

单片机

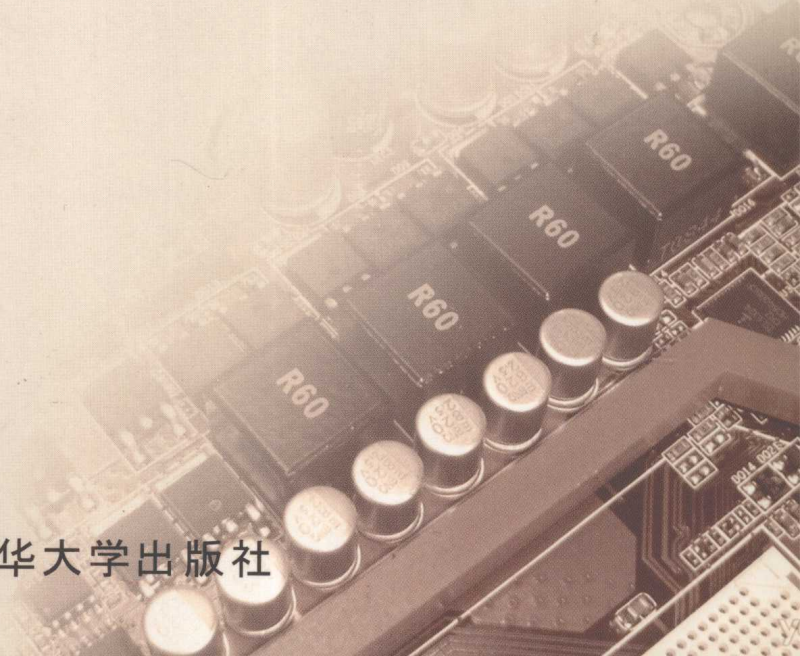
应用与项目实践

黄维翼 主编

眭碧霞 主审



清华大学出版社



内 容 简 介

全书共 8 个项目,项目 1~项目 4 为基础知识,内容包括单片机最小系统、抢答器、电子钟、密码键盘的设计,围绕 I/O 口、外部中断、定时器/计数器、串行通信等单片机内部资源的应用展开;项目 5~项目 8 为提高篇,内容包括 LED 点阵显示屏、多路测温系统、自动打铃系统、数据采集器的设计,涉及存储器、并行 I/O 口的扩展,字符/点阵液晶显示器、数字温度传感器、I²C 总线器件、并行/串行 A/D 转换器和 D/A 转换器的应用。

本书既可作为高职高专计算机与电子信息类相关专业的教材或教学参考书,也可供相关的工程技术人员参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

单片机应用与项目实践/黄维翼主编. —北京:清华大学出版社,2010.4

(国家示范性高职院校建设项目成果. 计算机专业系列)

ISBN 978-7-302-21832-6

I. ①单… II. ①黄… III. ①单片微型计算机—高等学校:技术学校—教材
IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2010)第 004405 号

责任编辑:田 梅

责任校对:袁 芳

责任印制:杨 艳

出版发行:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:三河市春园印刷有限公司

经 销:全国新华书店

开 本:185×260 印 张:17.75 字 数:423 千字

版 次:2010 年 4 月第 1 版 印 次:2010 年 4 月第 1 次印刷

印 数:1~4000

定 价:28.00 元

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

出版说明

特色教材建设是推动课程改革和专业建设的基础,是提升人才培养质量的重要举措,也是高职院校内涵建设的重点之一。

2007年,经教育部、财政部批准,常州信息职业技术学院进入100所国家示范性高职院校建设行列。开展示范院校建设以来,学院坚持以科学发展观为指导,针对市场设专业,针对企业定课程,针对岗位练技能,围绕区域经济建设、信息产业发展的实际需求,全面推进以“三依托、三合一”为核心的工学结合人才培养模式改革,强化职业素质和职业技能的培养,构建了具有学院自身特色的校企合作管理平台,在培养高素质技能型人才、为服务区域经济建设等方面取得了显著成效。

为展示课程建设成果,学院和清华大学出版社合作出版了常州信息职业技术学院特色教材30部,这也是学院示范院校建设的成果之一。作为一种探索,这套教材在许多方面还不尽成熟和完善,但它从一个侧面反映了学院广大教师多年来对有中国特色高职教育教学,特别是教材建设层面的创新与实践,希望能对深化以职业能力培养为核心的专业改革、切实提高教育教学质量发挥应有的作用。

在人才培养模式的创新、课程改革和教材建设中,我们始终得到教育部、财政部、江苏省教育厅、财政厅和国家示范性高职院校建设工作协作委员会等各级领导、专家的关心和指导,得到众多行业企业、兄弟院校和清华大学出版社的大力支持,在此一并致谢!

常州信息职业技术学院

清华大学出版社

2009.6

随着高等职业教育教学改革的深化,任务驱动、案例教学、项目导向等教学方法得到了越来越广泛的应用。本书依据任务驱动、项目导向的教学思路,以项目为载体组织教学内容,将单片机产品的设计与开发过程与相关的知识点有机结合在一起,使学生在学会单片机产品开发过程的同时,认识单片机,使用单片机,掌握单片机的基本原理、相关概念。

单片机又称为微控制器(Micro Controller Unit,简称MCU),它是将计算机的中央处理器、存储器、定时器/计数器、并行口和串行口以及中断系统等部件集成在一块芯片中构成的计算机。因其具有集成度高、体积小、功能强、使用灵活,性价比高等诸多优点,单片机在工业控制、智能仪表、数据采集、信息处理、家用电器等领域中得到了广泛的应用。目前单片机有几十个系列,数百个型号,MCS-51单片机是国内应用最为广泛的机型之一。经过20多年的推广与发展,形成了一个规模庞大、功能齐全、资源丰富的产品群,在今后很长的一段时间内,将依然活跃如故,在嵌入式系统产品的低端市场占据重要份额。

本书是在总结多年MCS-51单片机发展历程和开发实践的基础上,结合目前单片机发展的新技术编写而成。根据高等职业教育的特点,本书简化了单片机及其外围器件内部结构的说明,以“够用”、“会用”为度,侧重于单片机系统的应用方法;根据单片机发展的特点,压缩了程序存储器、并行输入/输出接口扩展的内容,突出了单总线、I²C总线、SPI总线技术的应用;目前单片机C语言已非常流行,绝大部分的应用系统可直接用C51来编写。目前企业实际应用中已很少有人使用汇编语言来编写程序,鉴于以上原因,我们直接使用C51来编程,使学生无需花大量的时间和精力去学习枯燥难懂的汇编语言,直接面对硬件电路进行程序的设计与调试。

本课程主要包括存储器、输入/输出接口、中断系统、定时器/计数器、串行口等资源及外围接口器件的运用,单片机C语言以及开发平台的运用。因此针对单片机不同部分的运用能力,分别以抢答器、电子钟、密码键盘、多路测温系统、自动打铃系统等单片机应用产品作为载体,将相关的知识、技能融入其中。

本书是作者在总结多年MCS-51单片机教学经验和开发实践的基础上,结合目前单片机发展的新技术编写而成,书中提供了大量的应用案例,并有详细的电路和程序设计、分析过程,在编排上循序渐进,由简单到复杂,由基础到综合,以方便读者自学。

教学中,首先根据项目需求导入相关的知识点,然后利用虚拟仿真软件Proteus或单片机实验仪对所学的知识进行验证,通过丰富的案例引导学生思考如何运用这些知识去实现设计要求,最后指导学生进行硬件电路、控制程序的设计与调试,完成产品的制作。在项目的实施过程中,边教边做、边做边学、学练结合,训练学生对所学知识的运用能力和工程实践能力,从而实现基于单片机产品开发过程的能力训练。

本课程建议教学学时为130学时,具体分配如下表。

序号	项 目	课 时		
		理论	实践	合计
1	项目 1 单片机最小系统的设计	6	6	12
2	项目 2 抢答器的设计	10	8	18
3	项目 3 电子钟的设计	8	10	18
4	项目 4 密码键盘的设计	8	8	16
5	项目 5 LED 点阵显示屏的设计	8	6	14
6	项目 6 多路测温系统的设计	6	8	14
7	项目 7 自动打铃系统的设计	8	10	18
8	项目 8 数据采集器的设计	8	8	16
9	机动		4	4
合 计		62	68	130

本书由黄维翼担任主编,项目 1、2 由居平编写,项目 3、4 由吕勇编写,项目 5、8 由孙飞编写,黄维翼编写了项目 6、7、附录 A 和附录 B,并对全书做了统稿及修改。为了与 Proteus 软件保持一致,书中部分电路图不符合国家标准处未做修改。

由于编者水平有限,书中难免存在错误或不妥之处,恳请读者批评指正。

编 者

2009.11

编委会成员

主任：邓志良

副主任：闵 敏 束传政

成 员：赵佩华 曹建庆 杨 诚 陈剑鹤

薛苏云 眭碧霞 陈必群 秦益霖

赖华清 周 敏 宋 卫 庞 杰

秘 书：赵佩华 田 梅

项目 1 单片机最小系统的设计	1
1.1 任务 1 认识最简单的单片机应用系统	1
1.1.1 初识单片机	1
1.1.2 认识单片机的硬件电路	2
1.1.3 单片机的控制程序	3
1.2 任务 2 认识 MCS-51 单片机结构	4
1.2.1 单片机的内部结构	4
1.2.2 CPU 的结构	5
1.2.3 MCS-51 单片机的引脚	7
1.3 任务 3 认识单片机的存储器	12
1.3.1 程序存储器	13
1.3.2 片内数据存储器	14
1.3.3 外部数据存储器	15
1.3.4 特殊功能寄存器	15
1.4 任务 4 单片机最小系统的制作	17
1.4.1 最小系统的硬件电路	17
1.4.2 输入源程序、产生目标代码	17
1.4.3 编程器的使用	22
1.4.4 调试方法与步骤	26
1.5 项目小结	26
习题 1	27
项目 2 抢答器的设计	28
2.1 任务 1 认识单片机 C 语言	28
2.1.1 C 语言的特点	28
2.1.2 C51 的数据类型	29
2.1.3 C51 的数据存储类型与 8051 存储器结构	30
2.1.4 定义 8051 特殊功能寄存器及 I/O 口	32
2.1.5 认识 C51 的内部函数及常用的宏	35
2.2 任务 2 P0~P3 口应用举例	36
2.3 任务 3 学习单片机仿真实验仪的使用	41

2.3.1	认识仿真器	41
2.3.2	USB 型单片机仿真实验仪的安装	43
2.3.3	Keil 调试器的使用	43
2.4	任务 4 Proteus 6.9 的使用	48
2.4.1	认识 Proteus 的操作界面	49
2.4.2	编辑原理图	50
2.4.3	仿真调试	53
2.5	任务 5 认识 MCS-51 单片机的中断系统	55
2.5.1	中断的概念	55
2.5.2	MCS-51 单片机的中断系统	55
2.6	任务 6 设计动态 LED 显示器	59
2.6.1	认识 LED 显示器	59
2.6.2	静态显示方式	61
2.6.3	动态显示方式	61
2.6.4	LED 显示程序的设计技巧	63
2.7	任务 7 设计电子计数器	65
2.8	任务 8 设计抢答器	67
2.9	项目小结	71
习题 2		71
项目 3	电子钟的设计	73
3.1	任务 1 认识单片机的定时器/计数器	73
3.1.1	定时器/计数器概述	74
3.1.2	定时器/计数器的控制寄存器	74
3.1.3	T0、T1 的工作方式	76
3.1.4	时间常数的计算	76
3.2	任务 2 定时器/计数器的应用	78
3.3	任务 3 电子钟的设计与调试	95
3.4	项目小结	102
习题 3		103
项目 4	密码键盘的设计	104
4.1	任务 1 认识串行通信	104
4.1.1	数据通信的概念	104
4.1.2	同步串行通信和异步串行通信	105
4.1.3	通信方向	106
4.1.4	波特率	106
4.1.5	通信线的连接	107
4.1.6	RS-232 简介	108

4.2	任务 2 认识 MCS-51 单片机的串行口	110
4.2.1	MCS-51 单片机的串行口结构	110
4.2.2	串行口的控制寄存器	110
4.3	任务 3 串行口的应用	112
4.3.1	方式 0	112
4.3.2	方式 1	113
4.3.3	方式 2	117
4.3.4	方式 3	118
4.4	任务 4 设计行列式键盘	121
4.4.1	行列式键盘的基本原理	121
4.4.2	行列式键盘扫描程序	121
4.5	任务 5 设计密码键盘	125
4.6	项目小结	130
	习题 4	131
项目 5 LED 点阵显示屏的设计		132
5.1	任务 1 扩展单片机的系统总线	132
5.1.1	单片机系统总线	132
5.1.2	系统扩展的方法	133
5.2	任务 2 扩展单片机的存储器	136
5.2.1	程序存储器的扩展	136
5.2.2	数据存储器的扩展	138
5.3	任务 3 扩展并行输入/输出接口	141
5.3.1	通用锁存器、缓冲器的扩展	141
5.3.2	扩展可编程并行 I/O 接口	144
5.4	任务 4 设计 LED 点阵显示屏	149
5.4.1	认识 LED 点阵模块内部结构	149
5.4.2	设计过程	150
5.5	项目小结	154
	习题 5	154
项目 6 多路测温系统的设计		155
6.1	任务 1 用 DS18B20 设计电子温度计	155
6.1.1	初识数字温度传感器	155
6.1.2	读/写时序	156
6.1.3	基本操作指令	157
6.1.4	电子温度计的设计	158
6.2	任务 2 单片机控制字符型液晶显示模块	162
6.2.1	RT1602 液晶模块引脚功能	162

6.2.2	显示模块的指令功能	163
6.2.3	液晶模块与单片机的连接	165
6.3	任务3 设计多路测温系统	169
6.4	项目小结	175
习题6		176
项目7	自动打铃系统的设计	177
7.1	任务1 认识 I ² C 总线	177
7.1.1	I ² C 总线概述	177
7.1.2	I ² C 总线协议	178
7.2	任务2 用 I/O 口模拟 I ² C 总线操作	180
7.3	任务3 读/写串行 E ² PROM	185
7.3.1	引脚的功能	185
7.3.2	单片机与 24C02 的连接	185
7.3.3	单片机对 24C02 的读写程序	185
7.3.4	调试方法与步骤	187
7.4	任务4 认识实时时钟芯片 PCF8563	189
7.4.1	引脚功能	189
7.4.2	PCF8563 的寄存器	189
7.4.3	电子钟的设计	192
7.5	任务5 单片机控制点阵液晶显示模块 TG12864B	194
7.5.1	TG12864B 模块引脚功能	194
7.5.2	液晶屏与显示存储器之间的对应关系	195
7.5.3	TG12864B 的指令	196
7.5.4	单片机与液晶模块的连接	197
7.5.5	汉字的显示	206
7.6	任务6 设计自动打铃系统	210
7.7	项目小结	220
习题7		221
项目8	数据采集器的设计	222
8.1	任务1 数据采集器的实现过程	222
8.1.1	A/D 转换器的基本概念	222
8.1.2	认识 ADC0808/0809 的内部结构	223
8.1.3	ADC0808/0809 的引脚功能	223
8.1.4	ADC0808/0809 与单片机的典型连接	224
8.1.5	数据采集器的设计过程	225
8.2	任务2 数字电压表的设计	229
8.2.1	认识 SPI 总线	229

8.2.2	认识 8 位串行 A/D 转换器 TLC549	230
8.2.3	数字电压表的设计	232
8.3	任务 3 信号发生器的设计	234
8.3.1	D/A 转换器的基本概念	234
8.3.2	认识串行 D/A 转换器 TLC5615	234
8.3.3	用 TLC5615 设计信号发生器	236
8.4	任务 4 频率与周期的测量	239
8.4.1	频率的测量	239
8.4.2	周期的测量	242
8.5	项目小结	244
	习题 8	245
附录 A MCS-51 单片机的指令系统及汇编语言程序设计		246
A.1	寻址方式	246
A.2	MCS-51 单片机指令集	248
A.3	伪指令	251
A.4	汇编语言程序设计	253
A.5	C 语言与汇编语言混合编程	257
附录 B USB 型单片机实验仪的使用		260
B.1	单片机实验仪的组成	260
B.2	实验仪原理图	261
B.3	实验仪元器件布局	266
B.4	使用中的常见问题	267
参考文献		268

项目1

单片机最小系统的设计

项目描述

本项目首先通过一个最简单的单片机系统的介绍,使大家对单片机硬件电路及控制程序形成初步的认识,随后介绍单片机的内部结构、引脚的功能、与外围电路的连接方法、存储器的配置等相关知识,在此基础上指导大家亲手做一个可以工作的单片机最小系统。该系统可以实现接通电源后控制发光二极管不停地闪烁,从而了解单片机的开发过程,掌握常用工具软件的使用,硬件电路的调试方法,为后续项目的学习打下基础。

1.1 任务 1 认识最简单的单片机应用系统

1.1.1 初识单片机

单片机全称单片微型计算机(Sing Chip Microcomputer),又称 MCU(Micro Contrroller Unit),就是将 CPU、系统时钟、RAM、ROM、定时器/计数器和多种 I/O 接口电路集成在一块芯片上的微型计算机。典型的单片机结构框图如图 1-1 所示。

目前市场上较有影响的单片机有 Intel 公司的 MCS-51、MCS-96 系列; Motorola 公司的 68HCXX 系列; Microchip 公司的 16C5X/6X/7X/8X 系列; Zilog 公司的 Z86EXXXP 系列类型的单片机。

其中,MCS-51 系列(简称 51 系列)单片机最早由美国 Intel 公司推出,随后 Intel 公司把相关技术卖给了 Philips、Atmel 等多家公司,这样 MCS-51 系列单片机就在世界各地迅速普及发展,成为单片机的主流系列。

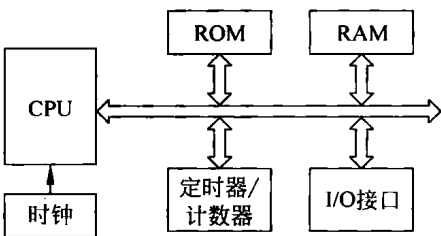


图 1-1 单片机的结构框图

MCS-51 系列单片机包括下列型号。

① 8031、8051、8751、8951 四种型号的单片机通常称为 8051 子系列,它们的区别仅仅在于片内程序存储器。8031 没有片内程序存储器,8051 内含 4KB 的 ROM,8751 片内有 4KB 的 EPROM,8951 片内有 4KB 的 E²PROM。目前应用最为广泛的是 89XX 型单片机,其他三种型号已很少见到。

② 8032、8052、8752、8952 是 8051 子系列的增强型单片机,常称为 8052 子系列。其内

部 RAM 为 256B, 片内程序存储器为 8KB, 比 8051 子系列各增加了 1 倍, 同时还增加了一个定时器/计数器和一个中断源。

③ 80C31、80C51、87C51、89C51 是 8051 子系列单片机的 CHMOS 型芯片, 两者功能兼容。CHMOS 型芯片的基本特点是功耗低。

1.1.2 认识单片机的硬件电路

当我们拿到一块单片机芯片, 想做一个电子产品时, 首先要了解芯片每个引脚的功能, 然后才能知道如何将其与外围器件连接。图 1-2、图 1-3 分别是 8051 单片机的实物图和引脚图。单片机最小系统所用到的主要引脚名称及连接方法如下:

① V_{CC} : 单片机电源输入端, 接 +5V。

② GND: 单片机的地线, 接地。

③ XTAL1、XTAL2: 用于产生单片机工作所需的时钟信号, 一般按图 1-4 接上晶振、电容就可以了。

④ RST: 复位信号输入端, 用于通电时对单片机内部寄存器进行初始化, 只需按图 1-4 接上相应的电阻、电容。

⑤ $\overline{EA}$: 通常直接将该引脚与电源 V_{CC} 相连。



图 1-2 单片机实物图

P1.0	1	40	V_{CC}
P1.1	2	39	P0.0/AD0
P1.2	3	38	P0.1/AD1
P1.3	4	37	P0.2/AD2
P1.4	5	36	P0.3/AD3
P1.5	6	35	P0.4/AD4
P1.6	7	34	P0.5/AD5
P1.7	8	33	P0.6/AD6
RST	9	32	P0.7/AD7
RXD/P3.0	10	31	$\overline{EA}/V_{pp}$
TXD/P3.1	11	30	ALE/PROG
INT0/P3.2	12	29	$\overline{PSEN}$
INT1/P3.3	13	28	P2.7/A15
T0/P3.4	14	27	P2.6/A14
T1/P3.5	15	26	P2.5/A13
$\overline{WR}/P3.6$	16	25	P2.4/A12
$\overline{RD}/P3.7$	17	24	P2.3/A11
XTAL2	18	23	P2.2/A10
XTAL1	19	22	P2.1/A9
GND	20	21	P2.0/A8

图 1-3 单片机引脚图

所谓单片机最小系统, 是指在尽可能少的外部电路条件下, 形成一个可以独立工作的单片机系统, 也就是为了保证单片机能够工作, 所必需的最小系统配置。

在进行单片机最小系统设计时, 应考虑以下问题。

首先, 要保证各电路能够工作, 必须有电源; 其次, 由于单片机是数字电路, 其工作离不开时钟, 因此必须给单片机配置时钟电路; 另外, 为保证单片机能可靠工作, 还必须配置复位电路。如果选用 8031 芯片, 由于片内没有程序存储器, 还必须外扩 ROM 芯片, 而对于 8051、8751、8951 这类芯片, 就不必扩展 ROM 芯片, 只需在 3 个必要条件的基础上加上系统所需的控制电路就可以了。

图 1-4 是一个用于控制发光二极管闪烁的单片机最小系统,该系统包含了电源、时钟、复位电路及发光二极管控制电路。连接好线路,通上电源,单片机就可以正常工作了。但仅有硬件电路是不够的,要完成应用系统的指定功能,必须编写相应的控制程序。

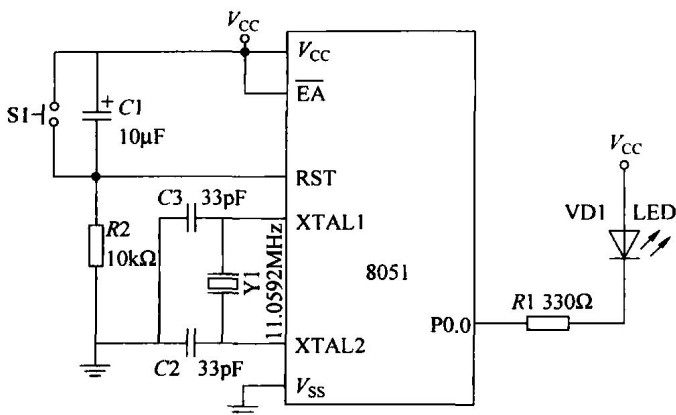


图 1-4 单片机控制发光二极管的电路图

1.1.3 单片机的控制程序

在 8051 单片机的 40 个引脚中, P0.0~P0.7、P1.0~P1.7、P2.0~P2.7、P3.0~P3.7 是可以直接控制的。图 1-4 中, 将 P0.0 通过限流电阻连接到发光二极管的阴极, 发光二极管的阳极接到电源 V_{CC} , 这样当 P0.0 为低电平时, 发光二极管导通而点亮, 而 P0.0 为高电平时, 发光二极管截止而熄灭。

在单片机 C 语言中, 可以将 P0.0 看作是一个变量, 假定用 LED 表示该变量, 用赋值语句就可以很方便地将 P0.0 设置为高电平或低电平。当执行语句“LED=0;”时, P0.0 引脚变为低电平, 而执行语句“LED=1;”时, P0.0 引脚变为高电平, 这样就可以控制发光二极管的点亮或熄灭。相应的程序如下:

```
#include <reg51.h>           //包含针对 51 系列单片机的基本声明
sbit LED = P0 ^ 0;           //用符号 LED 表示 P0.0 引脚

main( )
{
    LED = 0;                  //将 P0.0 设置为低电平, 点亮发光二极管
    LED = 1;                  //将 P0.0 设置为高电平, 熄灭发光二极管
}
```

为了让单片机能够运行以上程序, 还要做两件事。

- ① 将以上 C 语言源程序编译成单片机可以直接执行的目标代码。
- ② 用编程器将目标代码写入单片机内部的程序存储器中。

当将含有目标代码的单片机插入做好的线路板, 接通电源时, 可以看到发光二极管是熄灭的, 并没有闪烁, 什么原因呢? 原来程序中的“LED=0;”语句使发光二极管点亮, 紧接着“LED=1;”语句使发光管熄灭, 以后就再也不会执行“LED=0;”语句了, 这样发光二极管就始终处于熄灭状态。为此我们可以在单片机执行完第二条语句后, 再回到第一条语句继续

执行,让单片机不停地在两条语句之间循环,将程序改成:

```
#include <reg51.h>           //包含针对 51 系列单片机的基本声明
sbit LED = P0 ^ 0;           //用符号 LED 表示 P0.0 引脚
main( )
{ while(1)                   //让单片机反复执行以下两条语句
  { LED = 0;                  //将 P0.0 设置为低电平,点亮发光二极管
    LED = 1;                  //将 P0.0 设置为高电平,熄灭发光二极管
  }
}
```

重新产生目标代码,并写入芯片,通电后却发现发光二极管常亮但不闪烁,这又是什么原因呢?原来执行完“LED=1;”后发光二极管是熄灭了,但单片机运行速度是非常快的,在极短的时间(数微秒)后,又执行了“LED=0;”这样发光二极管又被点亮了,由于人类眼睛的视觉惰性,我们根本看不出发光二极管曾经熄灭过。为了解决以上问题,可以在发光二极管点亮后延时一段时间,再去熄灭发光二极管,发光二极管熄灭后同样也延时一段时间再去执行点亮操作,这样就可以看到发光二极管的闪烁了。

如何完成延时功能呢?虽然单片机执行一条语句的时间很短,如果让单片机循环执行一条语句,所需的时间(延时时间)是循环次数与单片机执行这条语句所需时间的积,当改变循环次数时,延时时间也就跟着改变了。下面的程序就是将变量 i 执行 20000 次加 1 操作实现延时的。

```
#include <reg51.h>           //包含针对 51 系列单片机的基本声明
sbit LED = P0 ^ 0;           //用符号 LED 来表示 P0.0 引脚
main( )
{ unsigned int i;            //定义一个用于延时的变量
  while(1)                   //让单片机反复执行以下四条语句
  { LED = 0;                  //将 P0.0 设置为低电平,点亮发光管
    for(i = 0; i < 20000; i++); //变量 i 从 0 加到 20000,完成延时
    LED = 1;                  //将 P0.0 设置为高电平,熄灭发光二极管
    for(i = 0; i < 20000; i++); //延时
  }
}
```

1.2 任务 2 认识 MCS-51 单片机结构

1.2.1 单片机的内部结构

MCS-51 系列单片机早期的型号包括 8051、8031、8751 等,由以下部件构成。

- ① 一个 8 位 CPU;
- ② 一个片内振荡器及时钟电路;
- ③ 4KB 的 ROM/EPROM 程序存储器(8031 没有);
- ④ 128B 的 RAM 数据存储器;
- ⑤ 可寻址 64KB 外部数据存储器 and 64KB 外部程序存储器的控制电路;

- ⑥ 32 位可编程的 I/O 口(4 个 8 位并行 I/O 端口);
- ⑦ 两个 16 位的定时器/计数器;
- ⑧ 一个可编程全双工串行口;
- ⑨ 5 个中断源、两个优先级嵌套中断结构。

MCS-51 系列单片机之后又有了 8052、8032、8752 等型号,它们与前 3 个型号的单片机相比,内部程序存储器和数据存储器容量都翻了一番,另外定时器增加到 3 个,中断源增加到 6 个,其他方面基本相同。

MCS-51 系列单片机内部结构如图 1-5 所示。

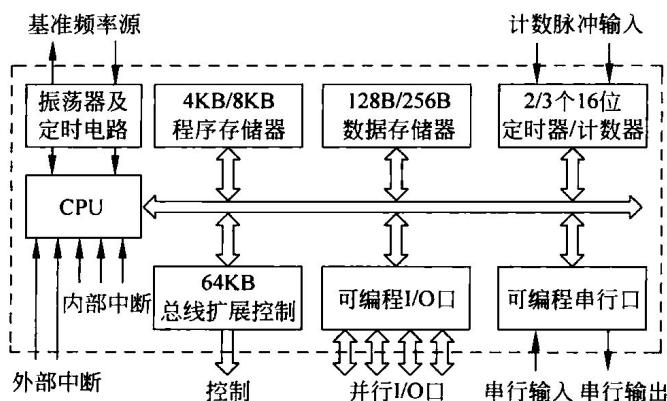


图 1-5 MCS-51 系列单片机内部结构图

随着 MCS-51 系列单片机进一步发展,其内部包含了更多的数据存储器以及由 E²PROM 构成的程序存储器,有的单片机还包括 A/D 转换电路、USB 接口电路、PWM 输出、看门狗计数器电路等各种的电路。MCS-51 系列单片机的内核甚至可以下载到 PLD(可编程逻辑器件)中,再在 PLD 器件上构造其他需要的电路,由此构成“单片系统”。

1.2.2 CPU 的结构

CPU 是单片机的核心,MCS-51 系列单片机内含一个高性能的 8 位中央处理器。CPU 的作用是从 ROM 中读取指令并进行分析,然后根据指令的功能控制单片机的功能部件执行指定的操作。CPU 由运算器和控制器两大功能部件组成。

1. 运算器

运算器的主要功能是进行算术运算和逻辑运算,由算术逻辑运算部件(ALU)、暂存器及部分特殊功能寄存器组成。

ALU 是运算器的核心部件,除了可以实现加、减、乘、除等算术运算和与、或、非、异或、循环、求补等逻辑运算外,还具有一定的位处理能力,如置位、取反、清 0、测试转移等操作,特别适合实时逻辑控制,这也是 MCS-51 系列单片机能够成为面向控制的微处理器的重要原因。

在 ALU 进行运算时,通常会用到 ACC、B、PSW 三个特殊功能寄存器。

ACC 寄存器(简称累加器 A):用于向 ALU 提供操作数和存放运算结果,还可实现与程序存储器、片外数据存储器及 I/O 接口的数据传递,是使用最频繁的寄存器。

B 寄存器:在乘除运算时存放另外一个操作数,乘除运算完成后,存放运算的一部分结果。如果不进行乘除运算,B 寄存器可作为一般寄存器使用。

PSW 寄存器:又称为程序状态字寄存器,当加、减、乘、除等指令执行完后,用来存储相应的状态信息。PSW 是一个 8 位的寄存器,各位的定义如下:

CY:进位标志位。进行运算时,如果操作结果在最高位有进位(加法运算)或借位(减法运算),CY=1,否则 CY=0。

AC:辅助进位标志位。如果操作结果的低 4 位有进位(加法运算)或借位(减法运算),AC=1,否则 AC=0。

F0:用户标志位。其功能由用户自行定义,用户可通过软件对它置位、复位或测试,以控制程序的流向。

RS1、RS0:工作寄存器选择位。用于选择工作寄存器 R0~R7 的实际位置,取值为 00~11,具体说明见 1.3.2 小节片内数据存储器。

OV:溢出标志位。如果两个操作数的运算结果超出了运算范围,OV=1,否则 OV=0。

P:奇偶标志位。如果累加器 A 中 1 的个数为奇数,P=1,否则 P=0。

PSW 寄存器各位定义见表 1-1。

表 1-1 PSW 寄存器各位定义

位编号	PSW. 7	PSW. 6	PSW. 5	PSW. 4	PSW. 3	PSW. 2	PSW. 1	PSW. 0
位名称	CY	AC	F0	RS1	RS0	OV	—	P
位地址	D7H	D6H	D5H	D4H	D3H	D2H	D1H	D0H

2. 控制器

控制器的作用是控制单片机各部件的协调动作。控制器由程序计数器 PC、指令寄存器 IR、指令译码器 ID、定时与控制逻辑电路、数据指针 DPTR、堆栈指针 SP 等组成。

程序计数器 PC 是一个 16 位的计数器,它总是存放着下一条指令所在的 16 位地址,单片机运行过程中,CPU 总是根据 PC 所指定的地址从程序存储器中取出指令,然后分析执行,同时 PC 的值自动加 1,为读取下一条指令做准备。单片机上电或复位时,PC 自动清 0,从地址 0000H 开始取指执行。

指令寄存器 IR 用来保存正在执行的指令代码。若要执行一条指令,首先要把它从程序存储器取到指令寄存器中。指令的内容一般包括操作码和操作数两部分,操作码送往指令译码器 ID,经译码后便确定了所要执行的操作,地址码则送往操作数形成电路以产生实际的操作数地址。

定时与控制逻辑电路是 CPU 的核心部件,用来控制取指令、分析指令、存取操作数等操作。它向其他部件发出各种操作控制信号,协调各部件的工作。

数据指针 DPTR 是一个 16 位的寄存器,由 DPH(数据指针的高 8 位)和 DPL(数据指针的低 8 位)组成,既能作为一个 16 位寄存器使用,也可作为两个独立的 8 位寄存器使用,