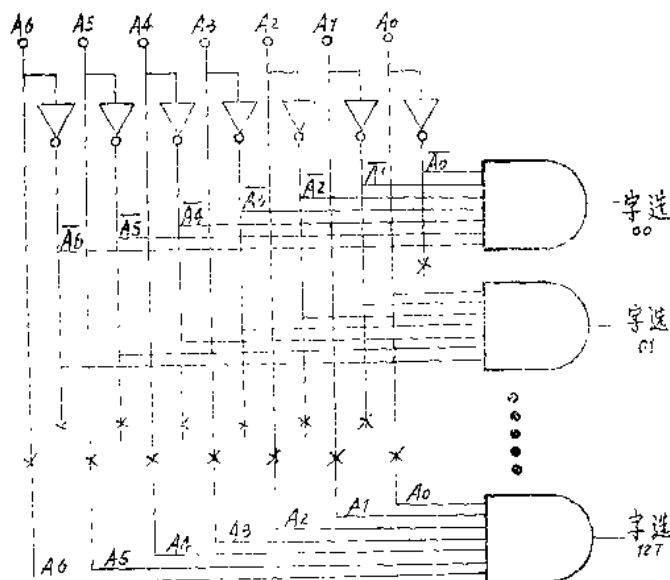


图 6.4 128 选 1 地址译码器

将 (图 6.3) 和 (6.4) 画成框图形式，再加上输入和输出三态缓冲器，就成为如 (图 6.5) 所示的 128×8 RAM 的电路框图。

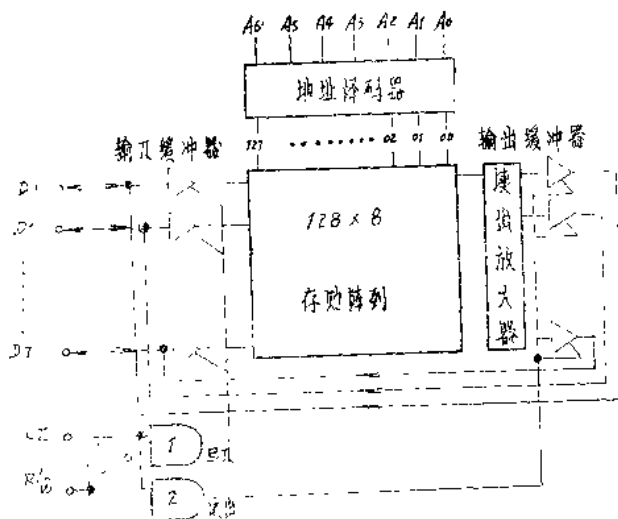


图 6.5 有双向数据总线 128×8 RAM

(图 6.5) 和 (图 6.2) 有两个显著差别：

1. (图 6.2) 输入和输出线是分开的，而 (图 6.5) 输入和输出线是公用的。一般说来，RAM 的数据线有两种形式，有单独的输入线 D_{IN} 和输出线 D_{OUT} 如 (图 6.2)；

有双向的数据线 (I/O)：该线既作数据输入线用，也作数据输出线用，这是可以实现的，因为 CPU 对 RAM 永远不会在同一时刻进行读出和写入操作。双向数据线结构可以减少引脚数。

(图 6.5) 左边 $D_0 \sim D_7$ (I/01~I/08) 是 8 根双向数据线，依靠在写入线上加高电平使三态输入缓冲器选通。该写入线由 $R/\bar{W}$ 线及片选 CE 线所控制。当 CPU 要向存储器写数据时，写入线为高，输入缓冲器选通，允许数据写入被选中的地址单元中去；此时，该读出线为低电平，输出缓冲器被禁止。

由 (图 6.5) 可见，读出和写入线受片选线 CE 及 $R/\bar{W}$ 线所控制 ($R/\bar{W}$ 接 CPU 的 $\bar{WR}$ 端， CE 接地址译码输出)。当 CE 输入高电平时 (正片选) 这块存储器被选中。如果 CE 为低电平，输入和输出缓冲器均被禁止。结果这块芯片与数据总线脱开。

当然，当 CE 为高，则由 $R/\bar{W}$ 线控制读出及写入信号。该 $R/\bar{W}$ 线是连到 CPU 的 $\bar{WR}$ 线上。当 CPU 将数据写入 RAM 时 $\bar{WR}$ 线为低电平。使 $R/\bar{W}$ 为低，使门 1 输出为高，即写入线为高，输入缓冲器选通。数据写入所选中的地址单元，当 CPU 将数据从 RAM 读出时， $\bar{WR}$ 线为高电平，门 2 输出变高电平，使输出缓冲器选通。而门 1 输出保持低，使输入缓冲器禁止。RAM 处于读出状态。

有些 RAM 只有一根单独的片选线。有些 RAM 则有几根片选线 (CE 或 CS)。有一些 RAM 芯片，片选线标为 $\bar{CE}$ 。其意是当片选线为低电平时，该片被选中。

2. (图 6.5) 地址译码是采用单级译码结构，或称字结构，即字线选择某个字的所有位。因为有 128 个字，故地址译码器输入线共七根 $A_6, A_5, A_4, A_3, A_2, A_1, A_0$ 。译码器的输出可以给出 $2^7 = 128$ 个状态。分别控制 128 根字选择线。对于每一种地址信号就选中其中相应的一根线。(图 6.2) 地址译码是采用双级译码结构。即分 x 向译码器和 y 向译码器。 x 向译码器的输入线五根，即 A_4, A_3, A_2, A_1, A_0 ； y 向译码器的输入线也五根即 A_9, A_8, A_7, A_6, A_5 。

采用双级译码，可以减少选择线的数目，因为地址译码器分成两个，若每一个译码器有 $n/2$ 个输入端，它可以有 $2^{n/2}$ 个输出状态，两个地址译码器共有 $2^{n/2} \times 2^{n/2} = 2^n$ 个输出状态。而译码器的输出线却只有 $2^{n/2} + 2^{n/2} = 2 \times 2^{n/2}$ 根，若以 (图 6.2) 为例， $n = 10$ ，则双译码器的输出状态为 $2^{10} = 1024$ 个状态，而译码线却只有 $2 \times 2^5 = 64$ 根。如果采用单级译码结构就要有 1024 根选择线。

关于存储单元阵列，是由存储单元电路组成的，存储单元电路是存储器的核心部分。一个 1024 位的 MOS RAM 芯片，就有 1024 个重复的单元电路。所以存储单元电路的面积、功耗以及稳定性对整个 MOS RAM 器件的性能有决定性的影响。MOS RAM 性能的不断f提高就是伴随着单元电路的不断改革而出现的。静态 RAM 的存储单元是六管的反相器交叉耦合而成的触发器组成的。利用触发器的两种稳态来表示存入信息 0 或 1。只要不断电，信息就永远存在。关于静态存储单元请参考计算机原理书。此处不再赘述。

二、动态 RAM 的组成

为了提高存储容量和降低功耗，要减少存储单元所占的面积，减小存储单元所占的

4

;

真 值 表

Ce	$R/\overline{W}$	D_{IN}	D_{OUT}	MODE
H	X	X	高阻	不选中
L	L	L	L	写
L	L	H	H	写
L	H	X	D_{OUT}	读

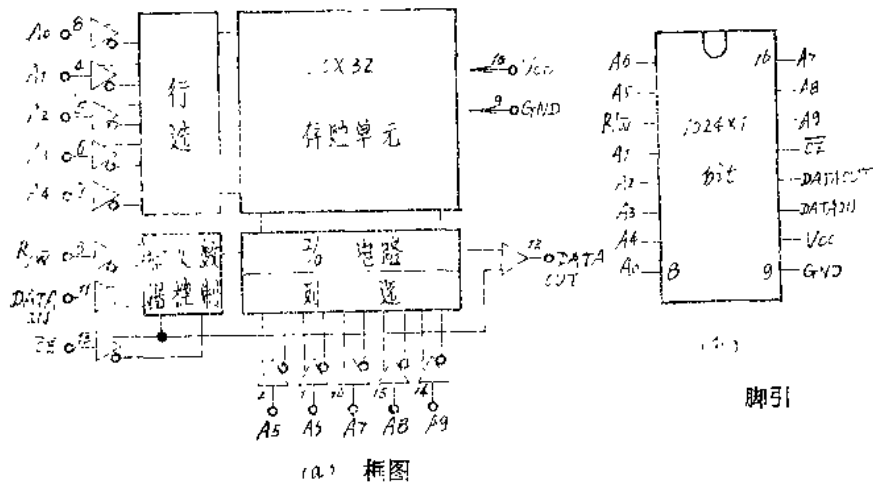


图 6.7 (2)

有效建立的时间称为读取时间 t_A 。而片选信号可以晚一些建立（不是必要的，但是一般系统中也是经常的）。片选信号建立后，数据输出的有效建立最晚不超过 t_{CO} 的时间。（当然此时还必须满足 t_A 的要求）一般在系统中，片选信号相对地址稳定的延迟不是很大。因而读取时间主要取决于 t_A 。

数据保持时间 t_{OH1} 及 t_{OH2} 说明在地址信号及片选信号撤销后, 输出数据能够保持的时间, 在系统设计中有时可以利用这个时间。

读周期与读取时间是两个不同的概念。读周期表示二次连续操作的必须间隔的时间，它一般要大于读取时间。

存贮器芯片的写操作是在地址稳定与片选信号有效后,在维持写脉冲($R/\bar{W}$ 控制线为低电平)的最小宽度 t_{WP} 条件下(在 t_{WP} 时间内,写入数据是稳定的)才能可靠的写入。所以数据表中给出 t_{WP} 的最小宽度。数据稳定建立时间 t_{WP} 是以写脉冲的后沿往前计算的。

片选信号的建立时间 t_{cw} 也是同样的，写脉冲的前沿必须比地址稳定建立时间晚 t_{AW} ，（用作内部地址译码时间，以保证不破坏不该写入的地址单元）。

存贮器芯片的读周期也可以是不同的，如有些芯片内部电路需要写的恢复时间。

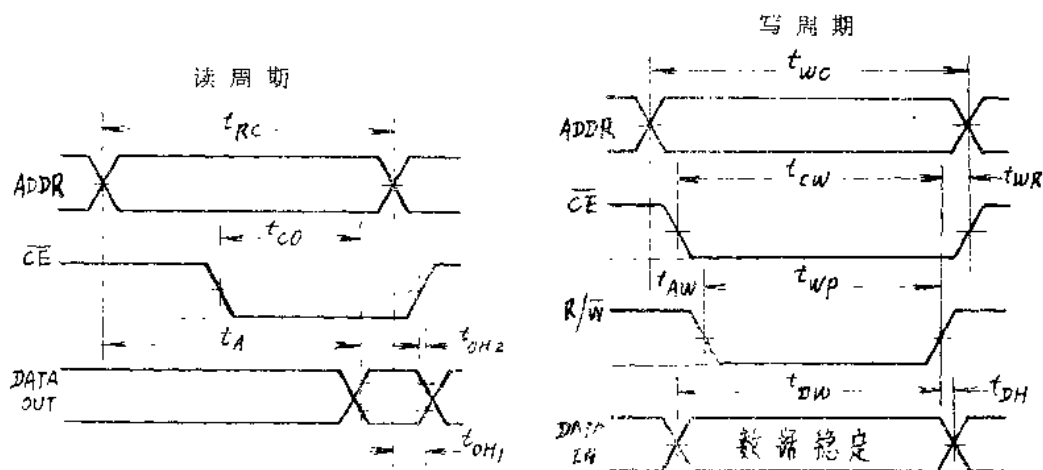


图 6.8 8102A 读写周期波形图

(二) (图 6.9) 是 INS 8111A—4 (256×4 位) MOS 静态 RAM 的功能框图, 管脚引线。

从框图看出, 这种 RAM 芯片有两个片选端 $\overline{CE}_1$ 、 $\overline{CE}_2$, (低电平有效) 和输出允许 OD 端。

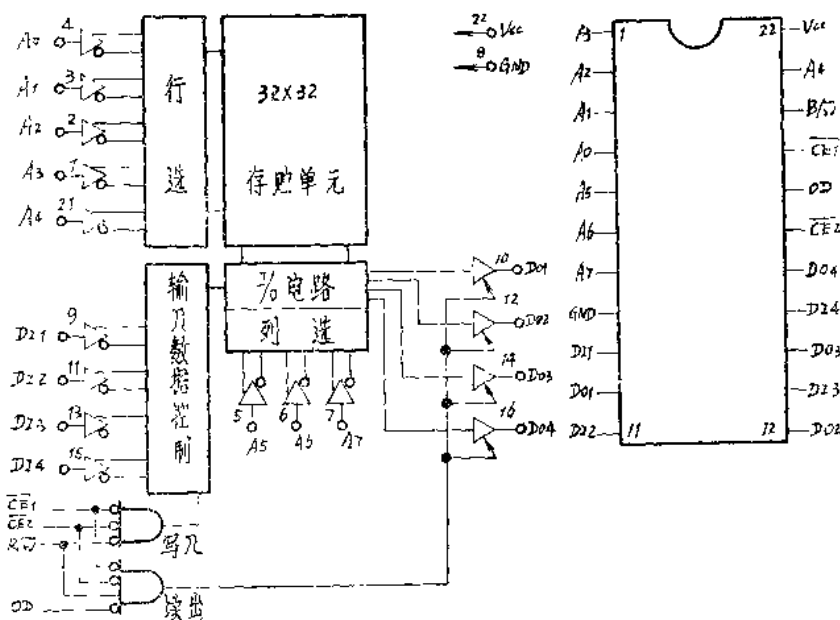
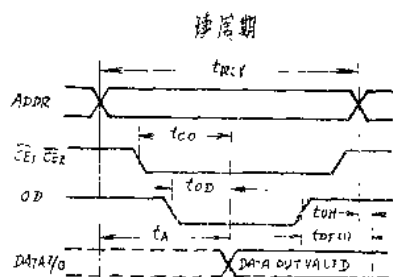


图 6.9

(图 6.10) 是 INS 8111A—4 的读写周期操作波形图。INS8111A—4 读周期数据的读出, 必须在地址稳定, 片选 $\overline{CE}_1$ 及 $\overline{CE}_2$ 同时有效、(R/W 为高电平) 输出允许 OD 为低电平后, 经过一段 t_{on} 时间, 数据才有效。写操作在地址稳定同时 $\overline{CE}_1$ 及 $\overline{CE}_2$ 有效,



(三) (图 6.11) 是 Intel 2114 4K (1024×4) MOS 静态 RAM 的功能框图^①和管脚引线。从框图看出这种 RAM 只有一个片选端 $\overline{CS}$ ，一个读/写控制端 $R/\overline{W}$ ，或标以 $\overline{WE}$ 。数据输入和输出合在一起。这称作“双向”的。

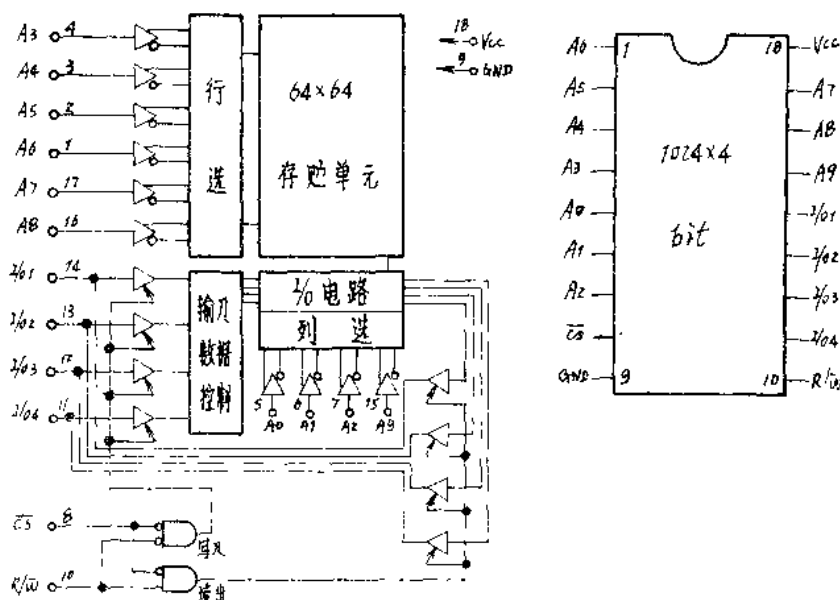


图 6.11

由(图 6.12) 读周期的波形图可见, 该周期从地址有效开始, 由于地址信息在 2114 RAM 中没有被锁存, 故在 t_{RC} 期间地址线上的信息必须保持稳定。地址信息出现以后, 紧跟着要出现片选信号 CS, 该信号是低电平有效。片选信号有效再经过 t_{CO} 时间, 被寻址的 RAM 单元的信息送上数据总线 D_{OUT} , 称为数据有效。在图中没有标出 $R/\overline{W}$ 信号, 因为这种芯片只有一个读写控制端, 当 $R/\overline{W} = 1$ 即表示读出周期, 数据被读出

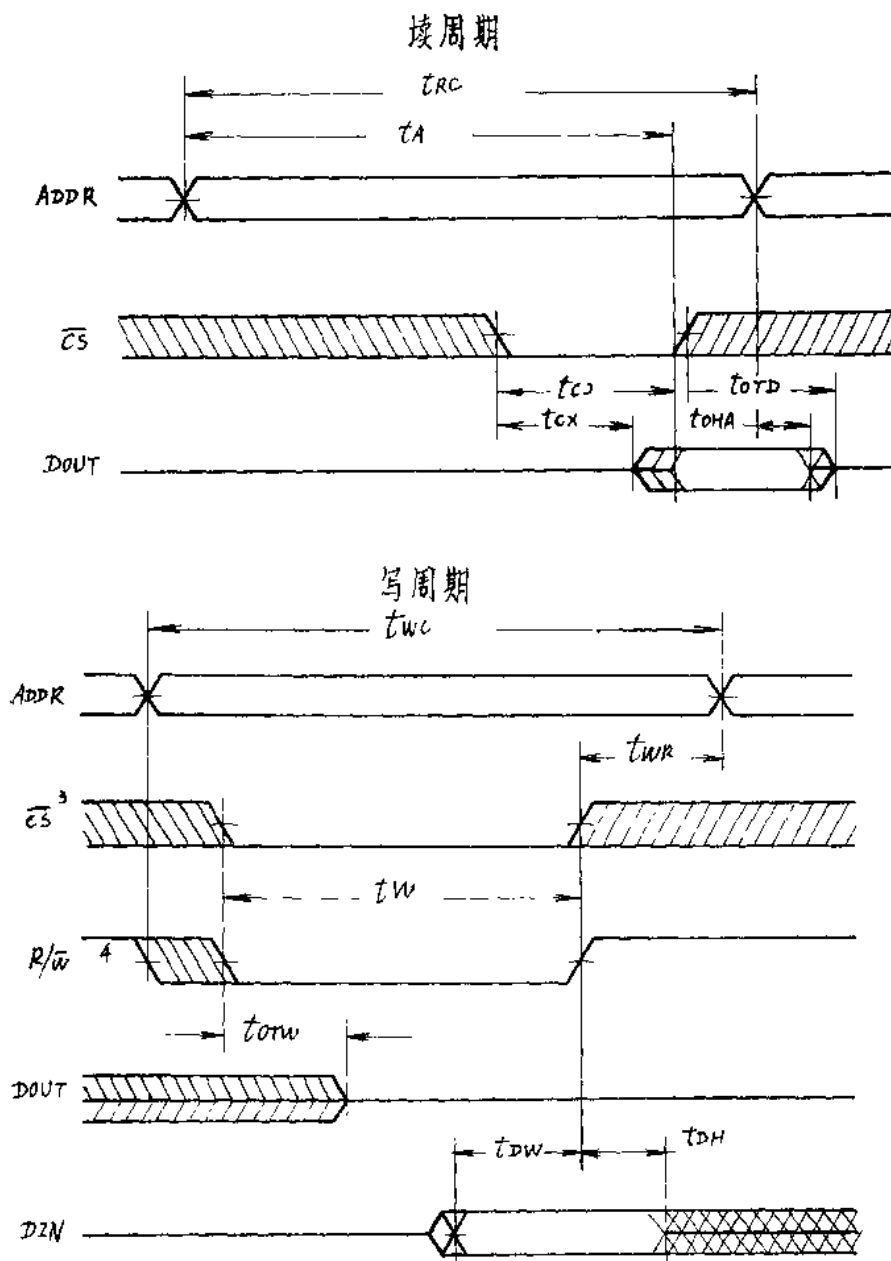


图 6.12

后一直保持在总线上，直至 $\overline{CS}$ 信号撤去，并且在 $\overline{CS}$ 信号失效后还有一段数据保持有效时间 t_{OD} 。（表 6-1）给出读周期的参数；写周期从地址有效开始，在写周期 t_{WC} 期间地址信息必须保持稳定，因为没有地址锁存器。地址稳定后片选 $\overline{CS}$ 有效，在整个 $\overline{CS}$ 有效期间， $R/\overline{W}$ 必须有效。 $R/\overline{W}$ 信号也允许与地址信息同时有效，因为 $R/\overline{W}$ 在进入芯片内部时，还有一段延时，这保证地址信息在数据写入之前就已经稳定了。在数据总线上被写入的数据，在 $R/\overline{W}$ 变高（失效）之前，必须保持 t_{OW} 稳定时间；在地址总线上的地址信

表 6-1 读周期参数表

符 号	参 数	2114A-1		2114-2		2114-3		2114		单 位
		min	max	min	max	min	max	min	max	
t_{RC}	读周期时间	100		200		300		450		ns
t_A	取数时间		100		200		300		450	ns
t_{CO}	片选有效到 输出出现		70		70		100		120	ns
t_{CX}	片选有效到 输出有效	10		0		0		0		ns
t_{ORD}	片选断开到 输出变为三态		30		40		80		100	ns
t_{OHA}	地址改变后输出 的保持时间	15		10		10		10		ns

表 6-2 写周期参数

符 号	参 数	2114A-1		2114-2		2114-3		2114		单 位
		min	max	min	max	min	max	min	max	
t_{WC}	写周期时间	100		200		300		450		ns
t_W	写时间	75		100		150		200		ns
t_{WR}	写恢复时间	0		20		0		0		ns
t_{OTW}	从写信号有效到 输出三态的时间		30		40		80		100	ns
t_{PDW}	写信号失效之前 数据有效时间	70		100		100		200		ns
t_{PH}	写信号无效后数 据保留时间	0		0		0		0		ns

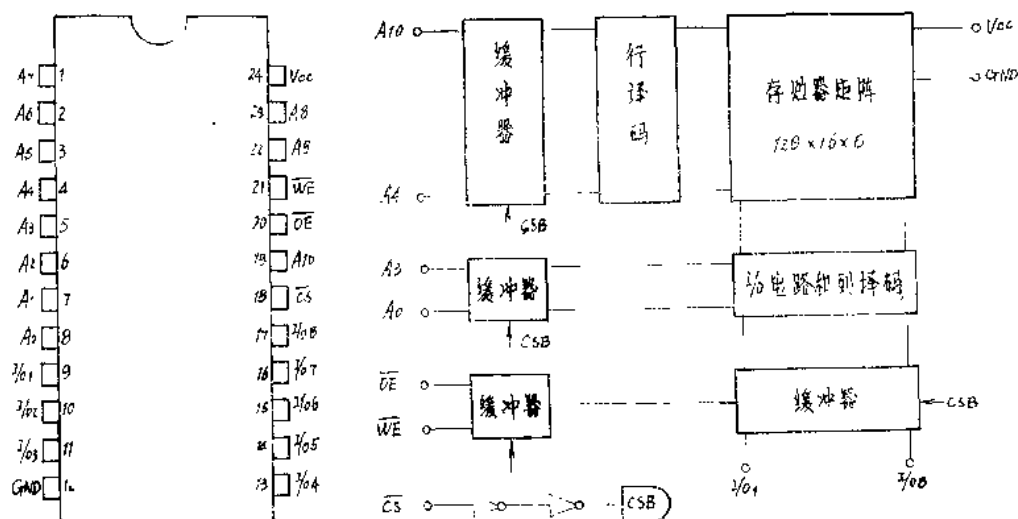


图 6.13

息，在 R/W 变高之后，必须保持 t_{WR} 时间。（表 6-2）给出写周期的参数。了解读/写周期参数，对于系统设计是十分重要的。在系统设计时必须保证芯片的读写速度与 CPU 的速度相匹配。

（四）（图 6.13）是 MB8416（2048×8）CMOS 静态 RAM 管脚引线和功能框图。

（图 6.14）是 MB8416 的读周期波形图。（图 6.15）是 MB8416 的写周期波形图。

（五）（图 6.16）是 MB8118（16, 384×1）动态 RAM 的功能框图和管脚引线图。

表 6-3 真值表

CS	OE	WE	方 式	I/O PIN
H	X	X	未 选 中	高 阻
L	H	H	输 出 无 效	高 阻
L	L	H	读	D_{OUT}
L	X	L	写	D_{IN}

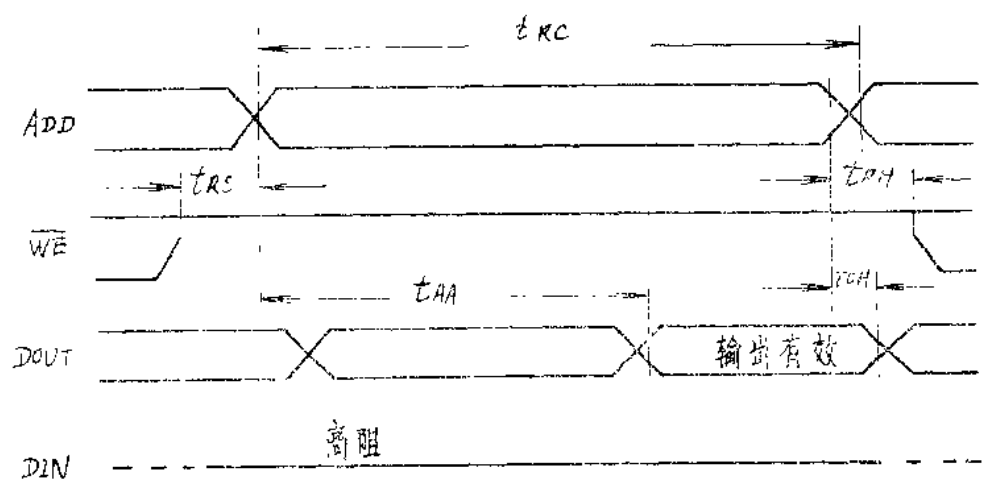


图 6.14

8118 是 16K×1 的动态 RAM，它具有能寻找 16,384 个地址的能力，按说芯片应有 14 根引线作为地址引线。但从管脚图看，只有 $A_6 \sim A_0$ 七根地址引脚，所以它的地址必须分时传送，即将 14 位地址信息分成行地址，（ $A_6 \sim A_0$ ），和列地址（ $A_{13} \sim A_7$ ）。用行选通信号 $\overline{RAS}$ ，和列选通信号 $\overline{CAS}$ 分别控制行地址和列地址的接收。当然在 8118 内部设行地址锁存器和列地址锁存器。当 $\overline{RAS}$ 有效（低电平）时，芯片接收 $A_6 \sim A_0$ 七位行地址，将其锁存入行地址锁存器；其后 $\overline{CAS}$ 变为有效，芯片又接收 $A_{13} \sim A_7$ 七位列地址，也将其锁存入列地址锁存器。这种结构给存贮器刷新带来方便；刷新时每次是

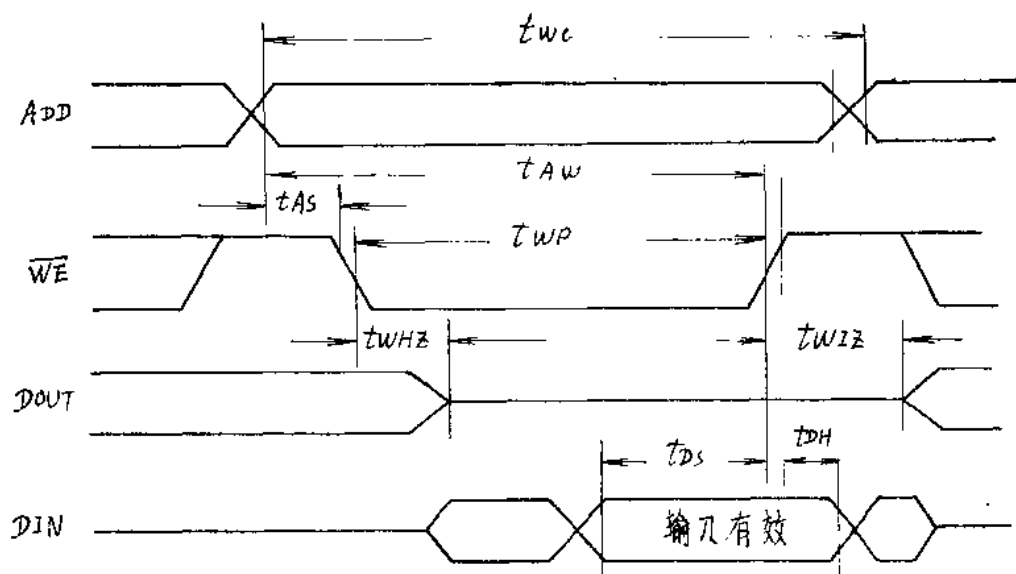


图 6.15

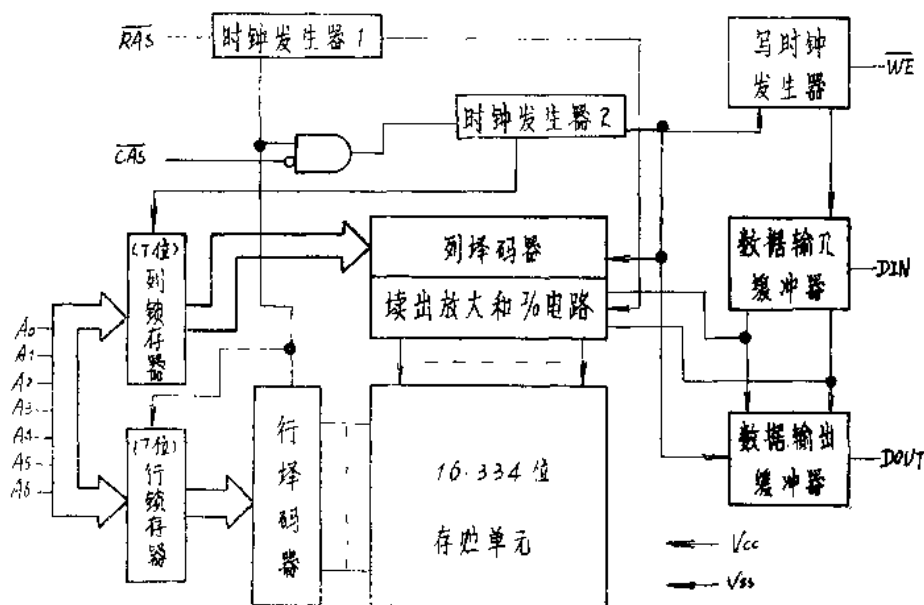


图 6.16 框图

刷新一整行。对于 16,384 单元的动态 RAM 来说, 每次刷新 128 位。所以刷新时只是 RAS 有效, 由 $A_6 \sim A_0$ 地址总线提供刷新地址。

由 (图 6.16) 功能框图可见, 行锁存器内的行地址信息送入行译码器, 每一种信息, 译码器 128 个输出端其中之一有效, 即选中其对应的 128 行中之一行, 每一行有 128 个单元, 它们被选通到读出放大器去。在那里每一个存储单元的逻辑电平都被鉴

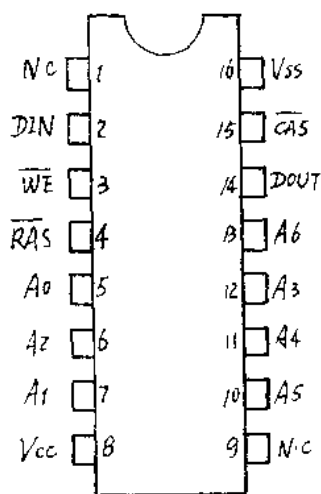


图 6.16 管脚

别，锁存和重写。但是列锁存器的信息也送入列译码器去译码，它的 128 个输出端只有一个有效，因此，只选通 128 个放大器的一个，把其信息送至输出缓冲器。

芯片的数据输入引脚和输出引脚是分开的，并各有各的缓冲器。此外 8118 只有一个控制信号端 $\overline{WE}$ ，没有设片选端 $\overline{CS}$ （或 $\overline{CE}$ ）。一般是另加外部逻辑电路，使用起片选作用的信号去控制 $\overline{RAS}$ 和 $\overline{CAS}$ 的产生，以代替 $\overline{CS}$ （或 $\overline{CE}$ ）的作用。详见 CPU 与动态 RAM 的连接。

下面分别介绍 8118 的读、写和刷新周期。

1. 8118 的读周期

8118 读周期的时序如（图 6.17）所示。周期从 $\overline{RAS}$ 脉冲变低开始。但地址必须在 $\overline{RAS}$ 变低前有效。

若 $\overline{RAS}$ 的冲脉宽度大于 t_{RAS} ，则可以在 $\overline{RAS}$ 有效后，经过时间 t_{RAC} ，使读出数据有效。但这必须要保证从 $\overline{CAS}$ 有效到输出数据有效之间的时间大于 t_{CAC} 。也即 $\overline{CAS}$ 必须在 $\overline{RAS}$ 自有效起至 $(t_{RAC} - t_{CAC})$ 时间前变为有效。而自 $\overline{RAS}$ 有效起经过时间 t_{RCL} ， $\overline{CAS}$ 即可变为有效，故两者之间为 $\overline{CAS}$ 的时间窗。只要 $\overline{CAS}$ 在这段时间内有效，就不影响读出数据变为有效的时间。读出数据的有效时间取决于 t_{RAC} 。否则，读出数据的有效时间，就要取决于 $\overline{CAS}$ 有效加上 t_{CAC} 。

行地址必须在 $\overline{RAS}$ 有效前 t_{ASR} 有效， $\overline{RAS}$ 有效后应保持时间 t_{RAH} ，这样才能保证行地址与列地址能正确选通到相应的锁存器，且有足够的时间译码。

整个的读周期时间为 t_{RC} 。

由于片中有地址锁存器，故在所要求的列地址保持时间以后，在读、写周期完成以前外界的地址总线可以改变，这与 2114 静态 RAM 是不同的。

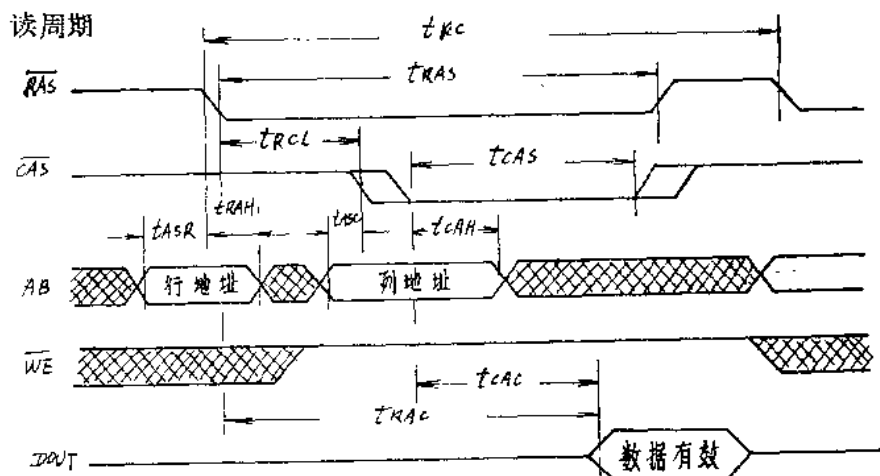


图 6.17