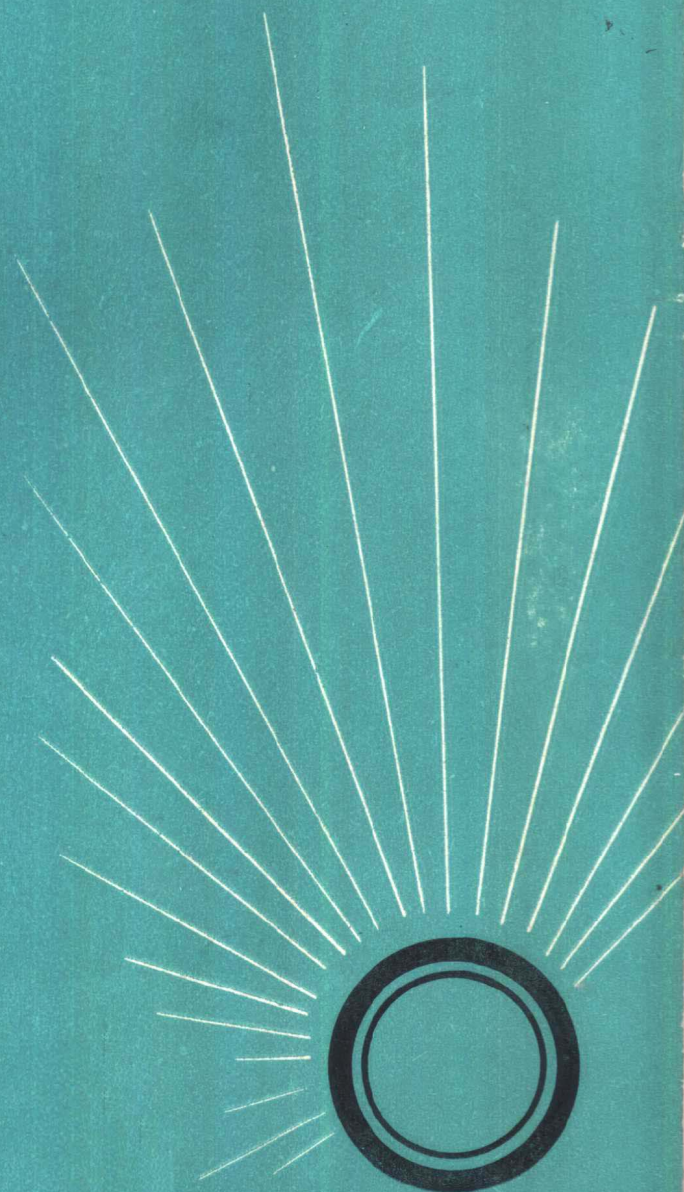


高等学校教材

微型计算机

黄遵熹 主编

接口与应用



西北工业大学出版社

高等学校教材

微型计算机接口与应用

黄遵熹 主编

西北工业大学出版社

1989年10月 西安

内 容 简 介

本书从微型计算机应用实际出发,对微机的接口、系统和应用作了较为完整和简明的介绍,以满足当前微机开发应用的实际需要。

全书分为三章。第一章介绍常用的接口芯片,包括存储器、定时器、并行接口芯片、串行接口芯片及DMA。第二章介绍微机系统与接口技术,包括A/D、D/A、显示、开关和键盘接口技术,以及微机系统设计和总线技术等。第三章介绍了微机应用的实例,并介绍了多微机系统和单片计算机。在附录中还编入了微机应用基础实验提纲。本书选材以工业应用的8位微机为主,并兼顾了16位微机的应用。

本书可作为已具备微机原理和数字逻辑等知识的非计算机专业大学本科、研究生的选修课教材,亦可供从事微机开发应用的工程技术人员参考。

高等学校教材

微型计算机接口与应用

主 编 黄遵熹

责任编辑 李 珏

责任校对 樊 力

*

西北工业大学出版社出版

(西安市友谊西路127号)

陕西省新华书店发行

空军导弹学院印刷厂印装

ISBN 7-5612-0177-X/TP·31 (课)

*

开本787×1092毫米 1/16 12.5印张 1插页 328千字

1989年10月第1版 1989年10月第1次印刷

印数1—7000册

定价: 2.76元

前 言

微型计算机的迅猛发展,吸引了成千上万的科技人员,竞先学习微型计算机技术。由于微型计算机在体积、成本和功能诸方面占有极大优势,使它的应用遍及科学技术、社会生产和经济生活各个领域。在我国,微机应用的类目已达上千种,并不断朝纵深发展,有效地推动着我国的现代化建设。目前,广大科技人员对微机学习的要求,不只是在原理上的普及,而是迫切需要得到关于微机接口、系统和应用方面的实际知识和引导。为了满足这种需要,本书试图从原理和实际应用两个方面介绍微机接口和应用技术,为微机的进一步普及作一些贡献。

本书谨向读者介绍三方面内容:第一章介绍Intel和ZiLog公司的通用接口芯片,包括存储器、定时器、并行接口、串行接口和DMA控制器;第二章介绍微机系统和接口技术,包括A/D、D/A、显示器、开关以及键盘的接口技术、微机系统组成和总线技术等;第三章给出微型计算机应用的三个方面例子,并介绍了多微机系统和单片计算机。所选内容以8位微机应用为主,在芯片和接口技术方面也兼顾了16位微机。本书还在附录中给出了应用基础实验提纲,以满足高校学生的学习要求。全书选材完整、简洁和实用,以使读者能较快地胜任微机开发和应用的实际工作。

本书可作为已具有微型计算机原理和数字逻辑知识的、非计算机专业大学本科和研究生的选修课教材,也可作为各类工程技术人员的参考书。本书的初稿曾在研究生和本科生中多次使用,并得到肯定。这次整理作了较大的修改。全书由黄遵熹主编,参加编写的还有黄静萍(§3-2、§3-5)、赵树仁(§3-1)、唐宏涛(§1-5)和陈翔(§1-4)等同志,其余部分由黄遵熹编写。全书插图由李容凡同志完成。西安电子科技大学高有行教授对书稿作了认真仔细的审查,并提出了宝贵的修改意见,在此一并表示感谢。

由于编者水平和实际经验所限,加之时间仓促,错误和不足在所难免,敬请读者评述和提出宝贵的意见。

编 者

1989年1月

目 录

第一章 基本接口电路	1
§ 1-1 半导体存储器系统	1
§ 1-2 并行接口芯片	10
§ 1-3 定时器/计数器	25
§ 1-4 串行接口芯片	35
§ 1-5 DMA 控制器	50
第二章 微型计算机系统与接口技术	70
§ 2-1 D/A转换器及其与 CPU 的接口	70
§ 2-2 A/D转换器及其与 CPU 的接口	83
§ 2-3 LED显示器	93
§ 2-4 开关、键盘的接口	99
§ 2-5 微型计算机系统	107
§ 2-6 微型计算机总线	116
第三章 微型计算机应用	130
§ 3-1 数据采集系统	130
§ 3-2 温度控制系统	144
§ 3-3 小规模集成电路功能测试	153
§ 3-4 多微机系统	159
§ 3-5 单片机及其应用	164
§ 3-6 微型计算机系统的抗干扰	184
附录	190
附录一 实验提纲	190
附录二 MCS-51指令系统	200
主要参考文献	211

第一章 基本接口电路

§ 1-1 半导体存储器系统

存储器是微型计算机三大组成部分之一，它完成二进制信息的存储功能。微型计算机内部都采用大规模集成电路制成的各种半导体存储器。存储器按功能可分为 ROM(只读存储器)和 RAM(随机存取存储器)两大类。ROM 用于存储固定的程序指令、数据等，它是微机系统必不可少的部分。RAM 用于存取随机信息。按电路形式不同又可分为静态存储器(SRAM)和动态存储器(DRAM)两类。有关这些内容已在微机原理课中介绍，这里不再重复。下面只对存储器与 CPU 的接口进行讨论，并以 TP801 单板机的存储器系统为例介绍存储器系统的配置。

一、存储器系统的设计问题

1. 存储器芯片的选择

您一定很清楚配置 ROM 和 RAM 的必要性。ROM 是必不可少的，再简单的微型计算机系统也一定要有 ROM，用以存放应用程序和必要的数。然而，RAM 却并不是非配置不可的，当你所用的系统在执行程序过程中无须接收新数据，或者只要取得极少的数据时，可以不要配置 RAM，而少数数据的接收可充分使用微处理器中的内部寄存器，这样可使系统硬件简化而更受欢迎。

存储器芯片型号的选择，大致应考虑三个因素：一是存取速度，二是容量和结构，三是价格和市场。最好选用与 CPU 时序相匹配的芯片，这样可以避免增加额外的电路。如果选用芯片的速度太慢，则应在系统设计中提供相应的 WAIT 信号的电路，势必增加麻烦。存储器芯片的容量有很多种，RAM 芯片有 256×8 、 $1K \times 4$ 、 $4K \times 1$ 、 $4K \times 8$ 、 $16K \times 1$ 等，EPROM 芯片有 $2K \times 8$ 、 $4K \times 8$ 、 $8K \times 8$ 、 $16K \times 8$ 等。选用不同结构的芯片，将直接关系到系统组成的形式。价格和市场来源问题也是我们不可忽视的选择原则，因为它将直接影响到产品或成果的推广。

RAM 芯片有静态 RAM 和动态 RAM 两类。静态 RAM 的选用将使电路结构简化，但大容量的存储系统将使芯片的数量大大增加。动态 RAM 的规模大，价格低，容易组成大容量的存储系统，但由于动态 RAM 要刷新，在电路上会增加一点麻烦。实际应用中，对小容量系统采用静态 RAM，而大容量系统总是用动态 RAM。

2. 高位地址译码

存储器的地址分配和高位地址译码是电路设计的主要问题。有两种译码方式可供选择：

一是线选技术。它是用高位地址线中一位作为各芯片组的片选信号，而用低位地址线直接与各芯片的地址线相连。例如图 1-1 中示出的由 $2K \times 8$ 芯片组成 $10kB$ 存储系统。显然，线选法的电路结构简单，不用增加译码电路，但是 CPU 的寻址能力得不到充分发挥，如此

图中的系统只能配置成10kB的容量，再大的容量就无法实现了。

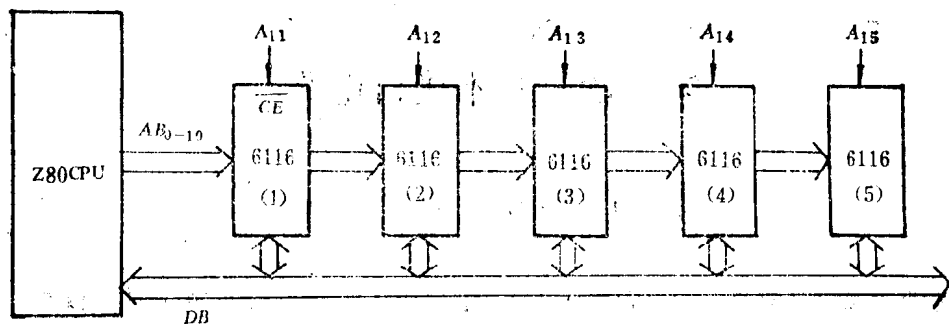


图 1-1 线选法片选

二是全译码方法。它是将低位地址总线直接连接各芯片的地址线，而余下的高位地址线全部译码，译码输出作为各芯片组的片选信号，以确定各组芯片的唯一地址范围。如图1-2是一个全译码系统例子。这种译码方式所构成的存贮系统容量为 2^N (N为地址线条数)。8位CPU的地址总线为16条，因而可组成的最大存贮系统容量为64KB。8088CPU的地址总线为20条，所以可组成1MB的存贮系统。

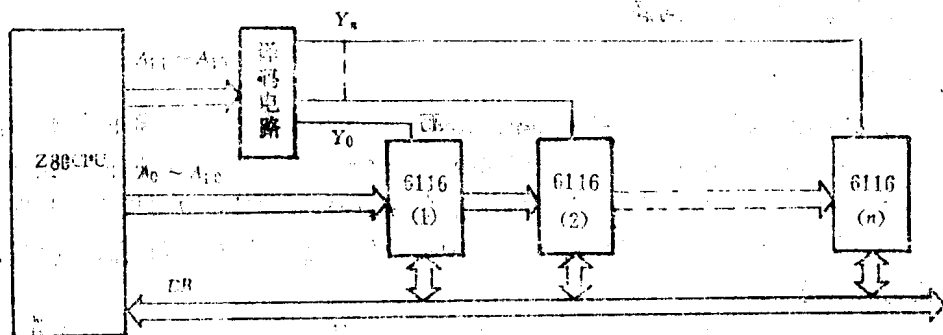


图 1-2 全译码片选

74LS138是常用的译码器芯片，它是三-八译码器，有一组包括三个引脚的选片逻辑，在不大的系统中用一片74LS138即可达到目的。如要组成较大的系统，可用几片74LS138组成两级译码电路来实现。

3. 动态存贮器的特殊问题

由于动态存贮器的存贮单元是利用FET管栅极电容的充放电状态来表示信息“0”和“1”的，但充电电荷要释放而将丢失信息，所以要“刷新”。刷新周期一般为2ms，即每隔2ms就要对所有存贮单元刷新一遍。对使用动态存贮器的系统必需预先考虑刷新电路，并与CPU配合来完成刷新功能。

动态存贮器的另一个问题是由于芯片集成度高而引脚数量不够而引起的。譬如一片16K的芯片，地址有14条，数据线8条，再加上其它控制信号，一片24~28引脚的封装就有困难，因而在制造芯片时总是将地址信号分为两组，公用几条地址线，并分两次打入芯片内部锁存。两组地址信号的打入是由行地址选通信号RAS和列地址信号CAS控制的。

图1-3为动态RAM控制逻辑框图。图中RAM芯片型号为2116，是16K×1位的。组成

16K×8的系统共用8片2116。地址引脚只有七条(MA₆~MA₀)，需将14位地址分两次打入：第一次在RAS选通下打入A₆~A₀，第二次在CAS选通下打入A₁₃~A₇。2116的选片信号由RAS代替。图中的选片信号逻辑是在M₁、MREQ、RFSH、Φ和2Φ控制下，产生RAS、CAS和SWITCH等时序控制信号。行/列多路器功能是在多路控制下，以规定时间先后与RAS、CAS配合，将行地址和列地址分时送到2116的MA₆~MA₀。刷新多路器的功能是在RFSH控制下，进行行地址和刷新地址的切换。RFSH=1时送行地址，RFSH=0时送刷新地址。由于RAS和CAS的时序安排与具体芯片型号有关。所以，整个控制逻辑应该在选定芯片的前提下才能设计。

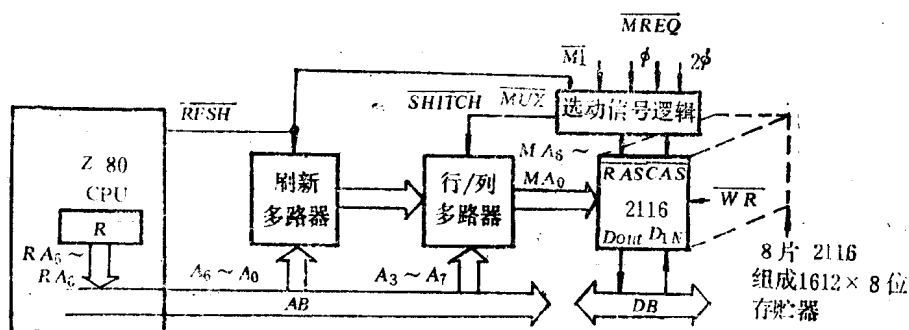


图 1-3 动态RAM控制逻辑

4. 负载驱动问题

一个存储器系统将由很多芯片组成，CPU的总线要与所有芯片连接。也就是说，总线将要带很重的负载。总线的负载能力一般只有一个TTL门，所以，能否带动这些负载将是一个问题。如果你所设计的整个微机系统，总线要带动的MOS门超过十几个，就应该考虑总线驱动问题。对单向传输的总线可选用象LS367、LS244等总线驱动器，而对双向传输的数据总线，可选用LS245这类总线收发器来驱动。还有一些专用总线驱动器当然也可以使用。图1-4是一个接有驱动器的系统框图。

二、存储器芯片及其与CPU的接口

1. ROM

ROM的种类包括掩模ROM、PROM和EPROM几类。在研制和批量产品中我们常用的是EPROM。典型的芯片有2716、2732、2764、27128等。这些芯片的透明窗口在紫外光照射下，可将全部信息擦除，这时各存储单元都为“1”。然后可在相应的EPROM写入器中将程序固化。

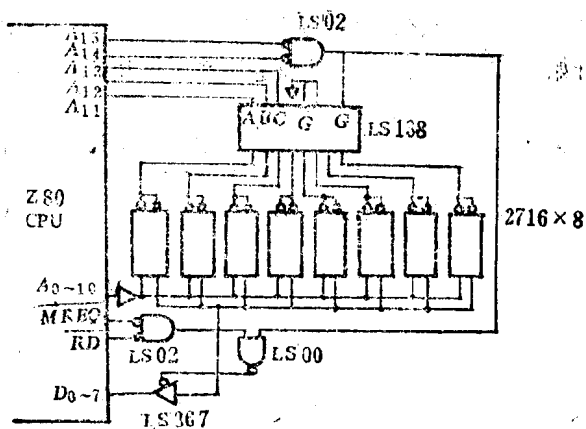


图 1-4 有总线驱动器的系统

图1-5是2716芯片的引脚排列与有关时序。2716是 $2k \times 8$ 位的EPROM，外形为双列直插式24脚封装。各引脚的功能如下：

$A_{11} \sim A_0$ 地址线。接相应的地址总线。

$O_7 \sim O_0$ 数据线。接数据总线。

$\overline{CE}(\overline{PD}/\overline{PGM})$ 片选线，编程时接编程脉冲。

V_{cc} +5V电源。

V_{pp} 编程电源，读操作时接+5V。

GND 地线。

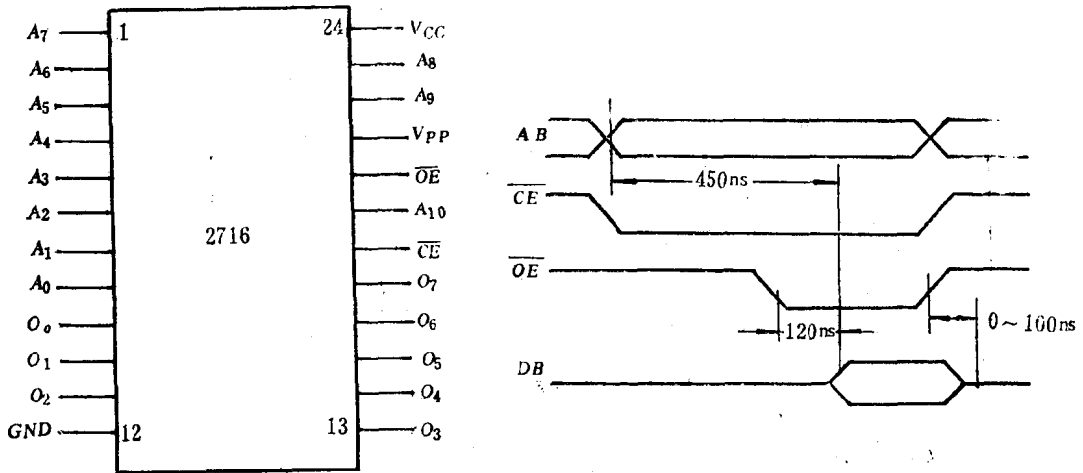


图 1-5 2716EPROM引脚及时序图

引脚 $\overline{CE}$ 、 $\overline{OE}$ 、 V_{pp} 、 V_{cc} 的不同信号组合，决定EPROM的五种操作方式，表1-1列出它们的关系。

关于编程电源 V_{pp} 的要求，一般写成+25V。实际上，不同产品的要求是有区别的，多数产品要求在20V~25V之间，也有的产品在20V以下，如果加高了电源电压，很容易将芯片烧坏，应严格按产品手册给出的数值加 V_{pp} 。或者，可用一个可变电源，从较低的电压开始，逐步加高 V_{pp} ，以确定该产品应加的 V_{pp} 值。

表 1-1 2716EPROM的操作方式

$\overline{CE}$	$\overline{OE}$	V_{pp}	V_{cc}	操 作 方 式
0	0	+5V	+5V	读
0	x	+5V	+5V	维持
1	1	+25V	+5V	编程
0	0	+25V	+5V	编程校验
0	1	+25V	+5V	编程禁止

2716与CPU的连接：地址线、数据线以及电源线直接连接，编程电源通常由开关控制。图1-6为基本连接方法框图。管脚 $\overline{CE}$ 和 $\overline{OE}$ 的连接可用三种方法：图1-6(a)是基本方法，它是用高位地址译码输出产生 $\overline{CE}$ 信号，而 $\overline{OE}$ 由 $\overline{MREQ}$ 信号控制，因此，数据较快地送到地址线上，满足CPU时钟频率较高的场合下工作。图1-6(b)是一种延迟访问时间以减少耗电的方法，适用于时钟频率较低的情况。图1-6(c)是一种简化的接法，地址译码器可得到有效地使用。

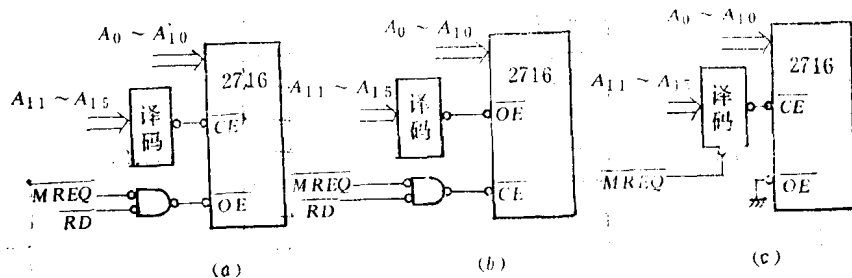


图 1-6 2716 EPROM的连接方法

2732是4K×8位的EPROM芯片，最大读出时间为450ns，封装和引脚与2716相同，其信号线仅仅把2716的 $\overline{OE}$ 和 V_{PP} 合为一条线，而把21脚改为 A_{11} 地址线。外部引脚示于图-7，表1-2为它的操作方式。

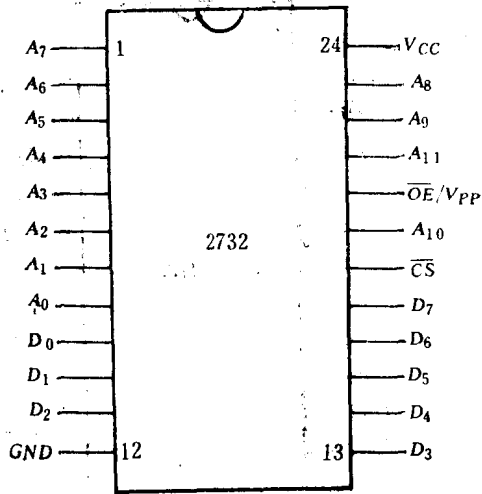


图 1-7 2732引脚图

表 1-2 2732操作方式

$\overline{CE}$	$\overline{OE}/V_{PP}$	V_{cc}	操作方式
0	0	+5V	读
1	x	+5V	维持
0	+25V	+5V	编程
0	0	+5V	编程校验
1	+25V	+5V	编程禁止

2764是8K×8位的EPROM芯片，双列直插式28脚封装，是一种高速大容量的芯片，读出时间为250ns，各引脚信号的定义与2716相同(只增加了地址线)。27128是16K×8位的EPROM芯片，管脚与2764兼容。图1-8是它们的引脚排列图。

EPROM的编程必须在专门的写入设备上进行。近几年来发展了一种EEPROM—电擦除可编程只读存储器，它能在微型计算机系统中进行“在线”修改，并能在断电情况下保持修改的结果，因而得到重视。这里简单介绍一下Intel公司的产品2817A EEPROM

芯片。

Intel 2817A是2K×8位EEPROM。+5V电源，最大工作电流为150mA，维持电流为55mA，读出时间为250ns。双列直插式封装，28引脚，HMOS-E工艺制造，片内有编程高压脉冲产生电路，因而不需外加编程电源。

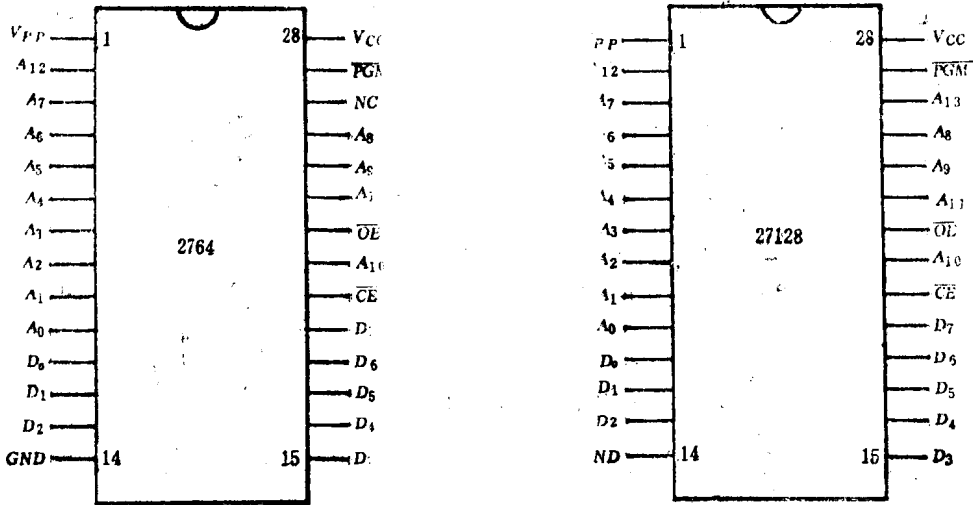


图 1-8 2764, 27128 EPROM 引脚图

图1-9是M2817A的引脚排列图。

引脚信号定义：

$A_0 \sim A_{10}$ 地址线

$I/O_0 \sim I/O_7$ 数据线

$\overline{CE}$ 片选线

$\overline{OE}$ 输出允许

$\overline{WE}$ 写允许

$RDY/BUSY$ 器件闲/忙状态线

2817A的工作方式列于表1-3。写入操作过程如下：当向2817A发出字节写入命令后，2817A便锁存地址、数据及控制信号，从而启动一次写操作。写入时间约16ms，在此期间 $RDY/BUSY=0$ ，表示2817A正在进行写操作，这时，数据总线处于高阻态，因而允许微处理器在写操作期间可执行其它任务。当一次字节写入操作完毕，2817A将 $RDY/BUSY=1$ ，以通知微处理器，这时可以对2817A进行新的字节读/写操作。

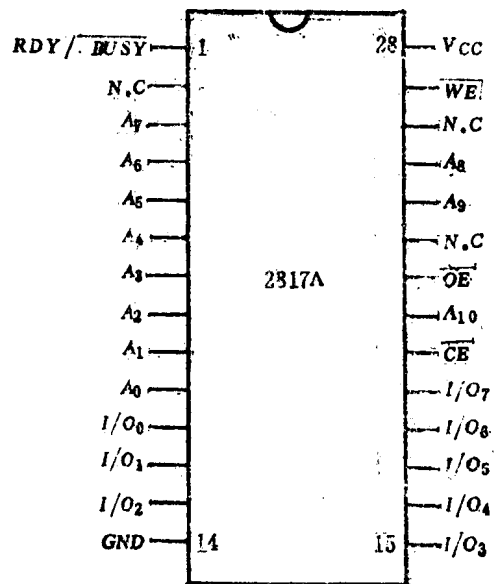


图 1-9 M2817A引脚排列图

2. 静态RAM

RAM与CPU的接口问题，除了增加一条读/写控制信号之外，其它没有什么多大区

表 1-3

2817A 操作方式

操作	CE	OE	WE	RDY/BUSY	I/O
读	0	0	1	高阻	D_{out}
维持	1	x	x	高阻	高阻
写入	0	1	0	0	D_{in}
擦除	字节写入前自动擦除				

别。其连接方法是：地址线接低位地址总线，地址总线的高位经译码后驱动选片端以确定地址范围，相应的控制总线控制读/写线。

常用的静态RAM芯片除以前介绍过的2114(1K×4位RAM)外，有6116、6264等。这里简略介绍一下这两种芯片。

6116为2K×8位的静态存储器，双列直插式24引脚封装，它与2716EPROM的引脚兼容，这在使用上是很利。图1-10列出了它的外部引脚排列图。RAM的读写操作由信号 $\overline{CE}$ 、 $\overline{WE}$ 和 $\overline{OE}$ 的组合来决定，如表1-4所示。

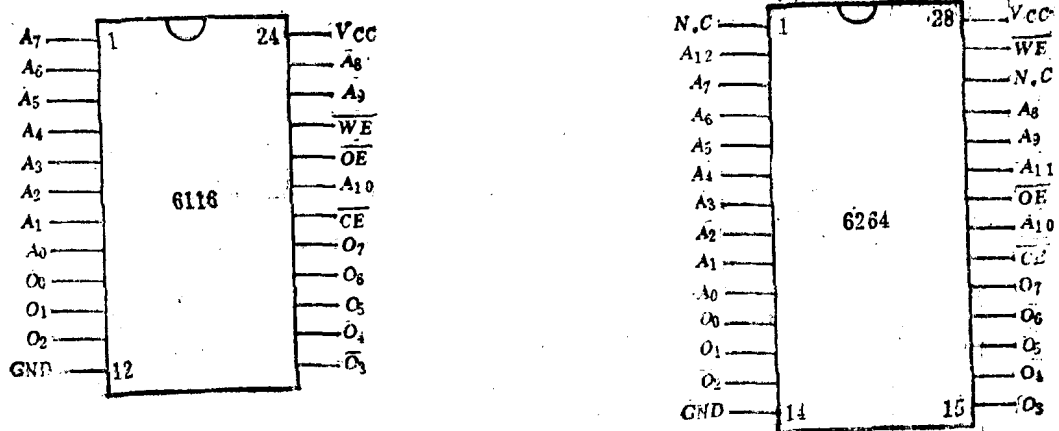


图 1-10 6116、6264RAM 引脚图

表 1-4

6116 的操作方式

CE	WE	OE	操作方式
0	0	1	写
0	1	0	读
1	x	x	非选

6264为8K×8位静态RAM芯片，双列直插式28脚封装，引脚与2764EPROM兼容，信号情况与6116类似。外部引脚图已画在图1-10中。表1-5为6264的操作方式。

表 1-5

6264 的操作方式

WE	CE ₁	CE ₂	OE	操 作
×	1	×	×	未选中(高阻)
×	×	0	×	未选中(高阻)
1	0	1	1	输出禁止(高阻)
1	0	1	0	读
0	0	1	1	写
0	0	1	0	写

三、TP-801单板机的存贮器结构

TP801的存贮器包括6KB只读存贮器、4KB随机存贮器并可以扩充6KB的系统。6KB的只读存贮器分为ROM(监控程序)、PROM1(用户程序)和PROM2(写EPROM)三个2KB的组件插座,所用的芯片为2716EPROM。RAM是由8片2114静态存贮器芯片组成的,每两片2114组成1KB的存贮区。图1-11为这个系统的框图。

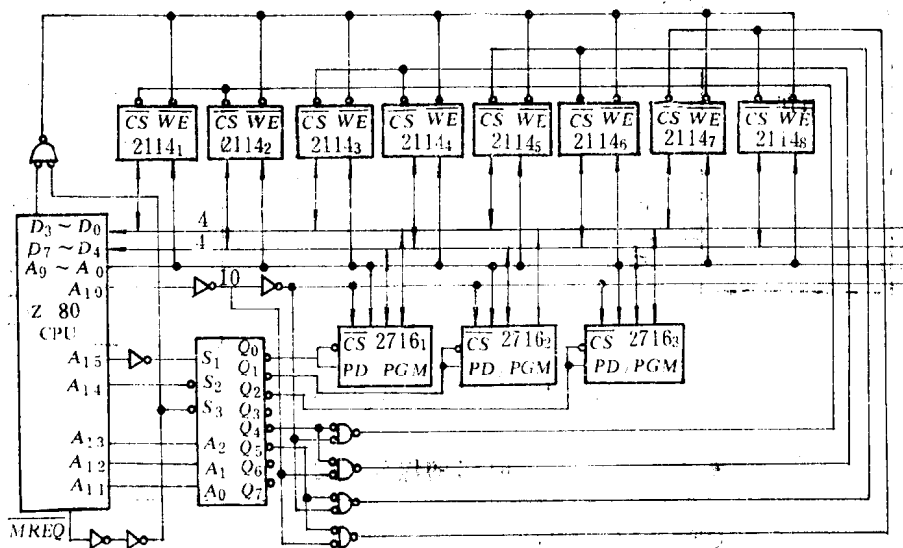


图 1-11 TP801存贮系统

图1-11中,高位地址译码采用全译码方式,以使用户可将存贮容量扩充到64KB的最大范围。译码器选用一片74LS138组件,地址总线的A₁₅、A₁₄接选片控制,由A₁₃、A₁₂、A₁₁译出八段存贮区,每段为2K的存贮空间。由图的连接关系可确定各ROM和RAM的地址区域如表1-6所示。

表 1-6

TP801存储器地址划分

	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉		A ₁	A ₀	地址区域(Hex)
ROM	0	0	0	0	0	×	×		×	×	0000~07FF
PROM1	0	0	0	0	1	×	×		×	×	0800~0FFF
PROM2	0	0	0	1	0	×	×		×	×	1000~17FF
RAM1	0	0	1	0	0	0	×		×	×	2000~23FF
RAM2	0	0	1	0	0	1	×		×	×	2400~27FF
RAM3	0	0	1	0	1	0	×		×	×	2800~2BFF
RAM4	0	0	1	0	1	1	×		×	×	2c00~2FFF

请注意RAM的译码，它是将地址总线A₁₀参与 $\bar{Y}_4$ 、 $\bar{Y}_5$ 的再次译码，因为2114是1K的芯片。译码芯片74LS138还有三条译码输出线未用，这给用户供6KB存储器的直接扩充。如果RAM芯片不用2114，而用规模较大的6116芯片，那末这种系统的芯片数量可减少，并可省去一些其他电路。这个问题留给读者自己去考虑，你可以设计出一个如意的更好的存储器系统。

四、存储器扩展

我们知道，8位的微处理器的地址总线是16条，因而它们的直接寻址能力是64K。如果某个应用系统的存储容量要求大于64K，那末并不是轻易能做到的，这就是所谓扩展问题。这里就扩展的基本思想和实现的基本方法作一简略介绍。

其基本思想是：以64K容量为一个存储体，该存储体是一个完整的子系统，具有上面所述的功能和逻辑结构。然后再以若干个这样的子系统组成一个应用系统所需的存储器。为了能正确地控制读写操作，在这个系统中要有体选控制逻辑。在对某个存储单元进行读写操作时，必须经过两次地址译码。首先送出一个体选控制信号，选中该存储单元所在的存储体，将这个存储体挂在CPU的总线上，然后二次译码去选中该存储单元，进行读写操作。

图1-12为一个512k存储器的扩展框图。每一个存储体的容量是64K，一共包括8个子系统M₀、M₁……M₇。系统由分段控制逻辑将某一个子系统挂到总线上（接通这个子系统的片选端），实现子系统的选择。分段控制逻辑的控制来自一个外设口的并行输出口，因而它是由一条输出指令来实现

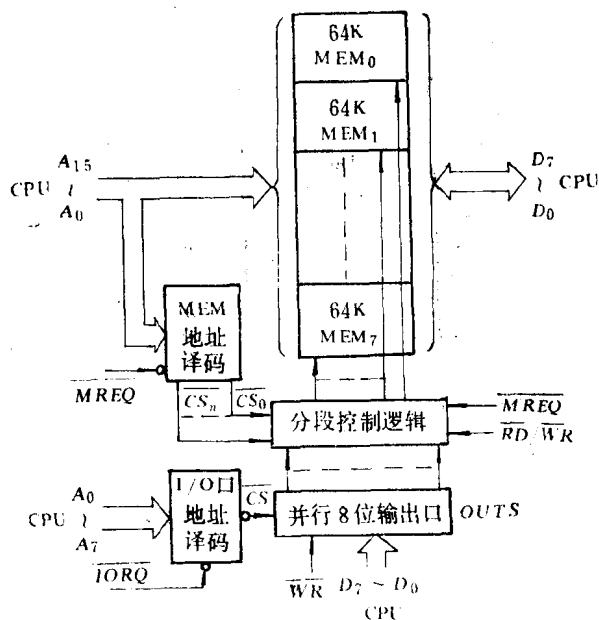


图 1-12 存储器扩展原理图

的。一旦被选中的子系统挂在总线上之后,就可以进行对这个子系统的读写操作。

在使用扩展的存贮系统时,需要由程序先执行体选指令,以达到切换存贮体的目的。所以在切换中应保证在程序上的连贯性,否则将出错。程序设计必须考虑它的特点和原理。

§ 1-2 并行接口芯片

在这一节中将介绍Zilog公司的Z80PIO和Intel公司的8255两种并行接口芯片。前者广用于Z-80系统,后者广泛用于80/85以及8086/88系统。因而它们不仅是8位微机的通用接口,也常用在16位微机中。

一、Z80 PIO

1. Z80 PIO的功能

Z80 PIO是一种通用的8位并行接口芯片,可直接与Z80 CPU或其它CPU相接,在CPU指令的控制下完成多种输入输出功能。基本功能是:

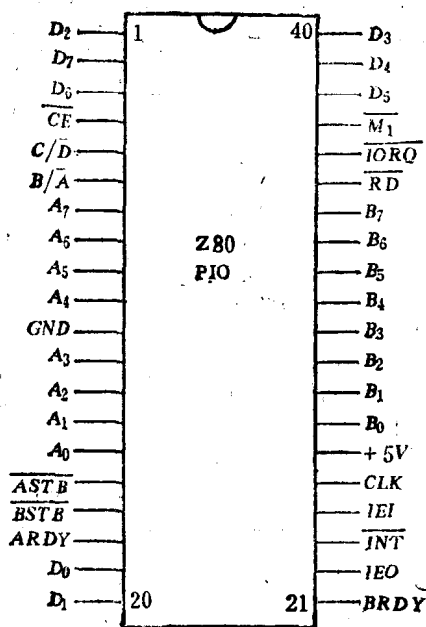
(1) 一片Z80 PIO包括两个相互独立的8位并行口:PIOA与PIOB。

(2) 四种工作方式由程序设定。这四种方式是输出方式M0,输入方式M1,双向方式M2和位控方式M3。

(3) PIO通过联络信号 $\overline{STB}$ 和RDY与外设通信。CPU与外设的数据传送可以用中断方式(规定为IM2方式)来实现,并在片内有链形中断优先权逻辑功能。

2. Z80 PIO引脚及与CPU的接口

Z80 PIO外形为40脚双列直插式封装(如图1-13)。40条引脚中包括8条数据线 $D_7 \sim D_0$,



它们直接与CPU的数据总线相接;控制信号 M_1 、 $\overline{IORQ}$ 、RD与CPU的系统控制线相接;选择信号 $\overline{CE}$ 、C/D、B/A与CPU的低8位地址总线相接,以确定PIO两个口中的各个寄存器地址。一般的连接方法及被选择的寄存器情况出于表1-7。

表 1-7 PIO 的选择信号

CE	C/D	A/B	选中的寄存器
0	0	0	PIOA数据寄存器
0	0	1	PIOB数据寄存器
0	1	0	PIOA控制寄存器
0	1	1	PIOB控制寄存器
1	x	x	未选中

图 1-13 PIO引脚图

三条中断控制线是: $\overline{INT}$ 为中断请求信号,它以IM2方式向CPU提出中断申请;IEI

和IEO是中断优先权链形电路的输入引脚和输出引脚，用以安排本片的中断优先级。在片内A口优先于B口。

两个口都有8条数据线 $PA_7 \sim PA_0$ 和 $PB_7 \sim PB_0$ ，以及两条联络线 $\overline{ASTB}$ 、 $ARDY$ 、 $BSTB$ 和 $BRDY$ 。这两条联络线的功能规定如下：

$\overline{STB}$ 选通信号，低有效。来自外设的选通脉冲，对不同方式的含义分别是： M_0 ——外设收到PIO提供的数据； M_1 ——外设来的数据已装入PIO的数据寄存器； M_2 —— $\overline{ASTB}$ 表示A口输出寄存器的数据进入数据总线， $\overline{BSTB}$ 表示将外设来的数据选通进入A口的数据寄存器； M_3 ——这时 $\overline{STB}$ 信号被封锁，不起作用。

RDY 给外设的准备就绪信号，寄存器待命，高有效。对不同方式的含义分别是： M_0 ——输出寄存器已有数据，通知外设可以取走数据； M_1 ——输入寄存器已空，准备接收外设的数据； M_2 —— $ARDY$ 表示A口输出寄存器有数据， $BRDY$ 表示A口输入寄存器已空； M_3 ——这时本信号被封锁，强制为低电平。

在应答方式传送数据时，可通过选通线 $\overline{STB}$ 申请中断。在无条件传送数据时， $\overline{STB}$ 可以挪用，即可由另外一个无关的外设使用(申请中断)，但此时不能用 RDY 信号。

Z80 PIO的引脚中没有复位端，但因其内部有一个接通电源即可清零的电路，一接通电源就复位。复位时的状态是：

- ① A、B两口的屏蔽寄存器复位，中断禁止；
- ② I/O线处高阻态， $RDY=1$ ；
- ③ 工作方式自动设定为 M_1 ；
- ④ 中断允许=0；
- ⑤ 数据输出寄存器复位。

此外，PIO还可以根据 $\overline{RD}$ 及 $\overline{IORQ}$ 处于高电平， M_1 处于低电平而清零，如图1-14的连接。这个复位电路中，由 $\overline{RESET}$ 或 M_1 组合加给PIO的 M_1 引脚，在无 $\overline{RESET}$ 信号时，起 M_1 的作用，而在 $\overline{RESET}$ 有效时，满足 $\overline{RD}=1$ 、 $\overline{IORQ}=1$ 的条件，故 $M_1=0$ 时起清零作用。

PIO一经复位，一直保持到CPU接受新的控制字为止。

3. PIO编程

Z80 PIO内部有两组寄存器：数据寄存器和控制寄存器。图1-15画出了所有寄存器。

数据寄存器包括输入寄存器和输出寄存器，分别承担输入数据的缓冲和输出数据的锁存。控制寄存器包括方式寄存器、I/O选择寄存器、中断控制寄存器、中断屏蔽寄存器和中断

矢量寄存器。这些寄存器的内容都应在初始化编程中设定。这些控制字的格式和功能分别叙述如下。

(1) 工作方式字

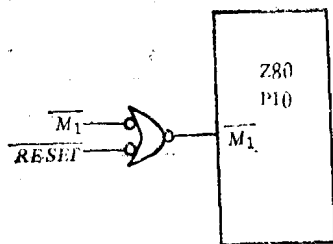


图 1-14 PIO 的复位

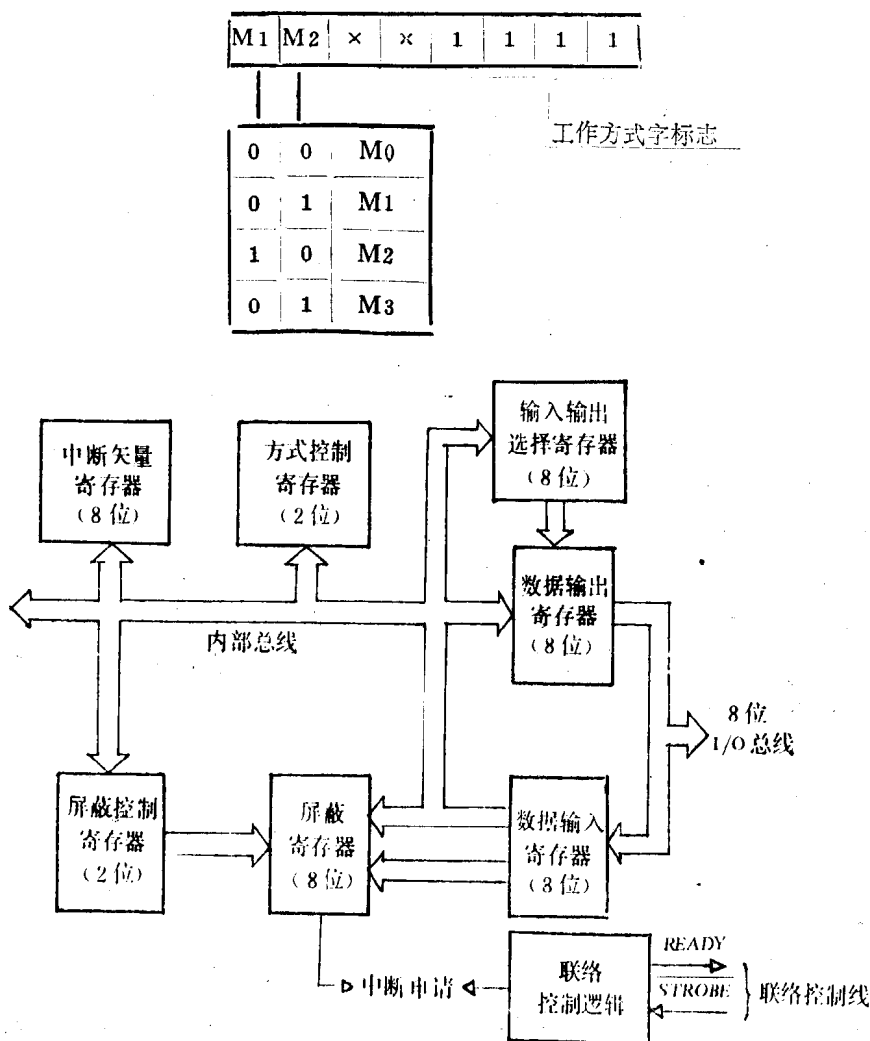


图 1-15 PIO 内部寄存器

(2) 方式3(M3)的输入输出选择字 如果在工作方式字中设定为M3方式,则紧跟着设置I/O选择字,以确定每一位的输入输出功能,并规定“1”表示输入,“0”表示输出。

I/O7	I/O6	I/O5	I/O4	I/O3	I/O2	I/O1	I/O0
------	------	------	------	------	------	------	------

(3) 中断控制字

