

第5章

半导体存储器

存储器是计算机硬件系统的基本组成部分,用于存储计算机工作所必需的数据和程序,分为内存储器和外存储器。本章要求在了解半导体存储器工作原理的基础上,着重掌握微机或微机应用系统内存储器的构成及与 CPU 的连接方法。

5.1 概 述

存储器是计算机系统记忆设备,用来存放程序和数据。随着计算机的发展,存储器在系统中的地位越来越重要,从某种意义上讲,存储器的性能已经成为计算机系统的核心。

5.1.1 存储器的分类

构成存储器的存储介质,目前主要是半导体器件和磁性材料。一个双稳态半导体电路、一个 CMOS 晶体管或磁性材料的存储元,均可以存储一位二进制代码。这个二进制代码位是存储器中最小的存储单位,称为一个**存储位**或**存储元**。由若干个存储元组成一个**存储单元**,多个存储单元组成一个**存储器**。

存储器的种类繁多,根据存储器的存储介质的性能及使用方法的的不同,可以从不同的角度对存储器进行分类。

1. 按存储介质分类

存储介质是指能寄存“0”、“1”两种代码并能区别两种状态的物质或元器件。存储介质主要有半导体器件、磁性材料和光盘等。

1) 半导体存储器

存储元件由半导体器件组成的叫半导体存储器。现代半导体存储器都用超大规模集成电路工艺制成芯片,其优点是体积小、功耗低、存取时间短。其缺点是当切断电源时,所存信息也随即丢失,它是一种**易失性**存储器。近年来已研制出用非挥发性材料制成的半导体存储器,克服了信息易失的弊病。

半导体存储器又可按其材料的不同,分为双极型(TTL)半导体存储器和 MOS 半导体存储器两种。前者存取速度快,后者集成度高,并且制造简单、成本低廉、功耗小。当前, MOS 半导体存储器已被广泛应用。

2) 磁表面存储器

磁表面存储器是在金属或塑料基体的表面上涂一层磁性材料作为记录介质,工作时磁层随磁体高速运转,用磁头在磁层上进行读/写操作,故称为磁表面存储器。按载磁体形状的不同,可分为磁盘、磁带等。由于采用具有矩形磁滞回线特性的材料作磁表面物质,它们按其剩磁状态的不同而区分“0”或“1”,而且剩磁状态不会轻易丢失,故这类存储器具有**非易失性**的特点。

3) 光盘存储器

光盘存储器是应用激光在记录介质(磁光材料)上进行读/写的存储器,具有非易失性的特点。由于光盘记录密度高、耐用性好、可靠性高和可互换性强等特点,光盘存储器越来越被广泛应用于计算机系统。

2. 按存取方式分类

按存取方式可把存储器分为随机存储器、只读存储器、顺序存储器和直接存取存储器 4 类。

1) 随机存储器

随机存储器(Random Access Memory, RAM)是一种可读/写存储器,其特点是存储器的任何一个存储单元的内容都可以随机存取,而且存取时间与存储单元的物理位置无关。计算机系统的主存都采用这种随机存储器。按照存储信息原理的不同,随机存储器又分为**静态随机存储器(Static RAM, SRAM)**和**动态随机存储器(Dynamic RAM, DRAM)**。

2) 只读存储器

只读存储器(Read Only Memory, ROM)是只能对其存储的内容读出,而不能对其重新写入的存储器。这种存储器一旦存入了原始信息后,在程序执行过程中,只能将内部信息读出,而不能随意重新写入新的信息。因此,通常用它存放固定不变的程序、常数以及汉字字库,甚至用于操作系统的固化。

3) 串行访问存储器

如果对存储单元进行读/写操作时,需按其物理位置的先后顺序寻找地址,则这种存储器叫做**串行访问存储器**。显然,这种存储器由于信息所在位置不同,从而使得读/写时间均不相同。例如磁带存储器,不论信息处在哪个位置,读/写时必须从其介质的始端开始按顺序寻找,故这类串行访问的存储器又叫**顺序存取存储器**。还有一种属于**部分串行访问的存储器**,如磁盘。在对磁盘读/写时,首先直接指出该存储器中的某个小区域(磁道),然后再顺序寻访,直至找到位置。故其前段是直接访问,后段是串行访问。

3. 按在计算机中的作用分类

按在计算机系统中的作用不同,存储器又可分为主存储器、辅助存储器和缓冲存储器。

主存储器的主要特点是它可以和 CPU 直接交换信息,主要由半导体存储器构成,也称为内存存储器。**辅助存储器**是主存储器的后援存储器,用来存放当前暂时不用的程序和数据,它不能与 CPU 直接交换信息。二者相比,主存速度快、容量小、每位价格高;辅存速度慢、容量大、每位价格低。**缓冲存储器**用在两个速度不同的部件之中,如 CPU 与主存之间可设置一个快速缓冲存储器,用来提高 CPU 访问存储器的速度。

存储器的分类如图 5.1 所示。

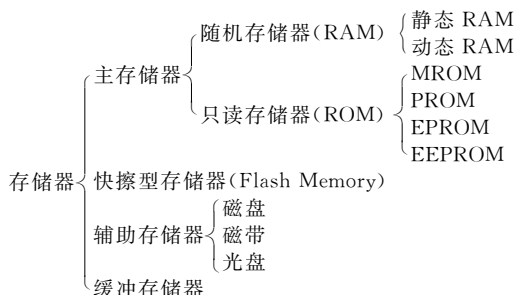


图 5.1 存储器的分类

本章主要介绍半导体存储器,即主存储器的有关内容。

5.1.2 存储器的性能指标

存储器的类型不同,其性能指标也不相同,在构成微型计算机硬件系统时需要全面考虑。通常应注意的问题有以下几点。

1. 存储器容量

在微型计算机中,存储器以字节为单元。每个单元包含8位二进制数,也就是1个字节。存储器的**容量**指的是存储器所能容纳的最大字节数。由于存储容量一般都很大,因此常以KB、MB或GB为单位。目前高档微型计算机的内存容量一般为32MB~4GB。存储器容量越大,存储的信息量也就越大,计算机运行的速度也就越快。

存储1位二进制信息的单元称为1个**存储元**。对于32MB的存储器,其内部有 $32\text{M} \times 8$ 个存储元。存储器芯片多为 $\times 8$ 结构,称为字节单元。也有 $\times 1$ 、 $\times 4$ 结构的存储器芯片,例如,Intel 2116为1位,2114为4位,6264为8位。这样的芯片应互相拼接成字节单元才能使用。在标定存储器容量时,经常同时标出存储单元的数目和每个存储单元包含的位数:

$$\text{存储器芯片容量} = \text{单元数} \times \text{位数}$$

例如,Intel 2114芯片容量为 $1\text{K} \times 4$ 位,6264为 $8\text{K} \times 8$ 位。

虽然微型计算机的字长已经达到16位、32位甚至64位,但其内存仍以一个字为一个单元,不过在这种微型机中,一次可同时对2、4或8个单元进行访问。

2. 存取周期

很多类型的存储器的读/写操作不能截然分开,比如,有的在读操作后要进行读后重写,有的在写操作前要先进行读操作。存储器的**存取周期**是指从接收到地址,到实现一次完整的读出或写入数据的时间,是存储器进行连续读或写操作所允许的最短时间间隔。计算机的运行速度与存储器的存取周期有着直接的关系,因此它是存储器的一项重要参数。一般情况下,存取周期越短,计算机运行的速度越快。半导体双极型存储器的存取周期一般为几至几百纳秒,MOS型存储器的存取周期一般为十几至几百纳秒,例如,常用的HM62256($32\text{K} \times 8$)的存取周期为120~200ns。

一个存储器系统的存取周期不仅与存储器芯片的存取周期有关,而且还与存取路径中的缓冲器以及地址/数据线的延时有关,往往是三者之和。目前SDRAM内存条的存取周期PC-100为8~12ns,PC-133为7ns,由此可计算出它们的最高工作频率分别为83~125MHz和143MHz。

3. 功耗

半导体存储器属于大规模集成电路,集成度高、体积小,但是不易散热,因此在保证速度的前提下应尽量减少功耗。一般而言,MOS型存储器的功耗小于相同容量的双极型存储器。例如,上述HM62256的功耗为40~200mW。

4. 可靠性

可靠性是指存储器对电磁场、温度变化等因素造成干扰的抵抗能力(电磁兼容性),以及在高速使用时也能正确地存取(动态可靠性)。半导体存储器采用大规模集成电路工艺制造,内部连线少、体积小、易于采取保护措施。与相同容量的其他类型存储器相比,半导体存储器抗干扰能力强。

5. 集成度

存储器由若干存储器芯片组成。存储器芯片的集成度越高,构成相同容量的存储器的芯片数就越少。半导体存储器的**集成度**是指在一块数平方毫米芯片上所制作的基本存储单元数,常以“位/片”表示,也可以用“字节/片”表示,MOS型存储器的集成度高于双极型存储器,动态存储器的集成度高于静态存储器,这也是动态存储器普遍用做微型计算机主存储器的原因。

6. 其他

其他还应考虑输入、输出电平是否与外电路兼容,对 CPU 总线负载能力的要求,使用是否方便灵活以及成本价格等。

5.2 随机读/写存储器

目前广泛使用的半导体存储器是 MOS 型半导体存储器。根据存储信息的原理不同,又分为静态 MOS 存储器和动态 MOS 存储器。半导体存储器的优点是存取速度快、存储体积小、可靠性高、价格低廉;缺点是断电后存储器不能保存信息。

5.2.1 静态 MOS 存储器

1. 基本存储元

基本存储元是组成存储器的基础和核心,它用来存储 1 位二进制信息。静态存储器的基本存储元电路如图 5.2 所示,它在 MOS 型双稳态触发器的基础上增添了两个门控管。图中 $T_1 \sim T_4$ 构成双稳态触发器,两个稳定状态分别表示 1 和 0,例如,A 点为高电平,B 点为低电平,表示存 1,相反则表示存 0。 T_5 、 T_6 为门控管,当行选择线 X 为高电平时, T_5 、 T_6 管导通,A 点和 B 点分别与内部数据线 D 和 $\bar{D}$ (也称**位线**)接通。 T_7 、 T_8 也是门控管,控制该存储单元的内部数据线是否与外部数据线接通。当列选择线 Y 也为高电平时, T_7 、 T_8 管导通,内部数据线与外部数据线接通,表示该单元的数据可以读出,或者把外部数据线上的数据写入到该存储单元。

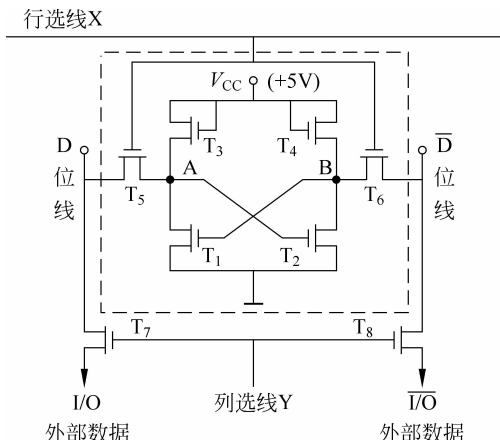


图 5.2 六管静态 RAM 存储电路

在读出时,X选择线与Y选择线均为高电平, T_5 、 T_6 、 T_7 、 T_8 管均导通,A点与D接通,B点与 $\bar{D}$ 接通, D 、 $\bar{D}$ 又与外部数据线接通。若原来存入的是1,A点为高电平,则D为高电平;B点为低电平,则 $\bar{D}$ 为低电平,二者分别通过 T_7 、 T_8 管输出到外部数据线,即读出1。相反,若A点为低电平,则D为低电平;B点为高电平,则 $\bar{D}$ 为高电平,二者分别通过 T_7 、 T_8 管输出到外部数据线,即读出0。读出信息时,双稳态触发器的状态不受影响,故为非破坏性读出。

在写入时,首先将要写入的数据送到外部数据线上。若该单元被选中,则X选择线与Y选择线为高电平, T_5 、 T_6 、 T_7 、 T_8 管均导通,外部数据线上的数据就分别通过 T_7 、 T_5 管和 T_8 、 T_6 管送到触发器的A点与B点。若写入的是1,则 T_2 导通,B点为低电平, T_1 截止,A点为高电平,写入结束,状态保持。若写入的是0,则状态相反,A点为低电平,B点为高电平。但如果电源掉电后又恢复供电,则双稳态触发器发生状态竞争,即掉电前写入的信息不复存在,因此SRAM被称为易失性存储器。

2. SRAM 的组成

静态读/写存储器的结构图如图5.3所示,存储体是一个由 $64 \times 64 = 4096$ 个六管静态存储电路组成的存储矩阵。在存储矩阵中,X地址译码器输出端提供 $X_0 \sim X_{63}$ 共64根行选择线,而每一行选择线接在同一行中的64个存储电路的行选端,故行选择线能同时为该行64个行选端提供行选择信号。Y地址译码器输出端提供 $Y_0 \sim Y_{63}$ 共64根列选择线,而同一列中的64个存储电路共用同一位线,故由列选择线可以同时控制它们与输入/输出电路(I/O电路)连通。很显然,只有行、列均被选中的某个单元存储电路,在其X向选通门与Y向选通门同时被打开时,才能进行读出信息和写入信息的操作。

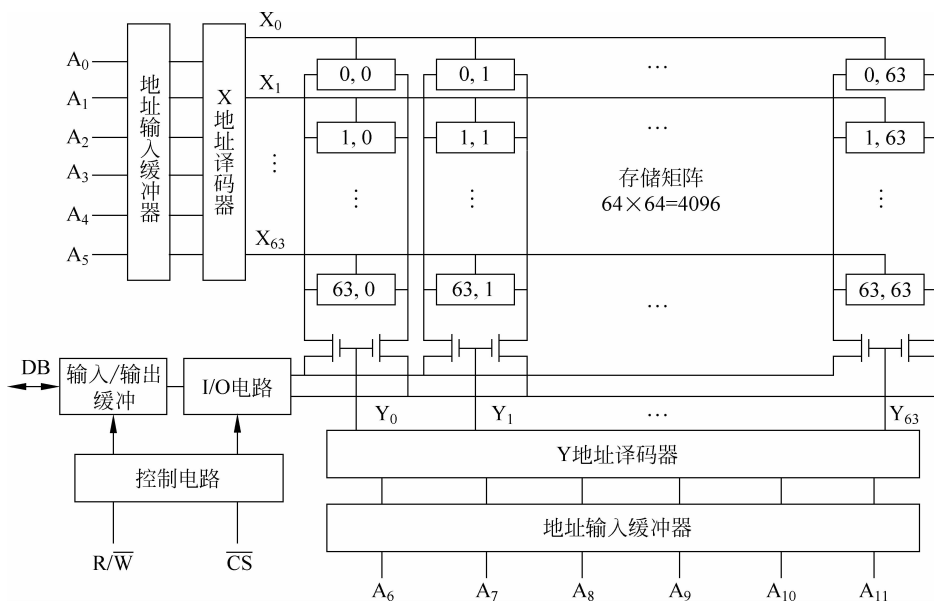


图 5.3 静态 RAM 结构组成原理图

图 5.3 中所示的存储体是容量为 $4K \times 1$ 位的存储器,因此,它仅有一个 I/O 电路。如果要组成字长为 4 位或 8 位的存储器,则每次存取时,同时应有 4 个或 8 个单元存储电路与外界交换信息,这种存储器中,将列按 4 位或 8 位分组,每根列选择线控制一组列向门同时打开;相应地,I/O 电路也应有 4 个或 8 个。每一组的同一位共用一个 I/O 电路。通常,一

个 RAM 芯片的存储容量是有限的,需要用若干片才能构成一个实用的存储器。这样,地址不同的存储单元可能处于不同的芯片中,因此,在选择地址时,应先选择其所属的芯片。对于每块芯片,都有一个片选控制端($\overline{CS}$),只有当片选端加上有效信号时,才能对该芯片进行读或写操作。一般地,片选信号由地址码的高位译码产生。

3. SRAM 的读/写过程

1) 读出过程

(1) 地址码 $A_0 \sim A_{11}$ 加到 RAM 芯片的地址输入端,经 X 与 Y 地址译码器译码,产生行选与列选信号,选中某一存储单元,该单元中存储的代码,经一定时间,出现在 I/O 电路的输入端。I/O 电路对读出的信号进行放大、整型,送至输出缓冲寄存器。缓冲寄存器一般具有三态控制功能,没有开门信号,所存数据还不能送到 DB 上。

(2) 在送上地址码的同时,还要送上读/写控制信号($R/\overline{W}$ 或 $\overline{RD}$ 、 $\overline{WR}$)和片选信号($\overline{CS}$),读出时,使 $R/\overline{W}=1$ 、 $\overline{CS}=0$,这时,输出缓冲寄存器的三态门将被打开,所存信息送至 DB 上。于是,存储单元中的信息被读出。

2) 写入过程

(1) 地址码加在 RAM 芯片的地址输入端,选中相应的存储单元,使其可以进行写操作。

(2) 将要写入的数据放在 DB 上。

(3) 加上片选信号 $\overline{CS}=0$ 及写入信号 $R/\overline{W}=0$ 。这两个有效控制信号打开三态门使 DB 上的数据进入输入电路,送到存储单元的位线上,从而写入该存储单元。

图 5.3 为静态 RAM $\times 1$ 结构的原理图,其电气特征是:只有一个 DB 及其电路。图中地址线与芯片内单元容量一一对应, $R/\overline{W}$ 为读/写控制, $\overline{CS}$ 为片选控制。

图 5.4 为静态 RAM $\times 2$ 结构的原理图,一条 Y 地址译码线控制相邻两列列选门控,它有两个 DB 及其电路。其余同 $\times 1$ 结构。 $\times 4$ 、 $\times 8$ 结构可类推。

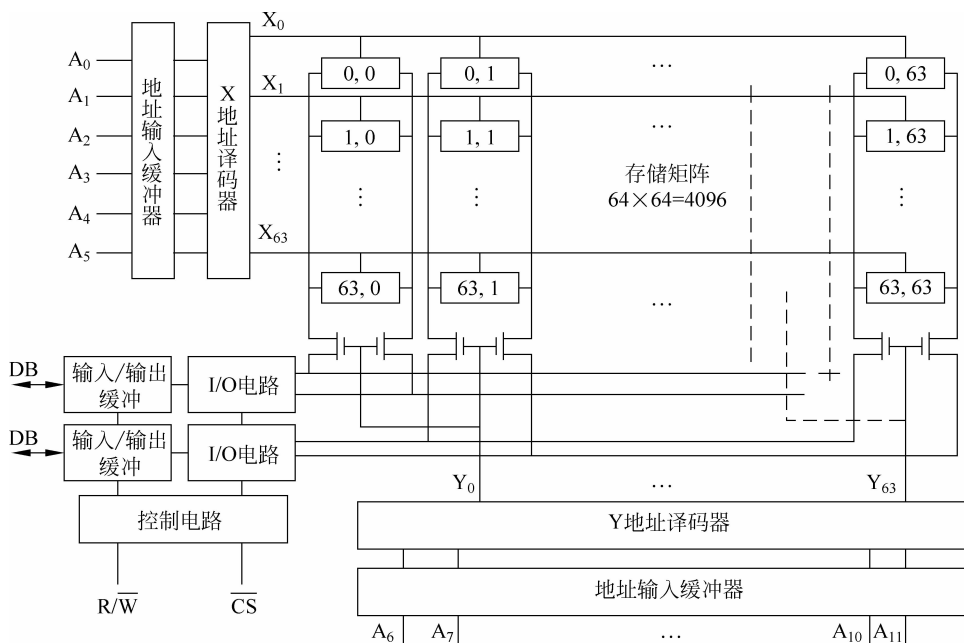


图 5.4 静态 RAM $\times 2$ 结构原理图

4. SRAM 芯片举例

常用的 SRAM 芯片有 2114、2142、6116、6264 等。

1) Intel 2114 存储器芯片

Intel 2114 是一个容量为 $1\text{K} \times 4$ 位的静态 RAM 芯片,其内部结构如图 5.5 所示,芯片的引脚图如图 5.6 所示。

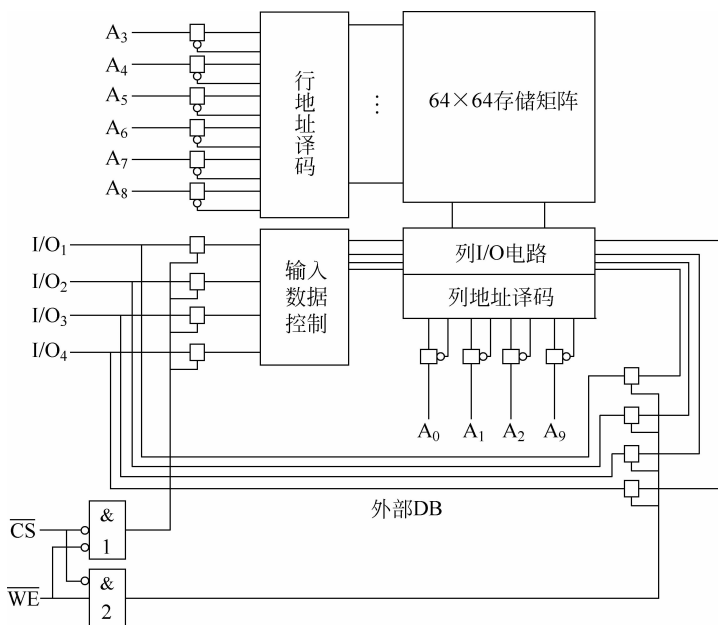


图 5.5 Intel 2114 内部结构图

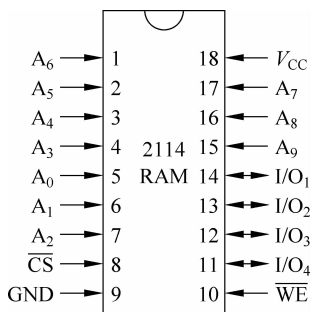


图 5.6 2114 引脚图

图 5.5 中, $A_0 \sim A_9$ 为 10 根地址线, 可寻址 $2^{10} = 1024$ 个存储单元。 $I/O_1 \sim I/O_4$ 为 4 根双向数据线。 $\overline{WE}$ 为写允许控制信号线, $\overline{WE} = 0$ 时为写入, $\overline{WE} = 1$ 时为读出。 $\overline{CS}$ 为芯片片选信号, $\overline{CS} = 0$ 时, 该芯片被选中。由于 2114 的容量为 1024×4 位, 故有 4096 个基本存储电路, 排成 64×64 的矩阵。用 $A_3 \sim A_8$ 6 根地址线作为行译码, 产生 64 根行选择线, 用 $A_0 \sim A_2$ 与 A_9 4 根地址线作为列译码, 产生 16 根列选择线, 而每根列选择线控制一组 4 位同时进行读或写操作。存储器内部有 4 路 I/O 电路以及 4 路输入/输出三态门电路, 并由 4 根双向数据线 $I/O_1 \sim I/O_4$ 与外部数据总线相连。当 $\overline{CS} = 0$ 与 $\overline{WE} = 0$ 时, 经门 1 输出线的高电平将输入数据控制线上的 4 个三态门打开, 使数据写入; 当 $\overline{CS} = 0$ 与 $\overline{WE} = 1$ 时, 经门 2 输出的高电平将输出数据控制线上的 4 个三态门打开, 使数据读出。

2) Intel 6264 存储器芯片

6264 是一种 $8\text{K} \times 8$ 的静态存储器, 其内部组成如图 5.7 所示, 主要包括 512×128 的存储器矩阵、行/列地址译码器以及数据输入/输出控制逻辑电路。地址线 13 位, 其中, $A_3 \sim A_{12}$ 用于行地址译码, $A_0 \sim A_2$ 和 A_{10} 用于列地址译码。在存储器读周期, 选中单元的 8 位数据经列 I/O 控制电路输出; 在存储器写周期, 外部 8 位数据经输入数据控制电路和列 I/O

控制电路,写入到所选中的单元中。

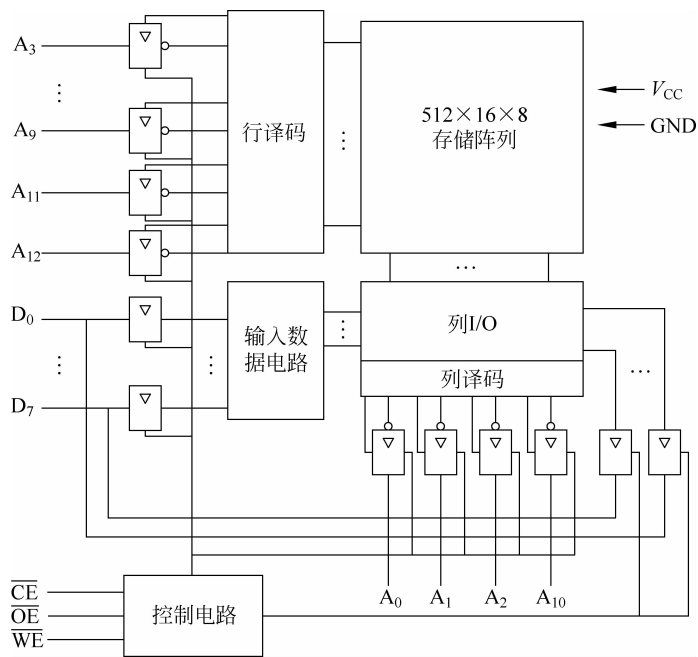


图 5.7 Intel 6264 内部结构

6264 有 28 个引脚,如图 5.8 所示,采用双列直插式结构,使用+5V 电源。

其引脚功能如下:

A₀~A₁₂: 地址线,输入,寻址范围为 8K。

D₀~D₇: 数据线,8 位,双向传送数据。

$\overline{\text{CE}}$: 片选信号,输入,低电平有效。

$\overline{\text{WE}}$: 写允许信号,输入,低电平有效,读操作时要求其无效。

$\overline{\text{OE}}$: 读允许信号,输入,低电平有效,即选中单元输出允许。

V_{CC}: +5V 电源。

GND: 接地。

NC 表示引脚未用。

6264 的工作方式如表 5.1 所示。

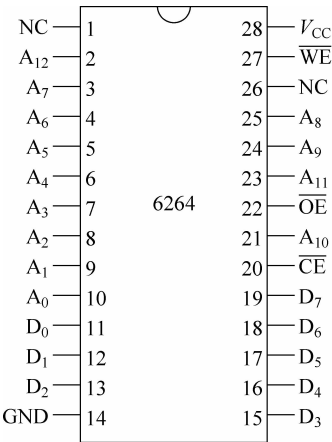


图 5.8 6264 引脚图

表 5.1 6264 的工作方式

$\overline{\text{CE}}$	$\overline{\text{WE}}$	$\overline{\text{OE}}$	方式	功能
0	0	0	禁止	不允许 $\overline{\text{WE}}$ 和 $\overline{\text{OE}}$ 为低电平
0	1	0	读出	数据读出
0	0	1	写入	数据写入
0	1	1	选通	芯片选通,输出高阻态
1	×	×	未选通	芯片未选通

5. SRAM 与 CPU 的连接

CPU 对存储器进行读/写操作,首先由地址总线给出地址信号,然后发出读操作或写操作的控制信号,最后在数据总线上进行信息交流。因此,存储器同 CPU 连接时,要完成地址线、数据线和控制线的连接。

目前生产的存储器芯片的容量是有限的,它在字数或字长方面与实际存储器的要求都有差距,所以需要进行扩充才能满足实际存储器的容量要求。通常采用位扩展法、字扩展法、字位同时扩展法。

1) 位扩展法

假定使用 $8K \times 1$ 的 RAM 存储器芯片,那么组成 $8K \times 8$ 位的存储器可采用如图 5.9 所示的位扩展法。此时只加大字长,而存储器的字数与存储器芯片字数一致。图中,每一片 RAM 是 $8K \times 1$,故其地址总线为 13 条($A_0 \sim A_{12}$),可满足整个存储体容量的要求。每一片对应于数据的 1 位(只有 1 条数据总线),故只需将它们分别接到数据总线上的相应位即可。在这种方式中,对片子没有选片要求,就是说片子按已被选中来考虑。如果片子有选片输入端($\overline{CS}$),则可将它们直接接地。在这种连接中,每一条地址总线接有 8 个负载,每一条数据总线接有一个负载。

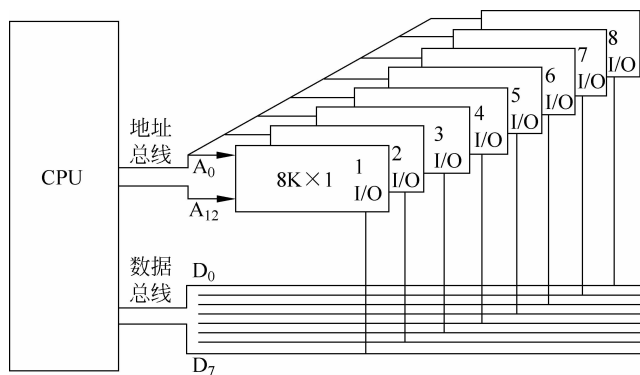


图 5.9 位扩展法组成 8K RAM

2) 字扩展法

字扩展是仅在字向扩充,而位数不变,因此将芯片的地址线、数据线、读/写控制线并联,而由片选信号来区分各片地址,故片选信号端连接到选片译码器的输出端。如图 5.10 所示

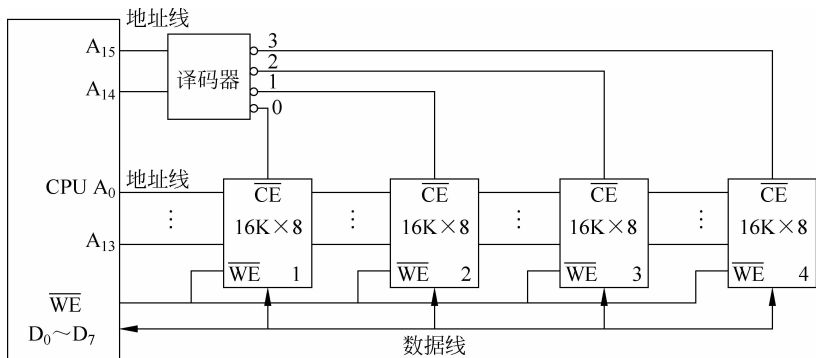


图 5.10 字扩展法组成 64K RAM

为用 $16\text{K} \times 8$ 位的芯片采用字扩展法组成 $64\text{K} \times 8$ 位的存储器连接图。图中 4 个芯片的数据端与数据总线 $D_0 \sim D_7$ 相连,地址总线低位地址 $A_0 \sim A_{13}$ 与各芯片的 14 位地址端相连,而两位高位地址 A_{14} 、 A_{15} 经译码器和 4 个片选端相连。地址空间分配表如表 5.2 所示。

表 5.2 地址空间分配表

地址 片号	片外 $A_{15} A_{14}$	片内 $A_{13} A_{12} A_{11} \cdots A_1 A_0$	说明
1	0 0	0 0 0 $\cdots$ 0 0	最低地址
	0 0	1 1 1 $\cdots$ 1 1	最高地址
2	0 1	0 0 0 $\cdots$ 0 0	最低地址
	0 1	1 1 1 $\cdots$ 1 1	最高地址
3	1 0	0 0 0 $\cdots$ 0 0	最低地址
	1 0	1 1 1 $\cdots$ 1 1	最高地址
4	1 1	0 0 0 $\cdots$ 0 0	最低地址
	1 1	1 1 1 $\cdots$ 1 1	最高地址

3) 字位同时扩展法

一个存储器的容量假定为 $M \times N$ 位,若使用 $1 \times k$ 位的芯片 ($1 < M, k < N$),则需要在字向和位向同时进行扩展。此时共需要 $(M/1) \times (N/k)$ 个存储器芯片。

4) 静态随机存取存储器的连接举例

在 64KB 地址空间中用 8 片 2114 构成 $4\text{K} \times 8$,即 4KB 存储区的全译码法连接方案:其地址范围为 $2000\text{H} \sim 2\text{FFFH}$ 。连接图如图 5.11 所示。

2114 的结构是 $1\text{K} \times 4$ 位,故可用两片 2114 按位扩充方法组成 $1\text{K} \times 8$ 位的存储器组,用 8 片可组成 4 组 $1\text{K} \times 8$ 位的存储器。 1K 芯片有 10 根地址线,可接地址总线 $A_0 \sim A_9$,每一组中的两片 2114 的数据线则分别接数据总线的高 4 位和低 4 位。

本例选用了译码器 74LS138,该芯片有 3 个片选端 G_1 、 $\overline{G_{2A}}$ 及 $\overline{G_{2B}}$,必须使 $G_1 = 1$ 、 $\overline{G_{2A}} = 0$ 及 $\overline{G_{2B}} = 0$,允许译码输出,芯片才能有效工作,否则,输出全为高电平。 A 、 B 、 C 为 3 位输入端,输出为 8 根选择线 $\overline{Y}_0 \sim \overline{Y}_7$,仅仅与输入代码对应的选择线为低电平(有效),其他的选择线为高电平。

线路中用 A_{15} 、 A_{14} 及存储器请求信号 $M/\overline{\text{IO}} = 1$ 作为 74LS138 的片选信号,如图 5.11 所接方式,译码器仅在 $A_{15} = 0$ 、 $A_{14} = 0$ 及 $M/\overline{\text{IO}} = 1$ 的情况下才能允许输出。 $M/\overline{\text{IO}}$ 参加译码控制是必要的,它使仅在访问内存时产生有效信号,保证正确地选中存储器地址,而不会与外部设备地址搞错。地址码的高 5 位 $A_{15} \sim A_{11}$ 进行译码, A_{10} 参与片选的二级译码,共同控制各芯片的片选端,以选中所寻址的某组芯片地址,这显然为全译码法。如果取 $A_{15} \sim A_{10}$ 分别为 001000、001001、001010 及 001011,经过译码器 74LS138 的 Y_4 、 Y_5 再与 A_{10} 组合控制,分别将输出的 4 个片选信号接到 4 组芯片的片选端,就可取得所需要的某组芯片地址区间,如表 5.3 所示,再由给定的低位地址 $A_9 \sim A_0$,即可选中某个地址单元。

图 5.11 中还画出了读/写信号产生电路,这里也加进了 $M/\overline{\text{IO}}$ 的控制,这样做可减少对 2114 的干扰。2114 的读/写控制信号只有一个 $\overline{\text{WE}}$, $\overline{\text{WE}} = 1$ 时为读操作, $\overline{\text{WE}} = 0$ 时为写操作。

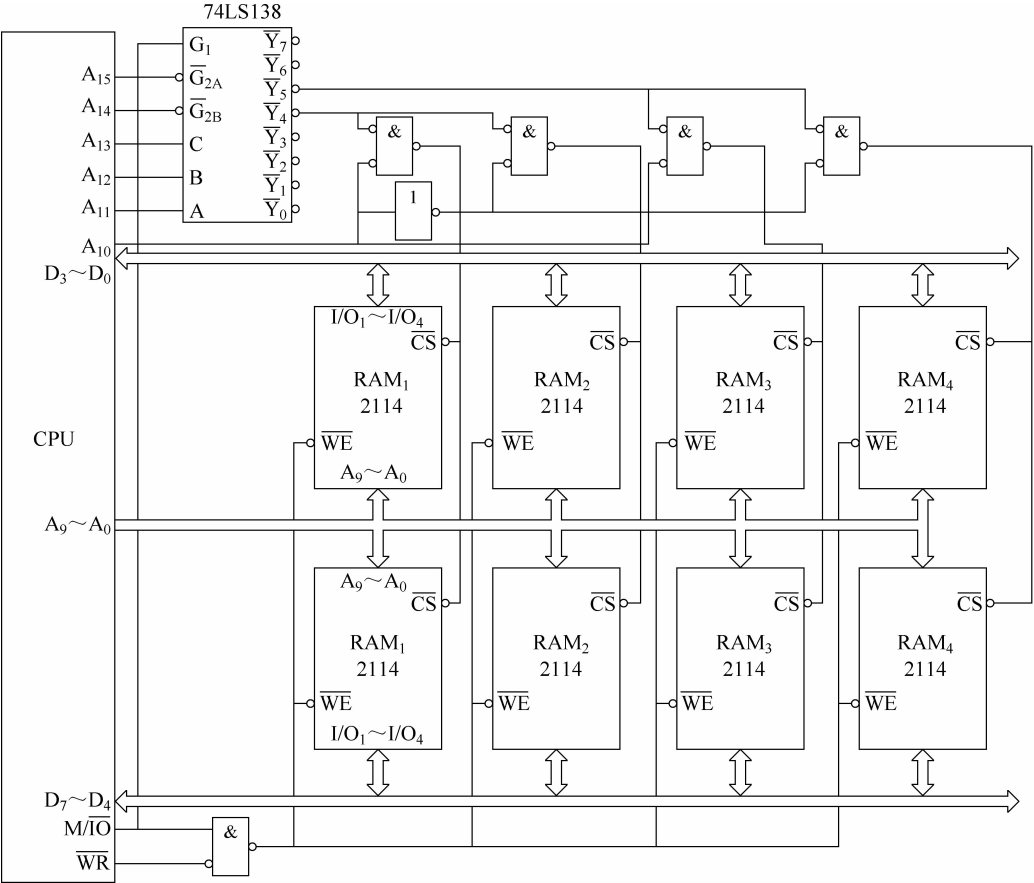


图 5.11 SRAM 2114 与 CPU 的连接

表 5.3 2114 的地址范围表

芯片	$A_{15} \sim A_{11}$	A_{10}	$A_9 \sim A_0$		地址
			最低地址	最高地址	
RAM ₁	00100	0	0000000000	1111111111	2000H~23FFH
RAM ₂	00100	1	0000000000	1111111111	2400H~27FFH
RAM ₃	00101	0	0000000000	1111111111	2800H~2BFFH
RAM ₄	00101	1	0000000000	1111111111	2C00H~2FFFH

5.2.2 动态 MOS 存储器

1. 四管动态存储元

在六管静态存储元电路中,信息暂存于 T_1 、 T_2 管的栅极,这是因为管子总是存在着一一定的电容。负载管 T_3 、 T_4 是为了给这些存储电荷补充电荷用的。由于 MOS 的栅极电阻很高,故泄漏电流很小,在一定的时间内这些信息电荷可以维持住。为了减少管子以提高集成度,把负载管 T_3 、 T_4 去掉,这样就变成了四管的动态存储电路,如图 5.12 所示。

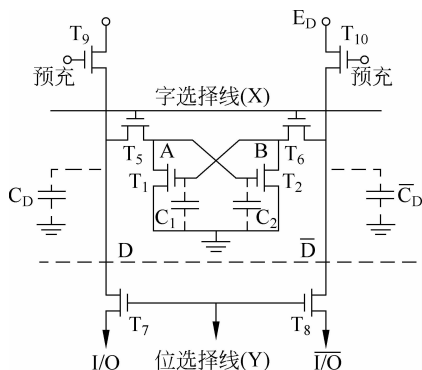


图 5.12 四管动态存储元

T_5 、 T_6 仍为控制管,由字选择线(X线)控制。当选择线为高电平时, T_5 、 T_6 导通,则存储电路的 A、B 点与位线 D、 $\bar{D}$ 分别相连,再通过 T_7 、 T_8 管与外部数据线 I/O、 $\bar{I/O}$ 相通。同时在一列的位线上接有两个公共的预充管 T_9 、 T_{10} 。

1) 写入操作

当写入时,I/O 与 $\bar{I/O}$ 仍加相反的电平(例如写入“1”时, $I/O = “1”$, $\bar{I/O} = “0”$),字选择线的高电平打开 T_5 、 T_6 管,因而所存的信息送至 A、B 端,将信息存储在 T_1 、 T_2 管的栅极电容上。当 T_5 、 T_6 截止时,靠 T_1 、 T_2 管栅极电容的存储作用,在一定时间内(如 2ms)可以保留所写入的信息。

2) 读出操作

当读出时,先给出预充信号,使 T_9 、 T_{10} 管导通,于是电源就向位线 D 和 $\bar{D}$ 上的电容充电,使它们达到电源电压。当字选择线使 T_5 、 T_6 管导通时,存储的信息通过 A、B 端位线输出。若原存信息为“1”,则电容 C_2 上存有电荷, T_2 管导通,而 T_1 管截止,因此 D 上的预充电荷经 T_2 管泄漏,故 $D=0$,而 $\bar{D}$ 仍为“1”,信号通过 I/O 和 $\bar{I/O}$ 线输出。与此同时,D 上的电荷可以通过 A 点向 C_2 补充。故读出过程也是刷新的过程。

3) 再生操作

由于存储的信息电荷终究是有泄漏的,电荷数又不能像六管电路那样由电源经负载管不断补充,时间一长,就会丢失信息。因此,必须设法同外界按一定规律不断给栅极进行充电,按需要补足栅极的信息电荷,这就是所谓的**再生或刷新**。

四管存储元的刷新过程并不复杂,在字选择线上加一个脉冲就能自动刷新正确的存储信息。设原存信息为“1”, T_2 管导通, T_1 管截止。经过一段时间, T_2 管栅极上漏失了一部分信息电荷,使 A 端的电压稍小于存“1”时的满值电压。当字选择线上加脉冲使 T_5 、 T_6 管导通后,A 端与位线 D 相接,就被充电到满值电压,从而刷新了原存的“1”信息。显然,只要定时给全部存储元电路执行一遍读操作,而信息不向外输出,那么就可以实现信息再生或刷新。

2. 单管动态存储元

为了进一步缩小存储器的体积,提高它们的集成度,人们又设计了单管动态存储元电路。

单管动态存储元电路如图 5.13 所示,它由一个管子 T_1 和一个电容 C 构成。写入时,

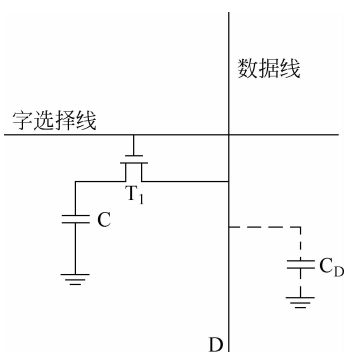


图 5.13 单管动态存储元

字选择线为“1”， T_1 管导通，写入信息由位线(数据线)存入电容 C 中；读出时，字选择线为“1”，存储在电容 C 上的电荷，通过 T_1 输出到数据线上，通过读出放大器即可得到存储信息。为了节省面积，这种单管存储元电路的电容 C 不可能做得很大，一般都比数据线上的分布电容 C_D 小。因此每次读出后，存储内容就被破坏。为此，必须采取恢复措施，以便再生原存的信息。

比较四管和单管存储元电路可知，它们各有优缺点。四管电路的缺点是管子多，占用的芯片面积大；优点是外围电路比较简单，读出过程就是刷新过程，故在刷新时不需要另加外部逻辑。单管电路的元件数量少，集成度高，但因读“1”和“0”时，数据线上的电平差别很小，需要高鉴别能力的读出放大器配合工作，所以外围电路比较复杂。

由单管动态存储元组成的存储体矩阵如图 5.14 所示。

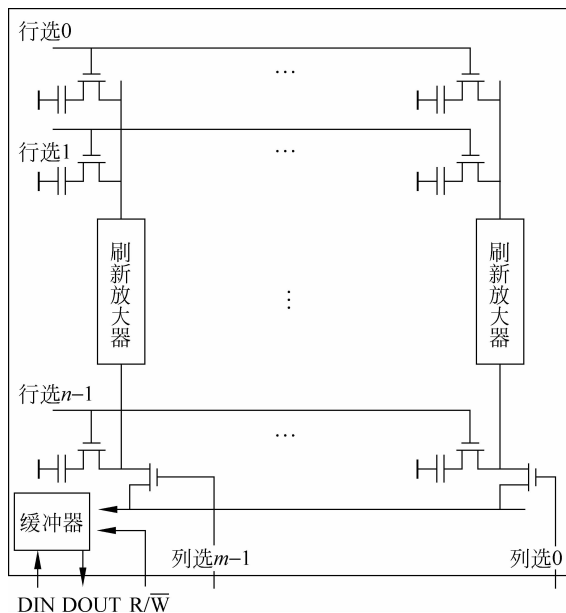


图 5.14 单管 DRAM 的存储矩阵

3. DRAM 的刷新和 DRAM 控制器

刷新的方法有多种，但最常用的是“只有行地址有效”的方法，按照这种方法，刷新时，存储体的列地址无效，一次只选中存储体中的一行进行刷新。具体执行时，依次选中存储芯片中的各行，被选中行中所有存储单元都分别和读出放大电路接通，在定时时钟作用下，读出放大电路分别对该行存储单元进行一次读出、放大和重写，即进行刷新。只要在刷新时限 2ms 中对 DRAM 芯片中所有行刷新一遍，就可以实现全面刷新。

为了实现刷新，DRAM 控制器具有如下功能：

1) 时序功能

DRAM 控制器需要按固定的时序提供行地址选通信号 RAS,为此,用一个计数器产生刷新地址,同时用一个刷新定时器产生刷新请求信号,以此启动一个刷新周期,刷新地址和刷新请求信号联合产生行地址选通信号 RAS,每刷新一行,就产生下一个行地址选通信号。

2) 地址处理功能

DRAM 控制器一方面要在刷新周期中顺序提供行地址,以保证在 2ms 中使所有的 DRAM 单元都被刷新一次;另一方面,要用一个多路开关对地址进行切换,因为正常读/写时,行地址和列地址来自地址总线,刷新时只是来自刷新地址计数器的行地址而没有列地址,总线地址则被封锁。

3) 仲裁功能

当来自 CPU 对内存的正常读/写请求和来自刷新电路的刷新请求同时出现时,仲裁电路要作出仲裁,原则上,CPU 的读/写请求优先于刷新请求。内部的“读/写和刷新的仲裁和切换”电路一方面会实现仲裁功能,另一方面会完成总线地址和刷新地址之间的切换。

图 5.15 是 DRAM 控制器的原理图。其中, $\overline{CAS}_0 \sim \overline{CAS}_n$ 和 $\overline{WE}$ 是传递的总线信号,与刷新过程无关。

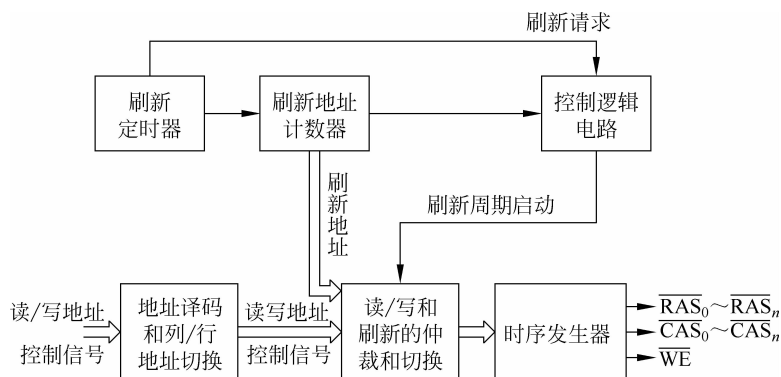


图 5.15 DRAM 控制器的原理图

4. 动态随机存取存储器举例

由于动态存储器电路简单,相对集成度要比静态存储器高得多,因此常作为微型计算机的主存储器。目前常用的有 4164、41256、41464 以及 414256 等类型,其存储容量分别为 $64K \times 1$ 、 $256K \times 1$ 、 $64K \times 4$ 和 $256K \times 4$ 。

414256 的内部组成如图 5.16 所示。

414256 的基本组成是 $512 \times 512 \times 4$ 的存储器阵列。在此基础上设有读出放大器与 I/O 门控制电路、行地址缓冲器/译码器、列地址缓冲器/译码器、数据输入/输出缓冲器、刷新控制/计数器以及时钟发生器等。存储器访问时,行地址和列地址分两次输入。首先由信号锁存地址线 $A_8 \sim A_0$ 输入的 9 位行地址,然后再由信号锁存地址线 $A_8 \sim A_0$ 输入的 9 位列地址,经译码选中某一存储单元,在读/写控制信号的控制下,即可对该单元的 4 位数据进行读出或者写入。

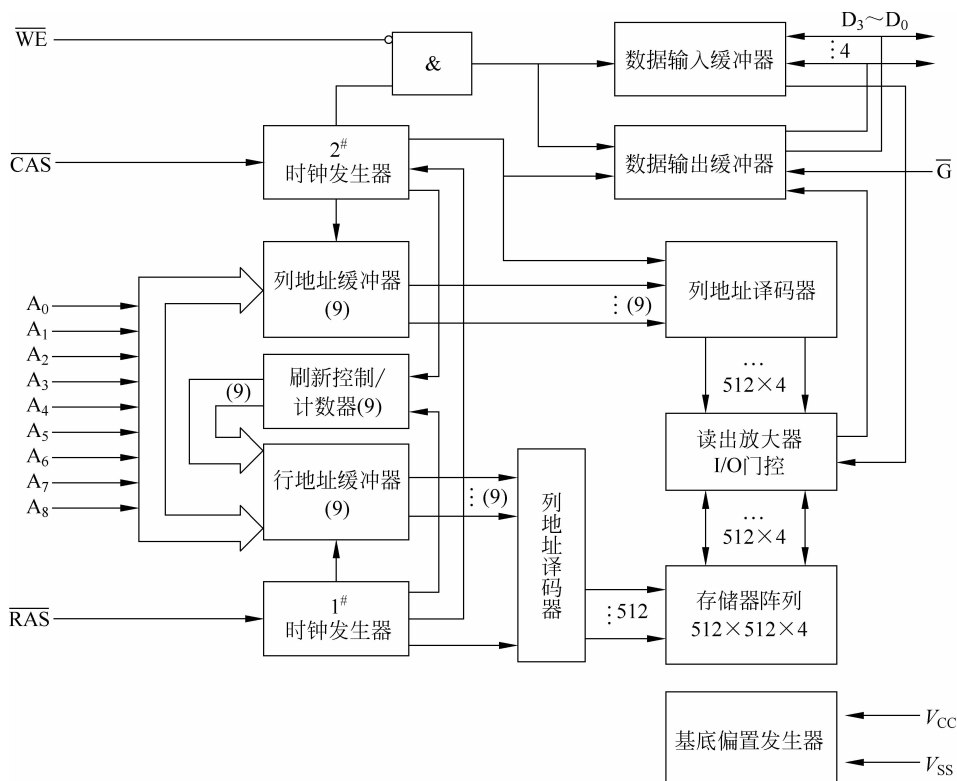


图 5.16 414256 内部结构

由于动态存储器读出时需预充电,因此每次读/写操作均进行一次刷新。MCM414256 需要每 8ms 刷新一次。刷新时通过在 512 个行地址间按顺序循环进行,可以分散刷新,也可以连续刷新。**分散刷新**也称为分布刷新,是指每 $15.6\mu\text{s}$ 刷新一行;**连续刷新**也称猝发方式刷新,它是对 512 行集中刷新。MCM414256 必须每 8ms 进行一次快速刷新, MCM41M256 每 64ms 进行一次快速刷新。

5. 高集成度 DRAM

由于微型计算机内存的实际配置已从 640KB 发展到高达 16MB 甚至 1GB,因此要求配套的 DRAM 集成度也越来越高。容量为 $1\text{M}\times 1$ 、 $1\text{M}\times 4$ 、 $4\text{M}\times 1$ 以及更高集成度的存储器芯片已大量使用。通常把这些芯片放在内存条上,用户只需把内存条插到系统板上提供的存储条插座上即可使用。例如有 $256\text{K}\times 8$ 、 $1\text{M}\times 8$ 、 $256\text{K}\times 9$ 、 $1\text{M}\times 9$ (9 位时有一位为奇偶校验位)及更高集成度的存储条。图 5.17 是采用 HYM59256A 的存储条。其中, $A_8\sim A_0$ 为地址输入线, $DQ_7\sim DQ_0$ 为双向数据线, PD 为奇偶校验数据输入, $\overline{\text{PCAS}}$ 为奇偶校验的地址选通信号, PQ 为奇偶校验数据输出, $\overline{\text{WE}}$ 为读/写控制信号, $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 为行、列地址选通信号, V_{DD} 为电源 (5V), V_{SS} 为地线, 30 个引脚定义是存储条通用标准。

另外,还有 $1\text{M}\times 8$ 的内存条, HYM58100 是用 $1\text{M}\times 1$ 的 8 片 DRAM 组成的,也可用两片 $1\text{M}\times 4$ DRAM 组成,更高集成度的内存条请参阅存储器手册。

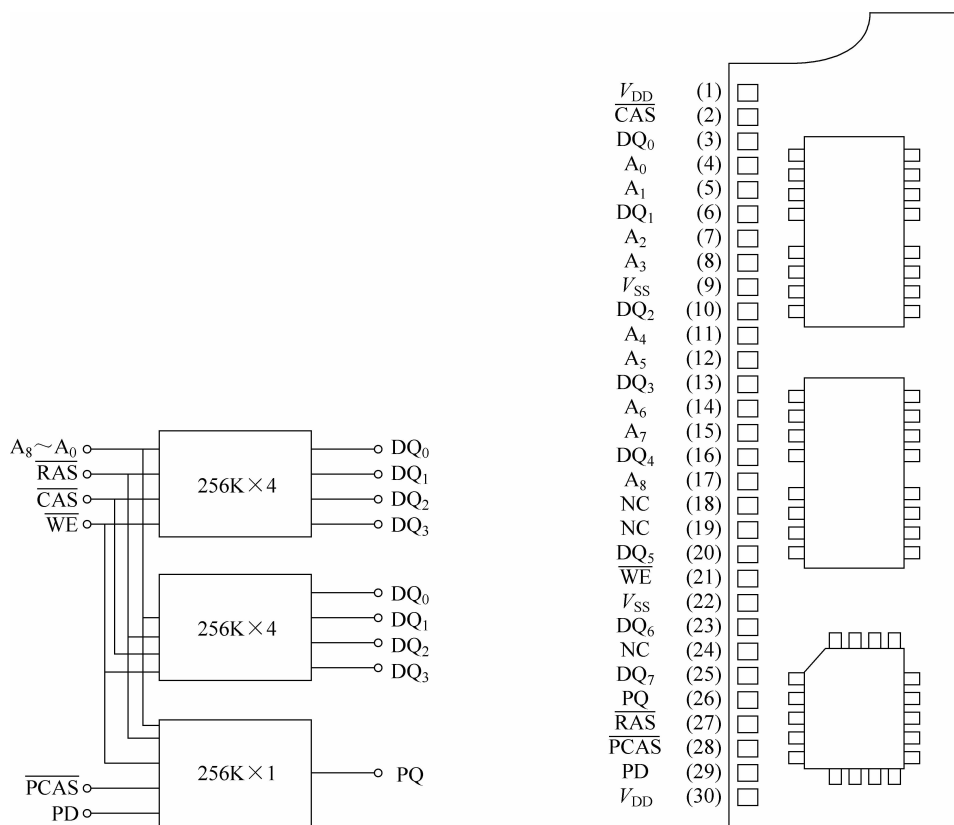


图 5.17 256K×9 存储条

5.3 只读存储器

只读存储器(ROM)的信息在使用时是不能被改变的,即只能读出,不能写入,故一般只能存放固定程序,如监控程序、IBM PC 中的 BIOS 程序等。ROM 的特点是非易失性,即掉电后再上电时存储信息不会改变。

早期只读存储器的存储内容根据用户要求,厂家采用掩膜工艺,把原始信息记录在芯片中,一旦制成后就无法更改,叫做掩膜只读存储器(Masked ROM, MROM)。随着半导体技术的发展和用户需求的变化,先后出现了可编程只读存储器(Programmable ROM, PROM)、可擦可编程只读存储器(Erasable Programmable ROM, EPROM)以及电可擦可编程只读存储器(Electrically Erasable Programmable ROM, EEPROM)。近年来还出现了快擦型存储器(Flash Memory),它具有 EEPROM 的特点,而速度比 EEPROM 快得多。

5.3.1 掩膜只读存储器

掩膜 ROM 制成后,用户不能修改,图 5.18 为一个简单的 4×4 位 MOS 管 ROM,采用单译码结构,两位地址线 A₁、A₀ 译码后可译出 4 种状态,输出 4 条选择线,可分别选中 4 个单元,每个单元有 4 位输出。

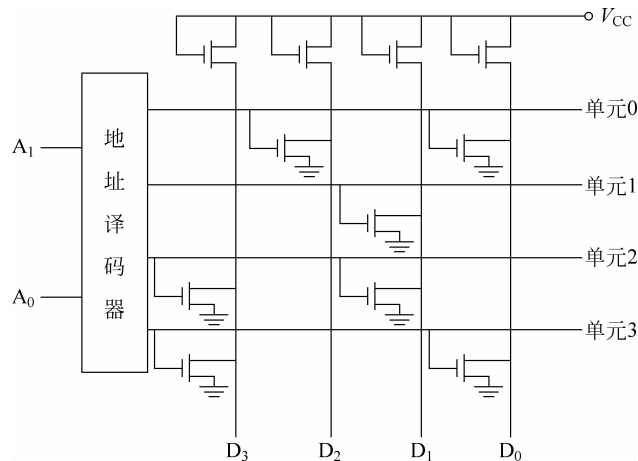


图 5.18 掩膜 ROM 电原理图

如图 5.18 所示的矩阵中,在行和列的交点处,有的连有管子,有的没有,这是由工厂根据用户提供的程序对芯片图形(掩膜)进行二次光刻所决定的,所以称为掩膜 ROM。

若地址线 $A_1 A_0 = 00$,则选中 0 号单位,即字线 0 为高电平,若有管子与其相连(如位线 D_2 和 D_0),则其相应的 MOS 管导通,位线输出为 0,而位线 D_1 和 D_3 没有管子与字线相连,则输出为 1。故存储器的内容取决于制造工艺,图 5.18 存储矩阵的内容如表 5.4 所示。

表 5.4 掩膜 ROM 的内容

单元	位			
	D_3	D_2	D_1	D_0
0	1	0	1	0
1	1	1	0	1
2	0	1	0	1
3	0	1	1	0

5.3.2 可擦可编程只读存储器

在某些应用中,程序需要经常修改,因此能够重复擦写的 EPROM 被广泛应用。这种存储器利用专门的编程器写入后,信息可长久保持,因此仍被归类于只读存储器。当其内容需要变更时,可利用专门的擦除器(由紫外线灯照射)将其擦除,各字节内容复原(为 FFH),再根据需要利用 EPROM 编程器写入新的数据,因此这种芯片可反复使用。

1. EPROM 的存储单元电路

通常 EPROM 存储电路是利用浮栅 MOS 管构成的,又称浮栅雪崩注入 MOS 管(Floating gate Avalanche Injection Metal-Oxide-Semiconductor, FAMOS 管),其构造如图 5.19(a)所示。

该电路和普通 P 沟道增强型 MOS 管相似,只是栅极没有引出端,而被 SiO_2 绝缘层所包围,故称为“浮栅”。在原始状态,栅极上没有电荷,该管没有导通沟道,D 和 S 是不导通的。如果将源极和衬底接地,在衬底和漏极形成的 PN 结上加一个约 24V 的反向电压,则

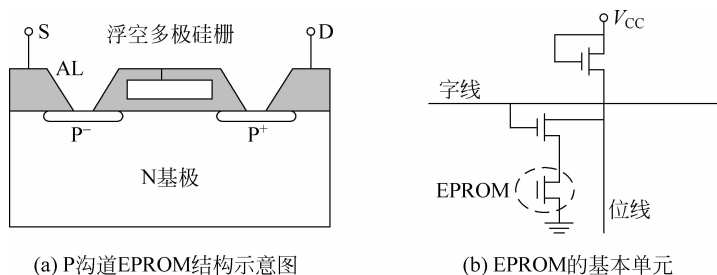


图 5.19 浮栅 MOS EPROM 存储电路

可导致雪崩击穿,产生许多高能量的电子,这样的电子比较容易越过绝缘薄层进入浮栅。注入浮栅的电子数量由所加电压脉冲的幅度和宽度来控制。如果注入的电子足够多,这些负电子在硅表面上就感应出一个连接源、漏极的反型层,使源漏极呈低阻态。当外加电压取消后,积累在浮栅上的电子没有放电回路,因而在室温和无光照的条件下可长期地保存在浮栅中。将一个浮栅管和 MOS 管串起来组成如图 5.19(b)所示的存储单元电路。于是浮栅中注入了电子的 MOS 管源、漏极导通,当行选线选中该存储单元时,相应的位线为低电平,即读取值为“0”,而未注入电子的浮栅管的源、漏极是不导通的,故读取值为“1”。在原始状态,即厂家出厂时,没有经过编程,浮栅中没有注入电子,位线上总是“1”。

消除浮栅电荷的办法是利用紫外线光照射,由于紫外线光能量较高,从而可使浮栅中的电子获得能量,形成光电流从浮栅流入基片,从而使浮栅恢复初态。EPROM 芯片上方有一个石英玻璃窗口,只要将此芯片放入一个靠近紫外线灯管的小盒中,一般照射 10 分钟左右,读出各单元的内容均为 FFH,就说明该 EPROM 已擦除。

2. 典型 EPROM 芯片介绍

EPROM 芯片有多种型号,如 2716(2K×8)、2732(4K×8)、2764(8K×8)、27128(16K×8)、27256(32K×8)等。下面以 2764A 为例,对 EPROM 的性能和工作方式作一介绍。

Intel 2764A 有 13 条地址线、8 条数据线、两个电压输入端 V_{CC} 和 V_{PP} 、一个片选端 $\overline{CE}$ (功能同 $\overline{CS}$),此外还有输出允许端 $\overline{OE}$ 和编程控制端 $\overline{PGM}$,其功能框图如图 5.20 所示。

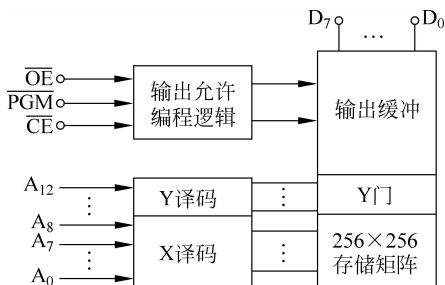


图 5.20 2764A 功能框图

1) 读方式

这是 2764A 通常使用的方式,此时两个电源引脚 V_{CC} 和 V_{PP} 都接至 5V, $\overline{PGM}$ 接至高电平,当从 2764A 的某个单元读数据时,先通过地址引脚接收来自 CPU 的地址信号,然后使控制信号 $\overline{CE}$ 和 $\overline{OE}$ 都有效,于是经过一个时间间隔,指定单元的内容即可读到数据总线上。

Intel 2764A 有 7 种工作方式,如表 5.5 所示。

表 5.5 2764A 的工作方式选择表

引脚 方式	$\overline{\text{CE}}$	$\overline{\text{OE}}$	$\overline{\text{PGM}}$	A_9	A_0	V_{PP}	V_{CC}	数据端功能
读	低	低	高	×	×	V_{CC}	5V	数据输出
输出禁止	低	高	高	×	×	V_{CC}	5V	高阻
备用	高	×	×	×	×	V_{CC}	5V	高阻
编程	低	高	低	×	×	12.5V	V_{CC}	数据输入
校验	低	低	高	×	×	12.5V	V_{CC}	数据输出
编程禁止	高	×	×	×	×	12.5V	V_{CC}	高阻
标识符	低	低	高	高	低	V_{CC}	5V	制造商编码
				高	高	V_{CC}	5V	器件编码

把 A_9 引脚接至 11.5~12.5V 的高电平,则 2764A 处于读 Intel 标识符模式。

要读出 2764A 的编码必须顺序读出两个字节,先让 $\text{A}_1 \sim \text{A}_8$ 全为低电平,而使 A_0 从低变高,分两次读取 2764A 的内容,当 $\text{A}_0=0$ 时读出的内容为制造商编码(陶瓷封装为 89H,塑封为 88H),当 $\text{A}_0=1$ 时,则可读出器件的编码(2764A 为 08H,27C64 为 07H)。

2) 备用方式

只要 $\overline{\text{CE}}$ 为高电平,2764A 就工作在备用方式,输出端为高阻状态,这时芯片功耗将下降,从电源所取电流由 100mA 下降到 40mA。

3) 编程方式

V_{PP} 接 12.5V, V_{CC} 仍接 5V,从数据线输入这个单元要存储的数据, $\overline{\text{CE}}$ 端保持低电平,输出允许信号 $\overline{\text{OE}}$ 为高,每写一个地址单元,都必须在 $\overline{\text{PGM}}$ 引脚端给一个低电平有效、宽度为 45ms 的脉冲,如图 5.21 所示。

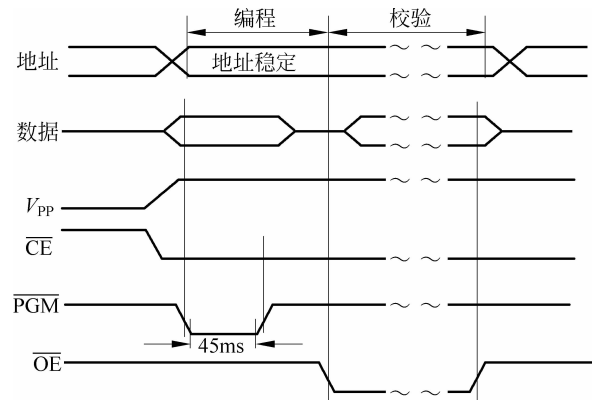


图 5.21 2764A 编程波形

4) 编程禁止

在编程过程中,只要使该片 $\overline{\text{CE}}$ 为高电平,则编程就立即禁止。

5) 编程校验

在编程过程中,为了检查编程时写入的数据是否正确,通常在编程过程中包含校验操作。

在一个字节的编程完成后,电源的接法不变,但 $\overline{\text{PGM}}$ 为高电平, $\overline{\text{CE}}$ 、 $\overline{\text{OE}}$ 均为低电平,则同一单元的数据就在数据线上输出,这样就可以与输入数据相比较,校验编程的结果是否正确。

6) Intel 标识符模式

当两个电源端 V_{CC} 和 V_{PP} 都接至 5V, $\overline{\text{CE}} = \overline{\text{OE}} = 0$ 时, $\overline{\text{PGM}}$ 为高电平,这时与读方式相同,另外,在对 EPROM 编程时,每写一个字节都需 45ms 的 $\overline{\text{PGM}}$ 脉冲,速度太慢,且容量越大,速度越慢。为此,Intel 公司开发了一种新的编程方法,比标准方法快 6 倍以上,其流程图如图 5.22 所示。

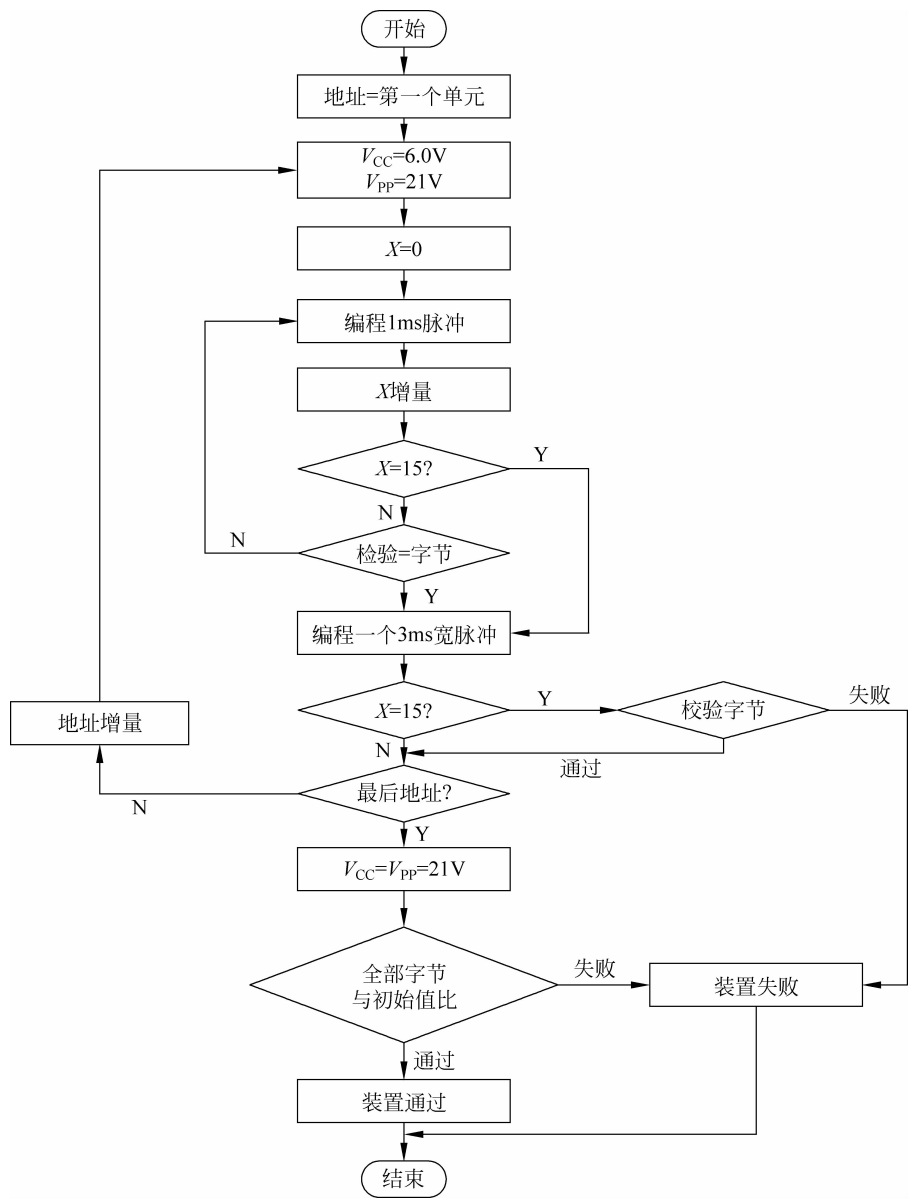


图 5.22 Intel 对 EPROM 编程算法流程图

实际上,按这一思路开发的编程器有多种型号。编程器中有一个卡插在 I/O 扩展槽上,外部接有 EPROM 插座,所提供的编程软件可自动提供编程电压 V_{PP} ,按菜单提示,可读、可编程、可校验,也可读出器件的编码,操作很方便。

3. 高集成度 EPROM

除了常用的 EPROM 2764 外,27128、27256、27512 等也是常用的 EPROM 芯片。由于工业控制计算机的发展,迫切需用电子盘取代硬盘,常把用户程序、操作系统固化在电子盘 (ROMDISK) 上,这时要用 27C010(128K×8)、27C020(256K×8)、27C040(512K×8) 等大容量芯片。关于这几种芯片的使用请参阅有关手册。

5.3.3 电可擦可编程存储器

EPROM 的优点是一块芯片可多次使用,缺点是即使整个芯片只写错一位,也必须从电路板上取下擦掉重写,这对于实际使用是很不方便的。在实际应用中,往往只要改写几个字节的内容即可,因此多数情况下需要以字节为单位的擦写。而 EEPROM 在这方面具有很大的优越性。

下面以 Intel 2816 为例,说明 EEPROM 的基本特点和工作方式。

1. 2816 的基本特点

2816 是容量为 $2K \times 8$ 的电擦除 PROM,它的逻辑符号如图 5.23 所示。

芯片的管脚排列与 2716 一致,只是在管脚定义上,数据线管脚对 2816 来说是双向的,以适应读/写工作模式。2816 的读取时间为 250ns,可满足多数微处理器对读取速度的要求。2816 最突出的特点是可以以字节为单位进行擦除和重写。擦或写用 $\overline{CE}$ 和 $\overline{OE}$ 信号加以控制。一个字节的擦写时间为 10ms。2816 也可整片进行擦除,整片擦除时间也是 10ms。无论字节擦除还是整片擦除均在机内进行。

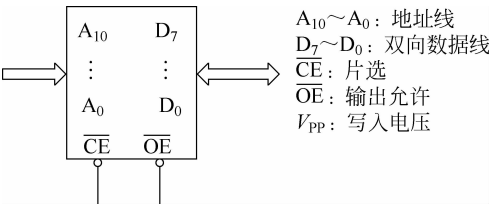


图 5.23 2816 的逻辑符号

2. 2816 的工作方式

2816 有 6 种工作方式,每种工作方式下各个控制信号所需电平如表 5.6 所示。从表中可见,除整片擦除外, $\overline{CE}$ 和 $\overline{OE}$ 均为 TTL 电平,而整片擦除时为 +9~+15V, V_{PP} 在擦或写方式时均为 21V 的脉冲,而其他工作方式时电压为 4~6V。

表 5.6 2816 的工作方式

管脚 方式	$\overline{CE}$	$\overline{OE}$	V_{PP} (V)	数据线功能
读方式	低	低	+4~+6	输出
备用方式	高	×	+4~+6	高阻
字节擦除	低	高	+21	输入为高电平
字节写	低	高	+21	输入
片擦除	低	+9~+15V	+21	输入为高电平
擦写禁止	高	×	+21	高阻

1) 读方式

在读方式时,允许 CPU 读取 2816 的数据。在 CPU 发出地址信号以及相关的控制信号后,与此相对应,2816 的地址信号和 $\overline{\text{CE}}$ 、 $\overline{\text{OE}}$ 信号有效,经一定延时,2816 就可以提供有效数据。

2) 写方式

2816 具有以字节为单位的擦写功能,擦除和写入是同一种操作,即都是写,只不过擦除是固定写“1”而已。因此,在擦除时,数据输入是 TTL 高电平。在以字节为单位进行擦除和写入时, $\overline{\text{CE}}$ 为低电平, $\overline{\text{OE}}$ 为高电平,从 V_{PP} 端输入编程脉冲,宽度最小为 9ms,最大为 70ms,幅度为 21V。为保证存储单元能长期、可靠地工作,编程脉冲要求以指数形式上升到 21V。

3) 片擦除方式

在 2816 需整片擦除时,当然也可按字节擦除方式将整片 2KB 逐个进行,但最简便的方法是依照表 5.6,将 $\overline{\text{CE}}$ 和 V_{PP} 按片擦除方式连接,将数据输入引脚置为 TTL 高电平,而使 $\overline{\text{OE}}$ 引脚电压达到 9~15V,则大约经过 10ms,整片内容就全部被擦除,即 2KB 的内容全为 FFH。

4) 备用方式

当 2816 的 $\overline{\text{CE}}$ 端加上 TTL 高电平时,芯片处于备用状态, $\overline{\text{CE}}$ 控制无效,输出呈高阻态。在备用状态下,其功耗可降到原来的 55%。

3. 2817A EEPROM

在工业控制领域,常用 2817A EEPROM,其容量也是 $2\text{K} \times 8$,采用 28 脚封装,它比 2816 多一个 $\text{RDY}/\overline{\text{BUSY}}$ 引脚,用于向 CPU 提供状态。擦写过程是:将原有内容擦除时,将 $\text{RDY}/\overline{\text{BUSY}}$ 引脚置于低电平,然后再将新的数据写入,完成此项操作后,再将 $\text{RDY}/\overline{\text{BUSY}}$ 引脚置于高电平,CPU 通过检测此引脚的状态来控制芯片的擦写操作,擦写时间约为 5ns。

2817A 的特点是片内具有防写保护单元,适于现场修改参数。2817A 引脚如图 5.24 所示。

图 5.24 中, $\text{R}/\overline{\text{B}}$ 是 $\text{RDY}/\overline{\text{BUSY}}$ 的缩写,用于指示器件的准备就绪/忙状态,2817A 使用单一的 5V 电源,在片内有升压到 21V 的电路,用于原 V_{PP} 引脚的功能,可避免 V_{PP} 偏高或加电顺序错误引起的损坏,2817A 片内有地址锁存器、数据锁存器,因此,可与 8088/8086、8031、8096 等 CPU 直接连接。

2817A 片内写周期定时器通过 $\text{RDY}/\overline{\text{BUSY}}$ 引脚向 CPU 表明它所处的工作状态,在写一个字节的過程中,此引脚呈低电平,写完以后此引脚变为高电平。2817A 中 $\text{RDY}/\overline{\text{BUSY}}$ 引脚的这一功能可在每写完一个字节后向 CPU 请求外部中断来继续写入下一个字节,而在写入过程中,其数据线呈高阻状态,故 CPU 可继续执行其程序。因此,采用中断方式既可在线修改内存参数,而又不致影响工业控制计算机的实时性。

2817A 读取时间为 200ns,数据保存时间接近十年,每个单元允许擦写 10^4 次,故要均衡地使用每个单元,以提高其寿命。2817A 的工作方式如表 5.7 所示。

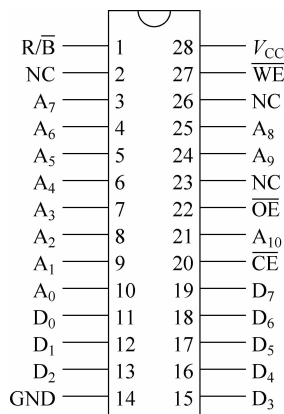


图 5.24 2817A 引脚图

表 5.7 2817A 工作方式选择表

引脚 方式	$\overline{\text{CE}}$	$\overline{\text{OE}}$	$\overline{\text{WE}}$	$\text{RDY}/\overline{\text{BUSY}}$	数据线功能
读	低	低	高	高阻	输出
维持	高	无关	无关	高阻	高阻
字节写入	低	高	低	低	输入
字节擦除	字节写入前自动擦除				

此外,2864A 是 $8\text{K}\times 8$ 的 EEPROM,其性能更优越,每一字节擦写时间为 5ns,整片擦除只需 2ms,读取时间为 250ns,其引脚与 2764 兼容。

5.3.4 快擦写存储器

只读存储器的特点是在不加电的情况下其中的信息可以长期保持。而快擦写存储器与 EEPROM 类似,除信息可以长期保持之外,也可在线擦除与重写。其集成度与价格已接近 EEPROM,因而有替代 EPROM 和 EEPROM 的趋势。

快擦写存储器(Flash Memory)的基本单元电路如图 5.24 所示,与 EEPROM 类似,也是由双层浮空栅 MOS 管组成。但是第一层栅介质很薄,作为隧道氧化层。写入方法与 EEPROM 相同,在第二级浮空栅加以正电压,使电子进入第一级浮空栅。读出方法与 EPROM 相同。擦除方法是在源极加正电压利用第一级浮空栅与源极之间的隧道效应,把注入浮空栅的负电荷吸引到源极。由于利用源极加正电压擦除,因此各单元的源极连在一起,这样,快擦存储器不能按字节擦除,而是全片或分块擦除。按照擦除和使用方式,快擦写存储器目前主要有以下 3 类。

1. 整体擦除快擦写存储器

整体擦除快擦写存储器除了一般只读存储器所具备的地址锁存器/译码器、片选电路、数据锁存器、输入/输出缓冲器和读控制电路之外,存储器阵列采用的是快擦写存储器电路,另外有擦除电压开关、编程电压开关、命令寄存器、停止定时器以及状态控制电路等。其中,命令寄存器用来写入命令字以及执行该命令所需要的地址和数据。

目前,整体擦除快擦写存储器已有多种型号,比如 28F010、28F020、28F256 和 28F512。

2. 对称型块结构快擦写存储器

对称型块结构快擦写存储器把存储器阵列划分成大小相等的存储块,每块可以独立地被擦除或者编程。擦除时可以擦除其中的任意一块。目前对称型块结构快擦写存储器已有多种型号,比如 28F008SA、28F016SA 和 28F032SA 等。

其中,28F008SA 是 8MB 存储器,其结构为 $1\text{M}\times 8$,划分成 16 个 64KB 存储块。内部设有状态机,在字节写和块擦除后自动进行验证操作,通过内部状态寄存器来指示状态机的状态,以及字节写和块擦除操作是否成功。

3. 带自举块快擦写存储器

带自举块快擦写存储器是在块结构的基础上增加了自举块,自举块用来存储自举程序。自举块具有数据保护特性,以在临界应用中保护自举程序代码。目前,带自举块快擦写存储器也有多种型号,比如 28F400BX 和 28F004BX 等,其中,28F004BX 是 8 位快擦写存储器;28F400BX 是 8 位/16 位快擦写存储器,可与 8 位/16 位数据总线连接。

28F004BX/28F400BX 是 4MB 块结构快擦写存储器,可构成 512KB 或 256K 字(16 位)存储器。在 4MB 存储器中包含 7 个独立的可快擦写存储块,一个可锁定的自举块(16KB),两个参数块(每块 8KB)和 4 个主块(1 块 96KB,3 块各为 128KB)。为了适应不同处理器自举代码位置协议,自举块位于地址映像的顶部(28F400BX-T 和 28F004BX-T)或底部(28F400BX-B 和 28F004BX-B)。自举块受引脚信号 PWD 控制,当 PWD 为 11.4~12.6V 时自举块不锁定,可进行编程和擦除操作;当 $PWD \leq 6.5V$ 时自举块锁定,即不能对其编程和擦除。

ROM、FLASH 存储器与 CPU 的连接、设计原则同 SRAM。但应注意 $\overline{OE}$ (输出允许)的使用,可酌情接 $\overline{RD}$ 信号或直接接直流地。

5.4 内存管理

5.4.1 80x86 系列 CPU 的工作模式

Intel 8086 CPU 只运行在实模式下,寻址范围只有 1MB。而 Intel 80286、80386、80486 则可以工作于实模式或者受保护的虚地址模式下,由它们组成的微型计算机大多配置了 4~16MB 的内存,有的还配置了 64MB 的存储器。它们的地址引线数目以及可寻址的最大空间如表 5.8 所示。

表 5.8 不同 CPU 的寻址范围

CPU	数据总线	地址总线	寻址范围	支持操作系统
8086	8 位	20 位	1MB	实方式
80286	16 位	24 位	16MB	实、保护方式
80386/80486/Pentium	32 位	32 位	4096MB	实、保护、V86 方式

实地址模式下,系统和程序适用 16 位“段基址:偏移量”格式,虽然用户的程序可按逻辑意义划分成代码段、数据段、堆栈段、附加数据段,但是它没有实现按段来分配存储空间。它只能在静态连接后,对整个程序分配连续的存储空间,当然也不支持虚拟存储。逻辑地址到物理地址的转换是由 CPU 自动完成的。

保护模式下,每个任务的虚拟存储空间可达 1024MB,CPU 自动把任务即将访问的部分映射到 16MB 的物理地址空间。

5.4.2 内存空间的管理

Intel 80x86 系列 CPU 组成的微机系统中的内存空间如图 5.25 所示。

1. 常规内存

常规内存(Conventional Memory)在内存分配表中占用最前面的位置,从 0KB 到 640KB(地址为 000000H~09FFFFH),共占 640KB 的容量。它在内存的最前面并且在 DOS 可管理的内存区,因此又称之为低 DOS 内存(Low Dos Memory)或基本内存(Base Memory),使用此空间的程序有 BIOS 操作系统、DOS 操作系统、外围设备的驱动程序、中断向量表、一些常驻的程序、空闲可用的内存空间以及一般的应用软件等。

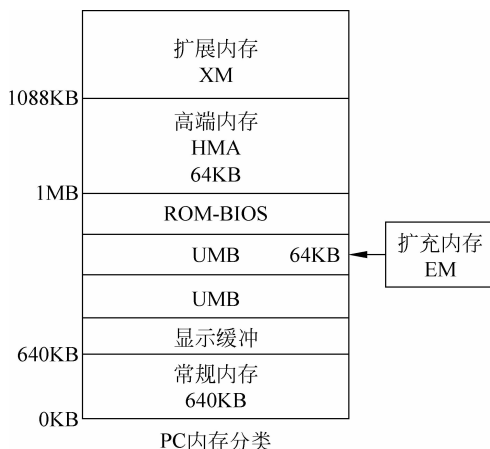


图 5.25 内存空间分布

MS-DOS 可以控制和管理 1MB 的内存空间,常规内存占了 640KB,其他的 384KB 保留给 BIOS ROM 及其他各种扩展卡使用。这 640KB 的常规内存基本上分为两部分:一部分给各种不同的操作系统程序使用;另一部分给数据、程序使用。

2. 上位内存

上位内存(Upper Memory Block, UMB)是常规内存上面一层的内存(640~1024KB),地址为 0A0000H~0FFFFFFH。在运行很多大的软件时,640KB 内存显然不够用,因此希望使用上位内存。在 384KB 的上位内存中,除了保留给 BIOS ROM、显示卡和其他各种扩展卡使用外,还有一些保留空间未使用,这些“空闲区”可被划分为 64KB 大小的块,称为上位内存块。所以在 DOS 5.0 以上的版本,允许应用程序突破 640KB 的限制,使用部分上位内存。

3. 高端内存区

高端内存区(High Memory Area, HMA)是 1024KB 至 1088KB 之间的 64KB 内存,其地址为 100000H~10FFEFH 或以上。CPU 在实地址模式下以“段地址:偏移量”方式来寻址,其寻址的最大逻辑内存空间为 (FFFFH: FFFFH),即 10FFEFH,此已超过 8088 CPU 的 20 条线所能寻址的 1MB 的上限,故 286 CPU 的地址线有 24 条,只要把 A₂₀ 地址信号线的“逻辑门”打开,即可使用此 64KB 范围的内存,这段内存仍在实地址模式下。

4. 扩展内存块

扩展内存块(Extended Memory Block, XMB)是指 1MB 以上的内存空间,其地址是从 100000H 开始,向上扩展的内存。扩展内存取决于 CPU 的寻址能力,286 CPU 可寻址到 16MB,386 CPU 以上至 Pentium II CPU 可寻址到 4GB。但是,有些主板上芯片组的实际地址译码电路并没有设计为可寻址那么大的地址空间,如 286 AT 的主板上最大寻址空间只到 4MB, Pentium 系列主板目前的最大扩展内存也到 1GB,距实际 CPU 的寻址空间还有一段距离。对于这些扩展内存,由于超过了 DOS 的寻址范围,因此并不能直接被实地址模式的 BIOS 或 DOS 操作系统所使用,而只能用于存放数据。但可以使用 DOS 的扩展器 (DOS Extender)或使用 Windows 3.1、Windows 95、Windows NT、OS2 等,在保护模式下供不同操作系统访问。要使计算机主机能使用扩展内存,还需要一些扩展内存驱动程序

(XMS)来加以驱动和设置。

5. 扩充内存

扩充内存利用 1MB 内存中 64KB 的内存区,此内存区为连续的 4 页,每页为 16KB 的实际页内存,它们映射到 EMS 卡上广大空间的逻辑页内存。Lotus、Intel、Microsoft 三家公司联合制订了扩充内存规范(Expanded Memory Specification,EMS)。对于 EMS 4.0 版本驱动程序,其映射的内存区为 1MB 内任意大小的内存,映射的扩充内存空间为 32MB,这是另一种扩充内存的方法。

5.5 例题解析

1. 若 256KB 的 SRAM 具有 8 条数据线,则它具有()条地址线。

- A. 10 B. 15 C. 20 D. 32

【解析】

$256\text{KB}=32\text{KB}\times 8$,而 $32\text{KB}=2^5\times 2^{10}\text{B}=2^{15}\text{B}$,故它具有 15 条地址线。正确结果是 B。

2. 对存储器进行访问时,地址线有效和数据线有效的时间关系应该是()。

- A. 同时有效同时无效 B. 数据线较先有效
C. 地址线较先有效 D. 以上均可

【解析】

正确结果是 C。

3. 8086 CPU 经过加电复位后,执行第一条指令的地址是()。

- A. FFFFH B. 03FFFH C. 0FFFFH D. FFFF0H

【解析】

在 8086/8088 的 1MB 存储空间中,从 FFFF0H 开始到 FFFFFH 共 16 个单元,一般用来存放一条无条件转移指令,转到系统的初始化程序,系统加电或复位后,会自动转到 FFFF0H 单元,开始系统初始化操作。正确结果是 D。

4. 在研制某一计算机应用系统的过程中,存储监控程序的存储器应选用()。

- A. RAM B. PROM C. EPROM D. ROM

【解析】

作为一个系统的监控程序,它是管理该系统的系统程序,在工作过程中一般不会改变,即工作在只读方式,因此应该选用只读存储器来存放。但是在系统的研制过程中,对系统的结构和功能等经常要修改,相应的监控程序也要改变,为此应该选用能够便于用户多次重写的只读存储器。EPROM 存储器可以使用紫外线照射,使其内容消失,然后再重新写入新的内容。正确结果是 C。

5. 写出下列容量的 RAM 芯片片内的地址线和数据线的条数:(1) $4\text{K}\times 8$ 位;(2) $512\text{K}\times 4$ 位;(3) $1\text{M}\times 1$ 位;(4) $2\text{K}\times 8$ 位。

【解析】

(1) $4\text{K}\times 8$ 位: 地址线 12 条,数据线 8 条。

(2) $512\text{K}\times 4$ 位: 地址线 19 条,数据线 4 条。

(3) $1\text{M}\times 1$ 位: 地址线 20 条,数据线 1 条。

(4) $2K \times 8$ 位: 地址线 11 条, 数据线 8 条。

6. 8086 CPU 执行“MOV [2003H], AX”指令, 从取指到执行指令最少需要多少时间? 设时钟频率为 5MHz, 该指令的机器码为 4 个字节, 存放在 1000H: 2000H 开始的代码段中。

【解析】

(1) 该条指令的机器码为 4 个字节, 存放在 1000H: 2000H 开始的 4 个单元中。取指令需要两个总线周期, 第一次取出 1000H: 2000H 与 1000H: 2001H 两个单元中的 16 位数据; 第二次取出 1000H: 2002H 与 1000H: 2003H 两个单元中的 16 位数据; 接着执行指令, 将 AX 中 16 位数传送到 DS: 2003H 与 DS: 2004H 两个存储单元中。因为是奇地址字, 所以需要两个总线周期才能完成。这样, 从取指到执行共需要 4 个总线周期。

(2) 在无等待周期的情况下, 从取指到执行共需要 $4 \times 4 \times 1/5 (\text{MHz}) = 3.2 \mu\text{s}$ (一个总线周期在无等待周期的情况下由 4 个时钟周期 T 组成)。

7. 什么是内存条? 采用内存条有何优点?

【解析】

内存条是一种以小型板卡形式出现的内存存储器产品, 在一个长条的印刷电路板上安装有若干存储器芯片, 印刷板长边上有 30、72 或 168 条引脚, 内存条可插在主板上的内存条插槽中。

采用内存条的优点是: 安装容易, 便于更换和易于增加或扩充内存容量。

习 题 5

- 试述 DRAM 的工作特点; 与 SRAM 相比有什么长处和不足之处; 说明它的使用场合。
- 试述 DRAM 刷新过程和正常读/写过程的区别。
- 设有一个具有 20 位地址和 32 位字长的存储器, 问:
 - 该存储器能存储多少个字节的信息?
 - 如果存储器由 $512K \times 8$ 位 SRAM 芯片组成, 需要多少片?
 - 需要多少片作芯片选择?
- 对由 $8K \times 8$ 位 RAM 组成的存储器系统, 若某组的起始地址为 08000H, 则其末地址为多少?
- 在 8088 最大方式系统总线上扩充设计 4K 字节的 SRAM 存储器电路。SRAM 芯片选用 Intel 2114, 起始地址从 00000H 开始。试画出此存储器电路与系统总线的连接图。
- 在 8088 系统总线上扩充设计 8K 字节的 SRAM 存储器电路。SRAM 芯片选用 Intel 6264, 起始地址从 04000H 开始, 译码电路采用 74LS138。
 - 计算此 RAM 存储区的最高地址是多少。
 - 画出此存储器电路与系统总线的连接图。
 - 编写程序实现对此存储器区域进行自检。
- 在 8086 最小方式系统总线上扩充设计 16K 字节的 SRAM 存储器电路。SRAM 芯片选用 Intel 6264, 起始地址从 04000H 开始, 译码电路采用 74LS138。
 - 计算此 RAM 存储区的最高地址是多少。
 - 画出此存储器电路与系统总线的连接图。