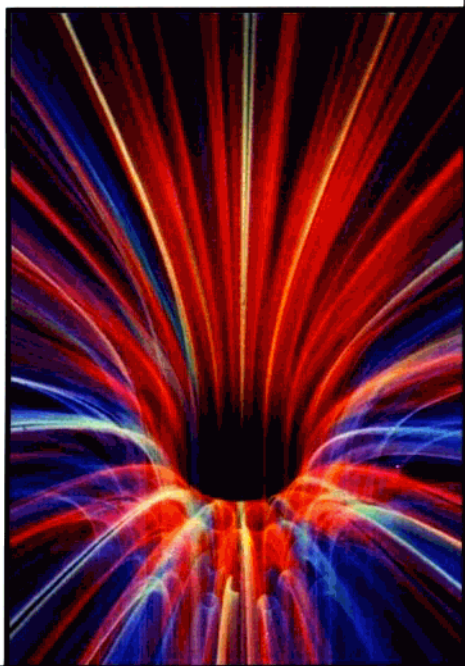


高等职业技术学院教材

微机接口技术 及应用

葛玻 武心 张华鹏 舒云星 编

微机接口技术及应用



武汉理工大学出版社

微机接口技术及应用

葛玻 武新 张华鹏 舒云星 编

武汉理工大学出版社

内 容 简 介

以 Intel 8086/8088 微处理器、MCS-51 系列单片机为对象讨论微型机的接口技术。全书共分为十二章。为适应课程内容更新的需要,本书增加了工控机接口技术及 32 位微机及多媒体基本知识。

在编写指导思想,坚持理论基础以“必需够用”为原则,特别突出工程实用性。既是教科书又具有一定工具书的特点,既面向教学,更注重面向应用。本书适用于大专院校学生,也可作为从事微机软硬件开发工作的科技人员参考。

图书在版编目 (CIP) 数据

微机接口技术及应用/葛玻等编.——武汉:武汉理工大学出版社,2000,9
ISBN 7-5629-1635-7

I.微… II.葛… III.微型计算机-接口 IV.TP364.7

武汉理工大学出版社出版发行

(武昌路 122 号 邮编:430070)

各地新华书店经销

荆州市鸿盛印刷厂印刷

开本:787×1092 1/16 印张:16.375 字数:398 千字

2000 年 9 月第 1 版 2003 年 6 月第 3 次印刷

印数:2501—3500 册 定价:19.50 元

(本书如有印装质量问题,请向承印厂调换)

前 言

微机接口技术是计算机系统开发的关键技术之一，以前该课程的内容只是 Intel8086/8088 为代表的微处理器及接口，或者是只介绍 MCS-51 单片机原理及应用技术。本书将两者进行重组及融合，采用对比式的方法全面进行介绍，这是本教材的一大特点。本书在理论阐述上力求简单明了，删去了以前过多的内部机理分析。更注重芯片的外特性及总线标准。每章开始均有“本章要点”，使读者了解学习本章应掌握的知识要点，每章最后均附习题供读者练习。

本书另一特点是突出工程实用性：①增加了工控机接口技术。②给出的应用程序实例中既给出 Intel8088/8086 汇编语言，又同时给出 C 语言程序。③接口内容比较全面，有一定“手册”的作用。④从接口技术的角度增加了 32 位微机、多媒体计算机的组成及应用。本书全部内容适合于 85 学时的教学要求，如果教学时数较少，有的章节可作为学生自学内容。葛玻编写第一章、第二章、第十章。武新编写第六章、第八章、第十一章。张华鹏编写第四章、第五章、第九章。舒云星编写第三章、第七章、第十二章。

编 者

2000 年 8 月

目 录

1 微型计算机接口技术概论.....	(1)
1.1 接口功能.....	(1)
1.2 CPU 和外设之间的数据传送方式	(2)
1.2.1 程序方式.....	(2)
1.2.2 中断方式.....	(2)
1.2.3 DMA(直接存储器存取方式)	(3)
1.3 端口编址方式.....	(3)
1.3.1 统一编址.....	(3)
1.3.2 独立编址.....	(3)
1.4 8086/8088、MCS-51 的 I/O 地址分配和地址译码	(4)
1.4.1 IBM-PC/XT 的 I/O 地址分配	(4)
1.4.2 MCS-51 的 I/O 地址分配	(4)
1.4.3 地址译码.....	(5)
1.5 接口设计与分析的基本方法	(7)
习题	(8)
2 中断技术.....	(9)
2.1 中断基本概念.....	(9)
2.2 中断源.....	(9)
2.2.1 8086/8088 系列微机的中断源	(9)
2.2.2 MCS-51 的中断源	(10)
2.3 8086/8088 中断系统	(10)
2.3.1 中断向量表.....	(10)
2.3.2 设置中断向量.....	(11)
2.3.3 8086/8088 中断过程	(12)
2.4 8259A 中断控制器.....	(13)
2.4.1 8259A 的引线、内部结构	(13)
2.4.2 8259A 的工作方式	(16)
2.4.3 8259A 的编程	(19)

2.4.4	8259A 的初始化编程举例	(25)
2.5	MCS-51 中断系统	(28)
2.5.1	中断的控制	(28)
2.5.2	中断响应的过程	(30)
2.5.3	应用程序	(31)
习题		(35)
3	可编程定时器/计数器	(37)
3.1	可编程定时器/计数器 8254/8253	(37)
3.1.1	8254/8253 内部结构与引脚信号	(37)
3.1.2	8254 在 PC 机中的应用	(43)
3.2	实时时钟电路 MC146818 及其应用	(52)
3.2.1	MC146818 内部寄存器与引脚信号	(52)
3.2.2	实时钟信息的读写操作	(55)
3.2.3	BIOS 实时钟的 I/O 调用	(55)
习题		(56)
4	MCS-51 单片机存贮器的扩展与设计	(57)
4.1	程序存贮器的扩展	(57)
4.1.1	程序存贮器扩展的最大空间及硬件结构图	(57)
4.1.2	常用 EPROM 程序存贮器芯片介绍	(58)
4.1.3	常用地址锁存芯片(8282 或 74LS373)	(59)
4.1.4	几种典型 EPROM 扩展电路	(59)
4.2	数据存贮器的扩展	(63)
4.2.1	外部数据存贮器的扩展	(63)
4.2.2	常用 RAM 芯片	(64)
4.2.3	典型外部数据存贮器的扩展方法	(65)
4.3	综合应用举例	(66)
习题		(71)
5	并行 I/O 接口	(72)
5.1	并行接口的特点	(72)
5.2	简单硬件连线并行接口	(72)
5.2.1	直接传送方式举例	(72)
5.2.2	查询传送方式举例	(73)
5.2.3	中断传送方式举例	(74)

5.2.4	DMA 传送方式	(75)
5.3	可编程并行输入输出接口 8255A	(75)
5.3.1	8255A 内部结构与引脚信号	(75)
5.3.2	8255A 控制字	(77)
5.3.3	8255A 工作方式	(78)
5.3.4	编程举例	(81)
5.4	可编程并行 I/O 接口芯片 8155	(84)
5.4.1	8155 内部结构与引脚功能	(85)
5.4.2	8155 控制字	(86)
5.4.3	8155 定时器	(87)
5.4.4	8155 应用举例	(88)
5.5	MCS-51 并行 I/O 口扩展及其应用	(90)
5.5.1	简单 I/O 口的扩展	(90)
5.5.2	可编程 I/O 口的扩展	(91)
5.5.3	应用举例	(93)
	习题	(97)
6	串行接口	(98)
6.1	串行通信基本概念	(98)
6.1.1	串行通信类型	(98)
6.1.2	传送方式	(100)
6.1.3	信号的调制与解调	(100)
6.1.4	传输率与串行通信接口	(101)
6.2	可编程串行接口 8250 及在 PC 机中的应用	(103)
6.2.1	8250 在 PC 机中的电路原理	(103)
6.2.2	8250 内部结构与引脚信号	(104)
6.2.3	8250 内部寄存器及编程	(106)
6.3	DOS 和 BIOS 串行通信功能调用	(115)
6.3.1	DOS 功能调用 INT 21H	(115)
6.3.2	BIOS 功能调用 INT 14H	(115)
6.3.3	用 C 语言编写串行口通信程序	(116)
6.4	MCS-51 串行口及应用	(119)
6.4.1	MCS-51 串行口的构成	(120)
6.4.2	串行口控制寄存器 SCON	(120)

6.4.3	串行口的工作方式	(121)
6.4.4	波特率的设定方法	(125)
6.4.5	应用举例	(127)
习题		(130)
7	微机常用总线介绍	(132)
7.1	概述	(131)
7.1.1	PC/XT 总线	(132)
7.1.2	ISA 总线	(132)
7.1.3	MCA 总线	(132)
7.1.4	EISA 总线	(132)
7.1.5	VESA 总线	(132)
7.1.6	PCI 总线	(133)
7.1.7	SCSI 接口标准	(133)
7.1.8	IDE 接口标准	(134)
7.2	ISA 总线	(134)
习题		(135)
8	人-机输入/输出接口	(136)
8.1	键盘接口技术	(136)
8.1.1	键盘工作原理与键码识别	(136)
8.1.2	PC 系列键盘	(139)
8.1.3	有关键盘的 DOS 和 BIOS 调用	(147)
8.2	鼠标器编程	(150)
8.2.1	鼠标器接口通信	(151)
8.3	显示器接口	(156)
8.3.1	显示器概述	(156)
8.3.2	EGA/VGA/TVGA 系统的组成	(160)
8.3.3	EGA/VGA/TVGA 显示模式	(163)
8.3.4	有关显示的 DOS 和 BIOS 调用	(168)
8.4	打印机接口	(173)
8.4.1	打印机与主机系统的连接	(173)
8.4.2	打印机接口电路	(174)
8.4.3	有关打印机的 BIOS/DOS 调用	(178)
习题		(179)

9 A/D 和 D/A 转换器接口	(181)
9.1 D/A 转换器的主要性能指标及输入输出特性	(181)
9.1.1 主要性能指标	(181)
9.1.2 D/A 转换器的输入输出特性	(182)
9.2 8 位 D/A 转换器 0832 及其应用	(183)
9.2.1 主要性能指标、内部结构和引脚功能	(183)
9.3 8 位 A/D 转换器 0809 及其应用	(187)
9.3.1 主要性能指标	(187)
9.3.2 内部结构及引脚功能	(187)
9.3.3 典型应用及时序图	(188)
9.4 12 位 A/D、D/A 及其应用	(191)
9.4.1 DAC1210 及其应用	(191)
9.4.2 AD574 及其应用	(193)
9.5 综合应用举例	(196)
9.5.1 题目要求	(196)
9.5.2 程序清单	(197)
习题	(199)
10 工控机接口技术及应用	(200)
10.1 工控机简介	(200)
10.2 工控机接口模板	(201)
10.2.1 开关量输入/输出板	(201)
10.2.2 脉冲量接口板	(206)
10.2.3 模拟量输入(A/D)	(208)
10.2.4 模拟量输出板(D/A)	(211)
10.2.5 通信模板	(215)
10.2.6 信号调理端子板	(217)
10.3 工控机应用举例	(221)
习题	(223)
11 磁盘文件管理	(224)
11.1 常用的软、硬盘机	(224)
11.1.1 软磁盘机的种类	(224)
11.1.2 硬盘机的种类	(224)
11.2 DOS 磁盘组织机构	(225)

11.2.1	DOS 磁盘组织机构	(225)
11.2.2	DOS 保留扇区布局及其作用	(227)
11.2.3	文件目录表及其应用	(229)
11.2.4	文件分配表及其应用	(233)
	习题	(235)
12	PC 系列微机结构	(237)
12.1	IBM PC/XT 机整机结构	(237)
12.1.1	8088/8086 CPU 的基本组成	(237)
12.1.2	IBM PC/XT 机系统结构	(237)
12.1.3	内存布局	(239)
12.1.4	I/O 端口的地址的分配	(239)
12.2	32 位微机介绍	(239)
12.2.1	Intel 80386 微处理器介绍	(239)
12.2.2	Intel 80486 微处理器介绍	(241)
12.2.3	Pentium 微处理器的新发展	(241)
12.2.4	主存储器的组织	(242)
12.3	常用多功能接口芯片	(243)
12.3.1	多功能芯片 82380	(243)
12.3.2	外围接口芯片 82C206	(244)
12.3.3	82350DT 系列外围接口芯片	(245)
12.3.4	外围接口芯片 82360SL	(246)
12.4	多媒体计算机的基本组成与应用	(246)
12.4.1	多媒体计算机概述	(246)
12.4.2	多媒体计算机硬件的标准配置	(247)
12.4.3	多媒体计算机的基本软件配置	(250)
	习题	(251)

1 微型计算机接口技术概论

本章要点: (1)掌握微机接口存在的意义,在微机系统中的作用。

(2) 掌 CPU 与外设之间交换信息方式。

(3) 掌握端口编址的意义、形式及实现方法。

(4) 掌握地址译码的几种形式,会根据硬件线路分析端口地址。

1.1 接口功能

微型计算机的 CPU 与外设之间经常需要交换数据,例如键盘、显示器、打印机、指示灯、按钮等都要通过输入/输出接口和主机相连。为了把高速的 CPU 与慢速的外设之间协调工作,需要在主机与外设之间建立相互联系的“纽带”或者说是桥梁。接口是介于主机与外设之间的连接部件,它是 CPU 与外设进行信息交换的中转站,如图 1.1 所示。

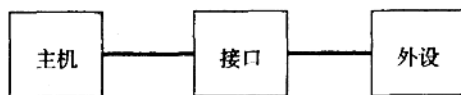


图 1.1 主机、外设与接口

接口的功能可总结为:锁存、隔离、变换、联络八个字。

锁存:因为大多数外部设备对信息的处理速度远低于计算机传送信息的速度,不可能在计算机输出指令起作用的时间内将信息处理完毕。另一个原因是对于多个输出外设而言,每一个外设都应有其对应的接口,以及与之相对应的地址,否则也无法实现一个 CPU 与多个外设之间的有序数据交换,计算机输出的数据可以先在数据锁存器中锁存,然后由外设慢慢进行处理。

隔离:由外设向计算机输入信息时,需要在外设与数据总线之间接入起隔离作用的缓冲电路。当 CPU 与多个输入外设进行数据输入时,若没有接口直接与数据总线相接,就有可能一方要提供高电平而另一方要提供低电平,这显然是不能允许的。加接隔离电路的作用是使外部信息只有在输入指令作用期间才与数据总线相连,在其它情况下,外部设备均与数据总线隔离,这样就可以避免上述情况的发生。隔离电路一般可用三态门组成。

变换:当外部输入电流、电压信号时,需要通过接口变换为计算机所能接收的数字信息。或者当需要计算机输出电压、电流信号时,需要通过接口把计算机的数字信号变换为外设所需的模拟信号。此外,串行传送与并行传送信号之间相互变换也属于变换范围。

联络:当计算机与外部设备之间进行数据交换时,收发双方有通过接口以一定方式进行“联络”的必要性。例如通过接口电路,使处理器得到所需要的时钟信号或接收外部的多个中断请求等,关于此类问题的进一步讨论将在后面进行。

接口电路是为了解决计算机和外设之间的信息变换问题而提出来的。接口是计算机和外设之间传送信息的部件,每个外设都要通过接口和主机相连。接口技术是专门研究 CPU 和外设之间的数据传送方式、接口电路的工作原理和使用方法的。

1.2 CPU 和外设之间的数据传送方式

计算机外设种类繁多,工作速度也差别甚大。外设与 CPU 之间的信息传送方式不同,CPU 对外设的控制方式也不同,CPU 何时从输入设备读取数据以及何时往输出设备写入数据就成为较复杂的问题。传送方式概括起来有如下三种方式:程序方式、中断方式、DMA 方式。

1.2.1 程序方式

程序方式传送是指在程序控制下进行信息传送,分为无条件传送方式和条件传送方式。

1.2.1.1 无条件传送方式

CPU 直接对外设输入或输出,不必查询外设的状态而直接进行信息传输,这种方式称为无条件传送方式。这种传送方式的优点是程序简单,硬件和软件很简单。但这种方式必须已知且确信外设已准备好的情况下才能用,否则就会出错。

1.2.1.2 条件传送方式(查询方式)

许多外设与 CPU 在速度上存在差异,同样传送一个数据,CPU 要快得多。查询方式是指 CPU 在传送数据之前,检查外设是否“准备好”,若没有准备好则继续查询其状态,直到准备好,才进行数据交换。CPU 每传送一个数据,需要费很多时间来等待外设准备,因此这种方式 CPU 的工作效率很低。但这种方式下的接口电路简单,硬件开销小,在 CPU 不太忙且传送速度要求不高的情况下可以采用。

1.2.2 中断方式

在上述的查询传送方式中,CPU 要不断地询问外设,当外设没有准备好时,CPU 要等待,不能干别的操作,这样就浪费了 CPU 的时间。为了提高 CPU 的效率,可采用中断的传送方式。

中断传送方式是计算机最常用的数据传送方式。CPU 启动外设后不再询问它的状态,依然执行自己的操作(主程序),即 CPU 与外设并行工作。外设完成操作后发出状态信息,经 I/O 接口电路转换成中断请求信号,向 CPU 申请中断,要求 CPU 暂时中断自己的主程序,若得到 CPU 的响应,则转入中断服务程序为外设服务。在中断服务程序中 CPU 执行输入或输出数据的操作,然后返回主程序继续执行原来被中断了的工作,上述整个过程称为中断。实现中断的硬件和软件称为中断系统。

利用中断技术管理外设服务后,CPU 从反复询问外设状态中解放出来,提高了工作效率。而且,可以同时为多个外设服务,只是在使用计算机时是分时操作。

利用中断技术,可以实现实时控制。在工业控制中,常常要对现场的各种参量进行控制,如控制温度和压力。当被控温度或压力达到一定值后,要作相应的处理。有了中断功能后,现场的检测机构通过接口电路在任何时候都可以发出中断请求,要求 CPU 进行实时处理。

CPU 响应中断请求就是执行中断服务程序。每个中断源均有相应的中断服务程序。通常在中断服务程序开头, 首先进行有关寄存器的现场保护, 在完成中断服务后, 还要恢复中断服务程序开始时所保存的各寄存器的内容, 以便返回主程序时使用。中断返回就是在中断服务程序末尾安排一条中断返回指令, 中断返回指令的任务就是把中断前保护的断点地址送回程序计数器 PC, 接着继续执行主程序。

1.2.3 DMA (直接存储器存取方式)

上述中断传送方式仍是由 CPU 通过程序来传送, 每次要保护断点, 保护现场需用多条指令, 每条指令要有取指和执行时间, 故通常传送一个字节需要几十微秒。这对于一个高速 I/O 设备, 以及成组交换数据的情况, 例如磁盘与内存间的信息交换, 就显得速度太慢了。

所以期望用硬件在外设与内存间直接进行数据交换 (DMA) 而不通过 CPU, 这样数据传送速度的上限就取决于存储器的工作速度。但是, 通常系统的 AB、DB 和 CB 三总线 (系统总线) 是由 CPU 管理的。在 DMA 方式时, 就希望 CPU 把这些总线让出来 (即 CPU 连到这些总线上的线处于第三态—高阻状态), 而由 DMA 控制器接管, 控制传送的字节数, 判断 DMA 是否结束, 以及发出 DMA 结束等信号。这些都是由硬件实现的。

通常 DMA 的工作过程是当外设把数据准备好以后, 发出一个选通脉冲给 DMA 控制器。于是 DMA 控制器向 CPU 发出总线请求信号, 在数据总线 DB 上给出数据, 并给出存储器读写命令, 就可在外设和存储器之间直接传送数据。在整个数据传送完后, DMA 控制器撤除总线请求信号, 并使总线响应信号无效, CPU 重新接管系统总线。

随着集成电路技术的发展, DMA 传送已不局限于存储器与外设间的信息交换, 还可以扩展为在存储器的两个区域之间, 或两种高速的外设之间进行 DMA 传送。

1.3 端口编址方式

微机通过 I/O 接口电路与外设相连, CPU 只有通过 I/O 接口电路才能与外部设备传递信息, 因此, 只要选中 I/O 接口电路, 就能找到相应的外部设备。一般每一个接口部件都包含一组寄存器, CPU 和外设进行数据传输时, 各类信息在接口中进入不同的寄存器, 一般称这些寄存器为 I/O 端口。每一个 I/O 端口应有其对应的地址, 就好像一栋教学楼内, 每一个教室都有其门牌号, 有了门牌号就可以找到相应的教室。对 I/O 端口编地址后, 就可以实现对这些端口的访问。端口编址有两种方式, 一种是端口地址与存储器地址统一编址, 即存储器映射方式; 一种是 I/O 地址和存储器地址分开独立编址, 即 I/O 映射方式。

1.3.1 统一编址

在这种方式中, 把每个 I/O 端口作为存储器的一个单元来对待, 故一个端口占有存储器一个单元的地址。CPU 从 I/O 接口电路取一个数据, 如同一次存储器读的操作, 而向 I/O 接口输出一个数据, 也如一次存储器写的操作。

这种编址的缺点是 I/O 接口占用了存储器的单元地址, 使内存容量减小; 由于使用访问存储器的指令来处理 I/O 数据, 故所写的程序不易区别出是访问 I/O 指令还是访问存储器的指令, 使程序很难看懂且不易调整。如 MCS-51 系列单片机就是采用这种方式。当 CPU 与

外设交换数据时采用 MOVX 指令, CPU 与片外数据存储器交换数据也采用 MOVX 指令且片外数据存储器单元地址与端口地址在同一空间编址。

1.3.2 独立编址

在这种编址方式中, I/O 接口的地址与存储器的地址是分开独立编址的, 对于这些 I/O 接口地址, 需要有专门的 I/O 指令去访问。一般有两条指令, 即输入(IN)和输出(OUT)指令。在这些指令中, 除了操作码外还有 I/O 接口地址编码, 当它是一个 8 位长的地址时, 可分别对 256 个 I/O 端口寻址。

这种方式的优点是 I/O 端口不占用存储器地址空间。由于对 I/O 端口操作和对存储器操作使用不同的指令, 所以程序中很容易区别出是 I/O 的数据处理还是存储器的数据存取。缺点是它需要专门的 I/O 指令, 其数量远不如访问存储器指令那么丰富和众多, 故程序设计不够灵活; 控制逻辑也因 I/O 接口和存储器的读、写控制命令分别设置和众多而比较复杂。

如: IBM-PC 系统、Z80 系统微机采用这种方式。

表 1.1 两种编址方式的端口访问指令

统一编址 (MCS-51)		独立编址 (IBM-PC)	
输入指令	输出指令	输入指令	输出指令
MOVX A, @DPTR	MOVX @DPTR, A	IN AX, PORT	OUT PORT, AX
MOVX A, @Ri	MOVX @Ri, A	IN AL, PORT	OUT PORT, AL
		IN AX, DX	OUT DX, AX
		IN AL, DX	OUT DX, AL

注: PORT 是一个 8 位的端口地址

1.4 8086/8088、MCS-51 的 I/O 地址分配和地址译码

若要进行计算机控制系统设计和开发, 必需清楚购买回的计算机 I/O 端口地址分配情况, 即哪些已分配给系统的外设, 还有哪些是空闲可选用的。以下用表格的形式介绍 IBM-PC 系列微机、MCS-51 系统微机 I/O 地址分配。

1.4.1 IBM-PC/XT 的 I/O 地址分配

8088CPU 为 16 位微型计算机系统, IBM-PC 是其最有代表性的主流机型。

PC 机系统提供了 10 位地址线作为 I/O 地址。即地址空间范围是 000 ~ 3FFH。但有些分配给系统基本 I/O 设备的接口, 有些分配给可选 I/O 设备的接口, 有的生产厂家建议保留。用户选择地址时要注意。前 256 个端口地址为 000 ~ 3FFH 分配给系统板上的 I/O 接口芯片占用。地址从 100 ~ 3FFH 为扩展槽上的 I/O 接口控制卡占用。在使用 I/O 端口时应注意: 已被系统配置占用了的地址不准使用; 系统未占用且计算机厂家并未申明保留的地址, 用户可以使用。

1.4.2 MCS-51 的 I/O 地址分配

MCS-51 系列单片机中, 51 子系列的单片机内部有四个 8 位的并行 I/O 口: P₀、P₁、P₂、

P₃，它们的地址依次是 80H、90H、A0H、B0H。MCS-51 单片机的内部四个 I/O 口的编址属于单片机内部 RAM 范围，地址为 8 位二进制数，若片内 I/O 口不够可通过扩展的方法，在片外扩展 I/O 口，口地址可以是 8 位或 16 位的，但对于外部扩展的 I/O 口的操作指令，必需为 MOVX 指令。另外 MCS-51 内部还有串行口，串口数据缓冲器地址为 99H；串口控制寄存器地址为 98H。

表 1.2 IBM-PC 机 I/O 地址分配表

分 类	地 址 范 围		I/O 端 口 名 称
	PC / XT	PC / AT	
系 统 板	000~00FH	000~01FH	DMA 控制器 1(8237)
		0C0H~0DFH	DMA 控制器 2
	080H~083H	080H~09FH	DMA 页面寄存器
	020H~021H	020H~03FH	中断控制器 1 8259
		0A0H~0BFH	中断控制器 2 8259
	040H~043H	040H~05FH	定时器 8253
	060H~063H		并行接口芯片 8255
		060H~06FH	键盘控制器
		070H~07FH	CMOS RAM
	0A0H		NMI 屏蔽寄存器
		0F0H~0FFH	协处理器
扩 展 槽	200~20FH	200~20FH	游戏控制卡
	210~21FH		扩展部件
	370~37FH	370~37FH	并行口控制卡 1
	270~27FH	270~27FH	并行口控制卡 2
	3F8~3FFH	3F8~3FFH	串行口控制卡 1
	2F0~2FFH	2F0~2FFH	串行口控制卡 2
	300~31FH	300~31FH	原型插件板
	3A0~3AFH	3A0~3AFH	同步通信卡 1
	380~38FH	380~38FH	同步通信卡 2
	3B0~3BFH	3B0~3BFH	单显 MDA
	3D0~3DFH	3D0~3DFH	彩显 CGA
	3C0~3CFH	3C0~3CFH	彩显 EGA/VGA
	320~32FH	1F0~1FFH	硬驱控制卡
	3F0~3F7H	3F0~3F7H	软驱控制卡

1.4.3 地址译码

有的外设有一个端口地址就可以了，但大多情况一个设备的接口要占用多个 I/O 端口地址。要实现地址控制必须有一个与分配的 I/O 地址相一致的地址译码电路。译码的方法很多，以下给出三种常用的译码方法：

1.4.3.1 门电路译码

门电路地址译码是采用一些简单门电路构成的译码电路，这种电路很简单，端口地址固定不变。图 1.2 为门电路译码电路。

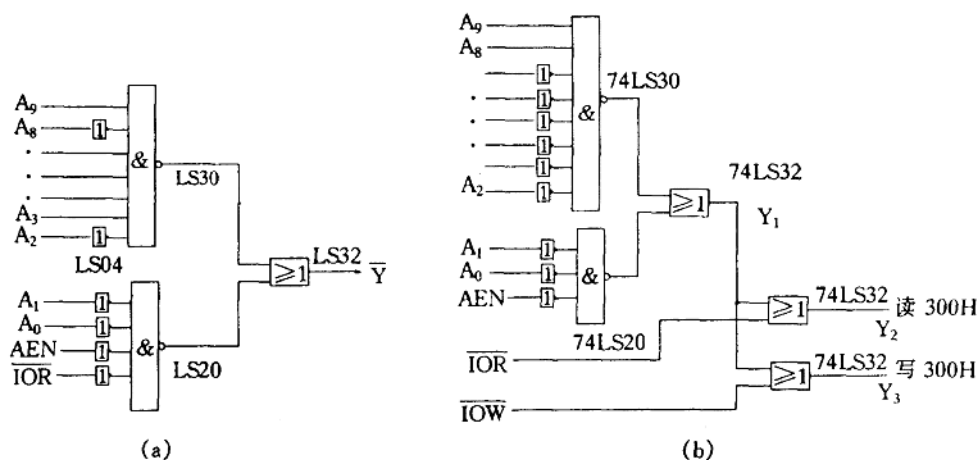


图 1.2 门电路译码电路图

图 1.2 (a) 中, $\bar{Y} = "0"$ 时, 地址码 $A_9 \sim A_0$ 依次为: 1011111000, 即口地址为: 2F8H, 这个译码电路形成一个端口选择信号, 用于输入指令读, 即当 $\overline{IOR}$ 有效, $AEN = "0"$, 地址码为 2F8H 即选中这个端口。AEN 是 PC 机中重要的控制信号, 当 $AEN = "0"$ 时, 对端口地址译码进行控制。

图 1.2 (b) 中当 $Y_2 = "0"$ 时, $\overline{IOR}$ 与 Y_1 必须同时为 "0", 即地址码 $A_9 \sim A_0$ 依次为: 1100000000, 即 300H, 同时 $AEN = 0$ 。这个译码电路形成两个端口选择信号, 一个用于输入指令读, 地址为 300H, 同时 $\overline{IOR}$ 有效; 一个用于输出指令写, 地址也为 300H, 同时 $\overline{IOW}$ 有效。

1.4.3.2 译码器译码

当接口电路中需要多个端口地址时, 采用译码器译码比较方便。译码器的型号很多, 如常用的 74LS138 是一个 3-8 译码器, 还有 74LS154 的 4-16 译码器及双 2-4 译码器 74LS139 和 74LS155 等。如图 1.3 是 PC 机系统板上地址译码电路。

例如当输入的地址码 $A_9 \sim A_5$ 分别为全 0, 且 $AEN = 0$ 时, Y_0 输出为低电平, 此时选中 DMA 控制器, 对应的地址码为 000H~01FH, 正好与表 1.2 符合。读者也可以自己进行多个 I/O 端口的验证, 与前表的地址是一致的。

1.4.3.3 线选法

当 I/O 端口个数不多, 为了减少译码器的数量及简化硬件系统, 在微机系统中有时采用线选译码, 下面以 MCS-51 为例, 简述其原理。如图 1.4。

图 1.4 (a) 为单片机 8031 与 8253 接口的硬件连线图, 因为 P_{27} 接至 8253 芯片的片选 $\overline{CS}$, 所以 8253 接口地址为 0XXX XXXX XXXX XXXXB, 可写为 7FFFH。图 1.4 (b) 中 P_{25} 与接口第一个 DAC0832 的片选 $\overline{CS}$ 相接, P_{26} 与第二个 DAC0832 的片选 $\overline{CS}$ 相接。为保证 8031 与两个接口进行准确无误地数据传输, 两个接口地址通过 P_{25} 和 P_{26} 分别进行选择。选中 1# 的 DAC0832 地址为 X10X XXXX XXXX XXXX, 可以写成 DFFFH, 选中 2# 的 DAC0832 地

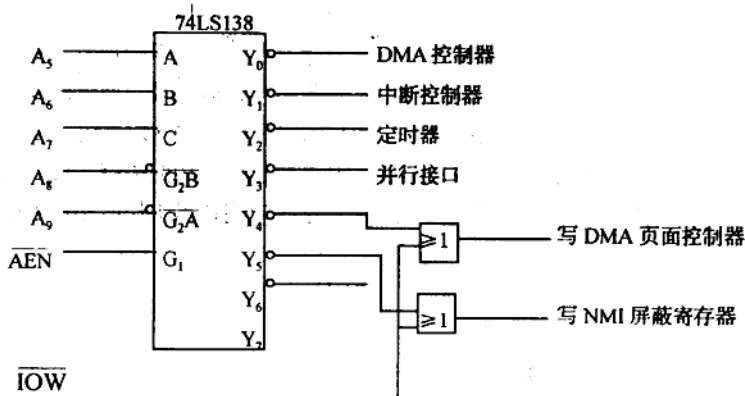


图 1.3 系统板上地址译码电路

址为 X01X XXXX XXXX XXXX，可以写成 BFFFH。这种通过地址直接选择接口地址的方法，称为线选法。在 MCS-51 系统中有许多这样的应用例子。

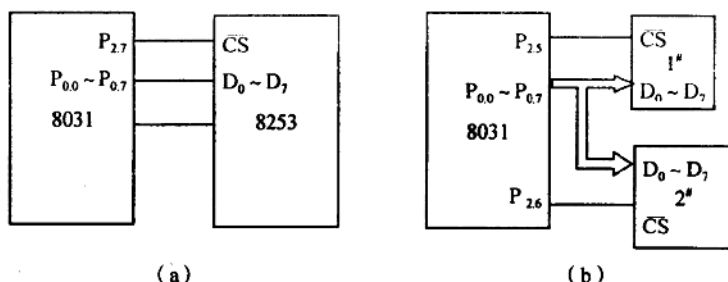


图 1.4 线选法举例

1.5 接口设计与分析的基本方法

外设都是通过接口与系统总线相连，在分析接口电路时，首先把信号线按地址总线、数据总线、控制总线进行分类分析。搞清控制信号是电平有效或是脉冲跳变有效，是高电平有效还是低电平有效，时序关系是什么。分析数据线的位数是 8 位、16 位、32 位或更高。依据地址线采用的地址译码方式，分析出接口的地址。设计的接口电路应能进行信号转换，使外设、CPU 信号通过接口得到转换，能够协调工作。在接口电路的设计中，一般不要做复杂的参数计算。选择实用熟悉的接口芯片，把其与系统总线正确地连接起来，并依硬件的实际接线，编写相应的接口驱动程序。采用可编程的接口芯片，功能完善，应用灵活，选用高性能集成电路代替一般电路以简化电路设计和减轻总线的负载。系统扩展时在满足系统功能要求的前提下，应留有余地，便于进行二次开发。仔细核查 CPU 与系统中相关器件的速度匹配和功率匹配，并留有余地，保证系统正常运行。

接口硬件电路提供了外设与 CPU 交换数据的工作条件或者说只是创造了外设工作的环境，真正要使外设工作、何时工作、如何工作，则要依靠相应的接口驱动程序，即微机系统工作需要“硬软兼施，方能奏效”。