

GH

高等学校
工科电子类 规划教材

刘乐善 胡盛斌 叶济忠

微型计算机接口技术

Weixing
JISUANTI

JIEKOU

JISHU

电子科技大学出版社

高等学校教材

微型计算机接口技术

刘乐善 胡盛斌 叶济忠

电子科技大学出版社

• 1994 •

930910

[川]新登字 016 号

内 容 简 介

本书在阐述微型计算机接口技术基本原理和有关技术(中断、DMA、总线和定时)的基础上,着重讨论了并行 I/O 接口、人—机接口、串行 I/O 接口、模拟接口和磁盘机接口的设计原理及相应接口芯片的接口方法和编程使用,并提供了应用实例。

本书可作电子精密机械专业教材,也可供其他专业本科生、研究生学习和工程技术人员参考。

微型计算机接口技术

刘乐善 胡盛斌 叶济忠

*

电子科技大学出版社出版

(成都建设北路二段四号)邮编 610054

郫县唐昌印制厂印刷

新华书店经销

*

开本 787×1092 1/16 印张 11.75 字数 280 千字

版次 1994 年 6 月第一版 印次 1994 年 6 月第一次印刷

印数 1—5000 册

中国标准书号 ISBN 7-81043-029-7/TP·14

定价: 8.80 元

出版说明

根据国务院关于高等学校教材工作的规定,我部承担了全国高等学校和中等专业学校工科电子类专业教材的编审、出版的组织工作。由于各有关院校及参与编审工作的广大教师共同努力,有关出版社的紧密配合,从1978~1990年,已编审、出版了三个轮次教材,及时供给高等学校和中等专业学校教学使用。

为了使工科电子类专业教材能更好地适应“三个面向”的需要,贯彻国家教委《高等教育“八五”期间教材建设规划纲要》的精神,“以全面提高教材质量水平为中心,保证重点教材,保持教材相对稳定,适当扩大教材品种,逐步完善教材配套”,作为“八五”期间工科电子类专业教材建设工作的指导思想,组织我部所属的九个高等学校教材编审委员会和四个中等专业学校专业教学指导委员会,在总结前三轮教材工作的基础上,根据教育形势的发展和教学改革的需要,制订了1991~1995年的“八五”(第四轮)教材编审出版规划。列入规划的,以主要专业主干课程教材及其辅助教材为主的教材约300多种。这批教材的评选推荐和编审工作,由各编委会或教学指导委员会组织进行。

这批教材的书稿,其一是从通过教学实践、师生反映较好的讲义中经院校推荐,由编审委员会(小组)评选择优产生出来的,其二是在认真遴选主编人的条件下进行约编的,其三是经过质量调查在前几轮组织编写出版的教材中修编的。广大编审者、各编审委员会(小组)、教学指导委员会和有关出版社,为保证教材的出版和提高教材的质量,做出了不懈的努力。

限于水平和经验,这批教材的编审、出版工作还可能有缺点和不足之外,希望使用教材的单位,广大教师和同学积极提出批评和建议,共同为不断提高工科电子类专业教材的质量而努力。

电子工业部
电子类专业教材办公室

前 言

本教材系按电子工业部的工科电子类专业教材 1991~1995 年编审出版规划,由电子机械教材编审委员会计算机教材编审组征稿,推荐出版,责任编委为张江陵教授。

本教材由华中理工大学刘乐善担任主编,东南大学郁慧娣担任主审。

本课程的参考学时数为 40 学时。全书共分八章。第一章概述,从微计算机系统的角度出发,引出接口技术问题,并着重讨论了 CPU 与接口之间传送信息的几种基本方式;第二章总线,介绍几种微机总线标准及其用法;第三章定时/计数,为构造微机控制系统的定时与计数提供基本方法;第四章并行接口,讨论了并行接口的特点及设计原理与方法;第五章人一机接口,论述了键盘与显示器接口的基本原理,并介绍几种交互式人机接口;第六章串行接口,讨论了串行通信的基本概念,接口标准及其接口电路的实现方法;第七章数/模、模/数接口,阐述了 A/D、D/A 转换器接口电路的任务、形式及设计方法;第八章磁盘机接口,论述了软磁盘子系统驱动器、控制器的工作原理及接口标准,介绍了温盘机的基本原理。

本教材是在使用多年的华中理工大学本科生、研究生教材的基础上,经修改和补充而成。其中第一、四、六、七章由刘乐善编写;第二、三、五章由胡盛斌编写;第八章由叶济忠编写。全书由刘乐善统稿。在本书的编写过程中,得到华中理工大学计算机外部设备教研室的大力支持,空军雷达学院王长胤教授对书稿的修改提出了宝贵意见。刘学清、王群同志承担了全部书稿的整理与录入工作。在此,一并表示最诚挚的感谢。

由于编者水平有限,书中难免存在一些缺点和错误,热忱希望广大读者批评指正。

编 者

目 录

第一章 微型计算机接口技术概述.....	1
§ 1.1 微机接口	1
§ 1.1.1 接口及接口的功能	1
§ 1.1.2 CPU 与接口之间传送信息的方式	3
§ 1.1.3 设计与分析接口电路的基本方法	4
§ 1.2 I/O 接口地址译码	5
§ 1.2.1 I/O 接口的寻址方式	5
§ 1.2.2 系统对 I/O 口地址分配	6
§ 1.2.3 I/O 口地址译码方法	7
§ 1.3 中断技术	9
§ 1.3.1 8088/8086 的中断系统	9
§ 1.3.2 8259A 可编程中断控制器	10
§ 1.3.3 8259A 在微机系统中的使用	18
§ 1.4 DMA 技术	20
§ 1.4.1 8237A—5 DMA 控制器	20
§ 1.4.2 DMA 控制器在系统中的使用	27
第二章 总线技术	31
§ 2.1 概述	31
§ 2.2 IBM PC 总线	36
§ 2.3 STD 总线	38
§ 2.3.1 STD 总线的特点	38
§ 2.3.2 STD 总线规范	38
§ 2.3.3 STD 应用举例	40
§ 2.4 几种常用的总线标准	42
§ 2.4.1 MULTIBUS 总线	42
§ 2.4.2 VME 总线	46
§ 2.4.3 RS—232C 总线	48
§ 2.4.4 IEEE 488 总线接口(即 GP—IB 总线)	49

第三章 定时/计数技术	51
§ 3.1 概述	51
§ 3.2 可编程定时/计数器 8253(8254)	52
§ 3.2.1 8253 的工作原理	52
§ 3.2.2 8253 的工作方式	54
§ 3.2.3 8253(8254)的编程	59
§ 3.3 8253 在系统中的应用	62
第四章 并行接口	65
§ 4.1 并行接口概念	65
§ 4.2 可编程并行接口	65
§ 4.2.1 8255A 工作原理及编程命令	66
§ 4.2.2 8255A 的 0 方式及连接方法	71
§ 4.2.3 8255A 的 1 方式及连接方法	74
§ 4.2.4 8255A 的 2 方式及连接方法	80
第五章 人一机接口	85
§ 5.1 概述	85
§ 5.2 键盘及接口电路	85
§ 5.2.1 微型机与非编码键盘的接口	85
§ 5.2.2 8279 键盘接口芯片	87
§ 5.2.3 PC 机的键盘接口	93
§ 5.3 LED 显示器及接口	95
§ 5.4 CRT 显示器接口	97
§ 5.5 计算机的图形接口	99
§ 5.5.1 视频控制器 MC6845	99
§ 5.5.2 几种常见视屏标准	104
§ 5.6 交互式人一机接口	106
§ 5.6.1 鼠标器	106
§ 5.6.2 光笔	109
§ 5.6.3 操纵杆	111
§ 5.7 汉字输入与输出	112
第六章 串行通信接口	115
§ 6.1 串行通信的基本概念	115

§ 6.1.1	串行通信的特点及通信方法	115
§ 6.1.2	数据传送模式	117
§ 6.1.3	信号的调制与解调	118
§ 6.1.4	传输速率与传送距离	119
§ 6.2	串行接口标准	119
§ 6.3	串行通信接口	124
§ 6.3.1	串行通信接口的基本任务	124
§ 6.3.2	用 8251 组成的串行接口电路	125
§ 6.3.3	8251 的内部逻辑与外部信号	125
§ 6.3.4	8251 的控制字和状态字	128
§ 6.3.5	8251 应用举例	131
§ 6.4	简述 IBM-PC/XT 异步通信适配器	134
第七章	数/模 模/数转换器接口	136
§ 7.1	数/模转换器接口	136
§ 7.1.1	D/A 转换器及其连接特性	136
§ 7.1.2	D/A 转换器与微处理器的接口方法	137
§ 7.1.3	D/A 转换器接口电路	138
§ 7.1.4	D/A 转换器应用的微机接口电路	142
§ 7.2	模/数转换器接口	144
§ 7.2.1	A/D 转换器及其连接特性	144
§ 7.2.2	A/D 转换器与微处理器的接口方法	145
§ 7.2.3	A/D 转换器接口电路举例	146
§ 7.3	数据采集系统的设计	152
第八章	磁盘驱动器接口	157
§ 8.1	软磁盘机	157
§ 8.1.1	软磁盘机的基本结构	157
§ 8.1.2	软磁盘机的种类及信号接口	158
§ 8.1.3	软磁盘机的格式化	159
§ 8.2	软磁盘机 BIOS 驱动程序	161
§ 8.2.1	软盘 BIOS 的调用	161
§ 8.2.2	BIOS 的结构与功能	162
§ 8.3	软盘控制器(FDC)专用集成电路	164
§ 8.3.1	μPD765 的主要功能及内部组成	164
§ 8.3.2	μPD765 的工作过程	166
§ 8.3.3	命令序列和结果状态序列	166

§ 8.4 IBM-PC 软盘控制器	167
§ 8.4.1 软盘控制器电原理图及主机侧接口电路	167
§ 8.4.2 驱动器侧的接口电路	168
§ 8.4.3 软盘控制器的应用	169
§ 8.5 温盘机	170
§ 8.5.1 温盘机的种类和规格	171
§ 8.5.2 温盘机信号接口	171
§ 8.6 硬盘 BIOS 驱动程序	173
§ 8.6.1 接口寄存器	173
§ 8.6.2 命令序列(设备控制块 DCB)	175
§ 8.6.3 主机与控制器之间的 I/O 实现过程	175
§ 8.7 温盘控制器	176
§ 8.7.1 WD1010 的内部结构	176
§ 8.7.2 任务文件寄存器	176
§ 8.7.3 命令寄存器及命令组	178
§ 8.7.4 WDC 与主机及驱动器接口	178
§ 8.7.5 温盘控制器设计的基本问题	179
参考文献	180

第一章 微型计算机接口技术概述

§ 1.1 微机接口

§ 1.1.1 接口及接口的功能

一个微机系统除了中央处理器 CPU 之外,还必须要有存储器、接口电路、外部设备、总线以及时钟、电源等部分,如图 1.1 所示。从图中可以看出,各类外部设备和存储器,都是通过各自的接口电路连到微机系统的总线上去的,因此用户可以根据自己的要求,选用不同类型的外部设备,设置相应的接口电路,把它们挂到系统总线上,构成不同用途、不同规模的应用系统。

所谓接口就是微处理器 CPU 与外界的连接部件(电路),是 CPU 与外界进行信息交换的中转站。例如源程序或原始数据要通过接口从输入设备送进来,运算结果要通过接口向输出设备送出去;控制命令通过接口发出去,现场状态通过接口取进来,这些来往信息都要通过接口进行变换与中转。微机接口技术是采用硬件与软件相结合的方法,研究微处理器如何与外部世界进行最佳耦合与匹配,以在 CPU 与外界之间实现高效、可靠的信息交换的一门技术。这里所说的“外部世界”,是指除 CPU 本身以外的所有设备或电路,包括存储器、I/O 设备、控制设备、测量设备、通信设备、A/D、D/A 转换器等。那么,CPU 与外设之间的接口应具有哪些功能呢?从解决 CPU 与外设的连接时存在有矛盾的观点来看,一般有如下功能:

一、数据缓冲功能

接口中一般都设置数据寄存或锁存器,以解决主机高速与外设低速的矛盾,避免因速度不一致而丢失数据。

二、设备选择功能

系统中一般带有多种外设,同一种外设中也可能有多台,而 CPU 在同一时间里只能与微机一台外设交换信息,这就要借助于接口的地址译码对外设进行寻址。高位地址用于芯片选择,低位地址进行芯片内部寄存器或锁存器的选择,以选定需要与自己交换信息的设备,只有被选定的设备才能与 CPU 进行数据交换或通信。

三、信号转换功能

由于外设所能提供的状态信号和它所需的控制信号往往同微机的总线信号不兼容,尤其是联接不同公司生产的芯片时,信号变换就不可避免。因此,信号转换就成为接口设计中的一个重要任务。这包括 CPU 的信号与外设的信号的逻辑关系、时序配合以及电平匹配上的转换。

四、接受、解释并执行微处理器的命令功能

CPU 对外设的各种命令都是以代码的形式发到接口电路,再由接口电路解读后,形成一系列控制信号去控制被控对象的。为了联络,接口电路还要提供寄存器“空”、“满”、或“准备好”、“忙”、“闲”等状态信号,向 CPU 报告寄存器的工作情况。

五、中断管理功能

当外设需要及时得到 CPU 的服务,特别是在出现故障需要 CPU 进行刻不容缓的处理时,就要求在接口中设置中断控制器,为 CPU 处理有关中断事务(提出中断请求,进行中断优先级

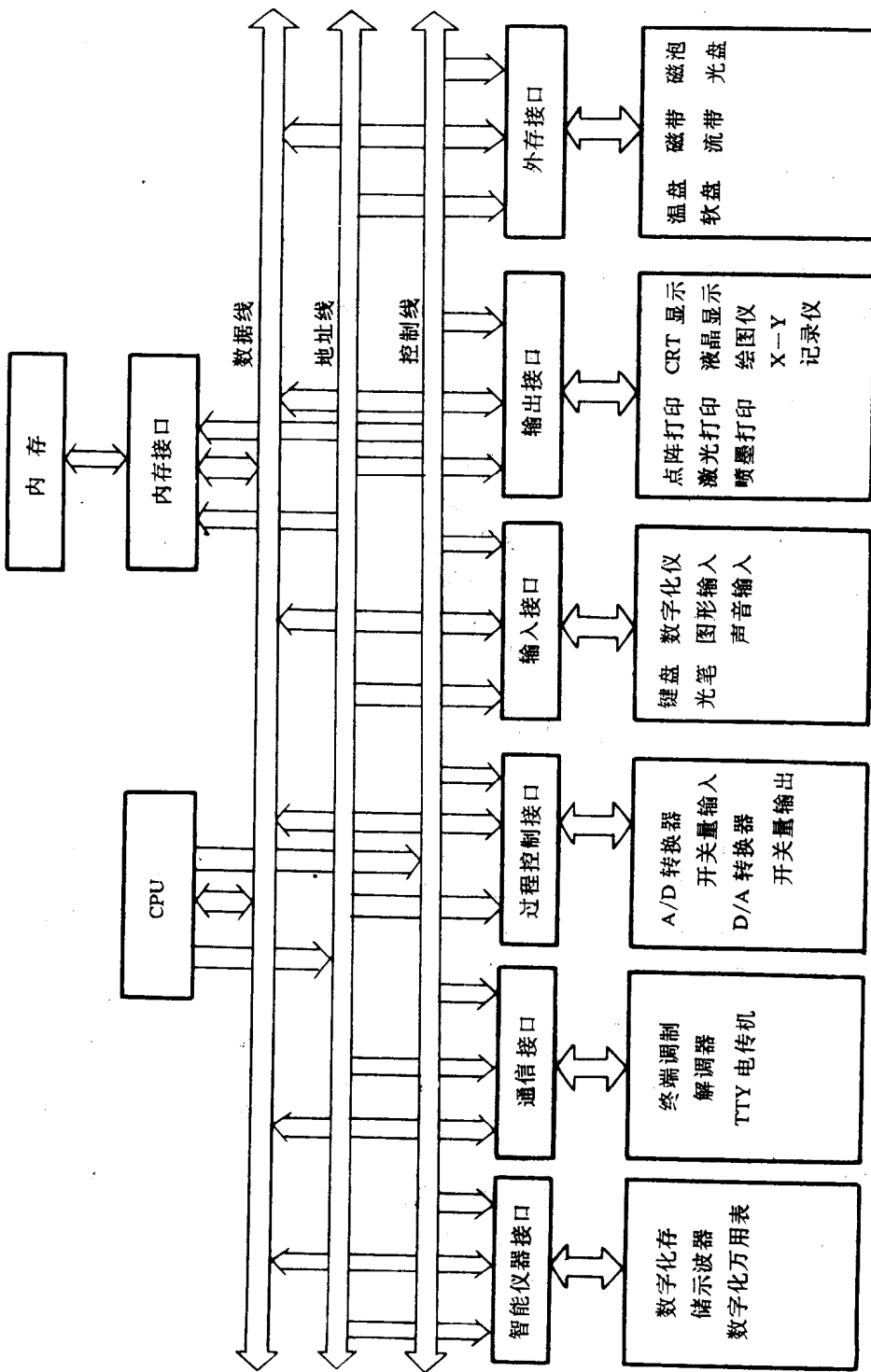


图 1.1 微机系统组成框图

排队,提供中断向量等),这样既增加了微机系统对外界的响应速度,又使 CPU 与外设并行工作,提高了 CPU 的效率。

六、数据宽度变换的功能

CPU 所处理的是并行数据(8 位、16 位或 32 位),而有的外设(如串行通信设备、磁盘驱动器等)只能处理串行数据,在这种情况下,接口就应具有数据并→串和串→并的变换能力。为此,在接口中设置移位寄存器。

七、可编程功能

现在的接口芯片基本上都是可编程的,这样在不改动硬件的情况下,只修改驱动程序就可以改变接口的工作方式,大大增加了接口的灵活性和可扩充性,使接口向智能化方向发展。

上述功能并非是每种接口都要求具备,对不同配置和不同用途的微机系统,其接口功能不同,接口电路的复杂程度大不一样。但前四种功能和可编程能力是一般接口都需要的。

§ 1.1.2 CPU 与接口之间传送信息的方式

外部设备与微机之间信息传送实际上是 CPU 与接口之间的信息传送,传送的方式不同,CPU 对外设的控制方式也不同,从而使接口电路的结构及功能也不同,所以接口电路设计者对 CPU 与外设之间采用什么方式传送信息颇为关心。一般有三种方式,即查询方式、中断方式和 DMA 方式。

一、查询方式

查询方式是主机在传送数据(包括读入和写出)之前,要检查外设是否“准备好”。若没有准备好,则再查其状态,直至外设准备好了,即确认外部设备已具备传送条件之后,才能进行数据传送。显然,在这种方式下,CPU 要传送一个数据,需花费很多时间来等待外设进行数据传送的准备,且 CPU 与外设不能同时工作,各种外设也不能同时工作,因此,信息传送的效率非常之低。但实现这种方式的接口电路简单,硬件开销小,在 CPU 不是太忙且传送速度不高的情况下,可以采用。

二、中断方式

采用中断方式传送信息时,无需反复测试外部设备是否准备好的状态。在外部设备没有作好数据传送准备时,CPU 可以运行与传送数据无关的其它指令。外设作好传送准备后,主动向 CPU 请求中断,CPU 若响应这一请求,就暂停正在运行中的程序,转入用来进行数据传送的中断服务程序,完成中断服务程序(即完成数据传送)后,自动返回原来运行的程序。外设重新进入工作状态,CPU 又与外设并行工作。这样就提高了 CPU 的效率。为了实现中断传送,要求在 CPU 与外设之间设置中断控制器,硬件结构较为复杂。中断方式用于 CPU 的任务比较多,传送速度不太高的系统中,尤其适合实时控制及紧急事件的处理。

三、直接存储器存取(DMA)方式

虽然中断传送方式可以在一定程度上实现 CPU 与外设并行工作,但是在外设与内存之间,或在外设与外设之间进行数据传送时,还是要经过 CPU 中转(即经过 CPU 的累加器读进和送出),这对高速外设(如磁盘)在进行大批量数据传送时,会造成中断次数过于频繁,不仅传送速度上不去,且耗费大量 CPU 的时间。为此,采用直接存储器存取方式,使 CPU 不参加数据的传送工作,由 DMA 控制器来实现内存与外设或外设与外设之间的直接快速传送,从而也减轻了 CPU 的负担。在小型计算机中,这种方式叫作数据通道传送方式。这种方式使计算机的:

硬件结构发生了变化,信息传送从以 CPU 为中心变为以内存为中心。若采用高速存储器,就使外设与 CPU 分时访问内存得以实现。

DMA 方式实际上是把输入输出过程中外设与主存交换信息的那部分操作与控制交给了 DMA 控制器,简化了 CPU 对输入输出的控制。这对高速度大批量数据传送特别有用。但这种方式要求设置 DMA 控制器,电路结构复杂,硬件开销大。

§ 1.1.3 设计与分析接口电路的基本方法

对一个已有接口电路进行分析解剖,或设计一个新的接口电路时,应如何入手呢?

一、分析接口两侧的情况

接口的一侧是 CPU,另一侧是外设。对 CPU 一侧,要搞清是什么类型的 CPU,以及它提供的数据线的宽度(8 位、16 位、32 位等)、地址线的宽度(16 位、20 位、24 位)和控制线的逻辑定义(高电平有效、低电平有效、脉冲跳变),时序关系有什么特点。其中数据与地址线比较规整,不同的 CPU 变化不大,而控制线往往因 CPU 不同其定义与时序配合差别较大,故重点要放在控制线的分析上。外设一侧的情况很复杂,因为外设种类繁多,型号不一,所提供的信号线五花八门,其逻辑定义、时序关系、电平高低差异甚大。对这一侧的分析重点放在搞清被连外设的工作原理与特点的基础上,找出需要接口为它提供哪些信号才能正常工作,它能反馈给接口哪些状态信号报告工作过程,以达到与 CPU 交换数据的目的。外设的种类甚多,从大容量快速磁存储器到指示灯、扬声器,不管其复杂程度如何,只要将它们的工作原理及各自原始的(本身所固有的)来去信号线的特性分析清楚了,那么对接口电路的剖析也就不难。

二、进行信号转换

由上述可知,要把 CPU 与外设两侧的信号线不加处理(改造)就直接连接是不行的。因此,经过对接口两侧信号的分析,找出差别之后,设法进行信号转换与改造,使之协调。这可以从 CPU 一侧做起,将 CPU 的信号进行转换来适合外设的要求,也可以从外设一侧做起,将外设的信号进行改造(逻辑处理)来适合 CPU 的要求。经过改造的信号线,在功能定义、逻辑关系、时序配合上,能同时满足两侧的要求,故可以协调工作。

三、接口驱动程序分析

接口的硬件电路只提供了接口工作的条件,要使接口真正发挥作用还要配备相应的驱动程序。对于微机系统中的标准设备(如 CRT、KB、PRINTER、HD、COMMUNICATION 等),在 ROM-BIOS 中都有相应的功能子程序供用户调用。但是对于接口设计者来说,常常碰到的是些非标准设备,况且在微机控制应用中,往往采用单板机或单片机,此时没有配置 BIOS,故无功能子程序可供调用,所以需要自己动手编制接口驱动程序,为此,必须了解外设的工作原理、接口电路的硬件结构。接口驱动程序也是模块化、结构化,一般是由初始化模块、功能模块几部分组成。

最后,在对具体接口电路进行设计与分析时要注意两点:

(1)尽量采用 LSI, VLSI 集成接口芯片,这样不仅开发周期短,而且可靠性高,特别是选用与 CPU 同一系列的接口芯片,就更为方便。

(2)软硬结合,综合考虑。在不同场合,实现同一种接口功能,可以采用硬件软化,也可以软件硬化的方式,根据需要与可能具体进行优化选择。不过一旦硬件结构定下来,相应的软件编程也就跟着定下来了。

总之,分析接口问题的基本方法可归纳为:首先分析接口两侧的信号及其特点,找出两侧进行联接时存在的差异,针对要解决两侧的这些差异,来确定接口应完成的任务;为了实现接口的任务,要考虑作哪些信号变换,选择什么样的元器件来进行这些变换,据此,进行接口电路功能模块化总体结构设计,这样就完成了对接口硬件的分析。对接口问题,仅有硬件分析,还不能真正了解接口,还必须对接口的软件编程进行分析,而软件编程是与硬件结构紧密相连的,硬件发生变化,接口的驱动程序也就随之改变。

随着集成电路集成度的增高,电子计算机向微型化和超微型化方向发展,微型计算机已成为导弹、智能机器人、人类宇航、探测太空奥秘复杂系统必不可少的智能部件。目前,微机不仅作为科学计算、实时控制、现代化通信和管理的手段,而且也成为人类进行学习、看病、咨询、智力游戏等生活服务和娱乐的工具。然而,在微机系统中,微处理器的这种神通广大的功能必须通过外部设备才能实现,而外设与微处理器之间的信息交换及通讯又是靠接口来实现的,因此,接口技术就成为直接影响微机系统的处理能力和微机推广应用的关键。可以这样说,微机的应用是随着外部设备不断更新和接口技术的发展而深入到各个领域的。所以,掌握微机接口技术就成为当代科技和工程技术人员应用微型计算机必不可少的基本技能。

§ 1.2 I/O 接口地址译码

§ 1.2.1 I/O 接口的寻址方式

外部设备接口中能被 CPU 直接访问的寄存器通常称之为端口(PORT)。CPU 通过这些端口发送命令,读取状态和传送数据,因此,一个接口可有几个端口,如命令口、状态口、数据口等。如何实现对这些端口的访问,就是所谓 I/O 接口寻址问题。有两种寻址方式。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 端口地址和存储器地址分开独立编址,即 I/O 映射方式。

一、统一编址方式

这种方式,是从存储空间划出一部分地址空间给 I/O 设备,把 I/O 端口当作存储器单元一样进行访问,不设置专门的 I/O 指令。凡对存储器可以使用的指令均可用于端口。6800 系列,6502 系列微型机和 PDP-11 小型机就是采用这种方式。

统一编址方式由于对 I/O 设备的访问也是使用访问存储器的指令,指令类型多,功能齐全,这不仅使访问 I/O 设备端口进行输入/输出操作灵活、方便,而且还可对端口内容进行算术逻辑运算和移位等等。另外,能给端口以较大的编址空间,这对大型控制系统和数据通信系统是很有意义的。其主要缺点是端口占用了存储器的地址,使存储器容量减小,还有指令长度比起专门 I/O 指令要长,因而执行时间较长。

二、独立编址方式

这种方式是接口中的端口地址单独编址而不占用存储空间,访问端口时要使用专门的 I/O 指令,可访问的端口地址一般为 256~1024 个。大型计算机普遍采用这种方式,有些微机,如 8088/8086、Z-80 系列机也采用这种方式。

这种方式的优点是不占用存储器地址,故不会减少存储器容量;单独用于端口的地址线少,地址译码方便,I/O 指令短,执行速度快,特别是由于专门 I/O 指令与存储器访问指令有明

显的差别,使程序中 I/O 操作和其它操作层次清晰,便于理解。这种方式的缺点是指令类型少,功能单一,只能对端口进行输入输出操作。这种方式要求 CPU 需设置两组读/写控制信号(存储器读/写,I/O 读/写)。例如 8086/8088 最小模式下要用 $\overline{IO}/\overline{M}$ 引脚和 $\overline{RD}$ 、 $\overline{WR}$ 构成两组控制信号;而在最大模式下,由于引脚不够用,没有直接输出 $\overline{IO}/\overline{M}$ 、 $\overline{RD}$ 、 $\overline{WR}$ 对外设和存储器进行读写操作的控制信号,而由 $\overline{S_2}$ 、 $\overline{S_1}$ 、 $\overline{S_0}$ 三个总线周期状态信号编码送至总线控制器 8288,经它解读后,再生成存储器读/写和 I/O 读/写两组控制信号。

§ 1.2.2 系统对 I/O 口地址分配

一、I/O 指令的应用

不同的微机系统对 I/O 口地址的分配方案是不同的。对于接口设计者来说,搞清楚系统 I/O 口地址分配十分重要,因为要把新的 I/O 设备加入到系统中去就要在 I/O 地址空间中占一席之地,搞清楚哪些地址已分配给了别的设备不能使用,哪些地址是空闲的,才能为我所用。下面介绍 IBM-PC 的 I/O 口地址分配情况。

PC 机 I/O 口地址是独立编址方式,由于采用专门的 I/O 指令访问端口,故 I/O 端口地址和存储器地址可以重叠,而不会相互混淆。

在专用 I/O 指令中,若采用直接寻址方式,即用一个字节作为端口地址,最多可访问 256 个端口。系统主板上的 I/O 口,就是采用的直接寻址,其指令格式为:

```
IN AL, PORT(输入指令)
OUT PORT, AL(输出指令)
```

若采用间接寻址方式,即用一个字作为端口地址,最多可寻址 $2^{16}=64K$ 个端口。例如,TP86 单板机上 8255A 的三个端口地址为 0FFF8H,0FFFAH 和 0FFFBH。其指令格式为:

```
MOV DX, 0FFF8H
IN AX, DX    或    IN AL, DX
OUT DX, AX   或    OUT DX, AL
```

二、I/O 口地址的分配

表 1.1 系统板 I/O 口地址使用表

地址空间	器 件/设 备	实际地址
0000~001F	DMA 控制器 8237A	00~0FH
0020~003F	中断控制器 8259A	20~21H
0040~005F	定时/计数器 8253A	40~43H
0060~007F	并行接口芯片 8255A	60~63H
0080~009F	DMA 页面寄存器	80~83H
00A0~00BF	非屏蔽中断屏蔽寄存器	A0H
00C0~01FF	未用	

实际上 PC 机仅用了 $A_9 \sim A_0$ 十位地址作为 I/O 口编码寻址,即可用 1024 个口地址。低端 512 个(0000~01FFH)供系统板电路使用,高端 512 个(200~03FFH)供扩充插槽使用。当 $A_9=0$ 时表示低端地址为系统板所用, $A_9=1$ 时表示高端地址为扩充插槽所用。因此,用户在设

计 I/O 设备接口卡时,一定要使口地址译码电路中的 $A_9=1$ 。

在 1024 个口地址中,有些已被系统占用,有些已被选配的 I/O 接口卡占用,还有一些被保留作今后开发使用。最后才是留给用户可用的 I/O 口地址。PC 机 I/O 口地址使用情况如表 1.1 所示,表示系统板 I/O 地址 000~1FFH 使用情况。表 1.2 表示扩充插槽 I/O 地址 200~3FFH 使用情况。原则上讲,凡未被占用的地址用户都可以使用,但要考虑系统的现有配置情况和计算机厂家为今后发展需要而留有的余地,以免发生 I/O 口地址重叠和冲突。一般用户可使用 300H~31FH 地址。它是留作实验卡用的。

表 1.2 插槽 I/O 口地址使用表

地址空间	器件/设备	地址空间	器件/设备
200~20F	游戏卡	380~38F	SDLC 通信卡
210~217	扩充部件	390~3AF	未用
218~2F7	未用	3B0~3BF	单显/打印卡
2F8~2FF	异步通信卡(二)	3C0~3CF	未用
300~31F	未用	3D0~3DF	彩显/图形卡
320~32F	硬盘卡	3E0~3EF	未用
330~377	未用	3F0~3F7	软盘卡
378~37F	打印卡	3F8~3FF	异步通信卡(一)

§ 1.2.3 I/O 口地址译码方法

CPU 为了对 I/O 口进行读/写操作,就需选定与自己交换信息的端口地址,如何通过 CPU 发来的命令代码(地址编码)来识别和确认这个端口,就是所谓口地址译码问题。口地址的译码方法灵活多样,通常可根据需要由地址和控制信号任意组合构成不同的译码电路来产生同一个口地址。从译码电路的形式可分为固定式和可选式译码,从译码电路采用的元器件,可分为门电路译码和译码器译码,目前,不少系统中采用 GAL 或 PAL 器件进行译码也十分方便。

一、固定式口地址译码

所谓固定式是指接口卡中用到的口地址不能更改,目前,接口卡中大部分都采用固定式译码。在固定式译码方式中,若仅需一个口地址时,采用门电路构成译码电路很简便。如图 1.2(a),可译出 2F8H 读操作口地址。图 1.2(b),能译出进行读/写操作的 2E2H 口地址。图中 AEN 参加译码,是用它来对口地址译码进行控制,只有当 $AEN=0$,即不是 DMA 操作时译码才有效;当 $AEN=1$,即当 DMA 操作时,使译码无效。避免在 DMA 周期,由 DMA 控制器对这些 I/O 口地址的外部设备进行读/写操作。

若接口电路中需使用多个口地址时,采用译码器译码比较方便。译码器的型号很多,如 3—8 译码器 74LS138,4—16 译码器 74LS154,双 2—4 译码器 74LS139 和 74LS155 等。采用 3—8 译码器 74LS138,可从它输入的 3 个代码(A,B,C)中解出 8 个输出来($Y_0 \sim Y_7$),它的 3 个输入控制端是 $G_1, \overline{G_{2A}}, \overline{G_{2B}}$,只有当 $G_1=1, G_{2A}=G_{2B}=0$ 时,才允许对输入 A、B、C 进行译码。若把 G_1 接高电平, $\overline{G_{2A}}$ 接 AEN, $\overline{G_{2B}}$ 接 $\overline{IOW}$ (或 $\overline{IOR}$),输入端 A、B、C 分别接 A_0, A_1, A_2 ,则当不是

DMA 操作时,由输入端 A,B,C 的编码来决定输出: $CBA=000$, 则 $Y_0=0$; $CBA=001$, $Y_1=0$; ...; $CBA=111$, $Y_7=0$ 。例如 PC 机系统板上的口地址译码,采用 74LS138 译码器,如图 1.3 所示。图中地址的高 6 位 $A_9 \sim A_4$ 参加译码,分别产生 DMAC(8237),PIC(8259),T/C(8253),PPI(8255A)的选片 $\overline{CS}$ 信号,而地址的低 4 位作芯片内部寄存器的访问地址。很明显,8237A 的口地址范围是 $000 \sim 01FH$,8259A 的口地址范围是 $020 \sim 03FH$ 等等,正好和前面表 1.1 所列出的口地址分配一致。

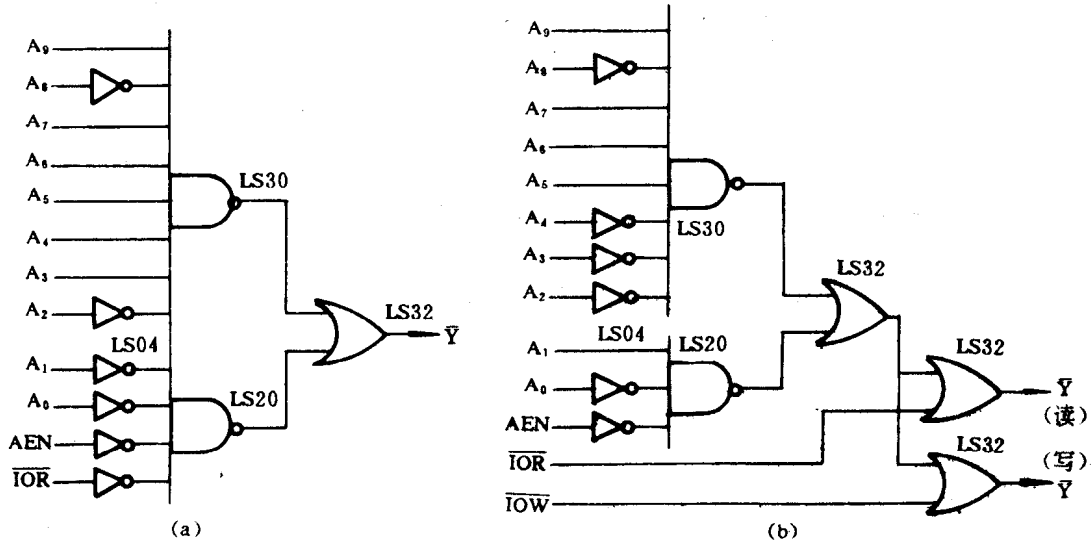


图 1.2 门电路译码

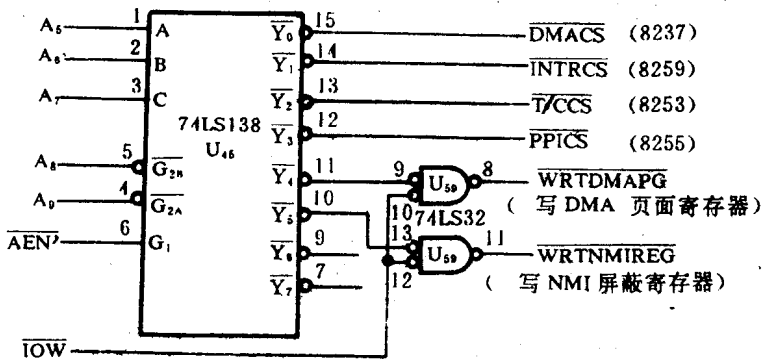


图 1.3 系统板口地址译码

二、开关式可选口地址译码

如果用户要求接口卡的口地址能适应不同的地址分配场合,或为系统以后扩充留有余地,则采用开关式口地址译码。这种译码方式使接口卡的 I/O 口地址可根据要求加以改变而无需改动线路,如图 1.4 所示。图中 DIP 开关状态的设置,就决定了译码电路的输出,若改变开关状态,就改变了 I/O 口地址。电路中使用了一片 8 位比较器 74LS688,当输入端 $P_0 \sim P_7$ 的地址与设置端 $Q_0 \sim Q_7$ 的开关状态一致时,输出为低。使用时可预置 DIP 开关为某一值,得到一组所要求的口地址。若考虑到读写分别控制,把 $\overline{IOR}$ 和 $\overline{IOW}$ 信号参加译码,使八个口地址作十六个