



职业资格培训教材
社会力量办学培训教材

(中级)

计算机

维修工

● 劳动和社会保障部教材办公室组织编写



中国劳动社会保障出版社

职业资格培训教材
社会力量办学培训教材

计算机维修工

(中 级)

劳动和社会保障部教材办公室组织编写

中国劳动社会保障出版社

图书在版编目(CIP)数据

计算机维修工：中级/张志强主编. —北京：中国劳动社会保障出版社，2001
职业资格培训教材、社会力量办学培训教材

ISBN 7-5045-3166-9

I. 计…

II. 张…

III. 电子计算机—维修—技术培训—教材

IV. TP307

中国版本图书馆 CIP 数据核字 (2001) 第 022061 号

中国劳动社会保障出版社出版发行

(北京市惠新东街 1 号 邮政编码：100029)

出版人：张梦欣

*

新华书店经销

北京地质印刷厂印刷 北京助学印刷厂装订

787 毫米×1092 毫米 16 开本 15 印张 373 千字

2001 年 8 月第 1 版 2006 年 1 月第 4 次印刷

印数：5000 册

定价：27.00 元

读者服务部电话：010-64929211

发行部电话：010-64911190

出版社网址：<http://www.class.com.cn>

版权专有 侵权必究

举报电话：010-64911344

前 言

《劳动法》和《职业教育法》明确规定，在全社会实行学历文凭和职业资格证书并重的制度。在国家劳动和社会保障行政管理部门的大力倡导下，取得职业资格证书已经成为劳动者就业上岗的必备的前提，同时，作为劳动者职业能力的客观评价，已经为人力资源市场供求双方普遍接受。取得职业资格证书不仅是广大从业人员、待岗人员的迫切需要，而且已经成为各级各类普通教育院校、职业技术教育院校毕业生追求的目标。

开展职业资格培训教材建设十分重要。为此，劳动和社会保障部教材办公室、中国劳动社会保障出版社组织编写了《职业资格培训教材》，用于规范和引导职业资格培训教学。第一批组织编写的有：制冷设备维修工、冷作钣金工、制冷空调工、家用视频设备维修工、客房服务员、汽车修理工、电工、办公设备维修工、电梯安装维修工、计算机操作员、计算机调试工、计算机维修工 12 个职业的教材。其他职业（工种）的教材将分期分批地组织编写。

职业资格培训教材的主要特点是：

1. 最大限度地体现技能培训的特色。教材以最新国家职业标准为依据，以职业技能鉴定要求为尺度，以满足本职业对从业人员的要求为目标。凡《标准》中要求的技能和有关知识，均作了详细的介绍。

2. 以岗位技能需求为出发点，按照“模块式”教材编写思路，确定教材的核心技能模块，以此为基础，得出完成每一个技能训练单元所需掌握的工艺知识、设备（工具）知识、相关知识和技能、专业知识、基础知识，并根据培训教学的基本规律，按照基础知识、专业知识、相关知识、设备（工具）知识、工艺知识、技能训练的次序组成教材的结构体系。

3. 服务目标明确。从教学形式上，主要服务于教育、劳动社会保障系统，以及其他培训机构或社会力量办学所举办的各种类型的培训教学，也适用于各

级各类职业技术学校举办的中短期培训教学，以及企业内部的培训教学；从培训教学时间上，服务于3~6个月不同等级的培训教学，即300~600授课学时的培训教学。

4. 在强调实用性、典型性的前提下，充分重视内容的先进性。尽可能地反映与本职业相关联的新技术、新工艺、新设备、新材料、新方法。

本书由张志强、李莉、顾巧论、杨煜、蔡振山（天津职业技术师范学院）、王卫军（天津市蓟县卫生学校）编写，张志强主编；赵国瑞（天津大学电子信息工程学院）审稿。

编写职业资格培训教材是一项探索性的事业，尽管参与编写的专家已经为此付出了艰苦的努力，但是由于缺乏可以借鉴的成功经验，加之时间仓促，存在缺点和不足在所难免，恳切希望广大读者提出宝贵意见和建议，以便今后修订，逐步完善。

劳动和社会保障部教材办公室

目 录

基础知识部分

单元 1 计算机基础知识	(1)
1.1 CPU 的组成	(1)
1.2 存储器	(2)
1.3 输入/输出方式	(4)
1.4 总线	(9)
1.5 接口	(12)
单元 2 DOS 操作系统	(15)
2.1 DOS 常用命令介绍	(15)
2.2 DOS 分区及操作系统的安装	(20)
单元 3 中文 Windows 98 操作系统	(30)
3.1 中文 Windows 98 操作系统简介	(30)
3.2 中文 Windows 98 操作系统的安装	(31)
3.3 控制面板	(35)
3.4 Windows 资源管理器	(49)
3.5 磁盘扫描程序	(57)
3.6 磁盘碎片的整理	(60)

专业知识部分

单元 4 计算机常用硬件	(64)
4.1 主板	(64)
4.2 CPU	(68)

4.3	内存	(70)
4.4	软驱	(73)
4.5	硬盘	(75)
4.6	光驱	(79)
4.7	显示适配器	(81)
4.8	显示器	(84)
4.9	声音适配器	(88)
4.10	打印机	(89)
4.11	调制解调器	(92)
4.12	电源	(94)
4.13	扫描仪	(96)
4.14	数码相机	(98)
4.15	摄像头	(99)
单元 5	计算机硬件检测	(101)
5.1	自检过程	(101)
5.2	自检中容易出现的错误及错误提示信息	(102)
单元 6	计算机软、硬件故障的维修	(105)
6.1	硬件维修的原则	(105)
6.2	故障诊断的方法	(107)
6.3	常见硬件故障	(111)
6.4	常见软件故障	(127)

相关知识部分

单元 7	数字电路	(134)
7.1	常用门电路	(134)
7.2	常用芯片介绍	(137)
单元 8	计算机病毒	(138)
8.1	计算机病毒简介	(138)
8.2	计算机病毒的分类	(140)
8.3	病毒的识别与预防	(141)
8.4	瑞星杀毒软件	(142)
8.5	杀毒软件 KV300 的使用	(146)

8.6 杀毒软件 KILL 与 KILL95 的使用	(149)
单元 9 INTERNET 的连接	(153)
9.1 Internet 简介	(153)
9.2 与 Internet 的连接	(157)
9.3 浏览器的简单使用	(164)

技能操作部分

单元 10 计算机组装与 CMOS 设置	(175)
10.1 计算机组装	(175)
10.2 CMOS 设置	(181)
单元 11 计算机主要部件的维修	(190)
11.1 电源的基本知识与维修	(190)
11.2 软盘驱动器的基本知识与维修	(193)
11.3 硬盘驱动器的基本知识与维修	(201)
单元 12 设备驱动程序的安装	(207)
12.1 DOS 下设备驱动程序的安装	(207)
12.2 Windows 下设备驱动程序的安装	(208)
单元 13 计算机系统测试软件	(214)
13.1 SysChk 测试软件	(214)
13.2 WinBench 测试软件	(222)

基础知识部分

单元 1 计算机基础知识

1.1 CPU 的组成

我们知道在传统的概念中计算机由运算器、存储器、控制器和输入输出设备这几个基本部分组成。其中，中央处理器即 CPU (Central Processing Unit) 是计算机的核心，它主要由运算器、控制器以及寄存器组成。CPU 具有指令控制、操作控制、数据运算、异常处理和中断处理等方面的基本功能。

指令控制就是对程序运行的控制。CPU 必须对指令的执行进行控制，以确保指令序列执行结果的正确性。操作控制就是对指令内操作步骤的控制，CPU 必须对完成每一条指令的时间和步骤进行控制，以协调各信号之间的相互关系。数据运算就是对数据进行算术运算和逻辑运算，这是 CPU 最基本的功能。异常处理和中断处理就是对运算中出现的错误情况以及对外围设备的服务请求等的处理。

(1) 运算器和寄存器

运算器是直接进行数据变换与运算的部件。它主要由寄存器、算术逻辑单元和组合逻辑电路三部分组成。其中算术逻辑单元 (ALU) 执行大多数的运算操作。运算操作包括算术运算和逻辑运算，算术运算是对数值数据进行算术计算，如进行加、减、乘、除四则运算和数据格式的转换。逻辑运算是指按位对数据进行与、或、非、比较、移位等运算。在进行算术和逻辑运算操作时，需要先从存储器中取出数据放到运算器中进行计算，然后再把计算结果存放到存储器中。

在运算器中有若干个临时存放数据的部件，我们把它称为寄存器 (Register)。寄存器由触发器构成，用于暂时存储最频繁使用的数据，如一些中间运算结果等。除此之外，寄存器中还可保存运算的状态，如数据是否有进位、是否为零、是否溢出、是否发生了其他错误等，为的是对运算中出现的各种情况进行处理。在运算器中一般有多种不同的寄存器。每个寄存器在指定时都必须给出一个指定的编号，叫做寄存器号，寄存器号在电路中可用一个二进制代码识别。

在运算器中由一个组合逻辑电路来完成算术和逻辑运算，它一般有两个输入端，可同时输入两个参加运算的操作数。操作数可以是寄存器的输出，也可以是存储器的输出。

(2) 控制器

控制器是计算机的指挥与控制中心，其主要功能是：

1) 从内存中取指令，并计算下一条指令在内存中的地址。控制器是在计算机指令的控

制下进行工作的，即按指令存储的顺序依次执行，或者根据指令决定执行的顺序。

2) 对指令进行译码，产生相应的操作控制信号。计算机指令是一种经过编码的操作命令，它指定需要进行的算术和逻辑操作，支配计算机中信息的传递以及在计算机与输入输出设备之间的信息传递。控制器对指令进行解码，并根据指令生成一系列时序控制信号，控制其他单元的工作。

3) 控制指令执行的步骤和数据流动的方向。控制器主要由程序计数器、指令寄存器、节拍及周期形成电路、操作控制器四部分组成。

1.2 存储器

存储器是计算机的记忆装置，其作用是存储程序、保存原始数据和中间结果。目前，主要的存储介质有半导体电路、磁性记录介质、光存储介质等。存储器中存储单元的数量称为存储器的容量。容量、价格和存取速度是评价计算机存储器的三大指标，但三者之间又互相制约。当存储容量大时，存取速度就低；如果采用存取速度高的元件，成本就高；成本高，就不允许做得容量太大。为了用最合理的成本实现最大的存储容量，采用两级存储，即把存储器分为主存储器和辅助存储器。

主存储器又称主存或内存，用来存储当前正在执行的程序指令和正在处理的数据。它一般采用半导体存储器件实现，其存储密度高，存取速度快，但成本较高，在电源断开时存储的内容会丢失。

主存储器又进一步划分为程序存储器和数据存储器。其中，程序存储器用来保存程序指令信息，也就是说，当计算机执行一个程序时，它是从程序存储器中读出指令来加以执行的。数据存储器用来保存正被计算机处理的数据。在计算机中，数据并不会自动地存在于数据存储器中，如同程序一样，它是从辅助存储器中读入主存储器的。

辅助存储器又称外存，用来存储当前暂不需要使用的程序和数据。一般采用光、磁等介质的存储器，其容量大，成本较低，但存取时间长。

为了进一步提高计算机的性能，在主存储器和 CPU 之间又增加了一级比主存储器速度更高的高速缓冲存储器 (Cache)，从而形成三级存储体系。

实现存储器的元器件有半导体存储器芯片、磁盘、磁带、光盘等。对存储器的访问是在控制器的控制下进行的。对外存储器的访问时间 with 数据在存储器中的位置有关，有的采用按顺序存取，有的是按部分顺序存取。

CPU 可以直接访问的存储器是主存储器。主存储器根据其性能和特点可分为只读存储器 (ROM) 和随机存储器 (RAM) 两大类。

(1) RAM

无论程序和数据存放在主存储器的任何位置，存储器被访问时都能够快速方便地存取其中的内容，存和取的速度与存储位置无关，我们把具有这种功能的存储器称为随机存储器 (RAM)。当存进新的数据时旧的数据才能去掉。这里所说的“存”和“取”数据的确切叫法应当是“写”和“读”。RAM 中的信息既可以读出，也可以写入，但断电时信息丢失。

实现 RAM 存储器的元器件主要有静态存储器 (SRAM) 芯片和动态存储器 (DRAM) 芯片。静态 RAM 的主要特点是用普通的触发器存放一位二进制信息，只要不切断电源，信息就可以长时间稳定地保存，其存取速度快并且不需要刷新。在高档计算机中，常用它作高速

缓冲存储器使用。动态 RAM 的主要特点是用电容器上的充电电荷表示一位二进制信息。由于电容上的电荷会随时间延长而泄漏，因此要定时对所存信息进行刷新，一般每秒要对所存信息刷新近百次。存储容量较大的系统一般都采用动态 RAM。

(2) ROM

我们把存储的信息只能读出而不能写入的存储器叫做只读存储器 (ROM)。其信息一旦写入计算机后就固定下来，即使切断电源，信息也不会丢失。它用于存储计算机中的一些固定的信息，如系统程序的核心部分。其中存储的数据在芯片的生产过程中写入，在计算机系统中不能进行修改。只读存储器分为普通 ROM、可编程 ROM、可擦除 PROM、电可擦写 ROM 和快闪存储器五类。

1) 普通 ROM 又称为掩膜 ROM。其优点是存储内容固定不变、可靠性高，但它只能写入一次而不能更改。

2) 可编程 ROM 又称为 PROM。其优点是允许用户写入自己的数据，但它只能被用户编程一次，以后就不能修改存储的数据了。

3) 可擦除 PROM 又称为 EPROM。其优点是允许用户存储内容被擦除并可多次编程。目前使用 EPROM 的较多。

上述 ROM、PROM 和 EPROM 只读存储器的数据要么不能擦除，要么必须采用专门设备进行擦除，而且为了擦除数据必须将存储器芯片从系统中拔下。

4) 电可擦写 PROM 又称 EEPROM。它是一种可用电子方式对存储器芯片中的某个存储单元进行擦除的非易失性（即关掉该电路的电源以后，所存储的信息并不丢失）存储器件，但它成本高，使用范围受到限制。

5) 快闪存储器 这是近年发展起来的一种非易失性存储器，又称快擦存储器。它是在 EPROM 和 EEPROM 的制造技术基础上发展起来的一种新型的电可擦除非易失性存储器元件。其擦除、重写的速度比 EEPROM 快，具有密度高、价格低、可靠性高的优点。

尽管快闪存储器是可写的，但擦除和重写能力有限，在计算机系统中的应用仅限于不需要经常重写的某些数据存储场合。

(3) Cache

Cache 是一个比主存储器速度更高的高速缓冲存储器，目前已被各种类型的计算机广泛采用。Cache 的容量较小，主要用来存储 CPU 最经常访问的指令或者操作数据。Cache 在最初时是没有数据或程序代码的，当 CPU 访问存储器时，从主存中读取的数据或代码在写入寄存器的同时还写入了 Cache 中，那么在此后 CPU 就可直接从 Cache 中访问这些数据或代码了，而没有必要再到主存中去访问。

(4) 内存地址

内存由大量的数据存储单元构成，为了寻找内存中某一个存储单元的位置，需要给不同的存储位置指定一个编号，这个编号就是内存地址。它是一串依次编排的数字编号，是识别存储器中不同存储单元的惟一标志，不同的存储单元有不同的地址。在进行读操作时，运算部件必须向存储器送一个数据地址以及读操作信号。在进行写操作时，运算部件必须向存储器送数据地址、数据以及写操作信号。写入和读出都是通过指定一个地址进行的。存储位置的指定可以是字节为单位，也可以是以字为单位，分别称为字节地址和字地址。

地址总线用来传送存储单元或输入、输出接口的地址信息。地址总线的根数一般反映了

一个计算机系统的最大内存容量。不同的 CPU 芯片，地址总线的数量不同。例如 8088CPU 芯片有 20 根地址线，可寻址内存单元数为 2^{20} ，即内存最大容量为 1MB。

1.3 输入/输出方式

计算机系统进行信息处理需要与外界交换信息，这一功能是由输入、输出系统完成的。输入、输出系统由输入、输出设备和输入、输出 I/O 接口构成。

典型的输入设备有键盘和鼠标等，通过它们把外界的数据传送给主机，键盘是将用户操作时的按键直接以 ASCII 码的形式送给主机，鼠标则将位置信息以数字形式输入到计算机中。其他的输入设备还有图像扫描仪等。

输出设备与输入设备相对应，其功能是将计算机的处理结果输出。典型的输出设备有显示器、打印机、绘图仪等。某些设备同时兼有输入和输出的功能，如电传打字机、数据终端等。

近年来随着计算机应用领域的不断拓展，对输入、输出设备也不断提出新的要求。新型的输入、输出设备有触摸屏、光笔等。

通常我们将计算机的输入、输出设备称为外围设备。外围设备是通过输入、输出接口与主机连接的。接受 CPU 输入、输出操作命令的是输入、输出接口。输入、输出接口的功能是：接收主机的命令，并将设备的状态存储在寄存器中，在需要时向主机提供设备的状态信息；在传输数据时为数据提供缓存空间，使主机与外围设备之间的速度匹配，必要时还将提供数据的转换功能；识别设备地址；提供适当的信号定时以完成数据和状态信息的传输，包括同步控制。

在输入、输出数据的过程中，数据传送的方式有程序控制方式、直接存储器访问方式等多种。其中程序控制方式可分为程序查询方式和中断方式两种。

(1) 程序查询方式

由于多数外围设备与 CPU 之间的数据传送都是随机的，因此要求 CPU 必须顾及输入、输出设备的工作状态。当计算机系统中只有一台外围设备时，CPU 要定时地对这台设备的状态进行查询，CPU 常常处于询问等待状态，或在执行主要功能的程序中穿插地进行询问。当有多台外围设备时，CPU 一般是循环地逐一进行询问；有些系统，将各个外围设备的状态标志位线“或”在一条公共检测线上，CPU 首先检测此线，有服务请求再去查询是哪一台设备。

程序查询方式是利用程序控制实现 CPU 与输入、输出设备之间的数据传送的，现以输入为例说明程序执行的过程。

1) CPU 通过地址总线将地址送出，设备选择电路根据此地址来判断是否为本设备，从而选中设备控制器。

2) CPU 查看正在“忙”的触发器是否为“0”。若为“0”，则发出命令字，请求启动外围设备进行数据输入，并置“忙”触发器为“1”，置“就绪”触发器为“0”，然后不断检测“就绪”触发器看它什么时候变为“1”。

3) 输入接口接到 CPU 的命令字后，立即启动外围设备工作，开始输入数据。

4) 外围设备启动后将输入数据送入数据缓冲寄存器存放，然后送往 CPU。

5) 外围设备完成数据输入后，置“就绪”触发器为“1”，通知 CPU 已经“准备好”。

6) CPU 从数据缓冲寄存器中读入输入数据，并将控制器状态标志复位，以便接口对外

设进行监视。一旦 CPU 用程序询问输入、输出设备时,则状态位信息取至 CPU 进行分析。

我们知道 CPU 的速度比外围设备的速度快得多,采用程序查询方式 CPU 只能等待外围设备,并且 CPU 在一段时间内只能和一台外围设备交换信息,不能使其他外围设备同时工作,因此,系统的效率较低。改进的方法是 CPU 周期性地调用多个设备的服务子程序,同时控制多个外围设备或者采用程序中断的方式。

(2) 中断方式

在实际应用中,多数事件是非寻常或非预期的。当这样的紧要事件发生时,外围设备向 CPU 发出请求,在请求允许响应时, CPU 应当暂停当前正在执行的程序,先转去为外设服务。紧急事件处理完毕之后,恢复原来的状态,再继续执行原来的程序。这种对要紧事件的处理方式,称为中断方式。外围设备发出的信号称为中断请求信号。这种处理要紧事件的服务子程序称为中断子程序,而引起中断的事件称为中断源。

1) 中断源 中断是由中断源发出的。常见的中断源有以下几种:

①输入、输出设备的中断 指输入设备或输出设备在工作过程中已做好了接收或发送的准备。

②数据通道中断 指磁盘、磁带等外围设备已准备同主机进行数据交换。

③实时时钟中断 指外部时钟电路工作到了规定的时间后所发出的中断请求。

④故障中断 指要求 CPU 进行紧急处理的异常情况。例如电源断电、设备故障等。

⑤系统中断 指由处理器内部引起的中断。例如,在运算过程中发生溢出、数据传送过程出现校验错误、控制器遇到非法指令等。

⑥为了调试程序而设置的中断 当我们编好程序以后,为了调试程序而在程序中设置断点,以便发现程序中的错误。

2) 中断请求的提出 当中断源发出中断请求信号时,与之对应的中断请求触发器置“1”,并一直保持到 CPU 响应了这个中断,才可以将这个中断请求清除。

3) 中断排队 由于中断请求的随机性,有可能遇到多个中断源同时发出中断请求的情况,这时必须决定先处理哪一个中断。为此 CPU 有一个裁决机制,在各种中断排队中,让最紧迫、最重要、处理速度较高的事件优先处理。判断中断优先级的原则是:

①当不同级别的中断发生时,按级别的高低依次进行处理。

②在多重中断系统中,处理一个中断过程中又发生了新的中断请求时,如果新中断的优先级更高,则在原中断程序未执行完时可以去进行新的中断处理。等到新的中断请求处理完成时再返回,继续原来的中断请求。这种重叠处理中断的现象称为中断嵌套。较低级中断不能使较高级中断过程再中断,同级中断过程也不能被同级中断再中断。

③同级中断源同时申请中断时,按事先约定的次序处理。

以上这些原则可以用电路或程序两种方法实现。

4) 中断屏蔽 中断屏蔽就是 CPU 在处理中断时阻止其他中断,使这些中断信号暂不被 CPU “感觉”到,但信号仍保留,等到条件允许时再响应。用中断屏蔽可使同一级及低一级中断不能中断同一级及高一级的中断服务子程序,因此可以利用中断屏蔽作用组成中断优先管理电路。

5) 中断的禁止和开放 禁止中断就是 CPU 拒绝任何中断。当系统处于两个程序的转换过程或正在执行某些不允许打断的程序时而禁止中断。为了避免造成混乱, CPU 设置了一个

中断允许触发器来控制是否开放中断和禁止中断。

6) 中断响应的条件和过程

① 中断响应的条件

a. 中断源有中断请求。

b. CPU 允许中断。

c. 一般要等到一条指令执行完毕才能响应中断。

② 中断响应的过程 当 CPU 响应中断的条件被满足时, CPU 会立即响应中断。

a. 对于比较简单的中断处理方式, CPU 响应中断的步骤是:

(a) 关中断, 此时 CPU 暂时不能响应中断。

(b) 保存现场信息, 把 CPU 响应中断前的那条指令的地址保存起来, 以便返回时能知道这个地址。我们把这个地址称为断点。另外, 还要在中断服务程序中保存那些需要使用的寄存器的原值, 以便使中断处理结束之后能恢复原来的状态继续执行程序。

(c) 判别中断条件, 确定中断服务程序入口。对于不同的中断源, CPU 要进行的处理不同, 中断服务子程序内容也不同。这些中断服务子程序分别放在内存的某些固定区域中。因此中断处理的关键是找出这些服务子程序的首地址。中断服务子程序入口地址可根据中断向量生成。

(d) 执行中断服务程序, 目的是完成中断请求的操作。

(e) 退出中断, 恢复现场信息。

(f) 开中断, 继续执行原程序。

b. 在多重中断系统中, CPU 响应中断的步骤是:

(a) 关中断, 此时 CPU 禁止所有中断。

(b) 保存现场信息。

(c) 判别中断条件, 确定中断优先级、中断源, 找出中断服务程序入口。

(d) 开中断, 设置 CPU 优先级为当前中断的优先级, 允许优先级更高的中断请求。

(e) 执行中断服务程序, 完成中断请求的操作。

(f) 关中断, 提高 CPU 优先级, 暂时禁止中断响应。

(g) 恢复现场信息。

(h) 开中断, 继续执行原程序。

在输入输出系统中, 按照中断处理方式可以把中断分为简单中断和程序中断。如果主机在响应中断请求后, 是通过执行一段服务程序来处理有关事件的, 则称为程序中断。这种方式要求 CPU 在响应中断后暂停原程序的执行, 并将断点和现场信息保存起来。简单中断是一种特殊的中断, 它不需要 CPU 执行服务程序, 可用在直接存储器存取传输中, CPU 让出若干访问存储器的周期供输入、输出设备与主存交换数据。

(3) 直接存储器访问方式

当在某一时刻, 几台外围设备同时发出中断请求信号时, CPU 可根据预先规定好的优先顺序, 按轻重缓急处理几台外围设备的数据传送, 从而提高计算机系统的工作效率。但是中断系统的保存与恢复现场需要一定的时间, 并且主机与外围设备, 例如磁盘、磁带等之间传输数据, 将引起 CPU 频繁干预, CPU 长时间为外围设备服务, 同时还可能引起数据丢失。

直接存储器存取 (DMA) 控制是在内存与外围设备之间开辟了一条直接数据传送通路,

CPU 将数据的地址和传输的数量告诉接口后, 传送过程就交给 DMA 控制器进行管理。这时就由接口来控制数据在主存储器和外围设备之间的传输而不需要由 CPU 来控制, 从而形成以存储器为中心的体系结构。DMA 和 CPU 是共享一个存储器的两个处理器, 只不过 DMA 的“头脑”比较简单而已。CPU 与 DMA 各自独立地工作, 从而进一步提高了系统的并行性。我们将数据的这种传输方式称为直接存储器访问 (DMA) 传输方式, 具有这种功能的接口称为 DMA 接口。

1) DMA 传送方式 DMA 控制器与 CPU 通常采用以下三种方式占用内存:

① CPU 暂停访问内存 对 CPU 来说, 一般 DMA 的优先级高于中断。CPU 暂停访问内存意味着 CPU 等待 DMA 的操作。在这种方式下, 当设备要求传送一批数据时, 由 DMA 控制器发一个信号给 CPU, 要求 CPU 放弃使用总线。DMA 控制器在获得总线控制权时开始进行数据传送。在一批数据传送完毕后, DMA 控制器把总线控制权交还给 CPU, 这时 CPU 才能使用总线。这种方法的优点是控制简单, 它适用于数据传输速率很高的设备进行数据块的传输。其缺点是在 DMA 控制器访问内存期间, CPU 基本处于不工作状态或称为保持状态。因为外围设备传送两个数据之间的间隔一般总是大于内存存储周期, 即使高速输入/输出 (I/O) 设备也是如此, 所以说 CPU 和内存都没有充分发挥效率, 相当一部分内存工作周期是空闲的。

② DMA 与 CPU 交替访问内存 DMA 与 CPU 交替访问内存就是 CPU 和 DMA 控制器轮流使用总线, 这样就不需要总线使用权的申请、建立和归还过程。在这种交替方式中, 总线和存储器的访问权是分时控制的, CPU “觉察”不到 DMA 的操作, 它既不停止程序的运行, 也不进入等待状态。如果 CPU 访问存储器的周期比内存的操作周期长, 那么在 CPU 两次访问存储器之间, 存储器就会有一段空闲时间可供 DMA 访问。这种控制方式的优点是在 DMA 传送时 CPU 能同时发挥最高的效率, 可以说它是一种高效的控制方式。它适合于 CPU 访存操作有固定周期的场合, 在采用 Cache 的 CPU 中, 这种访存规律一般是不存在的。它的缺点是硬件逻辑复杂。

③ 直接访问和周期挪用 在这种 DMA 传送方式中, 使用权一直在 CPU 那里, 当 DMA 控制器没有请求访问存储器时, CPU 按程序要求访问内存, 当 DMA 控制器需要使用存储器时, 则从 CPU 那里“挪用”存储器的周期。也就是说, DMA 控制器在 CPU 不使用存储器时向 CPU 请求访问存储器, 而当 CPU 访问存储器时则立即暂停对存储器的使用。因此这种方式称为“周期挪用”。周期挪用方式不影响 CPU 的正常工作, 我们把这种情况也称为 DMA 的直接访问存储器工作方式, 它是标准的 DMA 工作方式, 是一种广泛采用的方法, DMA 也因此而得名。但在这种方式中, 一个数据块的传输可能需要分若干次完成, 输入、输出设备每一次周期挪用时都要申请占用总线, 要有一个建立总线控制权和释放总线的过程。对内存来说, 即使传送一个字也要占用一个周期, 但对 DMA 控制器来说一般要长些, 这要视逻辑线路的延迟而定。通常周期挪用的方式适用于外围设备读写周期大于内存存储周期的情况。

2) DMA 控制器的组成 基本的 DMA 控制器主要由内存地址计数器、字计数器、数据缓冲寄存器、控制逻辑和标志寄存器组成, 其结构示意图如图 1—1 所示。

①内存地址计数器 内存地址计数器用于存放内存中要交换的数据的地址。操作开始时, 存放所要读写的存储字段的首地址。以后每传送一个字, 都将内存地址计数器加 1, 从

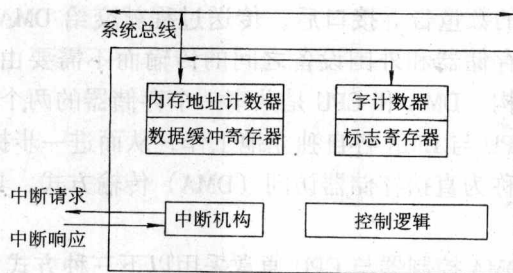


图 1-1 DMA 控制器结构示意图

而以增量方式给出每个内存中传输数据的地址。

②字计数器 字计数器用于对要传送的字数目计数。在 DMA 传送时，每传送一个字，字计数器就加 1，当计数器溢出即最高位产生进位时，表示这批数据传送完毕。

③数据缓冲寄存器 数据缓冲寄存器用于暂时存放每次传送的数据。当输入时，由设备将数据送往数据缓冲寄存器，再由数据缓冲寄存器通过数据总线送到内存。而输出时，则由内存将数据通过数据总线送到数据缓冲寄存器，然后再送到设备。

④控制逻辑 用于控制对内存地址计数器和字计数器的修改，指定传送类型，并对接口信号进行协调和同步。

⑤标志寄存器 标志寄存器用于存放请求标志等，每当设备准备好一个数据字后，控制逻辑就给出一个控制信号，设置“DMA 请求”信号。在周期挪用方式下，该标志置位后向控制逻辑发出总线使用请求，控制逻辑接收此信号后发出 DMA 响应信号，使“DMA 请求”标志复位，为交换下一个字做好准备。

3) DMA 控制器的操作 DMA 控制器执行的操作一般应包括：

① DMA 控制器接收由外设发出的 DMA 请求。

② DMA 控制器向 CPU 发出 DMA 请求，CPU 响应此请求后发回响应信号，这时 DMA 控制器在 CPU 的控制下接管总线。

③ 根据外部逻辑来决定数据传送的地址单元以及数据传送的长度，并执行数据的传送操作控制。

④ 指明 DMA 操作结束，恢复 CPU 对总线的控制。

4) DMA 传送过程 DMA 传送过程可分为 DMA 传送前预处理、DMA 数据传送、DMA 传送后处理三个阶段。

① DMA 传送前预处理 DMA 传送前预处理是由程序做的一些必要的准备工作。首先由 CPU 测试设备状态，然后把设备号送入 DMA 控制器的设备地址寄存器中并启动设备，再把交换数据的起始地址送入内存地址计数器中，最后把交换的数据个数送入数据字计数器中。在完成了这些工作之后，CPU 将继续执行原来的程序。

当外围设备已处理完上次接收的数据或已准备好发送的数据时，将通知 DMA 控制器发出 DMA 请求，申请主存总线。当有几个 DMA 申请时，需要根据轻重缓急通过硬件排队线路按预定优先级别排队。一旦设备得到了主存总线控制权就立刻开始传送数据。

② DMA 数据传送 DMA 数据传送可以以字或数据块为基本单位进行。以数据块为基本单位传送时，每次 DMA 占用总线后的数据输入和输出操作都是循环实现的。

③ DMA 传送后处理 一旦 DMA 的中断请求得到响应，CPU 马上暂停原来程序的执行，

而转去执行中断服务程序，最后还要校验送入主存的数据是否正确，并决定是否继续使用 DMA 方式传送数据。

1.4 总线

总线是计算机系统的骨架，是计算机中连接各个功能部件的纽带，是计算机各部件之间进行信息传输的公共通路。目前所有的计算机都采用了总线结构，总线结构的好坏将对系统性能有很大的影响。

(1) 总线的分类

总线的应用非常广泛，从不同的角度来划分，总线可有多种分类方法。如果根据所连接的部件可把总线分为元件级总线、板级总线、系统级总线、通信总线四种。

1) 元件级总线 又称片内总线，是指在 CPU 内部、寄存器之间、寄存器与运算部件之间的连接总线。

2) 板级总线 是指连接印刷电路板内部各芯片之间的总线。

3) 系统级总线 又称内总线，是指连接 CPU、主存储器、I/O 接口等的总线。

4) 通信总线 又称外总线，是指连接计算机系统与其他系统之间的总线。

(2) 总线的信息传输方式

在计算机总线中，信息的传输方式有串行传输、并行传输、复合传输和消息传输四种基本方式。

1) 串行传输 指数据在一条线路上按位进行传输。它只需一条数据传输线。在计算机中普遍适用于串行的通信线路连接终端、鼠标和调制解调器等慢速的外围设备。

2) 并行传输 指数据在为每个控制信号专门设置的信号线上同时进行传输。它需要很多信号线，为了减少线路数量，可以将并行传输方式与串行传输方式结合起来，构成并串行的传输方式。当数据线宽度不够时，数据可分多次传输。

3) 复合传输 指分时地传输同一数据源的不同信息，即不同的信号在不同的时间片中轮流地向总线的同一条信号线上发出。由于传输线上既要传输地址信息，又要传输数据信息，因此必须划分时间，以便在不同的时间段中完成地址和数据的传输。

4) 消息传输 消息是一种含有许多不同信息并且具有固定格式的数据。消息传输方式是将总线需要传送的数据信息、地址信息和控制信息等组合成一个固定的数据结构传输，一次可发送很多的信息，这样可减少总线同步操作对传输速度的影响。

(3) 总线的通信方式

虽然总线能传递多种信息，但每一时刻只能传输一种信号，所以只能分时地传输多种信号。每一个信号都是由发送端发出，接收端接收。为了正确地接收数据，要采用一定的方式让接收端知道发送端什么时刻开始发送，什么时刻发送完毕，这叫做总线通信的同步方式。通信过程可分为同步通信和异步通信两种过程。

1) 同步通信 同步通信就是各部件通过总线进行数据传送时，用一个公共的时钟进行同步控制。这就好像住在同一个屋子里的人都要遵守同一个作息时间表一样。所有信号与时钟的关系是固定的，主模块和从模块之间没有应答信号。这个公共的时钟可以由 CPU 总线控制部件发送到每一个部件，也可以让每个部件有各自的时钟发生器，但必须由总线控制部件发出的时钟信号对它们进行同步。