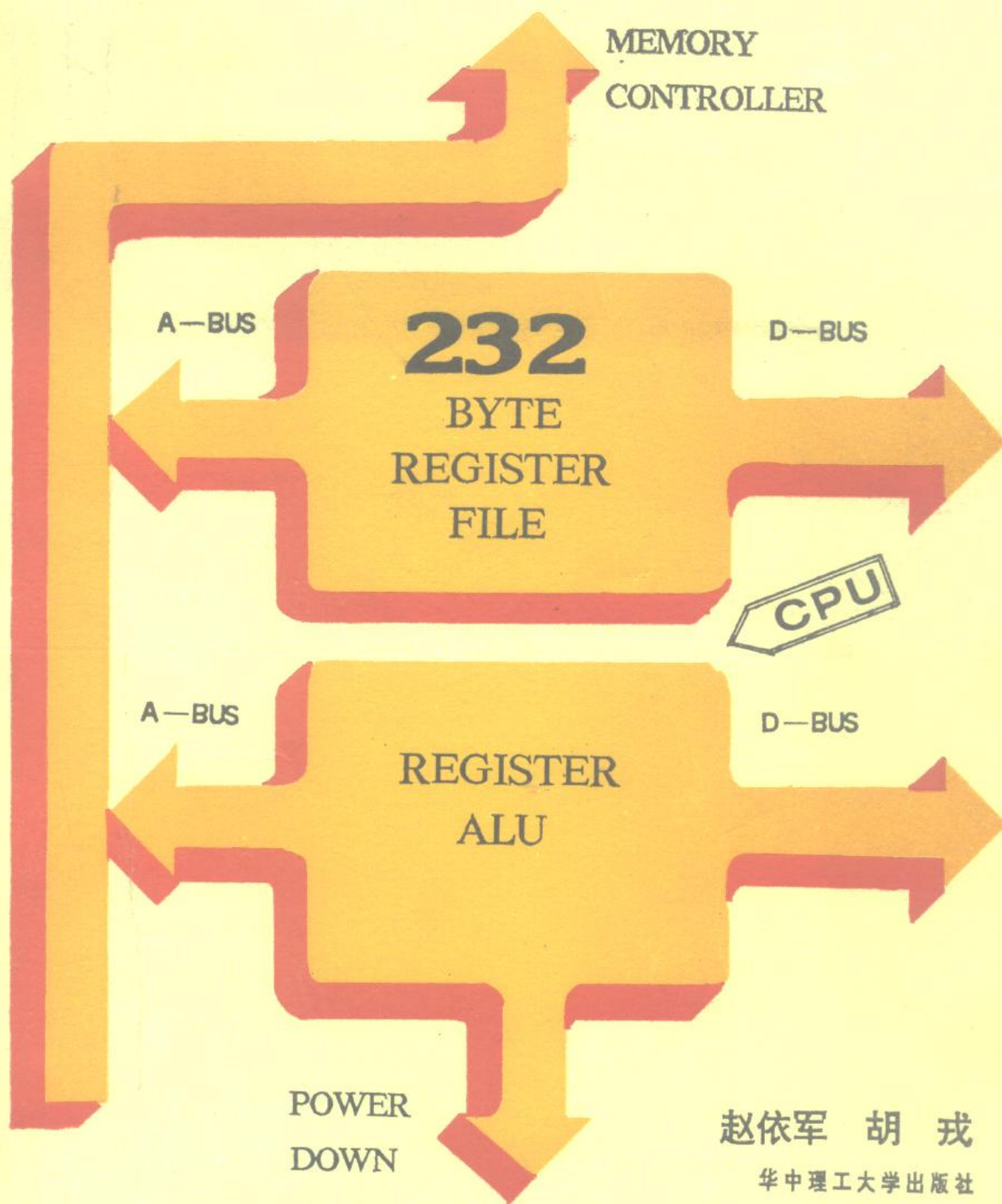




赵依军 胡 戎

华中理工大学出版社

8098

单片机原理及 系统设计技术

7316
=YD/1

8098 单片机原理及系统设计技术

赵依军 胡 戎

华中理工大学出版社

1990年3月·武汉

0021463

内 容 提 要

8098/8398/8798 是 Intel 公司于 1988 年第一季度新推出的准 16 位嵌入式微控制器。本书较为深入细致地介绍了 8098 的构成、CPU 操作、I/O 口、软件设计基础、中断系统、定时器、高速输入 HSI、高速输出 HSO、A/D、PWM(D/A)以及串行口原理及应用。在介绍原理的同时,结合作者的研究实践,给出了许多实用例子。本书的最后两章分别介绍了 8098 基本硬件系统设计和 8098 的综合应用。

该书对于从事单片机应用的工程技术人员来说,是一本不可多得的技术指导书。

它也适用于科研人员和大专院校师生。

JS302/50

8098 单片机原理及系统设计技术

编 者:赵依军 胡 戎

责任编辑:陈少华

出版者:华中理工大学出版社

发 行 者:华中理工大学出版社

经 销 者:新华书店湖北发行所

印 刷 者:华中理工大学出版社印刷厂

开 本:787×1092 1/16

印 张:16.5

字 数:423 千字

版 次:1990 年 3 月第 1 版

印 次:1991 年 9 月第 2 次印刷

书 号:ISBN7-5609-0463-7/TP·40

印 数:3 001—4 000

定 价:9.50 元

前 言

近几年来,单片机技术在我国推广和应用取得了较大的进展,8位单片机(主流机型为MCS-51)在工业自动化、智能仪器仪表、数据采集和处理、通信系统、家用电器等领域的应用日益普及;与此同时,新一代的16位单片机(主流机型为MCS-96系列)也开始受到广大科技人员的关注。尽管Intel的16位单片机8096前几年已被引进到国内,但由于其价格昂贵、且外部总线为16位,因而它在国内的推广和使用受到了限制。

1988年第1季度,Intel公司首次推出了高性能的“准16位”单片机8098。与8096相比,8098具有下述显著特点:(1)采用廉价的48脚双列直插式封装;(2)指令系统与8096完全兼容;(3)硬件上不仅保留了8096的特性,而且片内的10位A/D转换器还增加了采样保持电路;(4)外部数据总线为8位,因而易于与众多的8位外围器件接口;(5)价格低廉,人民币售价100元左右。由此可见,8098在我国计算机应用领域有着广泛的应用前景。

本书全面而又系统地介绍了8098的硬件结构、指令系统以及系统设计方法。全书共分十章,内容包括:8098结构综述、8098软件设计基础、中断系统及其应用、定时器/计数器及其应用、高速输入(HSI)及其应用、高速输出(HSO)及其应用、串行口及其应用、A/D和PWM(D/A)及其应用、8098基本硬件系统设计、8098综合设计实例。附录还给出了8098原文数据资料。

本书可作为《单片微机接口技术》一书的姊妹篇。《单片微机接口技术》是专门介绍8位单片机系统设计的,全书侧重硬件电路的设计。由于8098的外部总线为8位,因此,《单片微机接口技术》书中所介绍的硬件设计方法对8098系统同样适用。正因为如此,《8098原理及系统设计技术》主要介绍8098芯片本身如何应用,侧重点在软件设计上,同时,对《单片微机接口技术》中未介绍的一些新型芯片的应用(例如带有锁存器的EPROM、多功能端口扩展器87C75PF等)也作了适当补充。

本书系统性、先进性、实用性较强,可作为有一定微机基础、从事单片机设计及应用科技人员的参考书,也可作为大专院校及学习班的教学参考书或教材。

空军雷达学院院长王仕俊教授详细审阅了全书,陆绍雄付教授审校了全部书稿,他们提出了许多宝贵的修改意见,对此,我们表示衷心的感谢。Intel中国公司的朱旭东先生,航空航天工业部502所黎庆生高级工程师、王磊高级工程师,北方交通大学计算机系钟天戈教师,对本书的面世给予了大力的支持和帮助,在此一并致谢。

由于编者水平有限,加之时间仓促,书中难免存在一些缺点和错误,恳请广大读者来信批评斧正。

编 者

于武汉 空军雷达学院

1990年2月

目 录

第一章 8098 结构综述

第一节 8098 基本构成	1
一、芯片组成	1
二、引脚功能	1
第二节 CPU 操作	3
一、CPU 总线	3
二、CPU 寄存器阵列	3
三、寄存器算术逻辑单元 RALU	3
四、内部定时	4
第三节 存储器空间	5
一、寄存器阵列	5
二、特殊功能寄存器 SFR	6
三、掉电	7
四、保留的存储器空间	7
五、内部 ROM/EPROM	7
六、存储器控制器	8
七、系统总线	8
第四节 I/O 口	12
一、P0 口	12
二、P2 口	12
三、P3 口和 P4 口	13
四、I/O 口控制和状态寄存器	13
第五节 复位	14
一、复位信号	14
二、复位状态	14
三、复位电路	15

第二章 8098 软件设计基础

第一节 编程须知	18
一、操作数类型	18
二、寻址方式	19
三、程序状态字(PSW)	21
第二节 8098 指令系统详述	23
一、概述	23
二、8098 汇编语言指令和机器码指令格式	23
三、8098 指令系统	26

第三章 中断系统及其应用

第一节 8098 中断结构	64
一、中断源	64
二、与中断有关的寄存器	66
三、中断处理过程	66
四、中断优先级的控制	68
五、中断响应时间	69
第二节 中断系统编程实例	70
一、修改堆栈以实现改变中断服务程序的返回地址	70
二、用软件产生 EXTINT 和 HSI. 0 中断	71

第四章 定时器及其应用

第一节 8098 定时器概述	73
一、定时器 T1	73
二、定时器 T2	73
三、定时器中断	73
四、监督定时器	74
第二节 定时器应用实例	75
一、定时器 T1 的应用	75
二、定时器 T2 的应用	76
三、监督定时器的应用	78

第五章 高速输入 HSI 及其应用

第一节 HSI 的工作原理	79
一、HSI 的基本结构	79
二、HSI 的控制	80
三、HSI 中断	81
第二节 HSI 的使用方法	82
一、HSI 使用要点	82
二、查询法	82
三、中断法	83
第三节 HSI 应用实例	85
一、利用 HSI 测量脉冲信号	85
二、利用 HSI 测量转速	87
三、使用 HSI 测距	89

第六章 高速输出 HSO 及其应用

第一节 HSO 工作原理	92
一、HSO 的基本结构	92
二、HSO 的控制	93
三、HSO 的清除	94
四、采用定时器 T2 作 HSO 时基	95

五、软件定时器	95
六、HSO 中断	95
第二节 HSO 的使用方法	96
一、HSO 使用要点	96
二、HSO 的编程方法	96
第三节 HSO 应用实例	96
一、利用 HSO 产生脉冲	96
二、利用 HSO 产生 PWM 信号	99
三、用 HSO 驱动步进电机	102
四、用 HSO 实现高速可编程控制器	104
五、HSO 的其他应用	106

第七章 串行口及其应用

第一节 串行口的工作原理	111
一、串行口的工作方式	111
二、串行口的控制	113
第二节 串行口的使用方法	115
一、串行口的使用要点	115
二、串行口的编程方式	115
第三节 串行口应用实例	115
一、方式 0 的应用	115
二、方式 1 的应用	118
三、方式 2 和方式 3 的应用	121
第四节 利用 HSI 和 HSO 构成软件串行口	130

第八章 A/D 和 PWM(D/A)及其应用

第一节 A/D 转换器	141
一、A/D 转换器原理及其性能的提高	141
二、A/D 转换器的控制	144
三、A/D 转换器应用实例	146
四、利用 8098 构成温度控制器	147
第二节 PWM 输出(D/A)	148
一、PWM 结构和工作原理	148
二、PWM(D/A)应用实例	149

第九章 8098 基本硬件系统设计

第一节 典型的 8098 硬件系统	154
一、最简单的 8098 硬件系统	154
二、外接 EPROM 系统	155
三、外接 EPROM 和 RAM 系统	155
第二节 扩展 Latched EPROM	156
一、概述	156
二、Latched EPROM 的工作原理	156

三、可锁存 EPROM 系列	158
四、可锁存 EPROM 与 8098 的接口	160
第三节 扩展 87C75PF 端口扩展器	161
一、87C75PF 设计原理	161
二、87C75PF 的结构特点	163
三、87C75PF 的工作特性	164
四、87C75PF 的应用实例	172
第十章 8098 综合设计实例	
第一节 基于 8098 的分布式控制系统设计	174
一、单片机分布式控制系统的设计方法	174
二、单片机分布式控制系统的组成及通信方法	175
三、控制程序	177
第二节 智能探鱼器的设计实例	178
一、系统组成	178
二、主要外围硬件电路设计	180
三、声纳控制软件	182
第三节 低成本 8098 仿真器	183
一、概述	183
二、8098 仿真器硬件结构	183
三、存贮空间分配	185
四、系统软件结构	185
五、单步和断点功能的实现	185
第四节 电机运行控制的程序设计	186
一、引言	186
二、电机运行控制	187
附录 8098 原文数据资料	201
主要参考文献	257

第一章 8098 结构综述

第一节 8098 基本构成

8098/8398/8798 是 Intel 公司于 1988 年第一季度最新推出的准 16 位嵌入式微控制器(内部数据总线为 16 位,外部数据总线为 8 位)。与 MCS-96 系列中的其他芯片相比,它具有性能高、功能全、售价低廉、使用方便(48PIN DIP)等优点,因而非常适合在我国推广使用。

一、芯片组成

8098 组成框图如图 1.1 所示,它主要由一个高性能的 16 位 CPU 寄存器算术逻辑单元(RALU)、232 字节寄存器阵列以及一些外围子系统构成。在 RALU 控制下的外设子系统可分为以下一些部分: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、中断控制器和等待状态产生逻辑、一个同步/异步串行口、两个 16 位定时器和一个监督定时器、一个可供 D/A 转换器使用的脉冲宽度调制(PWM)输出。此外,片内还设有时钟脉冲发生器以支持整个芯片的正常运行。

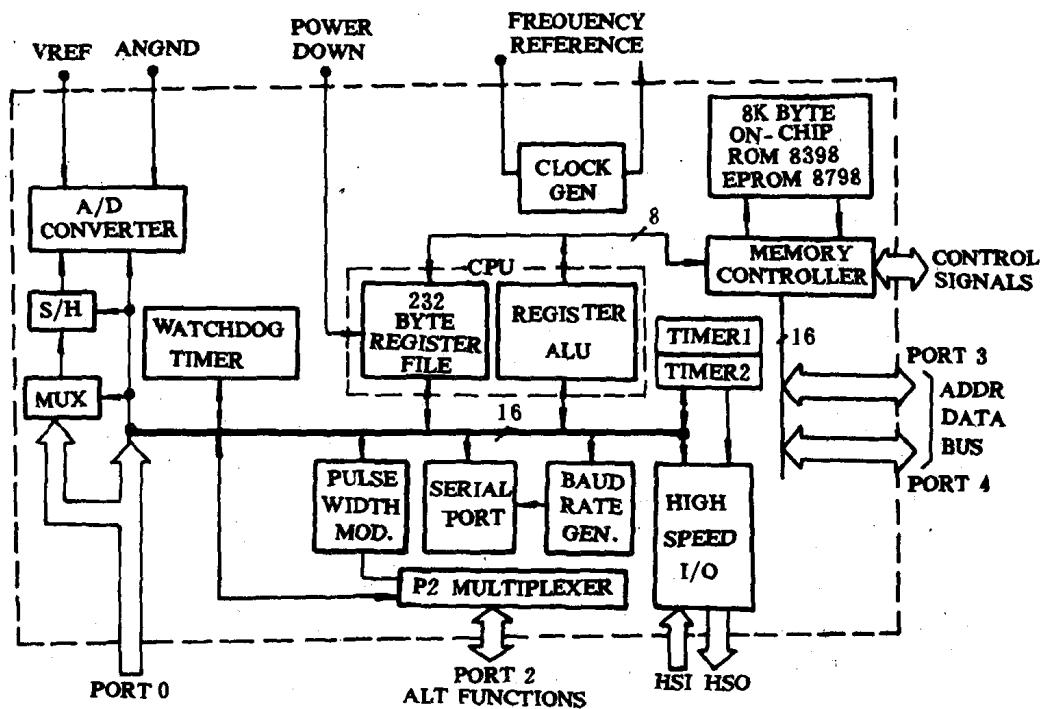


图 1.1 8098 组成框图

二、引脚功能

8098/8398/8798 均采用 48 引脚双列直插式封装,其引脚配置如图 1.2 所示。其中 8398 提供了 8K 字节的可加密 ROM,8798 提供了 8K 字节的可加密 EPROM。

8098 各引脚功能如下:

V_{CC} ——主电源(+5V)。

V_{SS} ——数字地(0V)。共有两个 V_{SS} 引脚,它们都必须接地。

V_{PD} ——RAM 备用电源(+5V)。在正常操作期间应加上此电源。在掉电情况下,若在 V_{CC} 下降到低于规定值之前(V_{PD} 应保持在规定范围之内)使 $\overline{RESET}$ 有效,则 8098 片内寄存器阵列顶部的 16 字节内容将保持不变,在掉电期间, $\overline{RESET}$ 应一直保持低电平,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,通常应与 V_{SS} 同电位。

V_{PP} ——EPROM 型芯片的编程电压。

XTAL1——片内振荡器中反相器的输入,也是片内时钟发生器的输入。通常接外部晶体。

XTAL2——片内振荡器中反相器的输出。通常接外部晶体。

$\overline{RESET}$ ——复位信号输入端。

$\overline{EA}$ ——存储器选择输入端。 $\overline{EA}=1$,CPU 寻址存储器 2000H~3FFFH 单元时,访问的是片内 ROM/EPROM; $\overline{EA}=0$,则访问的是片外存储器。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($\overline{ADV}$),它们由 CCR 寄存器选择。两者都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。ALE/ $\overline{ADV}$ 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)。

$\overline{WR}$ ——对外部存储器的写信号(输出)。

READY——准备就绪信号(输出)。用来延长对外部存储器的访问周期,以便使芯片能够与慢速或动态存储器接口;它也可用于总线共享。总线周期最多可延长至 1 μ s。当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——4 位高输入阻抗口。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(ACH4~ACH7)。

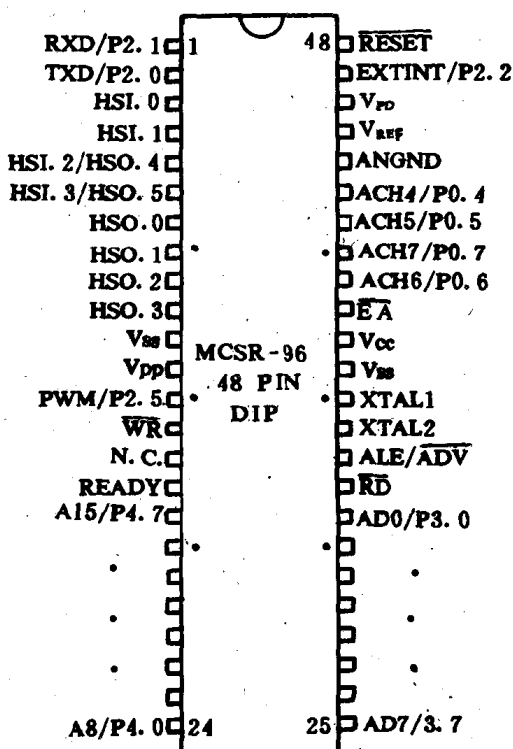


图 1.2 8098 引脚配置

P2 口——4 位多功能口。在 8098 中,它们除了可用作标准的 I/O 口外,还可用作其它特殊功能。

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地址/数据总线和地址总线,它们的引脚内部具有很强的上拉作用。

第二节 CPU 操作

8098 CPU 中的主要元件有:高速寄存器阵列、特殊功能寄存器(SFR)、存储器控制器和寄存器算术逻辑单元(RALU)。CPU 与外界通信是通过特殊功能寄存器 SFR 或存储器控制器进行的。8098 CPU 的主要特色是其 RALU 未采用其他类型 CPU 中常规的累加器结构,而是直接在由寄存器阵列和 SFR 所构成的 256 字节寄存器空间内进行操作。这些寄存器都具有累加器的特殊功能,它们可使 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能力和频繁的输入/输出访问能力,因此不存在使用通常的累加器结构时所出现的“瓶颈现象”。此外,通过 SFR 还可以直接控制 I/O、A/D、PWM、串行口等部件的有效运行。

一、CPU 总线

CPU 内部的一个控制单元和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。这两条总线是:8 位地址总线(A-BUS)和 16 位数据总线(D-BUS 总线)。D 总线仅在 RALU 与寄存器阵列或 SFR 之间传送数据,A 总线则用作上述数据传送的地址总线或用作与存储器控制器连接的多路复用地址/数据总线。CPU 无论是对片内 ROM 还是对片外存储器的访问都是通过存储器控制器进行的。

二、CPU 寄存器阵列

寄存器阵列共有 232 字节 RAM 单元,这些单元可按字节、字或双字存取。由于上述存储单元中的任何一个都可以为 RALU 所用,这就好象 CPU 有 232 个累加器一样,使用非常方便。需要指出的是,在寄存器阵列中的第一个字是专门留作堆栈指针使用的,因此,当需涉及到堆栈操作时,它不能用来存放数据。访问寄存器阵列和 SFR 的地址由 CPU 硬件控制,它们暂存在两个 8 位地址寄存器内。

三、寄存器算术逻辑单元 RALU

8098 的大多数运算都是由 RALU 完成的,RALU 的结构如图 1.3 所示。

RALU 包括一个 17 位的算术逻辑单元(ALU)、程序状态字(PSW)、程序计数器(PC)、循环计数器(LOOP COUNTER)以及三个暂存器(TEMPORARY REGISTER)。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 而单独进行一些简单的操作,例如,图 1.3 中为程序计数器而设置的增量器(INCREMENTOR)。然而,程序的转移必须由 ALU 控制。两个暂存器本身设有移位逻辑,可用于需要进行逻辑移位的一些操作中。例如规格化、乘、除等操作。低字寄存器(LOWER WORD REGISTER)仅在双字长数据移位时才使用,而高字寄存器(UPPER WORD REGISTER)则每逢执行移位或作为指令暂存器时就要使用。当执行循环移位操作时,由 5 位循环计数器完成循环计数。

图中延迟(DELAY)电路用以将 16 位总线转换成 8 位总线,当所有地址和指令送到 8 位地址总线上时,都需要通过此电路。此外,几个常数(如 0、1、2)被存放在 RALU 中,用以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。

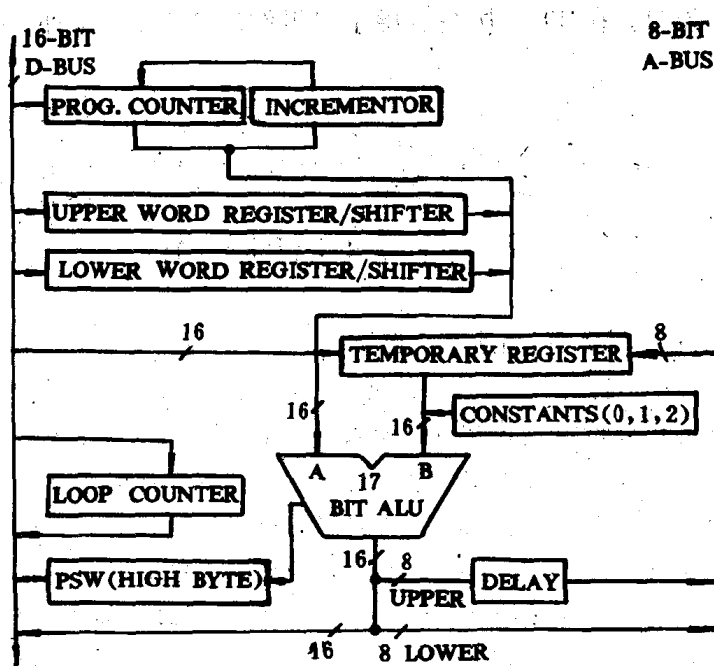


图 1.3 RALU 方框图

四、内部定时

8098 工作时所需的时钟可通过其 XTAL1 输入引脚从外部送入,也可采用芯片内部的振荡器。8098 的工作频率为 6MHz~12MHz。

8098 片内的振荡电路是一个单级非门电路,它与石英晶体配合使用,可组成一个稳定的晶体振荡器,如图 1.4 所示。图中的外接电容器 C_1 、 C_2 通常取为 30pF 左右。

来自晶体振荡器或外部振荡电路的信号经过三分频电路,便产生了如图 1.5 所示的三个不同相位的内部时钟。三个振荡周期构成一个状态周期 T (当 $F_{osc}=12\text{MHz}$ 时, $T_{osc}=83\text{ns}$, $T=250\text{ns}$),它是 8098 操作的基本时间单位,图 1.5 中 A、B、C 各相的占空比均为 33%,8098 的内部操作大多数都与上述三者之一同步。

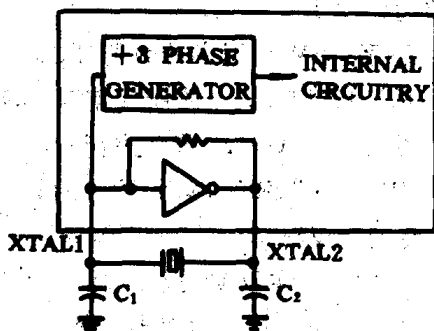


图 1.4 8098 晶体振荡器方框图

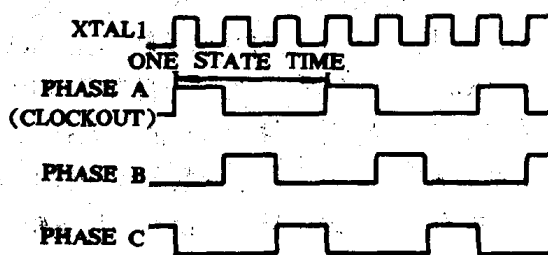


图 1.5 8098 内部定时波形图

第三节 存贮器空间

8098 可寻址外部存贮器空间为 64K 字节,它采用的是程序存贮器与数据存贮器合二为一的普林斯顿结构,因而不象采用哈佛结构的 MCS-48 和 MCS-51 系列那样有程序存贮器空间和数据存贮器空间之分。在 64K 字节的存贮器空间中,地址 0000H~00FFH 和 1FFFH~2080H 的存贮单元具有特殊用途,而其他的存贮空间既可用于存放程序或数据,也可用作 I/O 外设映象的存贮区。8098 的存贮器空间分布如图 1.6 所示。

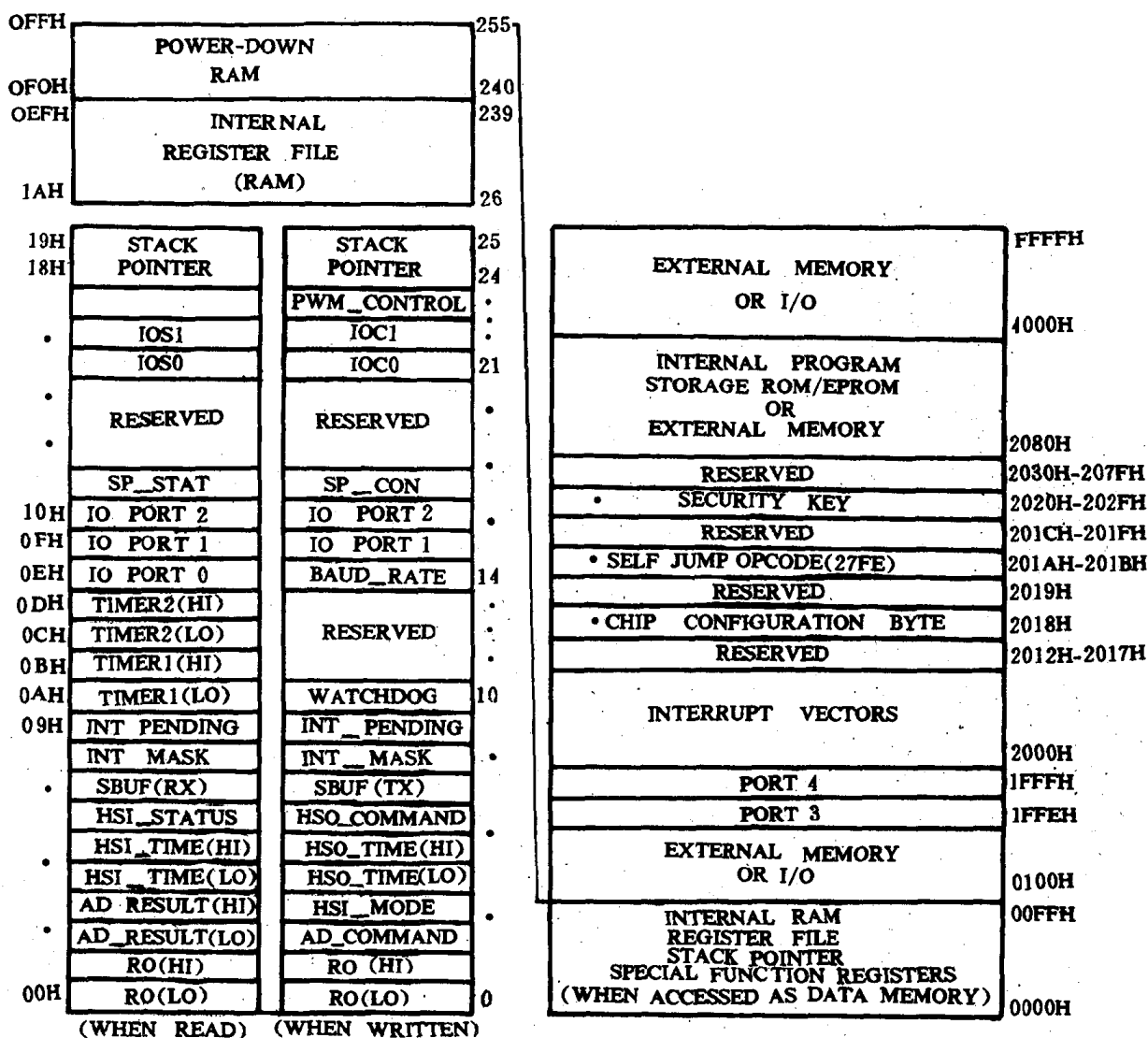


图 1.6 8098 存储器空间

一、寄存器阵列

00H~0FFH 存贮单元包括寄存器阵列和 SFR,在这部分内部 RAM 空间中,不能执行指令码,如果试图执行 00H~0FFH 存贮单元中的程序,只有将指令码存放在上述地址空间所覆盖的外部存贮器中。通常,外存的这一区域是为 Intel 开发系统所保留的。

RALU 能对 256 个内部寄存器单元中的任何一个进行操作,其中 00H~17H 这 24 个单元

用作特殊功能寄存器(SFR),18H和19H这两个单元则作为堆栈地址指针。如果不涉及到堆栈操作,则这两个单元也可作为普通的RAM使用。堆栈指针可由用户程序在初始化时设置,它可指向64K存储空间中的任何单元。除上述26个单元以外,剩下的230个单元作为用户的通用寄存器使用。

二、特殊功能寄存器 SFR

8098内部的各个功能部件均通过SFR进行控制。大多数SFR具有两种功能:读操作时为一种用途,写操作时为另一种用途。图1.6中给出了所有SFR的地址和名称,对图中标有“RESERVED”(保留)字样的寄存器用于未来器件的扩充和测试,用户不可对其进行读出和写入操作。

所有SFR的名称和用途如下:

R0——零寄存器,它的读出值总是零,在零寄存器寻址(变址寻址的一种)操作时,用作零基址;在计算和比较操作中,它用作常数。

AD_RESULT——A/D变换结果寄存器(高位/低位),用以存放A/D转换结束时的高位和低位转换结果。它只能按字节读出。

AD_COMMAND——A/D命令寄存器,用于控制A/D转换器的工作。

HSI_MODE——HSI方式寄存器,用来设置高速输入部件的工作方式。

HSI_TIME——HSI时间寄存器(高位/低位),存放触发高速输入部件的时间值。它只能按字读出。

HSO_TIME——HSO时间寄存器(高位/低位),存放高速输出部件的时间或计数值,以执行HSO命令寄存器中的命令。它只能按字写入。

HSO_COMMAND——HSO命令寄存器,它决定在HSO时间寄存器中的时间值所确定的时刻将发生什么事件。

HSI_STATUS——HSI状态寄存器,它指出HSI引脚的状态:即在HSI时间寄存器所记载的时刻,哪些引脚产生了事件以及HSI引脚当前的状态。

SBUF(tx)——串行口发送缓冲器,用于存放欲发送的字节(信息)。

SBUF(rx)——串行口接收缓冲器,存放刚从串行口接收到的字节(信息)。

INT_MASK——中断屏蔽寄存器。

INT_PENDING——中断悬挂寄存器,指示各中断源是否已产生了中断信号。

WATCHDOG——监督定时器寄存器,此寄存器被启动后,至少每隔64K状态周期,软件便要使它复位一次。若因故障未及时使它复位,则系统自动产生复位信号,从而使之脱离故障状态。

TIMER1——定时器1(高位/低位),只能按字读出。

TIMER2——定时器2(高位/低位),只能按字读出。

IOPORT0——P0口寄存器,指示P0口各引脚上的电平。

BAUD_RATE——波特率寄存器,用于存放串行口波特率。

IOPORT1——P1口寄存器,用于对P1口的读/写。

IOPORT2——P2口寄存器,用于对P2口的读/写。

SP_STAT——串行口状态寄存器,用于指示串行口的工作状态。

SP_CON——串行口控制寄存器,用于设置串行口的工作方式。

IOS1——I/O状态寄存器1,用于存放定时器以及HSI的状态信息。

IOC0——I/O 控制寄存器,用于控制 HSI 引脚的复用功能以及定时器的复位源和时钟源。

IOC1——I/O 控制寄存器,用于控制 P2 口引脚的复用功能以及定时器和 HSI 的中断。

PWM_CONTROL——脉宽调制控制寄存器,用于设置 PWM 脉冲的持续时间。

对上述 SFR 的操作应注意以下几点:

1. TIMER1、TIMER2 和 HSI_TIME 寄存器只能按字读出,对它们不能写或按字节读。
2. HSO_TIME 是只能按字写入的寄存器,对它不能读或按字节写。
3. R0 是按字或字节读写的寄存器,但写 R0 并不会改变它的值。
4. 所有其它 SFR 只能按字节访问。
5. 乘法和除法指令中的源操作数都不能是可写的 SFR。
6. 对于变址或间接寻址指令,所有 SFR 都不能作变址或间址寄存器使用。

三、掉电

8098 片内 RAM 空间顶部的 16 字节 RAM(0F0H~0FFH)由 V_{PD} 引脚供电。如果用户希望在掉电后保存上述 16 个单元的内容,那么掉电后 V_{PD} 引脚必须加接备用电源,并且此电压必须保持在规定的范围以内。8098 进入掉电状态时,RESET 引脚电平降低,经 2 个状态周期后,芯片进入复位状态,以防止掉电时出现写 RAM 的操作。此时电源由 V_{CC} 供给改为由 V_{PD} 供给,只要 V_{PD} 引脚上接有规定值的电源,上述 16 字节 RAM 中的内容就可以保持。

8098 脱离掉电状态时, V_{CC} 升至正常值,此时 RESET 仍保持低电平。当片内的振荡器起振时,8098 必须保持稳定供电,经 2 个状态周期后,振荡器稳定工作,RESET 引脚恢复到高电平。经 10 个状态周期后,8098 开始执行存放在 2080 单元的指令码。图 1.7 给出了掉电时序图。

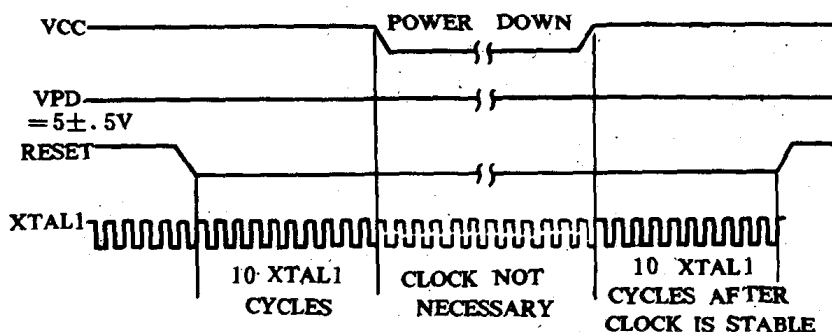


图 1.7 8098 掉电时序图

四、保留的存储器空间

表 1.1 列出了一些特殊存储单元地址表,标有“保留”字样的存储单元被 Intel 公司保留,用于测试或待开发的新产品中。这些单元必须被置为 0FFH 以确保现有芯片与新产品兼容。

对 8098 复位将使其从 2080H 单元中开始取指令,选择此地址的目的是为了使系统将来能够与多达 8K 字节的 RAM 和寄存器阵列相配接。

五、内部 ROM/EPROM

8398/8798 片内含有 8K 字节的 ROM/EPROM,地址范围为 2080H~3FFFH。芯片规定中断矢量、芯片配置寄存器(CCR)和保密字等均在 2000H~202FH 范围内。

只有当 $\overline{EA}$ 引脚接高电平并且地址在 2000H~3FFFH 之间时 8098/8798 才能从片内的 ROM/EPROM 中读取指令和数据。在其他情况下,8398/8798 均从外部存储器或内部 RAM 中读取指令或数据。

表 1.1 特殊存贮单元地址表

地 址	内 容
0000H~0017H	特殊功能寄存器(SFR)
0018H~0019H	堆栈指针(SP)
1FFEh~1FFFh	P3 口和 P4 口
2000H~2011H	中断向量
2012H~2017H	保留
2018H	芯片配置字节
2019H	保留
201AH~201BH	“跳转到自测试”操作码(27H FEH)
201CH~201FH	保留
2020H~202FH	保密代码
2030H~207FH	保留
2080H	复位地址单元

六、存贮器控制器

RALU 除了与片内 RAM 交换信息外,在与其他存贮器打交道时,都需通过存贮器控制器,它通过 A 总线和几条控制线与 RALU 相连。由于 A 总线是 8 位的,因此,为了避免频繁地从 RALU 取出指令单元的地址,加速取指速度,控制器中设有一个辅助程序计数器(PC),当顺序执行指令时,每次取指后,辅助 PC 值自动增加,只有执行跳转指令或子程序调用指令时,才把 RALU 中主 PC 的内容通过 A 总线加载到辅助 PC 中去。

除此而外,存贮器控制器中还有一个 4 字节的寄存器阵列,以加速执行过程,除在外总线周期期间强行插入等待状态外,此 4 字节寄存器阵列对 RALU 和用户来说都是随时可用的。

七、系统总线

8098 有多种运行方式,最常用的是标准总线方式。它使用 16 位多路复用地址和 8 位数据总线。此外,配合总线口进行总线存取操作的还有若干条控制信号。在标准方式下,CPU 通过与 8 位数据总线复用的 16 位地址总线 AD0~AD15 来对外部存贮器或 I/O 进行寻址,为了将总线上的地址和数据信号分离,8098 提供了地址锁存允许信号(ALE),利用其信号的下降沿给一个锁存器(例如 74LS373)送一个信号,用以锁存来自 AD0~AD7 的低 8 位地址。为了避免在解释存贮器系统时出现混乱,我们将 MA0~MA15 命名为地址信号,MA0~MD7 命名为数据信号。

(一)定时

图 1.8 给出了 8098 外部存贮器操作的理想定时波形。例如,当对外部存贮器取指时,ALE 变高,16 位地址从 AD0~AD15 引脚送出,其后 ALE 变低,当 $\overline{RD}$ 信号稳定后,外部存贮器将数据送往 8098。

1. 总线性能

由于 8098 外部数据总线是 8 位的。因此,当它存取 16 位数据时,性能将有所下降,尤其是在下列三种情况下,指令的执行速度将变慢:第一,将一个字节写到外部存贮器,所要执行的

指令将要额外花费两个状态周期才能完成；第二，从外部存储器读一个字，将使指令的执行时间延长一个状态周期；第三，在存取指令时，如果预先存取的指令序列出现空的（即有未占用的存储单元），那么对从外部获得的每个数据（8 位）所要执行的指令都要延长一个状态周期（最坏的情况是指令的字节数减一）。

2. 读

在读外部存储器期间，从外部存储器来的数据必须在 $\overline{RD}$ 信号上升沿出现之前放到总线上并稳定下来，这一稳定时间称为数据建立时间。8098 在 $\overline{RD}$ 信号的上升沿将总线上的数据读入片内。

3. 写

对外部存储器的写操作时序与读操作相同，在 $\overline{WR}$ 信号的下降沿，8098 从总线上撤消地址信号的同时送出所要写入的数据。当 $\overline{WR}$ 变为高电平时，则数据应被锁存到外部存储器，需要指出的是：8098 还有一个与众不同的特点，即利用其芯片配置寄存器 (CCR) 可将 $\overline{WR}$ 信号适当延时，以便使 8098 能与低速器件有效地配合使用。

4. 准备就绪

为了实现对低速存储器的存取和总线共享 (DMA)，8098 还提供了一个准备就绪 (READY) 信号用以展宽读和写信号的宽度。假如 READY 信号在 ALE 信号下降沿后的规定时间内变为低电平，那么，在 CLKOUT 信号 (相位 A) 的下降沿到来之前，8098 总线上的状态将保持不变 (即插入了等待状态)。当 READY 信号变为高电平后，在下一个 CLKOUT 信号的下降沿，总线操作将继续进行。READY 信号与 CLKOUT 是同步的。

8098 插入的等待状态个数可以是 1, 2 或 3。它们由 CCR 中的值所决定。如果复位时 READY 信号保持低电平，那么 CCR 取数周期中将插入三个等待状态。

READY 引脚内部设有上拉电路，只要外部不将其拉低，平时它呈现高电平。

(二) 操作方式

8098 支持多种操作方式以简化存储器系统、接口需求以及准备就绪控制。由于 8098 允许选择不同的总线控制信号约定，因而提供了总线的灵活性。此外，多种准备就绪控制方式降低了对低速外围器件以及存储器的外部硬件要求。所有操作方式信息存放在 CCR 中。

1. 芯片配置寄存器 CCR

CCR 是一个特殊的专用寄存器，它并不属于内部 RAM 中的专用寄存器，因此在系统工作时，无法通过对内部 RAM 的访问指令改变 CCR 的内容。CCR 用以设置芯片的操作方式，其中的五位用来定义总线控制方式以及准备就绪控制方式；二位给出了 ROM/EPROM 保护级别。

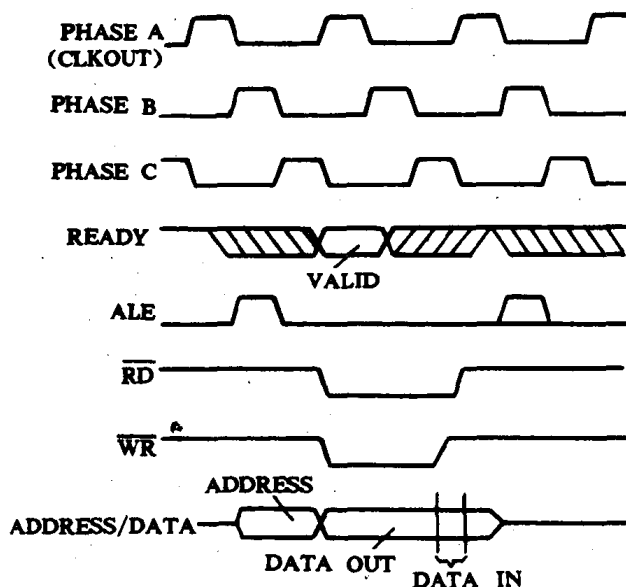


图 1.8 8098 总线定时波形图