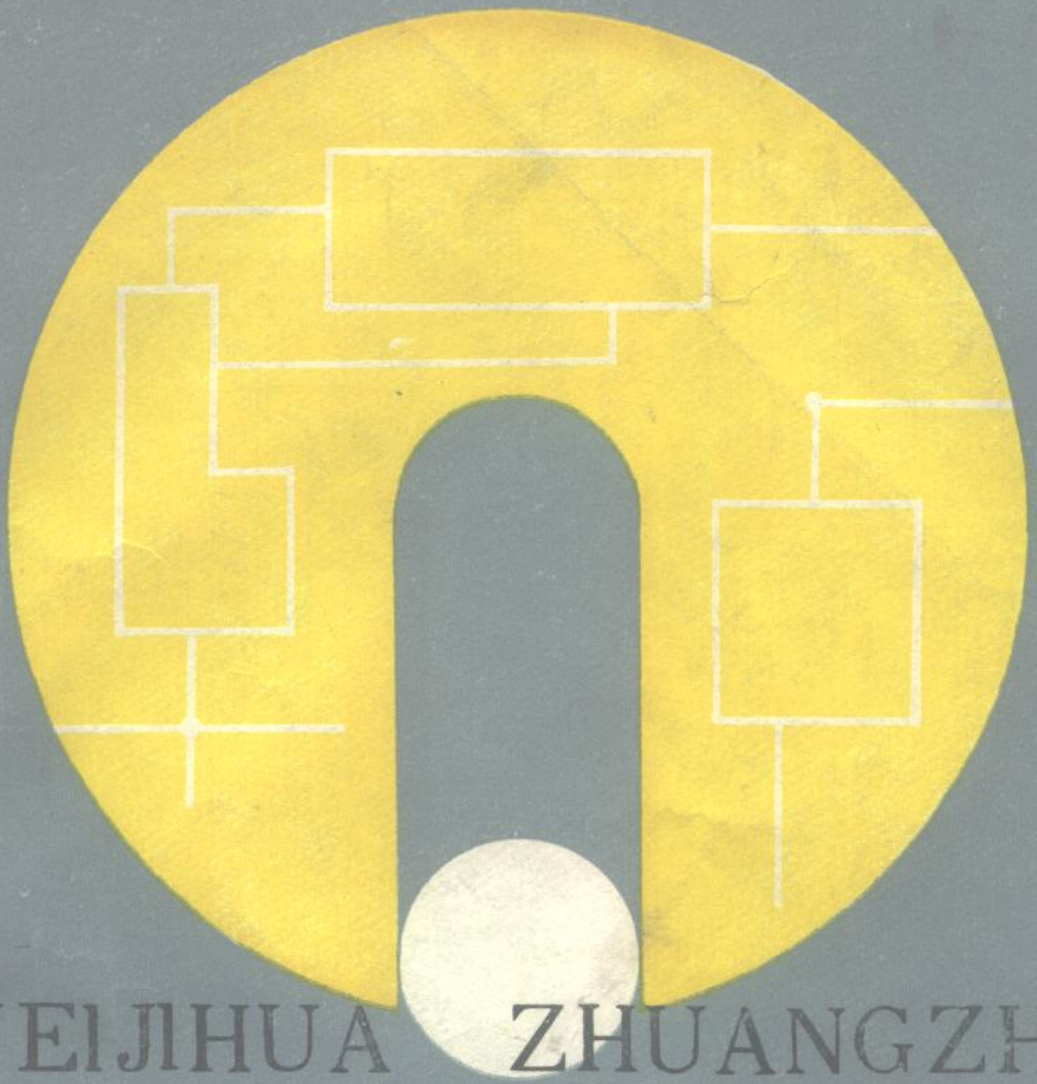


WEIJIHUA ZHUANGZHI
JIEKOU YU SHEJI

● 吴报鑫　张华宋 编著

● 上海交通大学出版社

微机化装置接口与设计

微机化装置接口与设计

吴报鑫　张华宋　编

上海交通大学出版社

内 容 简 介

本书着重从应用角度介绍微机化装置中的存储器、并行、串行、DMA、模拟转换等接口的硬件、软件典型设计方法以及汇编语言程序设计的一般规律与实用子程序库。既注意到内容的深化，也顾及到与微机基础知识的衔接。可作为大专院校有关专业的教材及从事这方面工作的科技人员的学习参考资料。

微机化装置接口与技术
上海交通大学出版社出版
（淮海中路 1984 弄 19 号）
新华书店上海发行所发行
常熟文化印刷厂印刷

开本 787×1092 毫米 1/16 印张 17.5 字数 430,000
1988 年 3 月第 1 版 1988 年 5 月第一次印刷
印数：1—5,000
ISBN 7-313-00113-4/TP 39 科目：168—300

定价：2.95 元

前　言

当前，以微机为主体的一代新型仪器装置已广泛应用于各种控制、测量、监测系统。在这类应用中，微机与外部世界进行信息交换是通过各类接口实现的。了解接口的原理和对接口的合理设计是充分发挥微机应用效能的关键。为使广大工程技术人员、科技工作者及有志于微机应用的同志能在了解微机原理的基础上尽快地掌握有关接口技术的知识，我们编写了这本书。

全书共分九章，主要阐述了接口的硬件构成和软件设计。既说明了接口技术的一般规律，也介绍了具体接口芯片及其应用。对程序设计方法和常用算法作出了分析，并提供了有实用价值的子程序库。

为使本书重点突出，避免篇幅冗长，书中对微处理器的原理、结构、指令系统、基本时序不再赘述。对书中阐述的各章内容力求避免泛泛介绍和广为罗列，仅以 Z80 系列为代表作深入介绍。书末附有 Z80 指令系统相应代码，常见术语的英汉对照及参考文献。

本书可作高等院校有关专业的教材，也可供有关专业人员参考。

由于水平有限，书中难免有错误或不足之处，恳请读者批评指正。

编　者

一九八七年九月

目　录

第一章 概 论

随着大规模集成电路的发展，微电脑技术已得到广泛的应用，并深入到各个领域。以微机为主体的一代新型的仪器、装置、设备，广泛应用于各种工业控制、测量、监测系统中。这些装置在性能上具有可编程序的自动化能力，能在程序控制下实现操作。对信息有自动提取、计算、处理、变换的能力；对工作结果有判断、分析、修正、显示、记录的能力；对装置本身有自检、自校、自诊断、自修复的能力。因此能完成高精度，高效率地测试或控制的任务。

现在人们常称这类带有微机的新型仪器装置为“智能化仪器装置”。严格说来，“智能”一词含有视觉方面的图形、色彩判别；听觉方面的语言和语音识别；思维方面的推理、判断、联想等等。能形成类似人的一部分智力劳动的能力。目前，已有不少以微机为基础设计制造的装置，但它们在这方面的能力还比较有限，为此，我们用“微机化装置”来广义地称呼这一类新型的仪器设备装置。

一、典型的微型计算机系统与接口

在微机的各种应用中，通常由微处理器(CPU)、存储器和外部设备组成各种微型计算机应用系统。

对于科学计算、企业管理等应用领域，一般采用通用微型计算机系统。系统中配置的外部设备主要是键盘、CRT 显示器、打印机、磁带、磁盘等，用于实现操作者与微机之间的人机对话，在人机之间交换、存取、显示或记录各种信息。这类应用主要采用高级语言编制软件。

对于“微机化装置”的应用领域，由于应用面广，涉及的控制、检测对象各不相同。在系统中除了配置供人机交换信息的外部设备以外，还需提供检测、控制对象与微机交换信息的外部设备。例如，各类传感器，各类执行机构、开关、指示器、显示器等等。所以需采用适合各具体应用场合的专用微机系统。这类应用的软件中较多地采用汇编语言或高级语言与汇编语言交叉使用。

在各种微机系统中，微处理器是系统的控制和处理中心，它将取自外部世界的信息进行存储、处理后，再去控制客观世界。CPU 在工作时不断地与存储器、各种外部设备进行联系，交换信息。外部设备的种类有机械式、电动式和电子式；交换信息的型式有数字量、模拟量或开关量。信息传送速度可以高达几十万 bit/s，也可以慢到秒、分或更长时间传送一次信息。信息传送方式可以是并行方式或串行方式。因而 CPU 与外部交换信息要比与存储器交换信息复杂得多。为了适应各种外部设备的需要，CPU 与外部设备之间除个别能直接联接外，大部分需要有特定的硬件连接和相应的软件控制。微机系统中，CPU 与外设之间这种特定的硬件连接与相应的控制软件总称为接口。对这些硬、软件的设计称为接口技术。

一个典型的微型计算机应用系统中，硬件一般包含微处理器，存储器、总线、接口、外设及电源。

按照不同的功能，接口可分为四种类型，用户交换接口、辅助操作接口、传感接口和控制

接口。

用户交换接口的主要功能是将来自用户的数据、信息传送给微型机，或将来自微机的数据、信息传送给外部设备。常见的打印机接口、键盘接口、终端显示接口属这一类。

辅助操作接口包括总线驱动器、总线接受器、时钟电路、磁带和磁盘系统接口。一般情况下，CPU 的扇出仅是 1～2 个 TTL 负载。如果 CPU 所带的存储器或接口较多，则 CPU 的驱动能力将不足。为使信息有效地可靠地传送，需要加接总线驱动器、总线接受器。对于大容量的外存储器，如磁带、磁盘和软盘系统的接口一般也归于这一类。

传感接口一般用于微机控制和检测系统中。由于客观世界是一个模拟世界，微机要控制和检测的外部信息例如压力、温度、流量、变形等一般都是模拟量，而微机所能接受和处理的都是数字量，因此常常通过传感接口去监视、感受外部世界被检测或控制对象的变化，将这种变化转换成电压或电流变化，再进一步转换成微型机能接受的数字量。

控制接口是当微机进行处理、运算后，需要控制外部执行机构动作时（例如驱动马达、启动阀门或点亮指示灯等），把数字量转换成模拟量，以实现对外部世界的控制。

接口单元是与微处理器，存储器等通过总线接在一起面向外设的，而不是与外部设备接在一起面向微机的。

在接口单元上，可与外设连接的部分称为“口”或“端口”。也称输入/输出(I/O)“口”或输入/输出(I/O)“端口”。

根据传递通道的数目，数据可以在外设与 I/O 口之间以三种方式传送：

1. 全双工方式　数据可在两个方向上同时传送；
2. 半双工方式　数据可在两个方向上传送，但一次只能在一个方向上进行；
3. 单工方式　数据只可在一个方向上传送。

全双工方式需要两个通道，而半双工及单工方式仅需要一个通道。

二、微机化装置中的典型接口方法

CPU 通过接口与外设之间的连接有各种不同方法。根据这些方法接口可分成：通用并行接口、异步串行接口、同步串行接口、标准通信接口、仪器接口和专用接口。图 1-1 列出了这些不同连接方法的接口。在一个系统中通常不需要用全部的接口方法，往往只需要其中的一种或几种。另一方面，有些外设则又能用同一类型的接法加以接口。在一个较小的应用系统中可能仅需 1～2 个并行 I/O 口或 1～2 个串行 I/O 口，但较大的系统也可能需十几个甚至几十个 I/O 口。

当外部设备直接与微机总线相兼容时，它可直接连向总线而不需要任何接口电路，如图 1-1最左边部分。最常见的例子是在微机系统中接入存储器扩充板。

并行接口中数据传送是多位同时传送的，接口的基本组成部分是双向数据寄存器、状态寄存器、控制寄存器、控制逻辑和地址比较器。通过对控制寄存器的设置可使这个口作输入口或作输出口。现在，已出现很多 LSI 构成的可编程并行接口芯片。可在 CPU 控制下由编程确定它的各种接口功能。

异步串行接口中，数据是在一条数据线上逐位依次传送的。这种接口常用于只有一条数据线的外部设备，或者用于计算机与远程外设的接口。接口中一般采用移位寄存器实现并行与串行之间的转换。在串行接收和发送时，分别需要外部时钟实现工作的同步。发送

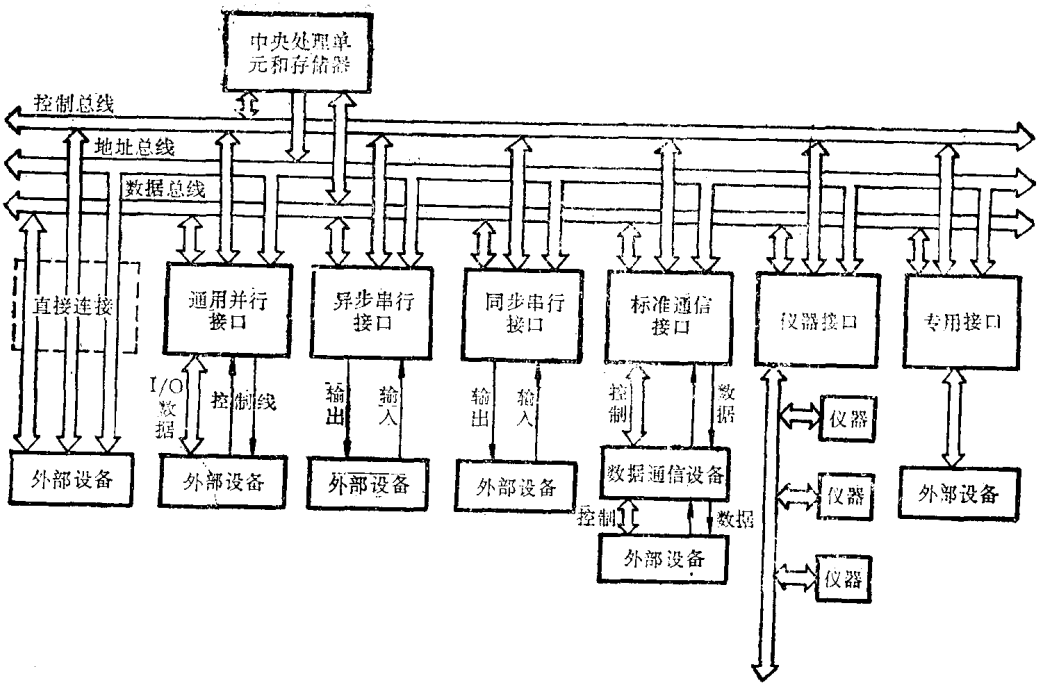


图 1-1　各种接口法方

部分(计算机输出)和接收部分(计算机输入)可以是独立的，两者可同时工作。通常对于由 LSI 电路构成的异步串行接口称为通用异步接收/发送器(UART)。

同步串行接口与异步串行接口的不同在于不需要外部时钟。但对于发送部分需要同步字符发生器，接收部分需要同步字符识别器，用以产生和识别同步数据传送中的同步字符。

标准通信接口最常见的是 RS-232-C。图 1-2 表示用于外部设备和计算机之间的 RS-232-C 标准接口。接口中包含控制逻辑及串行发送器和接收器，通过 25 芯接插件和外设相连。当外设与接口之间距离很远时，需通过调制解调器来防止信号的畸变和衰减。

最常见的仪器接口是 IEEE 488，如图 1-3 所示。该接口总线上可以连接多达15个仪器设备，组成一个计算机控制的仪器系统，系统中有控者、讲者和听者。通常用 24 芯接插件实现各信号的连接，其中有 8 条是数据线，3条数据传送控制线和 5 条通用接口管理线。

关于专用接口，是一些计算机生产厂为某些最适宜与他们的计算机系统配套的外部设备专门设计的接口。例如，打印机接口一般可以用通用并行接口来实现，但也可以设计一个专供打印机用的接口，这样做的好处是这个接口可以针对打印机所需的信号设计，而不必兼有顾通用性，因而简单、便宜。

三、微机化装置接口设计的一般方法

在微机化装置的接口中，需解决以下几方面的问题：

1. 输入　把外部设备送往微机的信息转换成与微机相容的格式。

2. 输出　把微机送往外部设备的信息转换成与外设相容的格式。这里的相容是指在信号类型、信号电平、信号传送方式等方面使外设与微机之间取得协调。

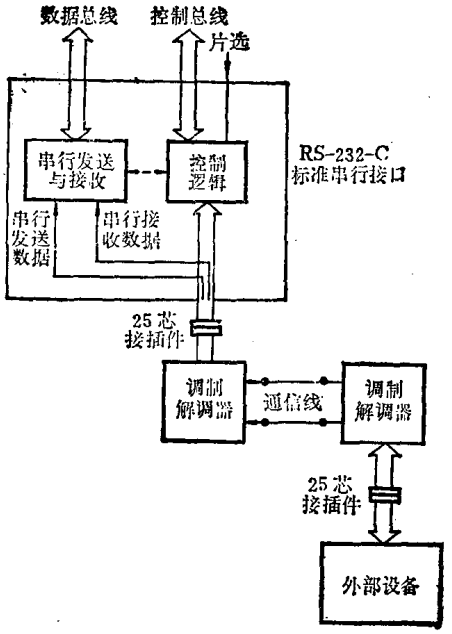


图1-2 RS-232-C标准接口

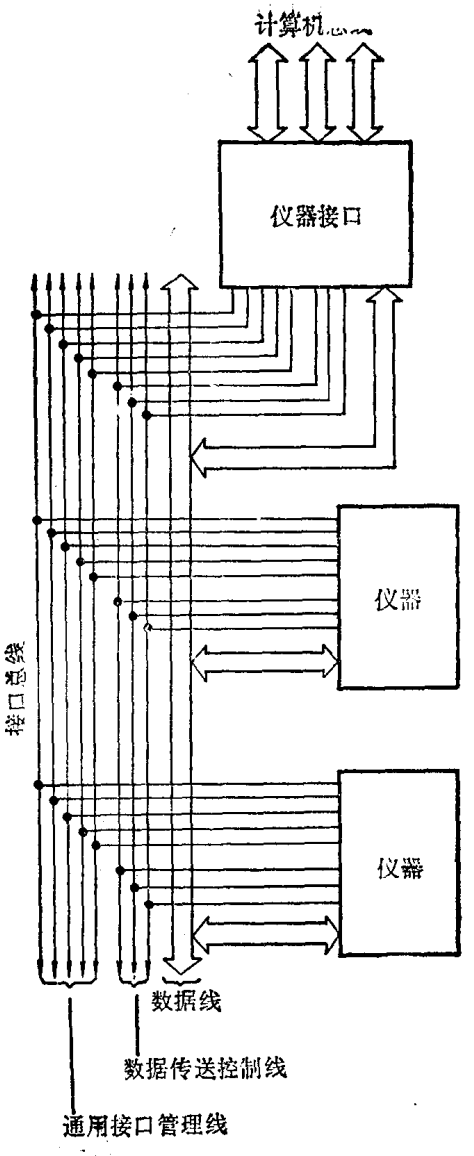


图1-3 IEEE488仪器接口

3．同步脉冲的产生 通常外部设备不止一个，微机在不同时刻和不同外设取得联系时，必须对外设寻址。一般需产生设备选择脉冲，以使相应外设与微机同步工作。

4．中断处理 检测并处理外部设备发到微机的中断请求信号。

实际上，一个完整的接口设计，包括机械的、电气的和功能的几方面要素：机械的要素包括接插件、电缆等；电气的要素包括发送与接受电路、信号形式、电平等；功能的要素包括接口的管理能力、接收、发送、控制等功能的逻辑，信息系统的编程等。

综上所述，为使微机有效地与客观世界或用户进行联系，充分发挥它的效能，各类"微机化装置"的设计重点已从以往的逻辑电路的设计调试转向专用的微机系统设计，这主要表现在对接口的硬件设计和相应的应用软件的开发。

随着大规模集成电路工艺水平的不断提高，微型机接口芯片的发展也很快。除了各种功能芯片外，已有各种功能模块出现。并进而又向着把CPU、存储器、接口等集成在一块芯

片上的单片机和专用微处理机发展。它们使接口设计中硬件设计的比重减小而软件设计比重相应增大，但不能全部替代应用中的接口设计工作。

设计中，既需要掌握各类芯片及设备的有关基本知识，也需理解装置中提出的各种算法过程，并能将这些算法过程编制成微型机应用程序。多个算法程序可集成一个子程序库供调用。设计时要善于将复杂的设计任务划分成许多便于实现的组成部分，选择和组织硬件，并在硬件与软件之间按需要进行协调。

研制一个微型计算机应用系统都有一个调研阶段，要根据系统所要完成的任务和目标、必须满足的性能指标、系统的成本等因素，仔细评价各种可供选择的微处理器，探索各种比较方案。通常，系统所要完成的许多功能，既可用硬件来实现，也可用软件来实现。例如，多个I/O设备的中断处理问题，如果用软件来实现，则中断处理所用的查询程序，不需要任何硬件逻辑的支持，只要求有一条“中断请求”线。但是，如果加上一个向量中断管理逻辑，则中断服务便会快得多。到底选用哪一种办法，取决于各种因素，如速度等。因此，必须在软件和硬件之间作出权衡。通过权衡比较，基本上确定是使用硬件还是使用软件为主，而且所要用的微处理器也要选择好。那么，主要的硬件特点如I/O器件、存储器大小和总线结构也能确定下来。这时，构成系统硬件的设计工作和编写软件的工作可并行进行。在大多数情况下，硬件设计相对地简单些，因为微处理器制造厂现在都能提供整套相互兼容的集成电路，可以按规定的方式连在一起构成一个系统，而对硬件研制的主要问题是测试问题。由于有了逻辑分析仪之类的硬件研制工具，大大方便了测试过程。在软件研制方面，重点是把要解决的应用问题抽象成算法，再用汇编语言或高级语言编写出相应的源程序，也就是用户需要研制的应用程序。通过调试，发现问题后，还必须进行修改，重新调试，经多次反复方能完成。

硬件的研制过程可归纳如下：

1．为系统每一个模块绘出详细的逻辑图(例如，控制模块、存储器模块、I/O接口模块等)，还要包括一些重要控制信号的定时图。

2．根据价格，速度和功耗选择系统所需要的各类器件。

3．将每一个模块都装配在测试板上(试验电路板)，便于调试和修改。

4．测试电路，描绘出信号流并与逻辑图和定时图进行比较，这可能要重复多次才能判明并改正设计和布线上的错误。

5．将各类模块组成一个完整的系统叫样机。在软件装入的前后，都要检测样机的硬件错误。

软件的研制过程可归纳如下：

1．如果要编写的软件比较复杂，可分为几个子程序以便几个人同时工作，这可以加快编写软件的速度并且易于调试和修改。

2．为每一个程序模块设计一个详细的流程图。

3．根据流程图，用汇编语言或高级语言编写程序。

4．调试每一个程序模块，如果检测到是逻辑上的错误，便应修改源程序并继续进行调试。

5．各程序模块调试好以后，将它们连接起来形成一个完整的软件包，然后，将该软件包作为一个整体进行调试。

软件的研制，不必等样机硬件构造好之后才进行，只要系统结构基本确定之后，软件和硬件便可并行研制。

最后一步是将样机硬件和软件合在一起作整体调试，在确认整个系统已没有错误之后，就可用印刷板来代替试验板。

第二章　存储器的接口方法与扩充

第一节　存储器概述

一、分类

存储器(Memory)是计算机的重要组成部分，用于存储二进制信息，从而使计算机具有记忆功能。当使用者将有关指令、数据、信息分别存入存储器后，计算机便可按人的意志自动地、有条不紊地工作。

按存储器与主机的关系，存储器有内存储器与外存储器之分。在主机内部的存储器称为内存储器，又叫"内存"。通常内存容量小，存取速度较快，可与 CPU 直接交换信息。在主机外部的存储器称外存储器，又叫"外存"。它属于计算机的外部设备。常见的外存有软磁盘、硬磁盘、磁带机等。它们的特点是容量较大，存取速度慢，不能直接与 CPU 交换信息，必须将外存的信息调入内存后才能被使用。

本章主要讨论微型计算机的内存储器接口。

大型或小型计算机的内存，以前通常采用磁芯存储器。近年来，随着大规模集成电路的发展，在大型或小型计算机中已开始采用半导体存储器。而微型计算机几乎全部采用半导体存储器。这种存储器具有体积小、重量轻、耗电少、速度快及非破坏性读出(即读出以后信息仍旧存在)等优点。

常见半导体存储器按其功能可分为只读存储器 ROM(read only memory)和随机存取存储器 RAM(random access memory)。

ROM 是只能读出的存储器，其中的信息在使用时不能改变，即使用时无法对它写入。当关断电源时，ROM 中的信息不会消失，通电后又可使用。因此 ROM 常用来存放永久性的程序、如监控程序、管理程序、调试、诊断程序、专用操作程序等。也可存放不变的常数和各种表格。

RAM 在使用时可以被读出，也可以被写入或对它进行改写。所以 RAM 又称读写存储器。它常被用来存放需要改变的程序或数据。当切断电源后，RAM 中存放的信息在几毫秒后便消失，电源恢复后，信息不再恢复，必须重新写入。即存在信息的易失性。解决半导体 RAM 信息遗失问题的一个方法，是设置后备电源。当主电源掉电时，备用电源的控制电路应保证在很短时间内把备用电源接到半导体 RAM 上去。使存在 RAM 中的信息不致丢失。一般的高能电池已足以作为备用电源。除了使用备用电源以外，也可采用新的工艺，如 MNOS (金属氮化物氧化物半导体)工艺、生产非易失性的存储器。这样，在一般的 RAM 中，设置一个 MNOS 工艺的存储器，一旦主电源掉电，就可以利用与主电源并联的电容器在短时间存储电荷的特性，在很短时间内把信息转存入非易失性的存储器中。

按存储信息的方法，ROM 和 RAM 还可作如下分类：

- ROM
  - 掩膜 ROM
  - 可编程 ROM(PROM)
    - 熔丝型 PROM
    - 可擦除 PROM(EPROM)
    - 电可改写 ROM(EEPROM)

RAM ┌静态 RAM
　　└动态 RAM

掩膜 ROM 由制造厂按用户给定的信息要求，掩膜制造而成，封装后不能改写，即用户只能读出无法改写。可编程 ROM(PROM)中信息可由用户根据需要写入。熔丝型 PROM 是用户把熔丝熔断或不熔断来表示"1"或"0"。信息写入后，不能再改写。可擦除 PROM，(即 EPROM)信息写入后，可用紫外线擦除，然后重新改写。但擦除或改写是需要设备和时间的，所以它们在装上微机应用时仍作只读存储器使用。EEPROM 可用电的方式擦除和改写。

静态 RAM 的每一位信息存储在一个触发器中。电源存在时，信息不会丢失。但电源失去几毫秒后，RAM 中的信息就会消失。动态 RAM 是利用 MOS 晶体管栅极与基底之间的寄生电容上的电荷来存储信息，这种电荷在几毫秒后便会漏失，所以每隔一定时间必须重新写入一次以保持原来的信息，这一重写过程称为刷新或再生。动态 RAM 的单位面积集成电路中存储信息较多，存取速度也比较快，功率耗散小，但需附加刷新电路。所以一般大容量存储器采用动态 RAM，小容量存储器则用静态 RAM。

根据半导体存储器的制造工艺，存储器还可分为双极型和 MOS 型两类。

综上所述，存储器的分类可归纳如表 2-1 所示。

表 2-1　存储器的分类

	按功能分类	按工艺分类	按存放信息方式分类
内存储器	随机存取存储器 (RAM)	双极型	静态 RAM
		MOS 型	静态 RAM
			动态 RAM
	只读存储器 (ROM)	双极型	一次掩膜成型 ROM
			熔丝型 ROM
		MOS 型	一次掩膜成型 ROM
			EEPROM(电可改写的 ROM)
			EPROM(可擦可编程的 ROM)
外存储器	软磁盘，硬磁盘，磁带，磁鼓		

二、存储器芯片举例

半导体存储器通常都做成双列直插式封装(DIP)的芯片。芯片中存储信息的每一个单元都对应一个地址。使用时是按地址号来选择各个存储单元的。对常见的具有 16 条地址线的微型机可以有 64K(2^{16} = 65536)寻址单元。即 16 条地址线具有 64 K 存储单元寻址能力。存储单元的地址号一般以 4 位十六进制数表示，64K 存储单元的地址范围是 0000H～FFFFH。

存储器芯片的容量一般用可寻址单元数×每单元数据位数表示。例如，1K×1(1024×1)表示有1024个可寻址单元，每个单元存放 1 位二进制数；2K×8(2048×8)表示有 2048 个

可寻址单元，每个单元存放 8 位二进制数。1K×4(1024×4)有1024个存储单元，每一单元存放 4 位二进制数等等。常见的 RAM 2114 的容量是 1K×4，这表明一块 RAM 2114 芯片有 1024 个可寻址的存储单元，每个存储单元有 4 位二进制数。即一块 RAM 2114 芯片可存储 1024×4=4096(4K)位信息。这一芯片需具有 10 条地址线和 4 条双向数据线。又如 EPROM 2716 的容量是 2 K×8。这表明一片 EPROM 2716 芯片有 2048 个可寻址的存储单元，每个存储单元有 8 位二进制数。即一块EPROM 2716 可存储 2048×8=16384(16K)位信息。这一芯片需具有 11 条地址线和 8 条双向数据线。

对于同样存储容量的存储器芯片可以有不同的做法。例如对于存储容量是 1024 (1K)位信息的存储器，可以做成有 1 K 个地址，每一地址只有 1 位二进制数，写成 1024×1。也可做成有 128 个地址，每一地址有 8 位二进制数，写成 128×8。对于前一种做法，1 K 个地址要 10 个地址引脚，还要一个双向数据引脚，共计 11 个引脚，对于后一做法，128 个地址要 7 个地址引脚，还要八个双向数据引脚，共计 15 个引脚。由此可知，存储器做成 1024×1 比做成 128×8 的引脚少。引脚少制造方便，所以存储器的数据往往做成 4 位或 1 位而较少做成 8 位。字长不足 8 位的存储器与 8 位的 CPU 相连时，可把几片存储器并联成 8 位数据。各类存储器的芯片除了必须有地址线，数据线以外，还需要有控制线。

存储器芯片中的存储单元排成矩阵形式，或者说排成阵列。例如，1024个存储单元可排成 32×32 (=1024) 的矩阵，即 32 行 32 列的矩阵。2048 个存储单元可排成 128×16(=2048)的矩阵等等。对存储器的寻址是通过行地址选择线和列地址选择线的重叠来实现的，如图 2-1 所示。这样做可以节省译码和驱动电路。以 1024 个存储单元而言，行地址译码器和列地址译码器各只需 32 条输出线，即总共译码输出线是 64 条。若不采用阵列，则译码输出线就需要 1024 条。

图 2-2 是 RAM2114 的框图及引脚定义。由图可见，芯片共有 18 只脚，除工作电源及地线外，还有 10 条地址线，4 条双向数据线，2 条控制线。其中$\overline{WE}$为写入允许控制，$\overline{CS}$为

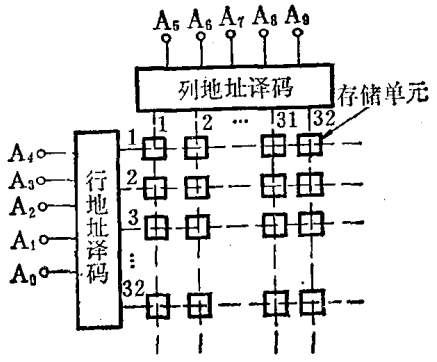


图 2-1 存储单元阵列的重叠选址

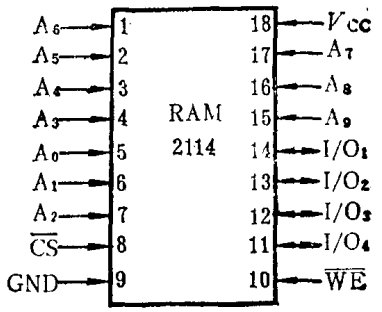


图 2-2 RAM2114 引脚图

表 2-2 RAM2114 控制脚$\overline{CS}$和$\overline{WE}$的逻辑关系

$\overline{CS}$	$\overline{WE}$	芯片状态
1	×	禁止
0	0	写入
0	1	读出

片选控制。它们的逻辑关系如表 2-2 所示。

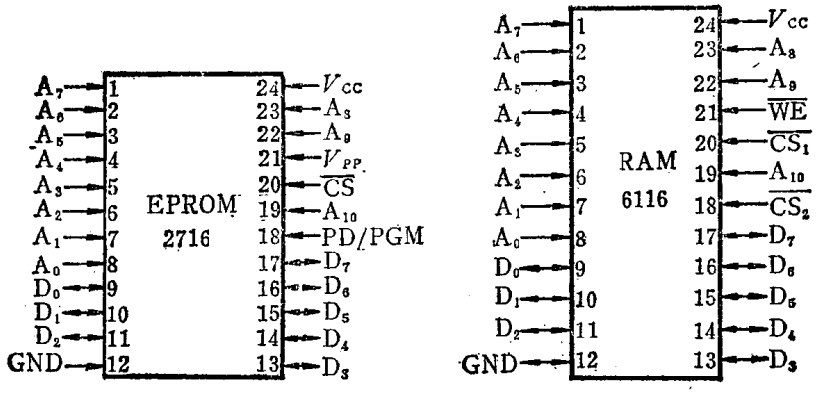
当 $\overline{OS}$ 是高电平时，芯片被禁止读写。当 $\overline{OS}$ 是低电平时，芯片才允许读写操作。当 $\overline{WE}$ 是低电平时，芯片处于写入状态；当 $\overline{WE}$ 是高电平时，芯片处在读出状态。

图 2-3(a) 示出了常用的 EPROM 2716 的框图及引脚定义。该芯片共有24条引脚，其中 11 条地址线，8 条双向数据线，11 条地址线中的 7 条用于行地址译码，以选择 128 行中的一行。4 条用于列地址译码，以选择 16 列中的一列。1 条片选线 $\overline{CS}$。V_{pp} 引脚在读操作时接 +5V，在编程操作时接 +25V。PD/PGM 引脚在读操作时可接低电平，在编程时需加上宽度为 50～55ms 的 TTL 高电平脉冲。在芯片未被选中时，PD/PGM 引脚可接高电平，这时芯片功耗可下降约 3/4，由 525 mW 下降为 132 mW。$\overline{CS}$、V_{pp}、PD/PGM 引脚在不同工作方式时的状态归纳见表 2-3。

图 2-3(b)示出了 RAM 6116 的引脚图。RAM 6116 容量是 2 K×8。每一块芯片可存储 2K 字节。使用时，片选端 $\overline{CS_1}$ 和 $\overline{CS_2}$ 均接向地址译码器；$\overline{WE}$ 端接向 $\overline{MEMW}$ 信号，在写操作时 $\overline{WE}$ 为"0"，读操作时 $\overline{WE}$ 为"1"。

表 2-3　EPROM2716的工作方式选择

工作方式	引脚			
	PD/PGM	$\overline{CS}$	V_{PP}	输出
读	低	低	+5V	输出
编程	50～55ms 宽 TTL 正脉冲	高	+25V	输入
未选中	无关(若接高，则可降低功耗)	高	+5V	高阻



(a)　　(b)

图 2-3　EPROM 2716 和 RAM 6116 引脚图
(a) EPROM 2716 引脚　(b) RAM 6116 引脚

RAM 6116 的一个重要特点，是它的引脚数与 EPROM 2716 相同，使用时的引脚接法与 EPROM 2716 读操作时引脚接法兼容。因此，应用 RAM 6116 调试好的程序，可固化到 EPROM 2716 中，然后将 EPROM 2716 替换 RAM 6116 而不必更改任何接线。使用很方便，因而 RAM 6116 的应用已越来越广。

第二节　存储器与 CPU 的接口方法

CPU 对存储器进行读写操作时，首先由地址总线给出存储单元的地址号，然后给出相应的读或写控制信号，最后才能在数据线上进行信息交换。所以 CPU 与存储器接口时，需要连接地址线、控制线和数据线。具有不同数量存贮单元的存储器芯片与 CPU 连接时，所需连接的地址线数目也不同。例如，有 1 K 存储单元的芯片需用 A_0～$A_9$10 条地址线。有 2K存储单元的芯片需用 A_0～A_{10}11 条地址线等等。

一般用于 8 位微机的存储器字长也应该是 8 位。当存储器字长不足 8 位时，可用几片存储器芯片组合成 8 位字长。例如，RAM 2114 芯片是 1 K×4，只有 4 条数据线，所以要用两片合在一起才能组成 8 位字长，共同在 1 K 地址范围内工作。由两片 2114 芯片共同形成 1 K 容量 8 位字长的存储器与 CPU 接口的示意图如图 2-4 所示。它们的地址线是并联在 CPU 的 A_0～A_9 地址线上。数据线分别接在 CPU 的 D_0～D_3 和 D_4～D_7 上。当地址线上出现某一地址时，两片 2114 选中同一个单元地址，它们的数据线共同形成一个字节，在片选和读写信号控制下与 CPU 交换信息。对于具有 8 位数据线的存储器，例如，2716，用一块芯片就可实现与 CPU 数据接口来实现 8 位字长的读写操作。

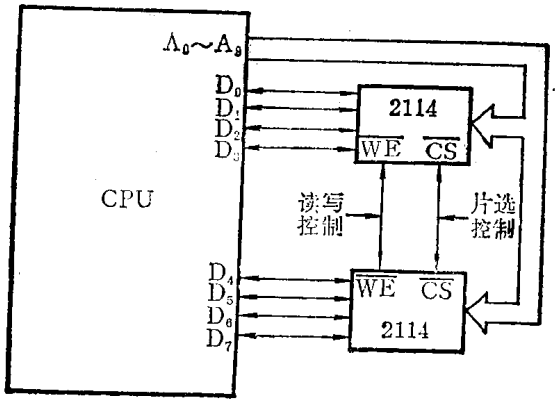


图 2-4　RAM2114 与 CPU 接口示意图

由于每一存储器芯片的容量是有限的，例如 2114 是 1 K×4，2716 是 2K×8。如果需要更大存储容量可由多片存储器芯片组合实现。这些芯片都接到 CPU 的相应总线上去。实际上，存储器芯片是通过内部缓冲器与总线连接的。这种连接使得不进行读写操作的存储器和总线之间处于高阻抗状态。而只有被选中需要进行读写操作的存储器芯片才能在总线上与 CPU 交换信息。所谓选中的存储器是指它的片选端($\overline{CS}$)处在有效状态，即低电平。对每一存储器芯片$\overline{CS}$端的控制通常称为片选控制。

一、存储器片选控制的方法

存储器的片选可以由地址线直接控制，也可由地址线经过译码逻辑来实现控制。前者称为线性片选，后者称为译码片选。

1. 线性片选法