

3543.26

计算机硬件基础与维修系列丛书

微型计算机实用接口技术

刘洪梅 薛永毅 伊 兵 编著
何希才 审校

机 械 工 业 出 版 社

本书介绍了微型计算机接口技术。内容包括通用 8 位、16 位微型机和 MCS-51 单片机的外特性；中断控制和 DMA 控制；A/D 和 D/A 转换器应用技术；数据传送技术；输入输出电路；接口电路；通信电路以及与 CPU 连接方式等。本书从实用角度出发，提供了较多性能优良的实用电路，资料齐全，内容丰富。

本书可供从事微机应用的科研、生产、教学和开发的科技人员使用，也可作为大专院校师生参考书。

图书在版编目 (CIP) 数据

微型计算机实用接口技术/刘洪梅等编著. - 北京: 机械工业出版社, 1998

(计算机硬件基础与维修系列丛书)

ISBN 7-111-06505-0

I. 微… II. 刘… III. 微型计算机-接口 IV. TP364

中国版本图书馆 CIP 数据核字 (98) 第 22133 号

出 版 人: 马九荣 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 蒋 克 王冠宇

北京市南方印刷厂印刷·新华书店北京发行所发行

1998 年 11 月第 1 版第 1 次印刷

787mm×1092mm¹/₁₆·15.75 印张

定价: 24.00 元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前 言

微型计算机技术发展很快，如何在最短时间内尽快掌握微型计算机（简称“微机”）应用技术，这是广大从事微机应用技术人员急待解决的问题，为此，编者根据参加“电子信息实训基地”的建设、“视频图像通信”等科研项目的体会以及多年的教学经验编写这本书，此书重点介绍微机应用中的接口技术，这会对从事微机应用的读者提供有益的帮助。

本书编写过程中，得到北京信息学院信息系主任鲍泓副教授的支持与帮助，在此表示感谢。由于编者水平有限，书中会有错误与不当之处，热切希望广大读者批评指正。

编者

1998年6月于北京

目 录

前言

第 1 章 通用微处理机及其辅助电路	1	第 3 章 中断控制	52
1.1 概述	1	3.1 中断原理	52
1.2 通用微处理机简介	2	3.2 中断优先权与中断判优	53
1.2.1 Z-80 CPU 的特征与控制信号	2	3.3 Z-80 的中断控制	54
1.2.2 8086 CPU 的特征与控制信号	5	3.4 8086 的中断控制	56
1.2.3 MCS-51 单片机的特征与控制 信号	10	3.5 MCS-51 的中断控制	62
1.3 时钟电路	15	第 4 章 DMA 控制	66
1.3.1 概述	15	4.1 DMA 控制原理	66
1.3.2 时钟振荡器	15	4.2 Z-80 DMA 控制器	68
1.3.3 与 CPU 的连接	17	4.3 8237A DMA 控制器	76
1.4 系统复位电路	17	4.3.1 8237A 功能与特征	76
1.5 地址译码器	20	4.3.2 8237A DMA 控制方法	78
1.6 存储器	22	第 5 章 微机应用系统的输入/输出 电路	82
1.6.1 概述	22	5.1 数字信号输入电路	82
1.6.2 静态 RAM (SRAM)	22	5.2 数字信号输出电路	85
1.6.3 动态 RAM (DRAM)	24	5.3 模拟信号输入电路	88
1.6.4 只读存储器 (ROM)	25	5.4 模拟信号输出电路	90
1.6.5 8086 与存储器的连接	28	5.5 异常诊断功能	91
1.6.6 MCS-51 单片机与存储器的 连接	29	第 6 章 并行接口电路	95
1.6.7 E ² PROM 及其应用	30	6.1 Z-80 PIO 并行接口	95
1.6.8 双端口 RAM 及其应用	34	6.1.1 概述	95
1.7 总线驱动器	37	6.1.2 Z-80 PIO 内部结构	96
第 2 章 微机的指令系统	39	6.1.3 PIO 工作方式	97
2.1 概述	39	6.1.4 方式 0 和方式 1 应用实例	100
2.2 寻址方式	39	6.1.5 方式 3 应用实例	101
2.3 数据传送指令	42	6.2 8255A 并行接口	102
2.4 算术运算指令	43	6.2.1 概述	102
2.5 逻辑类指令	45	6.2.2 8255A 与 CPU 的连接	103
2.6 数据串操作指令	46	6.2.3 8255A 的工作方式	104
2.7 转移指令	47	6.2.4 数据传送过程中各种方式的 应用	110
2.8 移位循环指令	49	6.2.5 8255A 工作方式的典型应用 实例	113
2.9 输入输出指令	50	第 7 章 串行接口电路	115
2.10 标志操作指令	50	7.1 8251A 可编程通信接口电路	115
2.11 其他指令	51	7.1.1 8251A 的功能	115

7.1.2 8251A 内部结构和引脚功能	115	10.5 采样保持电路	183
7.1.3 8251A 的使用	117	第 11 章 数据传送技术	186
7.2 Z-80 SIO 串行通信口	120	11.1 概述	186
7.2.1 Z-80 SIO 结构特点	120	11.2 数据传送的硬件接口	187
7.2.2 Z-80 SIO 引脚功能	121	11.2.1 并行与串行传送方式	187
7.2.3 Z-80 SIO 写入控制信号/读出 状态信号	123	11.2.2 外部设备数目与数据传送 方向	191
7.3 MCS-51 单片机的串行口	128	11.2.3 同步与异步传送方式	193
7.3.1 串行口的控制	128	11.3 数据传送的软件接口	196
7.3.2 串行口的工作方式	129	11.3.1 存储器与外围 LSI 间的数据 传送	196
7.3.3 应用实例	130	11.3.2 存储器间数据传送	199
第 8 章 定时器/计数器	133	11.3.3 CPU 间数据传送	200
8.1 概述	133	11.4 数据高速传送技术	204
8.2 Z-80 CTC	134	11.4.1 通用 RAM 当作双端口 RAM 使用	204
8.2.1 Z-80 CTC 的特性与结构	134	11.4.2 专用双端口 RAM 的使用方法	204
8.2.2 CTC 引脚功能	134	11.4.3 FIFO 的使用方法	207
8.2.3 CTC 的工作方式	135	11.5 RS-232 C 接口总线	208
8.2.4 CTC 应用实例	135	11.6 IEEE-488 接口总线	210
8.3 8253/8254 定时器/计数器	137	第 12 章 传感器接口电路	214
8.3.1 概述	137	12.1 常用传感器特性与参数	214
8.3.2 8253/8254 的编程	138	12.1.1 热敏传感器	214
8.3.3 8253/8254 的工作方式	140	12.1.2 湿敏传感器	216
8.4 MCS-51 的定时器/计数器	149	12.1.3 力敏传感器	218
8.4.1 结构	149	12.1.4 光敏传感器	219
8.4.2 工作方式	150	12.1.5 磁敏传感器	220
8.4.3 应用实例	152	12.2 传感器的信号处理电路	221
第 9 章 外设接口电路	154	12.2.1 传感器信号放大电路	221
9.1 键盘接口	154	12.2.2 信号变换电路	223
9.1.1 非编码键盘	154	12.2.3 滤波器	224
9.1.2 编码键盘	156	12.3 传感器的接口电路实例	226
9.2 LED 显示器接口	157	12.3.1 温度接口电路	226
9.2.1 LED 显示器原理	157	12.3.2 湿度接口电路	227
9.2.2 LED 显示器的基本用法	158	12.3.3 电压接口电路	227
9.3 CRT 显示器接口电路	159	12.3.4 光电接口电路	228
9.4 直流电动机的微机控制	160	12.3.5 磁力接口电路	228
9.5 步进电动机的接口	164	第 13 章 微机应用系统设计	229
第 10 章 A/D 和 D/A 转换器应用 技术	167	13.1 微机应用系统设计原则	229
10.1 概述	167	13.2 微机应用系统实例——微机温室 环境监控管理系统	229
10.2 A/D 转换器应用技术	168	13.3 微机应用系统的抗干扰设计	231
10.3 D/A 转换器应用技术	178		
10.4 模拟多路开关 (MUX)	181		

前 言

微型计算机技术发展很快，如何在最短时间内尽快掌握微型计算机（简称“微机”）应用技术，这是广大从事微机应用技术人员急待解决的问题，为此，编者根据参加“电子信息实训基地”的建设、“视频图像通信”等科研项目的体会以及多年的教学经验编写这本书，此书重点介绍微机应用中的接口技术，这会对从事微机应用的读者提供有益的帮助。

本书编写过程中，得到北京信息学院信息系主任鲍泓副教授的支持与帮助，在此表示感谢。由于编者水平有限，书中会有错误与不当之处，热切希望广大读者批评指正。

编者

1998年6月于北京

目 录

前言

第 1 章 通用微处理机及其辅助电路	1	第 3 章 中断控制	52
1.1 概述	1	3.1 中断原理	52
1.2 通用微处理机简介	2	3.2 中断优先权与中断判优	53
1.2.1 Z-80 CPU 的特征与控制信号	2	3.3 Z-80 的中断控制	54
1.2.2 8086 CPU 的特征与控制信号	5	3.4 8086 的中断控制	56
1.2.3 MCS-51 单片机的特征与控制 信号	10	3.5 MCS-51 的中断控制	62
1.3 时钟电路	15	第 4 章 DMA 控制	66
1.3.1 概述	15	4.1 DMA 控制原理	66
1.3.2 时钟振荡器	15	4.2 Z-80 DMA 控制器	68
1.3.3 与 CPU 的连接	17	4.3 8237A DMA 控制器	76
1.4 系统复位电路	17	4.3.1 8237A 功能与特征	76
1.5 地址译码器	20	4.3.2 8237A DMA 控制方法	78
1.6 存储器	22	第 5 章 微机应用系统的输入/输出 电路	82
1.6.1 概述	22	5.1 数字信号输入电路	82
1.6.2 静态 RAM (SRAM)	22	5.2 数字信号输出电路	85
1.6.3 动态 RAM (DRAM)	24	5.3 模拟信号输入电路	88
1.6.4 只读存储器 (ROM)	25	5.4 模拟信号输出电路	90
1.6.5 8086 与存储器的连接	28	5.5 异常诊断功能	91
1.6.6 MCS-51 单片机与存储器的 连接	29	第 6 章 并行接口电路	95
1.6.7 E ² PROM 及其应用	30	6.1 Z-80 PIO 并行接口	95
1.6.8 双端口 RAM 及其应用	34	6.1.1 概述	95
1.7 总线驱动器	37	6.1.2 Z-80 PIO 内部结构	96
第 2 章 微机的指令系统	39	6.1.3 PIO 工作方式	97
2.1 概述	39	6.1.4 方式 0 和方式 1 应用实例	100
2.2 寻址方式	39	6.1.5 方式 3 应用实例	101
2.3 数据传送指令	42	6.2 8255A 并行接口	102
2.4 算术运算指令	43	6.2.1 概述	102
2.5 逻辑类指令	45	6.2.2 8255A 与 CPU 的连接	103
2.6 数据串操作指令	46	6.2.3 8255A 的工作方式	104
2.7 转移指令	47	6.2.4 数据传送过程中各种方式的 应用	110
2.8 移位循环指令	49	6.2.5 8255A 工作方式的典型应用 实例	113
2.9 输入输出指令	50	第 7 章 串行接口电路	115
2.10 标志操作指令	50	7.1 8251A 可编程通信接口电路	115
2.11 其他指令	51	7.1.1 8251A 的功能	115

7.1.2 8251A 内部结构和引脚功能	115	10.5 采样保持电路	183
7.1.3 8251A 的使用	117	第 11 章 数据传送技术	186
7.2 Z-80 SIO 串行通信口	120	11.1 概述	186
7.2.1 Z-80 SIO 结构特点	120	11.2 数据传送的硬件接口	187
7.2.2 Z-80 SIO 引脚功能	121	11.2.1 并行与串行传送方式	187
7.2.3 Z-80 SIO 写入控制信号/读出 状态信号	123	11.2.2 外部设备数目与数据传送 方向	191
7.3 MCS-51 单片机的串行口	128	11.2.3 同步与异步传送方式	193
7.3.1 串行口的控制	128	11.3 数据传送的软件接口	196
7.3.2 串行口的工作方式	129	11.3.1 存储器与外围 LSI 间的数据 传送	196
7.3.3 应用实例	130	11.3.2 存储器间数据传送	199
第 8 章 定时器/计数器	133	11.3.3 CPU 间数据传送	200
8.1 概述	133	11.4 数据高速传送技术	204
8.2 Z-80 CTC	134	11.4.1 通用 RAM 当作双端口 RAM 使用	204
8.2.1 Z-80 CTC 的特性与结构	134	11.4.2 专用双端口 RAM 的使用方法	204
8.2.2 CTC 引脚功能	134	11.4.3 FIFO 的使用方法	207
8.2.3 CTC 的工作方式	135	11.5 RS-232 C 接口总线	208
8.2.4 CTC 应用实例	135	11.6 IEEE-488 接口总线	210
8.3 8253/8254 定时器/计数器	137	第 12 章 传感器接口电路	214
8.3.1 概述	137	12.1 常用传感器特性与参数	214
8.3.2 8253/8254 的编程	138	12.1.1 热敏传感器	214
8.3.3 8253/8254 的工作方式	140	12.1.2 湿敏传感器	216
8.4 MCS-51 的定时器/计数器	149	12.1.3 力敏传感器	218
8.4.1 结构	149	12.1.4 光敏传感器	219
8.4.2 工作方式	150	12.1.5 磁敏传感器	220
8.4.3 应用实例	152	12.2 传感器的信号处理电路	221
第 9 章 外设接口电路	154	12.2.1 传感器信号放大电路	221
9.1 键盘接口	154	12.2.2 信号变换电路	223
9.1.1 非编码键盘	154	12.2.3 滤波器	224
9.1.2 编码键盘	156	12.3 传感器的接口电路实例	226
9.2 LED 显示器接口	157	12.3.1 温度接口电路	226
9.2.1 LED 显示器原理	157	12.3.2 湿度接口电路	227
9.2.2 LED 显示器的基本用法	158	12.3.3 电压接口电路	227
9.3 CRT 显示器接口电路	159	12.3.4 光电接口电路	228
9.4 直流电动机的微机控制	160	12.3.5 磁力接口电路	228
9.5 步进电动机的接口	164	第 13 章 微机应用系统设计	229
第 10 章 A/D 和 D/A 转换器应用 技术	167	13.1 微机应用系统设计原则	229
10.1 概述	167	13.2 微机应用系统实例——微机温室 环境监控管理系统	229
10.2 A/D 转换器应用技术	168	13.3 微机应用系统的抗干扰设计	231
10.3 D/A 转换器应用技术	178		
10.4 模拟多路开关 (MUX)	181		

第 1 章 通用微处理机及其辅助电路

1.1 概述

微机应用系统的基本组成如图 1-1 所示，它由五大部分组成，即输入设备、存储器、运算器、输出设备和控制器。其中输入设备是人机进行交往的输入端口，常用的输入设备有板键、键盘、光电或磁输入机等。无论哪一种输入设备都以二进制数或代码的形式向微机输入。存储器是用来存放输入设备送来的代码或数据以及运算器送来的运算结果等。存储器分为内存储器与外存储器两大类，用作内存储器的有磁心存储器和半导体存储器，用作外存储器的有磁带机、磁鼓和磁盘等。运算器是对各种信息进行算术运算和逻辑运算的主要部件，它由很多逻辑电路组成，包括寄存器、加法器、移位器和一些控制电路等。输出设备是人机交往的输出端口，它把结果或各种信息以数字、字符、图形等形式表示出来。常用的输出设备有打印机、数码管和显示器等。控制器是微机的指挥部，它控制整个微机自动地、协调一致地工作。控制器由时序电路和逻辑电路组成，它对微机的控制是通过输出的电压和脉冲信号来实施的。习惯上人们把运算器和控制器看成一个整体，称为中央处理机（Central Processing Unit —— CPU），也称为微处理机或微处理器。

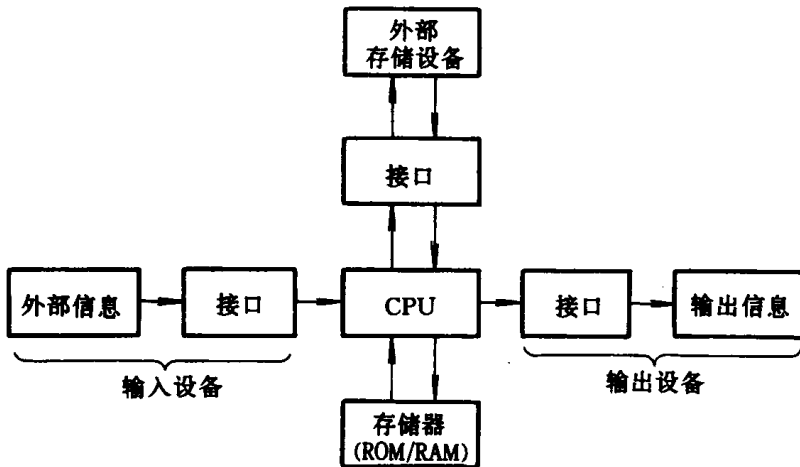


图 1-1 微机应用系统的基本组成

微处理机是微机应用系统的核心部分，自从 1971 年诞生以来，发展很快，据不完全统计，到目前为止已有 2000 多种，它们主要是字长、速度、容量和功能的不同。典型应用的是 INTEL 公司、MOTOLORA 公司和 ZILOG 公司生产的微处理机产品。本书主要介绍以 ZILOG 公司的 Z-80（8 位）微处理机、INTEL 公司的 8086（16 位）微处理机、INTEL 公司的 MCS-51 单片机组成的微机应用系统的接口技术。

把各种不同类型的集成芯片，例如微处理机芯片、各种规模型号的存储器芯片、各种类型的可编程的 I/O 接口芯片以及可选的 I/O 设备，以一定方式相互连接起来便可构成各种应用的微机应用系统，这些连接方式就是微机的接口技术。

1.2 通用微处理机简介

1.2.1 Z-80 CPU 的特征与控制信号

Z-80 是单片 8 位微处理机，功能较强，它由寄存器组、运算器、指令寄存器、译码器和定时及控制信号电路等组成，前两部分是运算器，后三部分是控制器，全集成于一块芯片内。它与外部系统的联系通过三组总线进行，其引脚功能如图 1-2 所示。

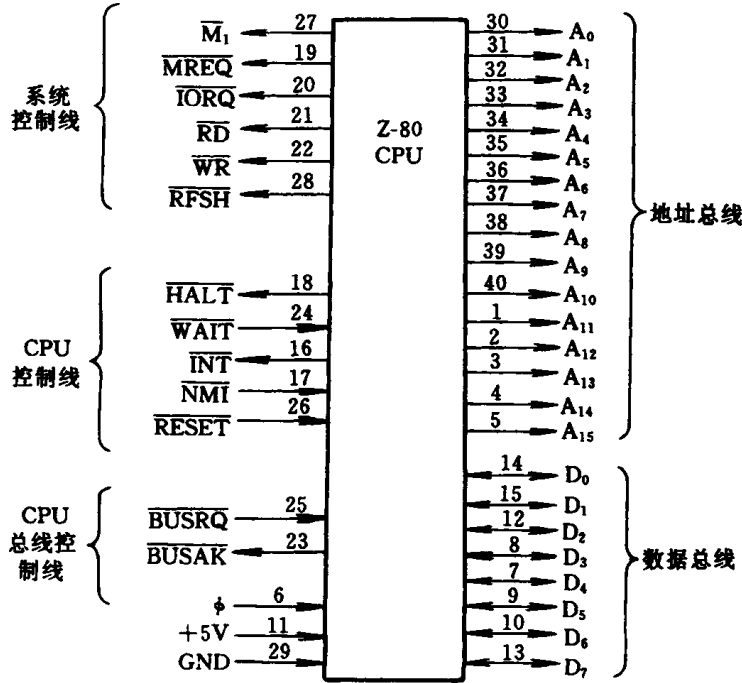


图 1-2 Z-80 CPU 引脚图

Z-80 有 40 条引脚，采用双列直插结构，其引线按功能可以分成三组：第 1 组是 16 根地址总线 (AB)，第 2 组是 8 根数据总线 (DB)，第 3 组是 13 根控制总线 (CB)，还有电源引线、地线和单相主时钟引线。

1. 地址总线

地址总线 16 根，编号为 A₀~A₁₅，高电平有效。地址总线为 CPU 与存储器、CPU 与外设、或者外设与存储器进行数据交换时提供地址。此总线作为存储器地址时，可选择的内存容量为 64KB。此总线信号代表外设地址时，只使用其低 8 位，因此，它能直接选择的外设可达 256 个。

2. 数据总线

数据总线 8 根，编号为 D₀~D₇，它为双向总线。数据总线上传送的是信息代码，它是 CPU 与存储器或 CPU 与外设进行交换的信息。

3. 控制总线

控制总线 13 根，可分为系统控制线、CPU 控制线及 CPU 总线控制线。

(1)*系统控制线 系统是指 CPU 以外的设备，如存储器及各个外设。系统控制线有 6

根,它是CPU对系统控制的信号线。

$\overline{M_1}$ (第1机器周期) 信号: 一输出信号, 低电平有效, 表示当前微机正在进行取指令码的操作。

$\overline{MREQ}$ (存储器请求) 信号: 三态输出信号, 低电平有效。这个存储器请求信号指示地址总线上保持有一个供存储器读或写操作的有效地址。

$\overline{IORQ}$ (输入/输出请求) 信号: 三态输出信号, 低电平有效, 表示地址总线上的信号是访问I/O设备的地址, 这时表示机器是对I/O设备的读写操作。

$\overline{RD}$ (读) 信号: 三态输出信号, 低电平有效, 表示CPU进行的是从存储器或I/O设备读取数据的操作, 存储器或I/O设备应利用此信号作为一门控制信号, 把数据送往数据总线。

$\overline{WR}$ (写) 信号: 三态输出信号, 低电平有效, 表示CPU进行的是向存储器或I/O设备写入信息操作。

$\overline{RFSH}$ (刷新) 信号: 输出信号, 低电平有效, 用它来直接对动态RAM刷新。

(2) CPU控制信号 它是外部设备对CPU进行控制的信号。

$\overline{HALT}$ (暂停状态) 信号: 输出信号, 低电平有效, 表示CPU正在执行HALT指令而处于暂停状态, 等待中断请求信号或启动信号的到来, 只有收到这些信号, CPU才能退出暂停状态, 恢复操作。

$\overline{WAIT}$ (等待) 信号: 输入信号, 低电平有效。 $\overline{WAIT}$ 告诉CPU所寻址的存储器或I/O设备, 尚未准备好数据传送。只要这个信号有效, CPU继续插入等待周期。这个信号能使CPU和任何速度的存储器与I/O设备同步。

$\overline{INT}$ (中断请求) 信号: 输入信号, 低电平有效。它由I/O设备发出接到CPU的 $\overline{INT}$ 端。CPU内有一中断允许触发器IFF, 在IFF=1, 并且没有 $\overline{NMI}$ 及 $\overline{BUSRQ}$ 请求时, CPU在执行完本命令就会响应I/O设备的中断请求, 转而为I/O设备服务。

$\overline{NMI}$ (非屏蔽中断请求) 信号: 输入信号, 低电平有效, 负边沿触发。这个非屏蔽中断请求信号, 具有比 $\overline{INT}$ 更高的优先权, 它不受IFF控制且总是在现行指令结束时被响应。

$\overline{RESET}$ (复位) 信号: 输入信号, 低电平有效。复位信号 $\overline{RESET}$ 迫使PC为0且初始化CPU。

(3) 总线控制信号 它是对三总线请求使用的控制。

$\overline{BUSRQ}$ (总线请求) 信号: 输入信号, 低电平有效。这个总线请求信号用于要求CPU把地址总线、数据总线和三态的输出控制信号处于高阻状态, 使别的设备能控制这些总线。这主要用于存储器与快速外设之间直接传送信息。当 $\overline{BUSRQ}$ 有效, CPU在现行的机器周期结束时立即响应, 使这些总线处于高阻状态。

$\overline{BUSAK}$ (总线响应) 信号: 输出信号, 低电平有效。它是CPU对系统提出的 $\overline{BUSRQ}$ 信号的应答, 如果有效, 表示CPU已让出了总线的使用权, 外部设备现可以使用这些总线了。

(4) 其他信号

1) ϕ (时钟) 信号: 单相系统时钟信号, Z-80时钟频率为2.5MHz, Z-80A时钟频

率为 4.0MHz。

2) 电源：一个输入直流 +5V 电源。

3) GND：地线。

(5) Z-80 控制信号与 CPU 状态之间关系 Z-80 控制信号与 CPU 状态之间关系如表 1-1 所示。

表 1-1 Z-80 控制信号与 CPU 状态之间关系

控制信号 CPU 状态	$\overline{M}_1$	$\overline{MREQ}$	$\overline{IORQ}$	$\overline{RD}$	$\overline{WR}$	$\overline{RFSH}$
取操作码	0	0	1	0	1	1
存储器读	1	0	1	0	1	1
存储器写	1	0	1	1	0	1
I/O 读	1	1	0	0	1	1
I/O 写	1	1	0	1	0	1
中断应答	0	1	0	1	1	1
刷新	1	0	1	1	1	0

众所周知微处理机执行指令的一系列动作都是在时钟脉冲 ϕ 的统一控制下一步一步进行的，都需要一定时间，各个指令需要时间虽有很大差别，但它们都是由存储器读或写、输入输出设备的读或写以及中断响应的这些基本机器周期所组成，图 1-3 示出 Z-80 的时序。

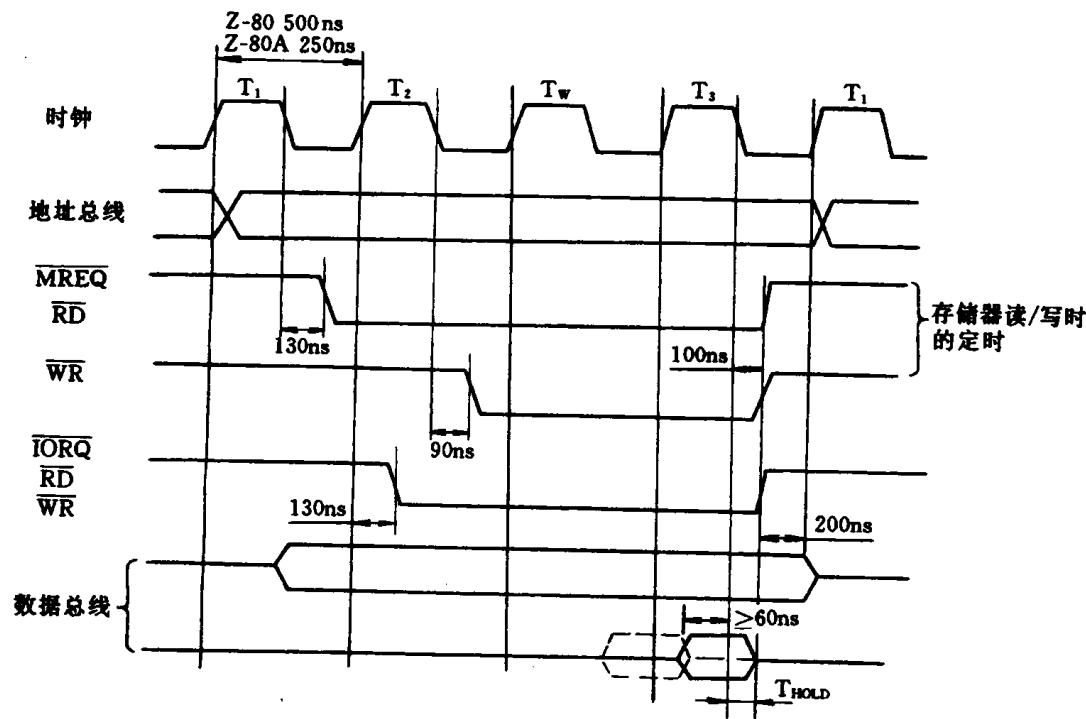


图 1-3 Z-80 CPU 的时序

图 1-3 中的 T_w 为等待周期，它不是应外部等待信号 $\overline{WAIT}$ 的请求而插入的，而是 CPU 内部自动插入的，它是当 CPU 与存储器以及外设连接时所考虑的相应时序配合问题，这样就可以使 CPU 与任何速度的存储器连接（当然存储器的速度还是要由系统的要求而选

定的)。

1.2.2 8086 CPU 的特征与控制信号

1. 8086 CPU 引脚功能

8086 CPU 是一种高性能 16 位微处理机，现已发展有 80186、80286、80386、80486，性能不断改善，但软件兼容，目前大多数微机系统都采用这种系列的 CPU。

8086 的引脚功能如图 1-4 所示，它是 40 条引脚的双列直插式封装。有 20 条地址线，16 条数据线（与地址线复用），以及必要的控制信号线。因芯片引脚数量太多，对部分引脚采用了分时复用方式。

8086 有最大和最小两种工作模式，8 条引脚（24~31）在两种工作模式中，具有不同的功能，图 1-4 括号中所示为最大模式下被重新定义的控制信号。8086 的引脚功能简述如下：

(1) AD₁₅~AD₀ 分时复用的地址数据总线。传送地址时三态输出，传送数据时可双向三态输入/输出。

(2) A₁₉/S₆、A₁₈/S₅、A₁₇/S₄、A₁₆/S₃ 分时复用的地址状态线。作为地址线用时，A₁₉~A₁₆与 AD₁₅~AD₀ 一起构成访问存储器的 20 位物理地址。当 CPU 访问 I/O 端口时，A₁₉~A₁₆保持为“0”。

作为状态线用时，S₆~S₃ 用来输出状态信息，其中，S₃ 和 S₄ 可用于表示当前使用的段寄存器号。当 S₄S₃=10 时，表示当前正使用 CS 对寄存器寻址，或是当前正在对 I/O 端口或中断矢量寻址，这时不需要使用段寄存器。S₄S₃ 状态编码如表 1-2 所示。

表 1-2 S₄S₃ 状态编码

S ₄	S ₃	段寄存器
0	0	ES
0	1	SS
1	0	CS (I/O、INT)
1	1	DS

S₅ 用来表示中断标志状态线，当 IF=1 时，S₅ 置“1”；S₆ 恒保持为“0”。

(3) BHE/S₇ 总线高位有效信号。三态输出，低电平有效，用来表示当前高 8 位数据总线上的数据有效。读/写存储器或 I/O 端口以及中断响应时，BHE 与 AD₀ 配合表示当前总线使用情况，如表 1-3 所示。

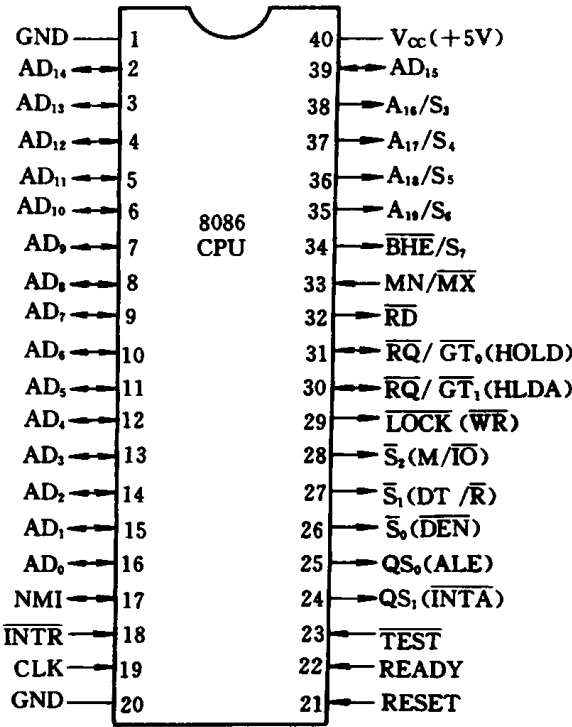


图 1-4 8086 CPU 引脚图

表 1-3 总线使用情况

BHE	AD ₀	总线使用情况
0	0	16 位数据总线上进行字节传送
0	1	高 8 位数据总线上进行字节传送
1	0	低 8 位数据总线上进行字节传送
1	1	无效

非数据传送期间， S_7 输出状态信息，在 CPU 处于保持响应期间被设置为高阻抗状态。

(4) $\overline{RD}$ 读信号，三态输出，低电平有效，表示当前 CPU 正在读存储器或 I/O 端口

(5) $\overline{WR}$ 写信号，三态输出，低电平有效，表示当前 CPU 正在写存储器或 I/O 端口。

(6) $M/\overline{IO}$ 存储器或 I/O 端口访问信号。三态输出，高电平时，表示当前 CPU 正在访问存储器；低电平时，表示当前 CPU 正在访问 I/O 端口。

(7) READY 准备就绪信号。由外部输入，高电平有效，表示 CPU 访问的存储器或 I/O 端口已准备好传送数据。

(8) INTR 中断请求信号。由外部输入，电平触发，高电平有效，INTR 有效时，表示外部向 CPU 发出中断请求，CPU 在每条指令的最后一个时钟周期对 INTR 进行测试，一旦测试到中断请求，并且当前中断允许标志 $IF=1$ 时，则暂停执行下条指令，输入中断响应周期。

(9) $\overline{INTA}$ 中断响应信号。向外部输出，低电平有效，表示 CPU 响应了外部发来的 INTR 信号，在中断响应总线周期，可用作读选通信号。

(10) NMI 不可屏蔽中断请求信号。由外部输入，正跳沿触发有效，不受中断允许标志的限制，CPU 一旦测试到 NMI 请求有效，待当前指令执行完就自动从中断入口地址表中找到类型 2 中断服务程序的入口地址，并输出执行。它是一种比 INTR 高级的中断请求。

(11) $\overline{TEST}$ 测试信号。由外部输入，低电平有效。当 CPU 执行 WAIT 指令时，每隔 5 个时钟周期对 $\overline{TEST}$ 进行一次测试，若测试到 $\overline{TEST}$ 无效，则 CPU 处于等待状态，直到 $\overline{TEST}$ 有效，CPU 才继续执行下一条指令。

(12) RESET 复位信号。由外部输入，高电平有效。

(13) ALE 地址锁存允许信号，向外部输出，高电平有效。在最小模式系统中用来作为 8282/8283 地址锁存器的片选信号。

(14) $DT/\overline{R}$ 数据发送/接收控制信号，三态输出。在最小模式系统中用来控制 8286/8287 数据收发器的数据传送方向。当 $DT/\overline{R}$ 为高电平时，表示数据从 CPU 向外部输出，即完成写操作； $DT/\overline{R}$ 为低电平时，表示数据从外部向 CPU 输入，即完成读操作。

(15) $\overline{DEN}$ 数据允许信号。三态输出，低电平有效。在最小模式系统中用作 8286/8287 数据收发器的选通信号。

(16) HOLD 总线请求信号。由外部输入，高电平有效。在最小模式系统中表示有其他共享总线的处理机向 CPU 请求使用总线。

(17) HLDA 总线请求响应信号。向外部输出，高电平有效。CPU 一旦测试到有 HOLD 请求时，就在当前总线周期结束时，使 HLDA 有效，表示响应这一总线请求，并立即让出总线使用权，CPU 中指令执行部件 (EU) 可继续工作到下一次要求使用总线为止，

一直到 HOLD 无效, CPU 才将 HLDA 置成无效, 并收回对总线的使用权, 继续操作。

(18) $\overline{MN}/\overline{MX}$ 工作模式选择信号。由外部输入, $\overline{MN}/\overline{MX}$ 为高电平时, 表示 CPU 工作在最小模式系统中; $\overline{MN}/\overline{MX}$ 为低电平时, 表示 CPU 工作在最大模式系统中。

(19) CLK 主时钟信号。由 8284 时钟发生器输入。8086 CPU 可使用的时钟频率随芯片型号不同而异, 8086 时为 5MHz, 8086-1 时为 8MHz, 8086-2 时为 10MHz, 8086 CPU 工作在最大模式系统中重新定义引脚功能如下:

1) $\overline{S_2S_1S_0}$ 总线周期状态信号。三态输出, 在最大模式系统中由 CPU 传送给 8288 总线控制器, 8288 对它们译码后, 代替 CPU 输出相应的控制信号。

2) $\overline{LOCK}$ 封锁信号。三态输出, 低电平有效。 $\overline{LOCK}$ 有效时, 表示 CPU 不允许其他总线主控者占用总线。这个信号由软件设置。当在指令前加上 LOCK 前缀时, 则在执行这条指令期间由 $\overline{LOCK}$ 保持有效, 即在此指令执行期间, 封锁其他主控者使用总线。

3) $\overline{RQ}/\overline{GT_0}$, $\overline{RQ}/\overline{GT_1}$ 请求/同意信号。双向, 低电平有效, 输入时表示其他主控者向 CPU 请求使用总线; 输出时表示 CPU 对总线请求的响应信号, 两条线可同时与两个主控者相连, 内部保证 $\overline{RQ}/\overline{GT_0}$ 比 $\overline{RQ}/\overline{GT_1}$ 有较高优先级。

4) QS_1 、 QS_0 指令队列状态。向外部输出, 用来表示 CPU 中指令队列当前的状态, 其含义如表 1-4 所示。

表 1-4 QS_1 、 QS_0 的含义

QS_1	QS_0	含 义
0	0	无操作
0	1	从队列中取第一个字节
1	0	队列已空
1	1	从队列中取后续字节

2. 8086 的最小模式系统

8086 构成的微机系统, 有最小模式和最大模式两种系统配置。最小模式是单处理机系统, 系统中所需控制信号全部由 8086 CPU 本身直接提供; 最大模式可构成多处理机系统, 系统中所需控制信号由总线控制器提供。CPU 工作模式的选择是由硬件决定的, 当 CPU 的引脚 $\overline{MN}/\overline{MX}$ 接高电平 (+5V) 时, 构成最小模式; 当 $\overline{MN}/\overline{MX}$ 接低电平 (地) 时, 构成最大模式。

图 1-5 示出 8086 CPU 构成的最小模式的基本配置, 它是一个以 8086 CPU 为主体的单处理机系统, 所有控制信号均由 CPU 提供。

系统中 8284A 是 8086、8088 专用时钟发生器集成芯片, 其引脚如图 1-6 所示, 其中 PCLK 为外设用 TTL 电平时钟, CLK 为 CPU 用时钟, $\overline{ASYN\overline{C}}$ 为准备就绪同步选择, CSYNC 为时钟同步用输入, EFI 为外时钟输入, RDY_1 、 $\overline{AEN_1}$ 、 RDY_2 、 $\overline{AEN_2}$ 用于低速存储器的等待信号。

图 1-5 系统中 8282 是带三态缓冲器的通用 8 位数据锁存器, 它的引脚如图 1-7 所示。当 STB 有效时, 输入端上的 8 位数据被锁存到锁存器中, $\overline{OE}$ 有效时, 锁存器中的数据输出; $\overline{OE}$ 无效时, 则输出端呈现高阻状态。它们在 8086 最小模式系统中作地址锁存器用, 当系统中所接存储器或 I/O 接口容量不超过 64KB 时, 只需要 2 片 8282 就够了。每片锁存 8

位地址码，当 CPU 的地址锁存允许信号 ALE 有效时，地址被锁存到锁存器中。由于 $\overline{OE}$ 被固定接地，所以 CPU 输出的地址码一旦被锁存后，立即稳定输出在地址总线上。

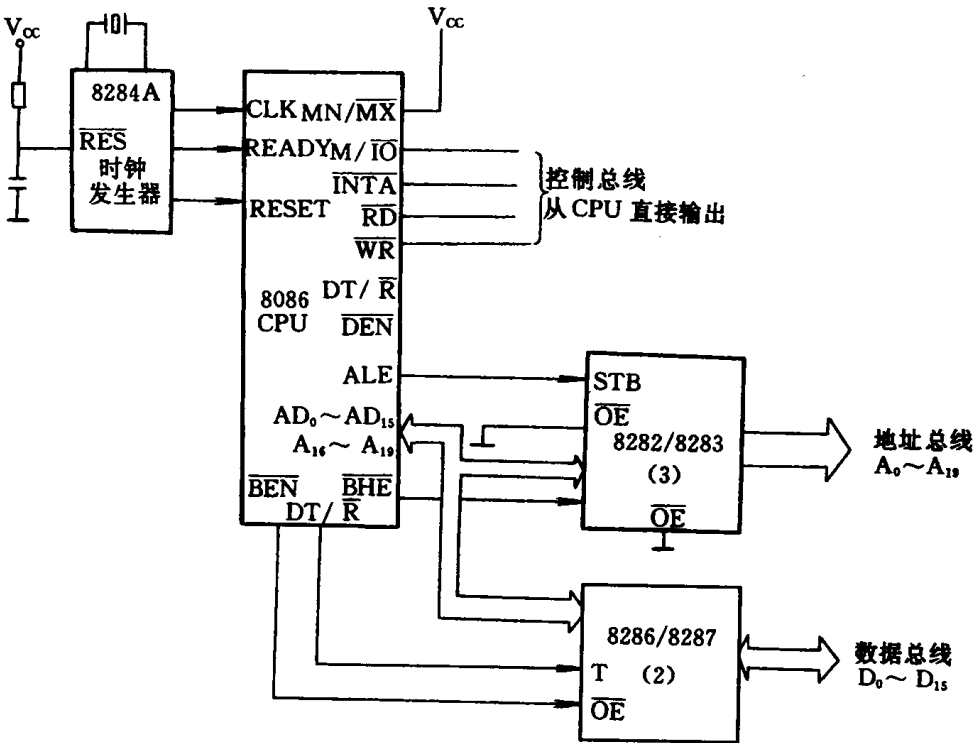


图 1-5 8086 CPU 最小模式系统配置

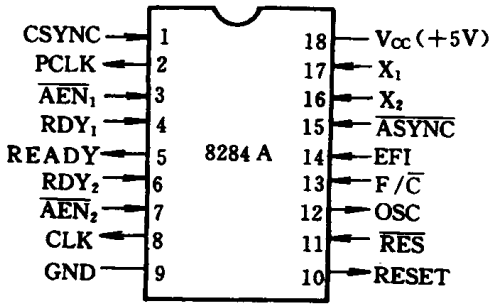


图 1-6 8284A 的引脚图

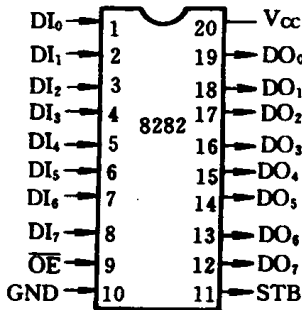


图 1-7 8282 引脚图

8286 是为数据总线接口设计的三套输出 8 位双向数据缓冲器，引脚如图 1-8 所示。 $\overline{OE}$ 是开启缓冲器的控制信号，当 $\overline{OE}$ 有效时，允许数据通过，当 $\overline{OE}$ 无效 ($\overline{OE}=1$) 时，禁止数据通过缓冲器，输出呈高阻状态。T 是数据传送方向控制信号，当 T 端为高电平时，8 位数据被正向传送，由 $A_7 \sim A_0$ 传送到 $B_7 \sim B_0$ ；当 T 端为低电平时，8 位数据被反向传送，由 $B_7 \sim B_0$ 传送到 $A_7 \sim A_0$ 。在 8086 最小模式系统中，8286 作为 8 位数据缓冲用，8286 的 $\overline{OE}$ 端与 8086 的数据允许信号 $\overline{DEN}$ 相连，当 8086 与存储器或 I/O 端口进行数据交换时， $\overline{DEN}$ 有效 ($\overline{DEN}=0$)，使 8286 的 $\overline{OE}$ 有效，允许数据通过，反之， $\overline{DEN}$ 无效时， $\overline{OE}$ 无效，禁止数据通过。8286 的 T 端与 8086 的数据发送/接

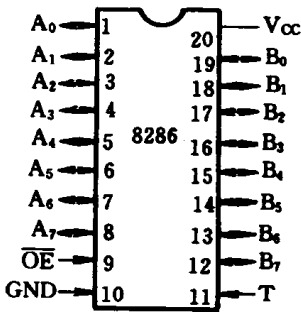


图 1-8 8286 引脚图

收信号 $\overline{DT/\overline{R}}$ 相连, 当 8086 写存储器或 I/O 端口时, $\overline{DT/\overline{R}}$ 为高电平, 使 8286 的 T 端为高电平, 8 位数据由 CPU 向存储器或 I/O 端口输出; 当 8086 读存储器或 I/O 端口时, $\overline{DT/\overline{R}}$ 为低电平, 使 8286 的 T 端为低电平, 8 位数据由存储器或 I/O 端口向 CPU 输入。在 8086 最小模式系统中, 可以不用数据收发 8286 缓冲器, 这时地址/数据总线可直接与存储器或 I/O 端口的数据线相连。

3. 8086 的最大模式系统

8086 的最大模式系统的基本配置如图 1-9 所示, 在结构上与最小模式系统的主要区别是系统中增设一个 8288 总线控制器, 这样可构成以 8086 CPU 为中心的多主控者微机系统。由 CPU 输出的状态信号 $\overline{S_2} \sim \overline{S_0}$ 同时送给 8288, 由 8288 输出 8086 CPU 系统所需要的控制信号。

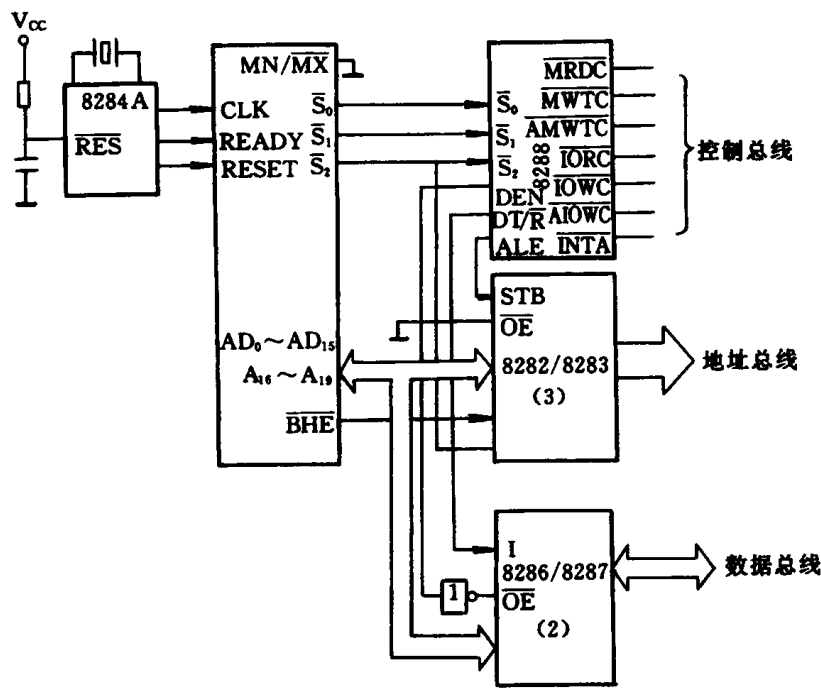


图 1-9 8086 的最大模式系统配置

8288 的引脚如图 1-10 所示, 来自 8086 CPU 的总线状态信号 $\overline{S_2} \overline{S_1} \overline{S_0}$ 经状态译码器译码后, 与输入控制信号 $\overline{AEN}$ 、 $\overline{CEN}$ 和 $\overline{IOB}$ 相互配合可产生一系列总线命令和控制信号。8288 提供的总线命令信号如表 1-5 所示。

现对有关控制信号作用说明如下:

(1) $\overline{MRDC}$ (读存储器) 命令: 它相当于最小模式系统中由 CPU 发出的控制信号 $\overline{RD}$ 和 $\overline{IO/\overline{M}}=0$ 的组合。

(2) $\overline{IORC}$ (读 I/O 端口) 命令: 它相当于最小模式系统中由 CPU 发出的控制信号 $\overline{RD}$ 和 $\overline{IO/\overline{M}}=1$ 的组合。

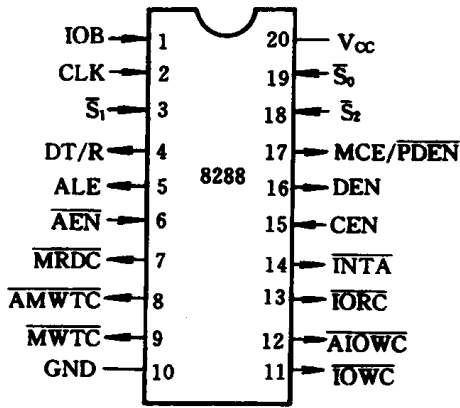


图 1-10 8288 引脚图