

在人工智能时代,随着数据规模的爆炸式增长和计算任务的日益复杂,主存储器的分类、工作原理及其与系统中其他组件(如 CPU)之间的高效协作显得尤为重要。高性能计算对存储系统提出了更高的要求,高速缓冲存储器、磁表面存储器等存储设备的结构与机制也在不断进化,以适应深度学习、大数据分析等场景对存储性能和效率的要求。本章旨在帮助读者深刻理解现代存储设备的工作原理以及如何构建层次化的存储系统,从而应对人工智能时代对计算机系统性能的全新挑战。

5.1 存储器的分类和层次结构

存储器(图 5.1)是计算机系统的记忆组件,负责存储程序指令和数据。随着计算机技术的发展,存储器在系统中的作用日益突显。特别是随着超大规模集成电路技术的进步,中央处理器(CPU)的处理速度已达到惊人的水平。然而,存储器的数据读写速度与 CPU 的处理速度之间存在差异,这种速度不匹配在很大程度上限制了计算机系统的整体性能。

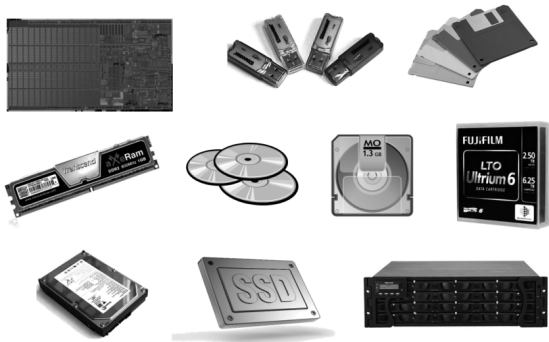


图 5.1 多样的存储器

随着 I/O 设备的数量不断增加,如果所有的信息交换都必须通过 CPU 来进行,那么 CPU 的工作效率将受到严重影响。因此,出现了 I/O 设备与存储器之间的直接存取方式(DMA),这进一步提升了存储器在计算机系统中的地位。特别是在多处理器系统中,各个处理器都需要与主存储器进行信息交换,并且在处理器之间的通信过程中,也需要共享存储在存储器中的数据。因此,存储器的作用变得尤为重要。可以说,在某种程度上,存储器的性能已经成为衡量计算机系统性能的一个核心指标。

5.1.1 存储器的分类

当今,存储器的种类繁多,从不同的角度对存储器可作不同的分类。

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

半导体存储器是由半导体器件构成的记忆设备。凭借超大规模集成电路技术,现代半导体存储器制成芯片,具备小尺寸、低功耗和快速存取的特点。它的主要缺陷是易失性,即在断电后信息会丢失。但最新研发的非挥发性半导体存储器已克服了这一缺点。根据材料类型的不同,半导体存储器可以进一步分为双极型(TTL)存储器和 MOS 存储器。双极型存储器特点是速度快,而 MOS 存储器则以高集成度、制造简便、成本低和低功耗著称,因此 MOS 存储器得到了广泛应用。

2) 磁表面存储器

磁表面存储器使用金属或塑料基底涂覆磁性材料作为记录介质。在使用时,磁层随载磁体高速旋转,磁头在其上执行读写操作。根据载磁体的形状不同,磁表面存储器可分为磁盘、磁带和磁鼓,而磁鼓已在现代计算机中较少使用。由于采用具有矩形磁滞回线的材料,这些存储器能够通过剩磁状态区分“0”和“1”,并且信息不易丢失,因此它们是非易失性的存储设备。

3) 磁芯存储器

磁芯存储器由硬磁材料制成的环状元件组成,其内部穿有用于激活的电流线和读出线,实现读写操作。由于磁芯是磁性材料,它是一种永久的非易失性存储器。然而,由于其体积庞大、制造工艺复杂和高功耗,磁芯存储器在 20 世纪 70 年代之后被半导体存储器所取代,目前已基本不再使用。

4) 光盘存储器

光盘存储器利用激光技术在磁光材料上执行读写操作。它们因高记录密度、耐用性、高可靠性和良好的互换性而日益流行于计算机系统中,是一种非易失性存储介质。

以上所述,半导体存储器属于易失性存储介质,而磁表面存储器、磁芯存储器和光盘存储器属于非易失性存储介质。

2. 按存取方式分类

存储器按照存取方式可分为以下几类:

(1) 随机存储器(Random Access Memory, RAM)。随机存储器允许在几乎相同的时间内访问存储单元中的任意位置,其访问时间与存储单元的物理位置无关。这种存储器通常用作计算机系统的主存储器。RAM 根据信息存储原理的不同,分为两大类:

- 静态 RAM(SRAM): 利用触发器存储信息,具有较快的存取速度,但成本较高,功耗较大。
- 动态 RAM(DRAM): 通过电容的充放电来存储信息,存取速度较慢,但成本较低,密度高,功耗较小。

(2) 只读存储器(Read Only Memory, ROM)。只读存储器是预先编程的存储器,在正

常操作下只允许读取数据,不允许写入新数据。它适合存放不经常变动的数据和程序,如启动固件(BIOS)或系统库。ROM的多种形式包括:

- 掩模型只读存储器(Masked ROM,MROM):在生产过程中将数据永久性地编入存储器。
- 可编程只读存储器(Programmable ROM,PROM):允许用户写入一次数据,之后变为只读。
- 可擦除可编程只读存储器(Erasable Programmable ROM,EPROM):可以使用紫外线照射来擦除数据,重新编程。
- 电可擦除可编程只读存储器(Electrically Erasable Programmable ROM,EEPROM):可以通过电信号擦除和重新编程数据,操作较为方便。
- 闪存(Flash Memory):结合了EEPROM的电擦除特点和高速读写性能,适用于大量数据存储。

(3) 串行访问存储器。串行访问存储器要求按照存储单元物理位置的先后顺序访问数据。这种存储器读/写操作的时间依赖于数据所在的位置,常见的串行访问存储器如下:

- 磁带存储器:必须从介质的开始处依次寻找数据,因此也称为顺序存取存储器。
- 磁盘存储器:属于半随机存储器,访问数据首先直接跳到一个大致区域(磁道),随后再在该区域内顺序查找数据。这类存储器结合了直接访问和串行访问的特点,因此也被称为直接存取存储器(Direct Access Storage Devices,DASD)。

3. 按读/写功能分类

存储器按照读/写功能可分为以下两类。

(1) 只读存储器(Read-Only Memory,ROM)。只读存储器是一种预先编程的非易失性存储器,其内容在生产过程中设定,并且在使用过程中不可被修改。ROM是理想的存储介质,用于存储不需要改变的程序和数据,如计算机的引导程序(Bootloader)和固件(Firmware)。ROM的内容通常由制造商在生产时编程,用户在正常使用条件下无法对其进行改写。

(2) 读/写存储器(Random Access Memory,RAM)。读/写存储器允许数据被反复读取和写入,是一种易失性存储器。在计算机运行过程中,RAM提供了临时数据存取的平台,允许系统快速访问和修改存储的信息。随机存储器是最常见的读/写存储器类型。

4. 按信息的可保存性分类

存储器按照信息的保存性质可分为易失性和非易失性两大类。

(1) 易失性存储器(Volatile Memory)。易失性存储器指的是存储器在断电后不能保持数据的类型。典型的易失性存储器包括:

- 静态随机存储器(Static RAM,SRAM):基于触发器技术,具有快速存取能力,普遍应用于高速缓存(Cache)。
- 动态随机存储器(Dynamic RAM,DRAM):依赖电容存储数据,容量大但存取速度较慢,是计算机主存储器的主要构成。

(2) 非易失性存储器(Non-Volatile Memory)。非易失性存储器指的是即使在断电情况下也能保持数据的存储器。常见的非易失性存储器包括:

- 磁存储器:使用磁性材料进行数据的长期存储,如硬盘驱动器(Hard Disk Drive,

HDD)和磁带(Magnetic Tape)。

- 光存储器：通过激光技术在光敏介质上进行数据存取，如光盘(CD)和蓝光光盘(Blu-ray Disc)。
- 非易失性随机存储器(Non-Volatile RAM, NVRAM)：结合了 RAM 的快速存取特性与数据持久保存的能力，即使在断电后也能保持信息。

5.1.2 存储器层次结构

存储器层次结构是一套设计用于协调存储容量、访问速度和成本之间关系的分层体系。此结构包含多个不同层级的存储器，主要如下。

(1) 高速缓存(Cache)：提供最快的数据访问速度，位于层次结构的顶层，距离中央处理器(CPU)最近。

(2) 主存储器(Main Memory)：通常指随机存取存储器(RAM)，在速度和容量上均衡，位于层次结构的中间。

(3) 辅助存储器(Secondary Storage)：如硬盘驱动器和固态驱动器，提供最大的存储容量但访问速度最慢，位于层次结构的底层。

1. 存储器三个主要特性的关系

存储器的性能可通过三个主要指标评估：速度、容量和成本(通常以每位价格，即位价表示)。这三个指标呈现出以下关系：

- (1) 速度与位价：速度越快，存储器制造和设计的复杂性越高，因此位价也越高。
- (2) 容量与位价：容量越大，单位存储成本越低，从而位价也越低。
- (3) 容量与速度：一般而言，容量越大，存储器的数据访问速度越慢。

尽管理想的存储器是容量大、速度快且位价低，但这三个指标之间的技术限制使得达到这一理想状态极具挑战性。图 5.2 提供了一个直观的说明，展示了这三个特性之间的相互关系和权衡。

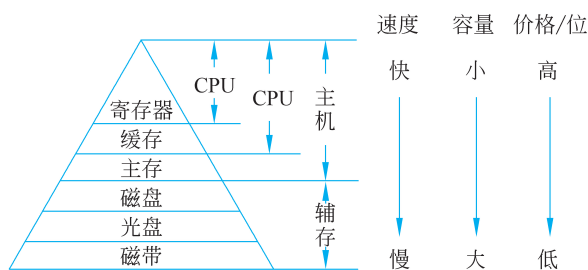


图 5.2 存储器三个主要特性的关系

如图 5.2 所示，存储器层次结构中不同层级的存储器以其速度、容量、位价和 CPU 访问频率的差异而有所区分。在这个层次结构中，从上到下，位价降低，访问速度减慢，容量增加，而 CPU 的访问频率也相应降低。

最顶层是寄存器，通常集成于 CPU 芯片内部，直接参与 CPU 的运算过程。寄存器数量有限，从几十到上百个不等，它们具有最高的访问速度和位价，但容量有限。

次一层级是主存储器(主存)，它存储即将被处理器执行的程序和数据。主存的访问速度低于寄存器，且与 CPU 的速度有较大差距。为了缓解主存与 CPU 之间速度的不匹配，

引入了高速缓存(Cache)。Cache 是一种速度快于主存但容量较小的存储器,它的位价高于主存,作为 CPU 与主存之间的缓冲区域,减少了 CPU 访问主存时的延迟。

寄存器、主存和 Cache 这三类存储器均由速度不同、位价各异的半导体存储材料制成,并且通常设置于主机内。在现代计算机架构中,Cache 也可能被整合到 CPU 芯片内。

位于层次结构底层的是辅助存储器,包括磁盘和磁带等。辅助存储器具有比主存更大的容量,主要用于存储暂时不被 CPU 直接访问的程序和数据文件。

如图 5.3 所示,存储器层次结构的设计反映了存储器技术中速度、容量、成本和访问频率之间的复杂权衡。通过这种分层,计算机系统能够以相对较低的成本提供足够的速度和容量,以满足不同的应用需求。

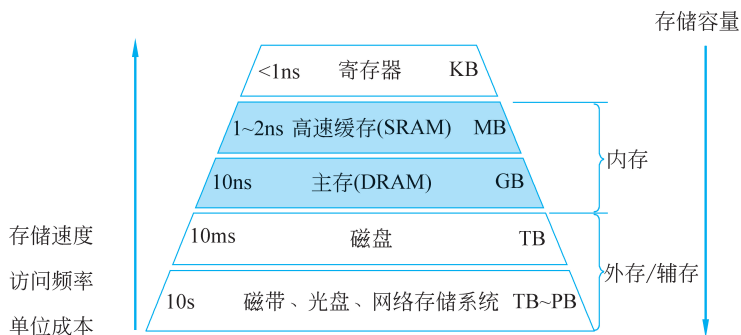


图 5.3 存储系统分层结构

在计算机架构中,主存是由半导体 MOS 存储器组成的关键组件,它负责存储和提供 CPU 执行所需的数据和指令。每个存储单元的大小通常与机器字长对应,这意味着在 32 位计算机中,主存的存储单元是 32 位宽,而在 64 位计算机中,则是 64 位宽。

当给出一个内存地址时,通过一个存储周期可以访问到对应的存储单元。值得注意的是,存储单元的大小与机器字长密切相关。以 32 位计算机为例,32 位宽的主存单元可以按照不同的粒度进行访问:可以是字节(8 位)、半字(16 位)或整字(32 位)。无论访问的数据大小如何,所需的时间都是一个存储周期,但访问的地址会根据字节、半字或整字进行相应的对齐。

具体来说,当以字节为单位访问时,可以直接使用字节地址。然而,当以半字为单位访问时,必须屏蔽掉字节地址的最低位,确保半字地址是按 2 对齐的。类似地,以整字为单位访问时,需要屏蔽掉字节地址的最低两位,以确保字地址是按 4 对齐的。这种对齐要求是因为非对齐的数据访问可能会跨越两个存储单元,导致需要两个存储周期来完成数据的读取,这在设计上是不被允许的。因此,如果出现非对齐访问的情况,计算机会产生数据未对齐异常,并中断当前的指令执行。

主存的设计和访问模式必须确保数据访问的对齐,以保证存取效率和处理器运行的正确性。这种设计原则在计算机架构的学术讨论和设计实践中都是至关重要的。32 位计算机主存示意图如图 5.4 所示。

为避免这个问题,高级语言中的变量也会考虑按存储单元边界对齐在主存中存放,也就是按边界对齐的方式存储数据。对齐存放与未对齐存放方式如图 5.5 所示。

2. 缓存、主存层次和主存、辅存层次

存储系统的层次结构是现代计算机设计中的一个核心概念,它主要包括缓存—主存层

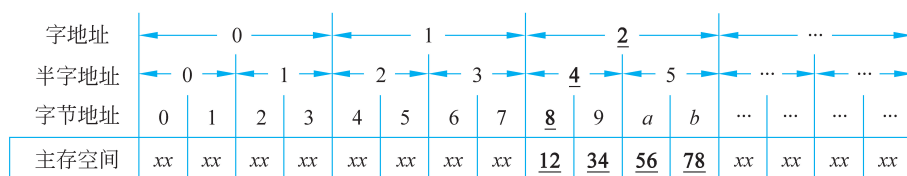


图 5.4 32 位计算机主存

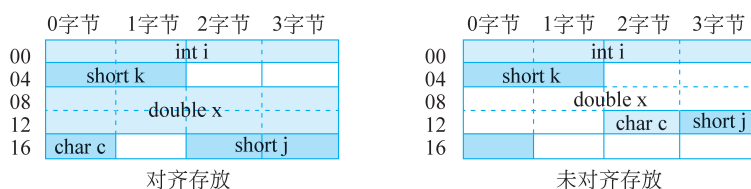


图 5.5 主存中的存放方式

次和主存—辅存层次。这种层次结构解决了处理器速度与存储器速度之间的不匹配问题，同时在成本和性能之间达到平衡，如图 5.6 所示。

缓存—主存层次的目标是缓解 CPU 与主存之间的速度差异。缓存存储器由于其更高的速度，可以存储 CPU 短期内需要的信息，从而允许 CPU 直接从缓存中访问数据，提高访问速度。由于缓存容量有限，主存中的数据必须定期调入缓存，同时旧的缓存数据被新数据所替换。缓存与主存之间的数据交换是由高级缓存管理硬件自动执行的，对程序员来说是不可见的，即“透明”的。

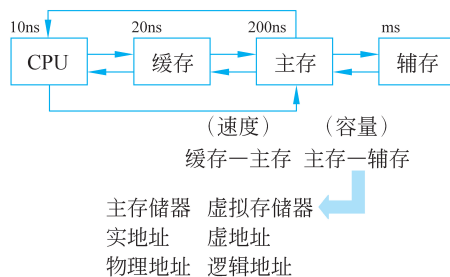


图 5.6 存储器的层次结构

主存—辅存层次则是为了解决存储容量的问题。辅助存储器（辅存）速度较慢，不能直接与 CPU 交换信息，但它提供了比主存更大的存储容量，用于存放大量暂时不需要的信息。当 CPU 需要这些信息时，它们会被调入主存，以便 CPU 能够直接访问。这个过程通常由硬件和操作系统协同完成。从 CPU 的角度看，缓存—主存层次的速度类似于缓存，而容量和成本接近主存。这样就在速度与成本之间找到了平衡点。

在主存—辅存层次，整体上看，其速度类似于主存，容量和成本则接近辅存。这种层次结构有助于解决速度、容量和成本之间的矛盾。现代计算机系统几乎都采用这种两级存储层次结构，构成了缓存、主存、辅存的三级存储系统。随着时间的推移，主存—辅存层次的不断演进促成了虚拟存储系统的发展。

在虚拟存储系统中，程序员所感知的地址空间（即虚拟地址空间）远大于物理主存的实际容量。机器指令的地址码，如 24 位，定义了虚拟存储器的地址空间，使得它大于主存的实际存储单元数量。这些地址码被称为虚地址（虚存地址或虚拟地址），而将主存的实际地址称为物理地址。在程序执行过程中，物理地址是实际访问的地址，也是真实存在的主存地址。

对于配备虚拟存储器的计算机系统而言，程序员在编程时使用的地址空间远超过实际的主存空间，这给程序员一种错觉，似乎他们有一个非常大的主存可用。这种现象正是虚拟存储器名称的由来。虚拟存储器的逻辑地址到物理地址的转换由计算机系统的硬件和操作系统自动完成，对程序员而言同样是透明的。当虚拟地址对应的内容已存在于主存中时，机

器可以立即使用该内容;如果内容不在主存中,则必须首先将其从辅存传输到主存的适当位置,之后才能供机器使用。

5.2 主存储器

5.2.1 主存储器概述

主存储器(通常简称为主存)是由一系列存储单元构成的存储系统,每个存储单元均分配有一个唯一的地址。主存与 CPU 之间通过数据总线和地址总线相连。CPU 通过发送特定的地址信号到地址总线上来定位主存中的相应存储单元,并通过数据总线执行数据的读取或写入操作。主存储器的关键技术指标包括容量、访问速度和数据宽度。其中,容量指的是主存可以存储的最大数据量,访问速度衡量的是数据在 CPU 与主存之间传输的速率,而数据宽度则指的是每次可以读取或写入的数据的位数。主存的基本组成、与 CPU 的交互机制、存储单元地址的配置以及技术指标,共同定义了主存的关键特性,这些特性为计算机系统提供了高效率的数据存储及访问功能。

1. 主存的基本组成

在实际操作中,当中央处理器利用存储器地址寄存器(MAR)中的地址访问特定存储单元时,该操作涉及一系列的电路活动。首先,地址信号经过地址译码器进行解析,以确定目标存储单元的位置。然后,通过相关驱动电路激活该存储单元。在读取操作中,选中的存储单元中的数据通过读出放大器的作用,被放大并传送到存储器数据寄存器(MDR)。在写入操作中,MDR 中的数据则必须通过写入电路才能被实际写入选中的存储单元。

以上过程确保了数据能够准确地从 CPU 传输到主存储器,并且从中读取到 CPU。这个复杂的过程是通过精细设计的硬件电路自动完成的,确保了数据访问的准确性和效率。主存的设计和实现是计算机架构中的一个关键部分,对整个系统的性能有着直接的影响。主存的实际结构如图 5.7 所示。

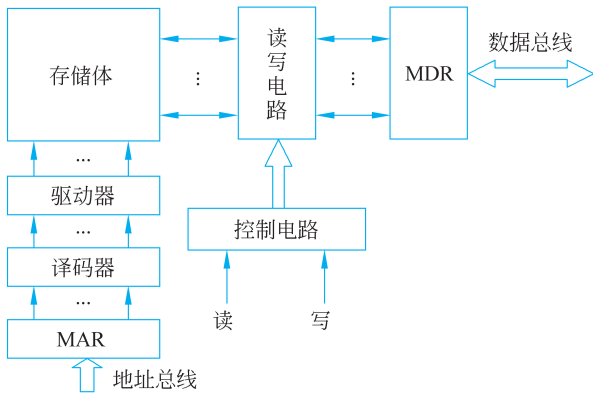


图 5.7 主存的基本组成

2. 主存与 CPU 的联系

在现代计算机系统中,主存储器通常由半导体集成电路构成,包括存储芯片内的驱动器、译码器以及读写电路。存储器地址寄存器(MAR)和存储器数据寄存器(MDR)则通常

集成于 CPU 的芯片内。存储芯片与 CPU 芯片之间的连接是通过一组总线实现的,这些总线负责地址和数据的传输,如图 5.8 所示。

在从主存中读取信息时,CPU 首先将目标数据字的地址送入 MAR,然后该地址通过地址总线传送到主存。随后,CPU 发出读取指令。主存接收到读取指令后,根据指定的地址将对应存储单元的内容读出,并通过数据总线发送。此时,信息字的后续处理(即将数据送至 CPU 内部的特定位置)不再是主存的职责,而是由 CPU 根据其内部逻辑进行管理。

在向主存写入信息字的过程中,CPU 首先将目标存储单元的地址从 MAR 发送至地址总线,并将待写入的信息字放入 MDR。接着,CPU 向主存发出写入指令。一旦主存接收到写入指令,它便将数据总线上的信息写入到由地址总线指定的存储单元中。

CPU 与主存之间的通信机制不仅体现了硬件设计的复杂性,也体现了系统在执行数据读写操作时必须保持的严格同步和协调。主存与 CPU 的高效互连对于整个计算机系统的性能至关重要。

3. 主存中存储单元地址的分配

在主存储器中,每个存储单元的物理位置是由其地址号唯一标识的。地址总线的作用是给定存储单元的地址号,根据这个地址号,存储器能够完成对一个存储字的读出或写入操作。

不同类型的计算机系统中,存储字的长度可能各不相同。为了便于字符处理,通常使用 8 位二进制数表示 1 字节,因此存储字的长度通常是 8 位的整数倍。在许多计算机系统中,可以按字(word)或按字节(byte)来进行寻址。以 IBM 370 系列计算机为例,其存储字长为 32 位,支持按字节寻址,意味着每个存储字含有 4 个可独立寻址的字节。在这种系统中,字地址通常使用该字的最高位字节的地址来表示,从而字地址是 4 的整数倍,这样就可以利用地址码的末两位来区分同一个存储字中 4 字节的具体位置。与此相反,PDP-11 系列计算机的存储字长为 16 位,字地址是 2 的整数倍,且它使用最低位字节的地址来标识字地址。

如图 5.9(a)所示,对于一个具有 24 位地址线的主存储器系统,按字节寻址时,可以访问的地址范围是 16MB(兆字节),而按字寻址时,可访问的范围缩减为 4MB。对于图 5.9(b)中的系统,尽管按字节寻址的范围仍然是 16MB,但由于存储字长短于前者,按字寻址的范围则增加到了 8MB。

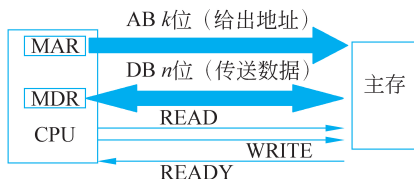


图 5.8 主存和 CPU 的关系示意图

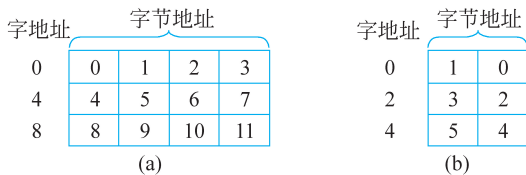


图 5.9 存储单元地址的分配

在设计主存储器地址分配方案时,必须考虑到系统的寻址能力与存储字长度之间的关系,从而确保存储器的空间能够被有效利用,并且可以通过地址线准确地访问每个存储单元。这种设计对于优化存储器的性能和存取效率至关重要。

4. 主存储器的技术指标

主存储器的关键技术指标包括存储容量和存储速度,这两项指标决定了存储器的性能和效率。

1) 存储容量

存储容量是指主存能存放二进制代码的总位数,即存储容量=存储单元个数×存储字长。

存储容量也可用字节总数来表示,即存储容量=存储单元个数×存储字长/8。

在当代计算机系统中,存储容量通常以字节为单位表示。例如,若一个计算机系统的主存储容量为 256MB,那么按字节寻址所需的地址线位数应为 28 位,因为 2^{28} 字节等于 256MB。

2) 存储速度

存储速度指标包括存取时间和存取周期两个方面。

- 存取时间(Memory Access Time):是指启动一次存储操作(读取或写入)到完成该操作所需的时间。它包括读出时间和写入时间:读出时间是指从存储器接收到有效地址到产生有效输出所需的时间;写入时间是指从存储器接收到有效地址到数据被成功写入选中单元所需的时间。
- 存取周期(Memory Cycle Time):是指存储器进行连续两次独立存储操作(例如,连续两次读取操作)所需的最短时间间隔。通常,存取周期是大于存取时间的。

现代存储器的存取周期可达到不同的速度。例如,MQS 型存储器的存取周期可以是 100ns,而双极型 TTL 存储器的存取周期可能接近 10ns。

3) 存储器带宽

存储器带宽是另一个与存取周期密切相关的指标,它表示单位时间内存储器能够存取的信息量。带宽可以用字/秒、字节/秒或位/秒表示。例如,若存取周期为 500ns,且每个存取周期能够访问 16 位信息,那么带宽为

$$\text{带宽} = 16 \text{ 位} / 500\text{ns} = 32\text{M 位/秒}$$

带宽是一个衡量数据传输速率的重要指标。存储器带宽决定了以存储器为中心的系统获取信息的速度。提高存储器带宽是解决系统瓶颈问题的关键。为了增加存储器带宽,可以采取以下措施:

- (1) 缩短存取周期。
- (2) 增加存储字长,以便每个存取周期能够读取或写入更多的二进制位数。
- (3) 扩展存储体的数量或宽度。

这些指标对于理解和评估存储器性能至关重要,并且对于存储器的设计和应用有着直接的影响。

5.2.2 半导体存储芯片概述

1. 半导体存储芯片的基本结构

半导体存储芯片的核心结构之一为控制线,主要分为读/写控制线和片选线。这些控制线在不同的存储芯片中可能呈现不同的配置。

(1) 读/写控制线:这些线路负责指示芯片进行读取或写入操作。例如,2114 型芯片的读/写控制线可能只用一根线共享,而 6264 型芯片可能分别用两根线来控制读和写操作。

(2) 片选线:片选线被设计用来选择或激活特定的存储芯片。一些芯片,如 2114 型,可能只需要一根片选线,而如 6264 型芯片可能需要两根片选线。

在一个半导体存储器系统中,常常需要多个芯片协同工作以提供大容量存储。这种情况下,片选信号便显得尤为重要,它确保了系统能够识别并选中正确的芯片进行数据存取。例如,一个存储器系统可能包含 32 片单位的存储芯片,如图 5.11 所示。然而,在读取一个存储字时,只有 8 片芯片被选中参与此次数据访问。

如图 5.10 和图 5.11 所示,展示半导体存储芯片的结构,包括控制线、地址线、数据线以及存储单元的布局等。

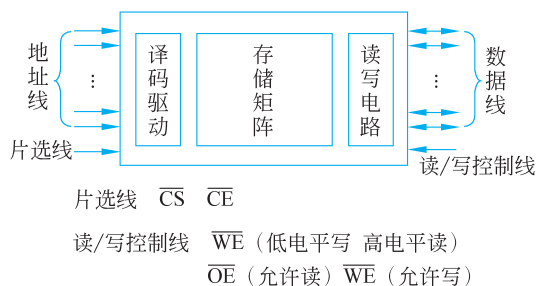


图 5.10 半导体存储芯片的基本结构

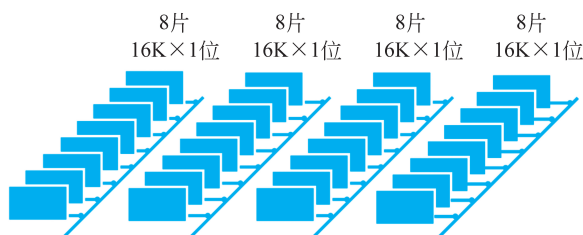


图 5.11 65K×8 位的存储器

2. 半导体存储芯片的译码驱动方式

半导体存储芯片的译码驱动方式有两种：线选法和重合法。

1) 线选法

如图 5.12 所示,是一个 16×1 字节线选法存储芯片的结构示意图。它的特点是用一根字选择线(字线),直接选中一个存储单元的各位(如 1 字节)。这种方式结构较简单,但只适于容量不大的存储芯片,如当地址 A_3 、 A_2 、 A_1 、 A_0 为 1111 时,则第 15 根字线被选中,对应图中的最后一行 8 位代码便可直接读出或写入。

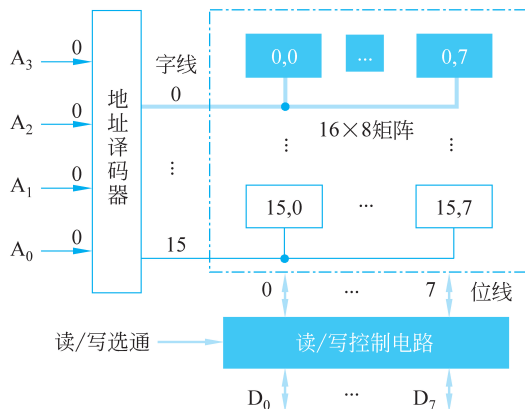


图 5.12 线选法示意图