

Intel 96 系列

单片微型机应用详解

● 金磐石 王永明

● 电子工业出版社

Intel 96 系列

单片微型机应用详解

金磐石 王永明

电子工业出版社

(京)新登字 055 号

内 容 提 要

随着单片机应用领域的不断拓宽,它越来越被人们所重视。本书根据作者多年从事开发、研制单片机系统的实际经验,详细介绍了 Intel 96 系列单片微型机的基本结构、工作原理、指令系统、基本硬件的设计和编程方法。书中运用大量的实例,讲解了如何正确掌握和使用 8×9× 的中断系统、定时器、HSI、HSO、A/D 与 PWM(D/A)以及串行口等。全书例证丰富,理论联系实际,适合从事微机应用的广大科技工作者阅读,也可作为大、专院校的教材,同时还可供计算机爱好者参考。

Intel 96 系列单片微型机应用详解

金磐石 王永明

责任编辑 贾贺

电子工业出版社出版(北京市万寿路)

电子工业出版社发行 各地新华书店经销

电子工业出版社计算机排版室排版

民族印刷厂印刷

开本:787×1092 毫米 1/16 印张:17.25 字数:441 千字

1992 年 4 月第一版 1992 年 4 月第一次印刷

印数:10 100 册: 定价:14.00 元

ISBN7-5053-1930-2/TP·465

前 言

Intel 96 系列单片微型机(包括 $8\times 9\times$ 和 $8\times C19\times KB$)简称 96 系列单片机。它是 1983 年后陆续推出的,国内的开发、应用自 80 年代末才开始,但其发展速度却很惊人。目前,几乎占领了我国 16 位及准 16 位单片机的全部市场,而且发展潜力很大。

96 系列单片机之所以受到用户的青睐,与它的高性能、高速度分不开。其片内的寄存器阵列兼有多种功能,彻底克服了面向累加器而引起的“瓶颈”问题;它的中断系统也进行了改造,可以方便地进行软件优先级的设置;其它诸如以先进先出队列(FIFO)为基础的高速输入(HSI)、以内容寻址存储器(CAM)为基础的高速输出(HSO)、以比较器为基础的脉宽调制(PWM)和以 16 位定时器为基础的监视定时器(WATCHDOG)均属首次引入单片机的功能部件;其 10 位多路采样保持和 A/D 转换器也非常具有特色。再配上高效指令系统,使 96 系列成为同类单片机中的佼佼者。

为适应人们学习、使用 96 系列单片机的需要,我们根据多年从事开发、研制单片机系统的经验,参考了大量国内外有关资料,编写了《Intel 96 系列单片微型机应用详解》一书。

全书共分十一章。第一章为 $8\times 9\times$ 的基本结构和工作原理;第二章较详细地介绍了其指令系统;第三章给出了实际应用中经常用到的计算程序;第四章说明了基本硬件的设计和 $879\times$ 的编程方法;第五、六、七、八、九、十章中列举了大量实例,从原理、应用两方面讲解了如何正确掌握和使用 $8\times 9\times$ 的中断系统、定时器、HSI、HSO、A/D 与 PWM(D/A)以及串行口;第十一章则重点叙述了 96 系列的高档芯片 $8\times C19\times KB$ 。

本书内容新颖、系统,理论与实际结合较强,有许多内容是以往同类书籍所没有或很少介绍的。不但可作为大、专院校及培训班学习 96 系列单片机原理的教材,还为从事微机应用的科研和工程技术人员提供了参考资料。

本书第十一章由王永明执笔,其它各章由金磐石执笔。

在成书过程中,得到了北京东方计算机研究所的大力支持,孙腾湛总工程师审阅了全书,并提供了 ECI UD98 交叉窗口调试器。在此,表示衷心的感谢。

限于水平和经验,本书还会有许多缺点和不足,望广大读者批评指正。

作 者

目 录

第一章 8×9×的基本结构和工作原理	(1)
1.1 8×9×的基本构成	(2)
一、芯片的组成	(2)
二、引脚的功能	(3)
1.2 CPU 操作	(7)
一、CPU 总线	(7)
二、CPU 寄存器阵列	(7)
三、寄存器算术逻辑单元 RALU	(7)
四、CPU 的基本操作	(8)
五、内部定时	(8)
1.3 存储器空间	(9)
一、寄存器阵列	(10)
二、特殊功能寄存器 SFR	(10)
三、掉电	(11)
四、保留的存储器空间	(12)
五、内部 ROM/EPROM	(13)
六、内部可执行 RAM(XRAM 仅 8×9×JF 具有)	(13)
七、存储控制器	(13)
八、系统总线	(13)
1.4 I/O 口	(19)
一、P0 口	(19)
二、P1 口	(19)
三、P2 口	(19)
四、P3 口和 P4 口	(20)
五、I/O 控制和状态寄存器	(20)
1.5 复位	(21)
一、复位信号	(21)
二、复位状态	(22)
三、复位同步方式	(22)
第二章 8×9×单片机指令系统	(24)
2.1 基础知识	(24)
一、操作数类型	(24)
二、8×9×的寻址方式	(25)
三、8×9×的程序状态字	(28)

2.2	指令系统概述	(30)
2.3	8×9×指令系统详解	(33)
	一、数据传送指令	(33)
	二、算术指令	(35)
	三、逻辑指令	(41)
	四、堆栈操作指令	(43)
	五、跳转和调用指令	(45)
	六、条件跳转指令	(46)
	七、位测试并跳转指令	(47)
	八、循环控制指令	(47)
	九、单寄存器指令	(48)
	十、移位指令	(49)
	十一、专用控制指令	(50)
	十二、规格化指令	(51)
第三章	计算程序的设计	(52)
3.1	定点数运算	(53)
	一、定点数基本概念	(53)
	二、定点数加减运算	(54)
	三、定点数乘法运算	(56)
	四、定点数除法运算	(58)
3.2	浮点数运算	(61)
	一、浮点数表示方法	(61)
	二、浮点数加减法运算	(64)
	三、浮点数乘法运算	(69)
	四、浮点数除法运算	(70)
	五、定点数转换成浮点数	(72)
3.3	数制转换	(73)
	一、整数十翻二	(73)
	二、小数十翻二	(74)
	三、整数二翻十	(75)
	四、小数二翻十	(75)
3.4	多项式运算及其应用	(76)
	一、浮点多项式计算	(76)
	二、多项式的应用	(79)
3.5	常用函数计算程序	(83)
	一、正弦函数	(83)
	二、指数函数 e^x	(85)
	三、自然对数函数 $\ln x$	(87)
	四、平方根	(90)
第四章	设计方法及 879×的编程	(92)
4.1	基本硬件的连接	(92)

一、电源	(92)
二、振荡器	(93)
三、复位	(93)
4.2 8×9×与外部存储器的连接	(95)
一、地址译码	(95)
二、8×9×存储系统扩展实例	(96)
4.3 输入/输出	(100)
一、P0 口	(100)
二、P1 口	(100)
三、P2 口	(101)
四、P3 和 P4 口	(102)
五、P3 和 P4 口的重构	(103)
六、准双向口的编程要求及硬件考虑	(104)
4.4 端口扩展器 87C75PF	(104)
一、87C75PF 的设计原理	(104)
二、87C75PF 的特点	(106)
三、87C75PF 的工作特性	(107)
四、87C75PF 的应用实例	(114)
4.5 879×的 EPROM 编程	(116)
一、自动配置字节编程方式	(117)
二、自动编程方式	(117)
三、从机编程方式	(120)
四、运行编程方式	(122)
五、变型的快脉冲编程算法	(123)
第五章 中断系统及其应用	(124)
5.1 8×9×的中断结构	(124)
一、中断源	(124)
二、与中断有关的寄存器	(126)
三、中断处理过程	(127)
四、中断优先级的控制	(128)
五、中断响应时间	(130)
5.2 中断系统编程实例	(131)
一、修改堆栈以改变中断服务程序的返回地址	(131)
二、用软件产生 EXTINT 和 HSI.0 中断	(132)
第六章 定时器及其应用	(134)
6.1 8×9×定时器	(134)
一、定时器 T1	(134)
二、定时器 T2	(134)
三、定时器中断	(135)
四、监视定时器	(135)
6.2 定时器应用实例	(136)

一、定时器 T1 的应用	(136)
二、定时器 T2 的应用	(138)
三、监视定时器的应用	(140)
第七章 高速输入 HSI 及其应用	(141)
7.1 HSI 的工作原理	(141)
一、HSI 的结构	(141)
二、HSI 的控制	(142)
三、HSI 引脚产生的中断	(143)
7.2 HSI 的使用方法	(144)
一、HSI 使用要点	(144)
二、查询法	(144)
三、中断法	(145)
7.3 HSI 的几个应用实例	(147)
一、利用 HSI 测量脉冲信号	(147)
二、HSI 测量转速	(150)
三、使用 HSI 测距	(152)
第八章 高速输出 HSO 及其应用	(155)
8.1 HSO 工作原理	(155)
一、HSO 的基本结构	(155)
二、HSO 的控制	(156)
三、HSO 的清除	(157)
四、采用定时器 T2 作 HSO 时间基准	(158)
五、软件定时器	(158)
六、HSO 中断	(159)
8.2 HSO 的使用方法	(159)
一、HSO 使用要点	(159)
二、HSO 的编程方法	(159)
8.3 HSO 应用实例	(159)
一、事件的触发	(159)
二、利用 HSO 产生脉冲	(161)
三、用 HSO 产生 PWM 信号	(165)
四、用 HSO 驱动步进电机	(168)
五、用 HSO 实现高速可编程控制器	(170)
六、HSO 的其它应用	(172)
第九章 A/D 和 PWM(D/A)及其应用	(177)
9.1 $8 \times 9 \times$ A/D 转换器及应用	(177)
一、 $8 \times 9 \times$ A/D 转换器原理及其性能的提高	(177)
二、A/D 转换器的控制	(181)
三、查询式 A/D 转换	(182)
四、利用 $8 \times 9 \times$ 构成温度控制器	(184)
9.2 PWM 输出(D/A)	(185)

一、PWM 结构和工作原理	(185)
二、产生 PWM 波形	(186)
第十章 串行口及其应用	(190)
10.1 串行口的工作原理	(190)
一、串行口的工作方式	(190)
二、串行口控制	(192)
10.2 串行口的使用方法	(194)
一、串行口的使用特点	(194)
二、串行口的编程方式	(195)
10.3 串行口应用实例	(195)
一、方式 0 的应用	(195)
二、方式 1 的应用	(197)
三、方式 2 和方式 3 的应用	(200)
10.4 利用 HSI/HSO 作软件串行口	(207)
第十一章 96 系列高档芯片 8×C19×KB	(217)
11.1 8×C19×KB 的概述	(217)
一、80C196KB 的特点	(218)
二、CPU 与内部定时	(218)
三、特殊功能寄存器窗口	(218)
四、保留存储器空间	(221)
五、系统总线	(221)
六、复位状态与最小硬件连接	(221)
七、管脚与封装	(223)
11.2 80C196KB 指令集	(224)
11.3 中断系统	(226)
一、特殊中断	(226)
二、中断控制	(227)
三、中断优先级	(228)
四、中断响应时间	(228)
11.4 定时器	(230)
一、定时器 1	(230)
二、定时器 2	(230)
三、定时器中断	(231)
11.5 高速输入和输出	(232)
一、高速输入(HSI)	(232)
二、高速输出的 CAM 和有关寄存器	(232)
三、清除 HSO 和锁定命令	(233)
四、使用 HSO 的注意事项	(234)
11.6 串行口	(234)
一、串行口状态与控制	(234)
二、串行口中断	(236)

11.7 模拟接口和 I/O 口	(236)
11.8 特殊运行方式	(237)
一、空闲方式	(237)
二、掉电方式	(238)
三、ONCE 和测试方式	(239)
11.9 $\overline{\text{HOLD}}/\overline{\text{HLDA}}$ 协议	(239)
附录 1 96 系列指令系统一览表	(243)
附录 2 96 系列指令操作码与状态时间表	(246)
附录 3 $8 \times 9 \times$ 特殊功能寄存器	(250)
附录 4 $8 \times \text{C19} \times \text{KB}$ 的新特点	(257)
参考文献	(263)

第一章 8×9×的基本结构和工作原理

微型计算机的出现,使电子计算机更广泛地应用于人类生活中的各个领域。单片微型计算机作为微型计算机的一员,在工业控制、仪器仪表及家用电器的智能化等方面,更具有特殊的意义,发挥着重要的作用。

单片微型计算机简称单片机,它是在一个芯片上集成了微型计算机系统的许多功能,主要包括中央处理单元(CPU)、存储器(RAM 和 ROM)及外围输入/输出接口(I/O 口)。它与通常所说的微处理器不同,微处理器(如 Intel 公司的 8086)只是一个 CPU 和一些辅助功能部件,其本身并不能构成一个微型计算机系统。而单片机则不同,它具有许多适用于控制的软件指令和相应的硬件支持,因而广泛地应用于各种设备和仪器的控制之中。

单片机问世以来,应用日趋广泛,性能不断地改善和提高,在许多应用场合取代了现有的多片微型计算机系统,但其性能价格比却更为优越,体积却大为减小。单片机的潜在能力愈来愈为人们所注意。

随着应用的进一步深入,8 位单片机已不能满足需要。例如,51 系列单片机只能用于速率为 300 波特的马达控制,对速率为 2400 波特新型马达的控制就显得无能为力了;又如,每秒 100~200 个字符的打印机,可用 51 系列单片机加以控制,但每秒 200~400 个字符的打印机控制则需要性能更强的 96 系列单片机。

96 系列单片机有许多种,表 1.1 所示的单片机通称为 8×9×,这类单片机特别适用于要求很高的实时控制。表 1.2 列出了其应用的范围。

表 1.1 HMOS 8×9×单片机

	掩膜 ROM			CPU			用户可编程芯片					
							EPROM			运行编程 EPROM		
	68-Pin	64-Pin	48-Pin	68-Pin	64-Pin	48-Pin	68-Pin	64-Pin	48-Pin	68-Pin	64-Pin	48-Pin
有 A/D	8397BH 8397JF	8397BH 8397JF	8395BH 8398	8097BH 8097JF	8097BH 8097JF	8095BH 8098	8797BH		8795BH 8798	8797BH 8797JF	8797BH 8797JF	8798
无 A/D	8396BH			8×9×								

表 1.2 8×9×的应用范围

工业方面:	示波器	导弹制导	打印机
电机控制	消费品方面:	鱼雷制导控制	汽车方面:
工业机器人	录像机	智能武器装置	点火控制
离散与连续过程控制	激光盘驱动	航天导航系统	变速器控制
数字控制	高级电视游戏	数据处理方面:	燃料控制
智能传感器	电讯方面:	图形终端	防滑刹车
仪器仪表方面:	调制解调器	彩色与黑白复印机	排气控制
医疗器械	智能线路控制	温式硬盘驱动	
液体和气体色谱仪	导航与控制方面:	磁带机驱动	

本章以 8×9×(包括 8×9×BH、8×9×JF 和 8×98 系列芯片)为例,对 Intel 公司近年来

推出的 16 位和准 16 位单片机的基本构成、CPU 操作功能、存储器空间、I/O 口功能以及工作方式,进行了简要的说明。有关功能的详细描述见以后各章。

1.1 8×9×的基本构成

一、芯片的组成

8×9×组成原理如图 1.1 所示。为方便地描述其功能,可将它分成以下几块:

- 一个高性能的 17 位 CPU(RALU 和 232 个字节的寄存器阵列);
- 一整套存储及存储控制系统;
- 高速输入/输出(HSI/HSO);
- 带有采样/保持(S/H)电路的 4 通道 10 位 A/D 转换器;
- 中断控制和等待状态产生逻辑;
- 一个全双工同步/异步串行口;
- 两个 16 位硬件定时器和一个监视定时器(WATCHDOG);
- 一个可用作 D/A 转换的脉冲宽度调制输出(PWM);
- 支持整个芯片正常运行的时钟产生逻辑。

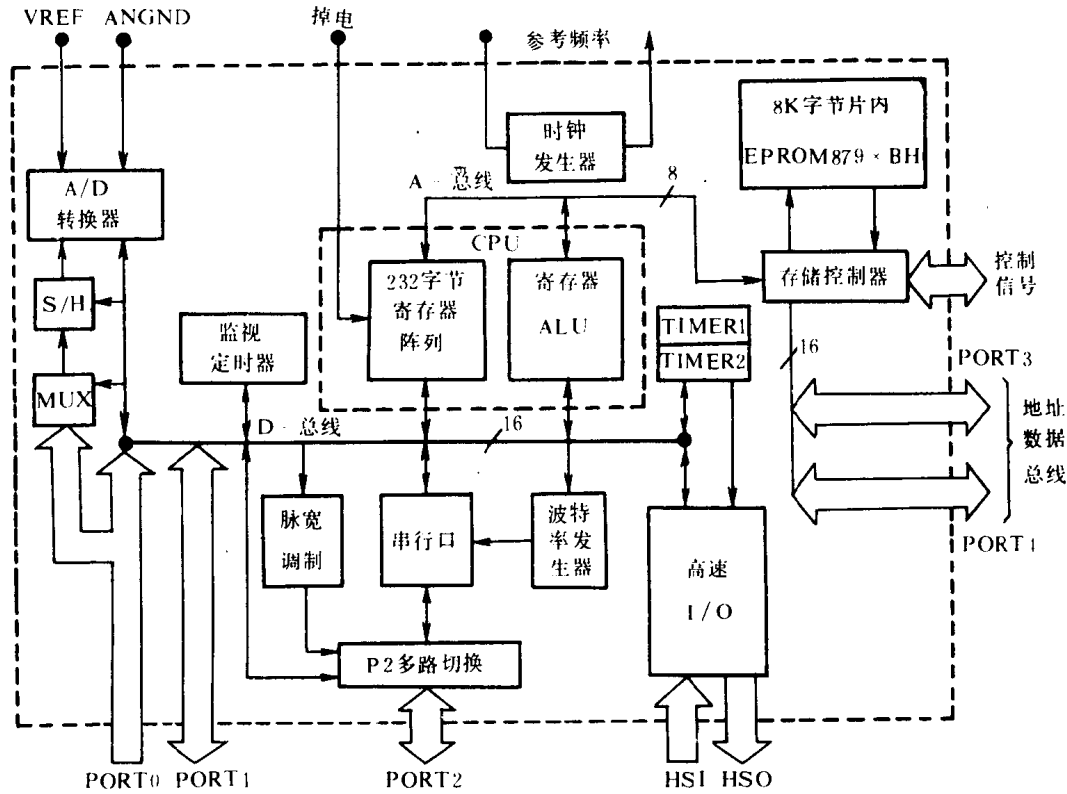


图 1.1 8×9×的组成框图

二、引脚的功能

8×9×共有 DIP、PGA、PLCC 和 SDIP 四种不同形式的封装。根据引脚个数分为 48 脚芯片、64 脚芯片和 68 脚芯片。8×98 是只有 48 引脚的 DIP 结构。引脚配置如图 1.2、1.3、1.4、1.5、1.6 所示。

与 68 脚相比,8×9×48 脚的芯片没有如下引脚:

- P0 口中的 $P_{0.0} \sim P_{0.3}$ 四个引脚;
- P1 口的全部八根引脚 $P_{1.0} \sim P_{1.7}$;
- P2 口中的 $P_{2.3}$ 、 $P_{2.4}$ 、 $P_{2.6}$ 和 $P_{2.7}$ 四个引脚;
- 控制信号中的 CLOCKS、INST、NMI、BUSWIDTH、 $T_2CLK(P_{2.3})$ 、 $T_2RST(P_{2.4})$ 。

表 1.3 是各种封装引脚的对照表。

8×9×各引脚功能如下:

- V_{CC} ,主电源电压(+5V)。
- V_{SS} ,数字地(0V)。有两个 V_{SS} 引脚,它们都必须接地。
- V_{PD} ,RAM 备用电源(+5V)。正常操作时,应加上此电源。掉电情况下,在 V_{CC} 未降到规定值之前(V_{PP} 应保持在规定的范围内),使 $\overline{RESET}$ 有效,则 8×9×片内寄存器阵列顶部的 16 个字节内容保持不变,直至 V_{CC} 恢复到规定范围内且振荡器达到稳定时为止。
- V_{REF} ,片内 A/D 转换器的参考电压(+5V)。它也是 A/D 转换器模拟部分的电源电压以及读 P0 口操作所需的逻辑电压。
- ANGND,A/D 转换器的参考地,应与 V_{SS} 同电位。
- V_{PP} ,EPROM 型芯片的编程电压。编程时,应接+12.75V 的电压,其它情况下应浮空为高电平。在含 ROM 或不含 ROM/EPROM 的芯片中,此引脚上的电压不能高于 V_{CC} ;在含片内 EPROM 的应用系统中,该引脚必须浮空。
- XTAL₁,片内振荡器反相器的输入,也是片内时钟发生器的输入。通常接外部晶体。
- XTAL₂,片内振荡器反相器的输出。通常接外部晶体。
- CLKOUT,片内时钟发生器输出,48 脚芯片无此引脚。输出频率为振荡器频率的 1/3,占空比为 33%。
- $\overline{RESET}$,复位信号输入端。
- BUSWIDTH,总线宽度选择输入端,48 脚芯片无此引脚。如果芯片配置寄存器 CCR 的 D_1 位为“1”,则运行中总线宽度由 BUSWIDTH 的逻辑值决定,若 BUSWIDTH 引脚为高电平,选择 16 位总线,否则选择 8 位总线,如果 CCR 的 D_1 位为“0”,则总线宽度总是 8 位。
- NMI,非屏蔽中断,48 脚芯片无此引脚。此引脚有正跳变时,形成一个指向外部存储器 0000H 单元的中断矢量,而这部分外部存储器(00H~0FFH)保留给 Intel 开发系统使用。
- INST,取指令操作指示,48 脚芯片无此引脚。读外部存储器期间,此脚输出高电平表示是取指操作。整个总线周期内,INST 都有效。

$\overline{EA}$,存储器选择输入端。 $\overline{EA}=+5V$,CPU 寻址 2000H~3FFFH(对于 8×9×JF 为 2000H~5FFFH)的存储器时,访问的是片内 ROM/EPROM; $\overline{EA}=0V$ 时,则访问的是片外存

存储器; $\overline{EA}=+12.75V$ 时, 则可对内部有 EPROM 的芯片编程。此引脚内部有下拉作用, 若外部无驱动, 它总保持低电平。

- $ALE/\overline{ADV}$, 地址锁存允许(ALE)或地址有效输出($\overline{ADV}$)。由 CCR 寄存器选择。两者均提供了一个锁存信号, 以便把地址从地址/数据总线中分离出来。当选择 $\overline{ADV}$ 功能时, 在总线周期结束时, 此引脚变高。 $\overline{ADV}$ 可作为外部存储器的片选信号。该引脚仅在外存储器访问期间才有效。
- $\overline{RD}$, 对外部存储器的读信号(输出)
- $\overline{WR}/\overline{WRL}$, 对外部数据存储器或对外部数据存储器的低位字节的写操作输出信号, 由 CCR 寄存器选择。 8×98 对两种功能未加区分。选择 $\overline{WR}$ 功能时, 每次写外部存储器时, $\overline{WR}$ 都变低; 选 $\overline{WRL}$ 功能时, 只有写外部存储器的偶数字节(低位)时, $\overline{WRL}$ 才变低。
- $\overline{BHE}/\overline{WRH}$, 总线高位字节($\overline{BHE}$)或写外部存储器高位字节($\overline{WRH}$)的输出信号。 8×98 无此引脚, 两种功能由 CCR 寄存器选择。 $\overline{BHE}$ 为低电平, 选择连接至数据总线高位字节的存储器; 地址线 A_0 为低电平, 选择连接至数据总线低位字节的存储器。因此, 访问一个 16 位宽的存储器时, 若 $(A_0=0) \wedge (\overline{BHE}=1)$, 访问的是低位字节; 若 $(A_0=1) \wedge (\overline{BHE}=0)$, 访问的是高位字节; 若 $(A_0=0) \wedge (\overline{BHE}=0)$, 则同时访问 2 个字节。如果选择 $\overline{WRH}$ 功能, 当写存储器奇数字节(高位)时, 此引脚变低。
- **READY**, 准备就绪信号。输出引脚。用来延长对外部存储器的访问周期, 以使芯片能够与慢速或动态存储器相连; 它也可用于总线共享。总线周期最多可延长至 $1\mu S$ 。不使用外部存储器时, 此信号无效。通过 CCR 寄存器, 可控制插入到总线周期中的等待状态周期数。READY 引脚内部有微弱的上拉作用, 当无外部驱动时, 此引脚为高电平。
- **HSI**, 高速输入部件的信号输入端。共有四个 HSI 输入引脚: HSI. 0、HSI. 1、HSI. 2、HSI. 3, 其中两个引脚(HSI. 2、HSI. 3)与 HSO 部件共用。
- **HSO**, 高速输出部件的信号输出端。共有六个 HSO 输出引脚: HSO. 0、HSO. 1、HSO. 2、HSO. 3、HSO. 4、HSO. 5, 其中两个引脚(HSO. 4、HSO. 5)与 HSI 部件共用。
- **P0 口**, 8 位高输入阻抗输入口。这些引脚既可作为数字输入口, 也可用作 A/D 转换器的模拟输入口(AHC₄~ACH₇)。对内部含有 EPROM 的芯片编程时, 作为选择方式(PMOD. 0~PMOD. 3)的输入。48 脚芯片只有 4 位。
- **P1 口**, 8 位准双向 I/O 口。48 脚芯片无此口。
- **P2 口**, 8 位多功能口。除了可用作标准的 I/O 口外, P_{2.0}~P_{2.5} 还可用作其它特殊功能; P_{2.6}~P_{2.7} 为准双向口。EPROM 编程时也要用到这些引脚。48 脚芯片只有 4 位。
- **P3 口和 P4 口**, 具有漏极开路输出的 8 位双向口。这些引脚用作多路复用的地址/数据总线和地址总线, 引脚内部具有很强的上拉作用。EPROM 编程时, P3 和 P4 口在编程方式下, 用作命令、地址和数据的通路。
- **TXD/P_{2.0}**, 串行口发送端, 是 P_{2.0} 引脚的替换功能。此引脚还用作编程方式下的 PVER 或 SALE 端。
- **RXD/P_{2.1}**, 串行口接收端, 是 P_{2.1} 的替换功能。此引脚用作编程方式下的 PALE。

- EXTINT/P_{2.2}, 外部中断申请端, 是 P_{2.2} 的替换功能。此引脚还用作编程方式下的 $\overline{\text{PROG}}$ 。
- T₂CLK/P_{2.3}, 定时器 2 的外部时钟输入端, 是 P_{2.3} 的替换功能。
- T₂RST/P_{2.4}, 定时器 2 的复位端, 是 P_{2.4} 的替换功能。
- PWM/P_{2.5}, 脉宽调制器输出, 是 P_{2.5} 的替换功能。此引脚还用作编程方式下的 $\overline{\text{PDO}}$ 或 $\overline{\text{SPROG}}$ 。

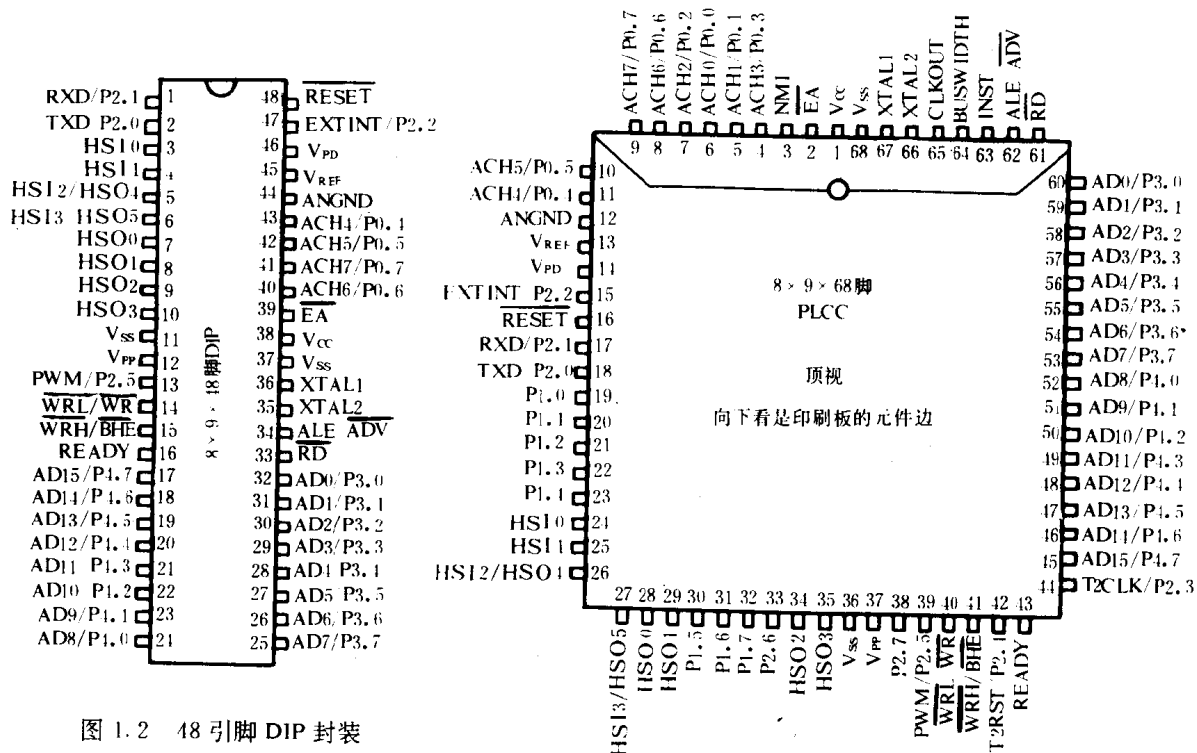


图 1.3 68 引脚 PLCC 封装

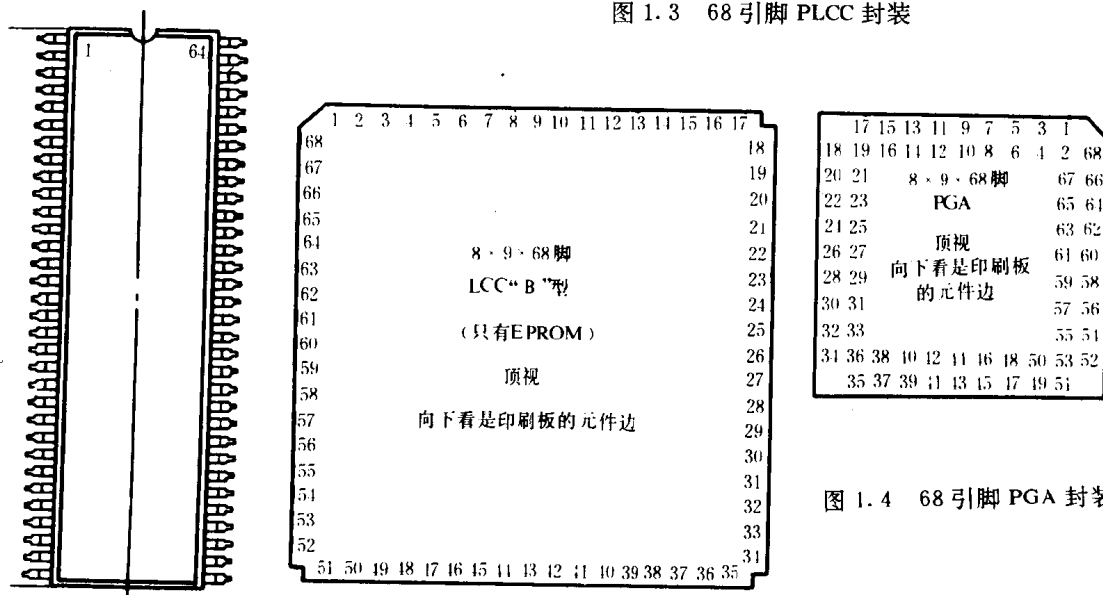


图 1.5 68 引脚 LCC 封装

图 1.6 64 引脚 SDIP 封装

表 1.3 各种封装引脚的对照表

名 称	68 脚 PLCC	68 脚 PGA	48 脚 DIP	64 脚 SDIP	名 称	68 脚 PLCC	68 脚 PGA	48 脚 DIP	64 脚 SDIP
ACH ₀ /P _{0.0}	6	4	—	4	P _{1.6}	31	47	—	29
ACH ₁ /P _{0.1}	5	5	—	3	P _{1.7}	32	46	—	30
ACH ₂ /P _{0.2}	7	3	—	5	P _{2.0} /TXD/PVER	18	60	2	16
AHC ₃ /P _{0.3}	4	6	—	2	P _{2.1} /RXD/PALE	17	61	1	15
ACH ₄ /P _{0.4} /MOD. 0	11	67	43	9	P _{2.2} /EXTINT	15	63	47	13
ACH ₅ /P _{0.5} /MOD. 1	10	68	42	8	P _{2.3} /T ₂ CLK	44	34	—	42
ACH ₆ /P _{0.6} /MOD. 2	8	2	40	6	P _{2.4} /T ₂ RST	42	36	—	40
ACH ₇ /P _{0.7} /MOD. 3	9	1	41	7	P _{2.5} /PWM/PDO	39	39	13	37
ALE/ADV	62	16	34	60	P _{2.6}	33	45	—	31
ANGND	12	66	44	10	P _{2.7}	38	40	—	36
BHE/WRH	41	37	15	39	P _{3.0} /AD0 PVAL	60	18	32	58
BUSWIDTH	64	14	—	—	P _{3.1} /AD1 PVAL	59	19	31	57
CLKOUT	65	13	—	—	P _{3.2} /AD2 PVAL	58	20	30	56
EA	2	8	39	1	P _{3.3} /AD3 PVAL	57	21	29	55
EXTINT/P _{2.2} /PROG	15	63	47	13	P _{3.4} /AD4 PVAL	56	22	28	54
HSI. 0	24	54	3	22	P _{3.5} /AD5 PVAL	55	23	27	53
HSI. 1	25	53	4	23	P _{3.6} /AD6 PVAL	54	24	26	52
HSI. 2/HSO. 4	26	52	5	24	P _{3.7} /AD7 PVAL	53	25	25	51
HSI. 3/HSO. 5	27	51	6	25	P _{4.0} /AD8 PVAL	52	26	24	50
HSO. 0	28	50	7	26	P _{4.1} /AD9 PVAL	51	27	23	49
HSO. 1	29	49	8	27	P _{4.2} /AD10 PVAL	50	28	22	48
HSO. 2	34	44	9	32	P _{4.3} /AD11 PVAL	49	29	21	47
HSO. 3	35	43	10	33	P _{4.4} /AD12 PVAL	48	30	20	46
HSO. 4/HSI. 2	26	52	5	24	P _{4.5} /AD13 PVAL	47	31	19	45
HSO. 5/HSI. 3	27	51	6	25	P _{4.6} /AD14 PVAL	46	32	18	44
INST	63	15	—	—	P _{4.7} /AD15 PVAL	45	33	17	43
NMI	3	7	—	—	RD	61	17	33	59
PWM/P _{2.5} /PDO	39	39	13	37	READY	43	35	16	41
PALE/P _{2.1} /RXD	17	61	1	15	RESET	16	62	48	14
PROG/P _{2.2} /EXTINT	15	63	47	13	RXD/P _{2.1}	17	61	1	15
PVER/P _{2.0} /TXD	18	60	2	16	SALE/PVER/P _{2.0}	18	60	2	16
P _{0.0} /ACH ₀	6	4	—	4	SPROG/PDO/P _{2.5}	39	39	13	37
P _{0.1} /ACH ₁	5	5	—	3	TXD/P _{2.0} /SALE	18	60	2	16
P _{0.2} /ACH ₂	7	3	—	5	T ₂ CLK/P _{2.3}	44	34	—	42
P _{0.3} /ACH ₃	4	6	—	2	T ₂ RST/P _{2.4}	42	36	—	40
P _{0.4} /ACH ₄ /MOD. 0	11	67	43	9	V _{PP}	37	41	12	35
P _{0.5} /ACH ₅ /MOD. 1	10	68	42	8	V _{CC}	1	9	38	64
P _{0.6} /ACH ₆ /MOD. 2	8	2	40	6	V _{PD}	14	64	46	12
P _{0.7} /ACH ₇ /MOD. 3	9	1	41	7	V _{REF}	13	65	45	11
P _{1.0}	19	59	—	17	V _{SS}	68	10	11	34
P _{1.1}	20	58	—	18	V _{SS}	36	42	37	63
P _{1.2}	21	57	—	19	WR/WRL	40	38	14	38
P _{1.3}	22	56	—	20	WRH/BHE	41	37	15	39
P _{1.4}	23	55	—	21	XTAL ₁	67	11	36	62
P _{1.5}	30	48	—	28	XTAL ₂	66	12	35	61

1.2 CPU 操作

8×9×CPU 主要由寄存器阵列和寄存器算术逻辑单元(RALU)组成。CPU 通过特殊功能寄存器(SFR)或存储器控制器与外界进行通信。其 RALU 没有采用 CPU 中通常使用的累加器结构,而是直接在 256 个字节的寄存器空间(由 24 个字节的 SFR 和 232 个字节的寄存器阵列组成)内进行操作。这些寄存器都具有累加器的功能,可使 CPU 对运算前后的数据进行迅速变换,同时又提供了高速的数据处理和频繁的输入/输出能力,因而不存在以往累加器结构的“瓶颈现象”。此外,通过 SFR 还可以直接控制 I/O、A/D、PWM、串行口等部件,使之高效率运行。

一、CPU 总线

CPU 内部的“控制单元”用两条总线将寄存器阵列和 RALU 连接起来。图 1.1 中给出了 CPU 与两条总线的连接情况。两条总线是 8 位地址总线(A-BUS)和 16 位数据总线(D-BUS)。D 总线用于 RALU 同寄存器阵列或 SFR 之间的数据传送,A 总线则作为上述数据传送的地址总线或用作与存储控制器(MEMORY CONTROLLER)连接的多路复用地址/数据总线。无论是对片内 ROM 还是对片外存储器,CPU 都是通过存储控制器进行访问的。

二、CPU 寄存器阵列

寄存器阵列(REGISTER FILE)共有 232 个字节的 RAM 单元,它们可按字节、字或双字存取。由于上述任何一个存储单元都可以被 RALU 使用,这就好象 CPU 有 232 个累加器,使用非常灵活方便。需要指出的是,寄存器阵列中的第一个字单元是专门留作堆栈指针使用的,因此,软件涉及到堆栈操作时,不能用该单元存放数据。访问寄存器阵列和 SFR 的地址由 CPU 硬件控制,该地址暂存在两个 8 位地址寄存器内。

三、寄存器算术逻辑单元 RALU

8×9×的大多数运算由 RALU 完成,其结构如图 1.7 所示。

RALU 由一个 17 位的算术逻辑单元(ALU)、程序状态字(PSW)的高位字节、程序计数器(PC)、循环计数器(LOOP COUNTER)以及三个暂存寄存器(TEMPORARY REGISTER、UPPER WORD REGISTER/SHIFTER、LOWER WORD REGISTER/SHIFTER)组成。上述所有寄存器都是 16 位或 17 位(16 位加符号扩展位)的,此外,有的寄存器还能脱离 ALU 单独进行一些简单的操作。程序计数器还增设了增量器(INCREMENTOR),顺序执行指令时,程序计数器的值靠增量器修改,但是,程序的转移还需 ALU 控制。两个暂存寄存器本身设有移位逻辑,可用于需要逻辑移位的一些操作,如规格化、乘、除等操作。低字寄存器仅在双字长数据移位时才使用,而高字寄存器,则每逢执行移位或作为指令暂存寄存器时,就要使用。双操作数指令中,第二个操作数存放在 TEMPORARY REGISTER 中,包括乘法中的乘数和除法中的除数。作减法时,该寄存器的输出取补后,再送到 ALU 的 B 端。重复移位的次数,由 5 位循环计数器完成计数。

图 1.7 延时(DELAY)电路用来将 16 位总线转换成 8 位总线形式,所有地址和指令都需变为 8 位后才能送到 A-BUS 上。存放在 RALU 中的几个常数(如 0、1、2),用来加快某些运算的速度,例如地址自动增量、求 2 的补码、执行加 1 或减 1 指令等。