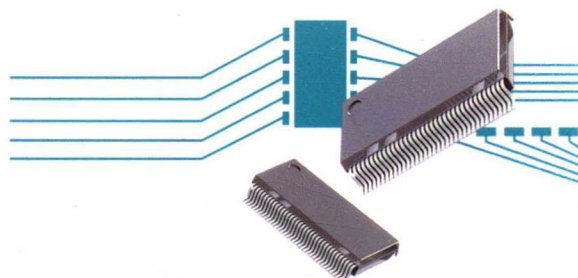
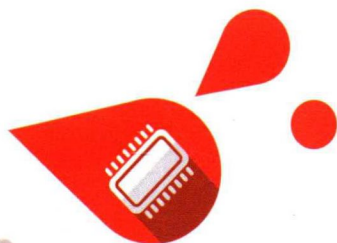


◎ 徐爱钧 著



含光盘 1 张

# 单片机 C语言编程 与 Proteus 仿真技术



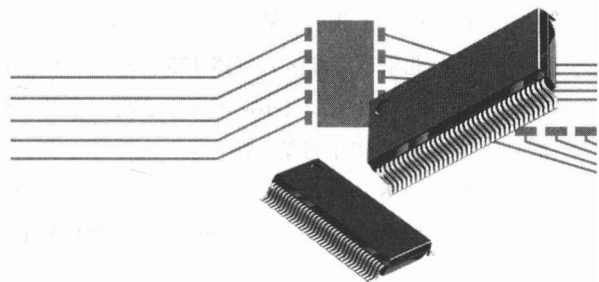
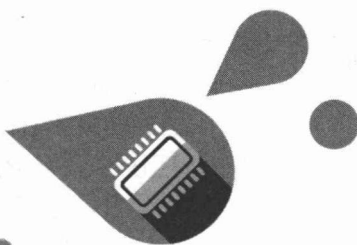
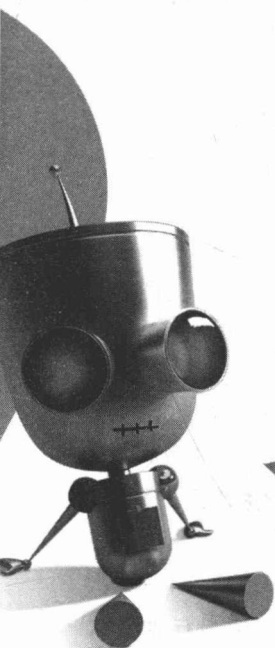
中国工信出版集团



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

◎ 徐爱钧 著

# 单片机 C语言编程 与 Proteus 仿真技术



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书在介绍8051单片机组成原理的基础上,结合目前流行的Keil C51编译器和Proteus虚拟仿真环境,阐述单片机C语言编程和虚拟仿真应用技术,包括单片机中断系统、定时器/计数器、串行口等片内资源的工作原理、单片机系统扩展、DAC与ADC、键盘与显示器接口技术,详细介绍了单片机片内、片外资源的C语言程序设计及其Proteus虚拟仿真应用方法,给出了大量单片机C语言程序范例和Proteus原理电路图,所有范例均在Proteus软件平台上调试通过,可以直接运行。

为帮助读者更好地学习和掌握C51应用编程方法和Proteus仿真技术,本书附赠一张CD-ROM光盘,包含Keil公司全功能C51评估软件包、Proteus仿真电路图及各章所有范例的程序源代码。

本书适合从事单片机应用系统开发研制的广大工程技术人员阅读,也可以作为高等院校相关专业大学生或研究生的教学参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有,侵权必究。

### 图书在版编目(CIP)数据

单片机C语言编程与Proteus仿真技术 / 徐爱钧著. —北京:电子工业出版社,2016.1

ISBN 978-7-121-27538-8

I. ①单… II. ①徐… III. ①单片微型计算机—C语言—程序设计 ②单片微型计算机—系统仿真—应用软件 IV. ①TP312 ②P368.1

中国版本图书馆CIP数据核字(2015)第267039号

责任编辑:富军

印刷:三河市华成印务有限公司

装订:三河市华成印务有限公司

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编 100036

开本:787×1092 1/16 印张:22.5 字数:576千字

版次:2016年1月第1版

印次:2016年1月第1次印刷

印数:3000册 定价:59.00元(含光盘1张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。



## P R E F A C E

8051是目前国内外工业测量控制领域内使用极为广泛的一类8位单片机。它在一块芯片上同时集成CPU、ROM、RAM及多种外围功能接口,具有体积小、价格低、功能强、可靠性高、使用方便灵活等特点,以单片机为核心设计各种智能化电子设备,周期短,成本低,易于更新换代,维修方便,已成为电子设计中最为普遍的应用手段。世界上许多大半导体厂商,如Atmel、Analog Device、Infineon、NXP、TI、SiLAB等公司都推出了各具特色的8051系列单片机。

早期,单片机应用开发大多采用汇编语言编程,编程效率不高,程序不易移植和维护。随着Keil C51编译器的流行,现在已经普遍采用C语言进行单片机应用编程。C语言具有类似自然语言的特点,既能直接操作机器硬件,又可以进行方便灵活的高级语言编程。在单片机应用系统开发过程中,除了编程工具之外,硬件平台也必不可少。目前,各种单片机开发平台层出不穷,英国Labcenter公司推出的Proteus软件是一款极好的单片机虚拟硬件平台,以其特有的仿真技术很好地解决了单片机及其外围电路的设计和协同仿真问题,可以在没有单片机实际硬件的条件下,利用PC机进行虚拟仿真实现单片机系统的软、硬件设计。Proteus虚拟硬件平台可以与Keil C51完美结合,在原理图中直接进行单片机C语言程序的源代码仿真调试,实现对系统性能的综合评估,验证各项技术指标。Proteus平台涵盖了8051等多种微处理器模型及各种常用电子元器件,包括74系列、CMOS4000系列集成电路、A/D和D/A转换器、键盘、LCD显示器、LED显示器,还提供示波器、逻辑分析仪、通信终端、电压/电流表、I<sup>2</sup>C/SPI终端等各种虚拟仪表,可以直接用于虚拟仿真,结合原理图和源码级程序调试,能够立即观察到单片机应用系统的输入、输出效果,极大地提高了应用系统的设计效率。

本书在构思及选材上符合单片机应用发展要求,突出先进性和实用性,对C51应用编程方法、Proteus虚拟仿真技术等进行详尽阐述,给出了大量单片机C语言程序和Proteus仿真设计范例。所有范例均已在Proteus平台上调试通过,可以直接运行。

全书共9章:

第1章阐述8051单片机基本组成、存储器结构、CPU时序、并行I/O端口及指令系统。

第2章阐述Proteus虚拟硬件平台,介绍在ISIS集成环境中绘制原理电路图、与Keil C51联机实现源代码仿真调试的方法。

第3章阐述Keil C51应用程序设计,介绍C51的基本语句、数据类型、Keil C51对ANSI C的扩展、与汇编语言程序接口及C51库函数等。

第4章阐述单片机片内资源应用,介绍C51编程的基本原则方法,给出中断系统、定时器/计数器、串行口等功能部件的C语言应用编程实例。

第5章阐述系统扩展与低功耗应用,介绍存储器、并行I/O端口的扩展及单片机低功耗应用方法,给出并行接口扩展芯片和低功耗工作方式的C语言应用编程实例。

第6章阐述键盘与显示器接口应用,介绍矩阵接盘、数码管、点阵字符和图型液晶显示

器等与单片机的接口方法，给出C语言应用编程实例。

第7章阐述数模与模数转换接口应用，介绍传统并行及新型串行D/A、A/D转换器芯片及其与单片机的接口方法，给出C语言应用编程实例。

第8章阐述I<sup>2</sup>C总线接口应用，介绍I<sup>2</sup>C总线结构与数据传输，给出I<sup>2</sup>C接口存储器芯片、A/D-D/A转换芯片及时钟芯片的C语言应用编程实例。

第9章给出5个单片机Proteus虚拟仿真设计实例及其完整的C51源程序。

本书在编写过程中得到广州风标信息技术有限公司 (<http://www.windway.cn>) 匡载华总经理的大力支持和热情帮助，电子工业出版社柴燕、富军编辑提出了许多宝贵意见，徐阳、彭秀华等参加了部分章节的编写和程序调试工作，在此一并表示感谢。

由于作者水平有限，书中难免会有错误和不妥之处，恳请广大读者批评指正，读者可通过电子邮件：[ajxu@tom.com](mailto:ajxu@tom.com)、[ajxu41@sohu.com](mailto:ajxu41@sohu.com)直接与作者联系。Proteus的DEMO软件可到官方网站<http://www.labcenter.co.uk>下载，或者与国内代理商广州风标信息技术有限公司联系购买正版软件。

徐爱钧 于长江大学

# 目 录

## CONTENTS

第1章 8051单片机基础	1
1.1 8051单片机的特点与基本结构	1
1.2 8051单片机的存储器结构	5
1.3 CPU时序	8
1.4 复位信号与复位电路	10
1.5 并行I/O端口结构	11
1.6 指令系统	13
1.7 指令的寻址方式	15
1.7.1 寄存器寻址	15
1.7.2 直接寻址	15
1.7.3 立即寻址	15
1.7.4 寄存器间接寻址	16
1.7.5 变址寻址	16
1.7.6 相对寻址	17
1.7.7 位寻址	18
1.8 指令分类详解	18
1.8.1 算术运算指令	18
1.8.2 逻辑运算指令	20
1.8.3 数据传送指令	21
1.8.4 控制转移指令	23
1.8.5 位操作指令	25
1.9 汇编语言程序设计	26
第2章 Proteus虚拟仿真	31
2.1 集成环境ISIS	31
2.2 绘制原理图	35
2.3 创建汇编语言源代码仿真文件	37
2.4 在原理图中进行源代码仿真调试	39
2.5 原理图与Keil环境联机仿真调试	42
第3章 Keil C51应用程序设计	49
3.1 Keil C51程序设计的基本语法	49
3.1.1 Keil C51程序的一般结构	49
3.1.2 数据类型	50



3.1.3	常量、变量及其存储模式	51
3.1.4	运算符与表达式	52
3.2	C51程序的基本语句	56
3.2.1	表达式语句	56
3.2.2	复合语句	56
3.2.3	条件语句	56
3.2.4	开关语句	57
3.2.5	循环语句	57
3.2.6	goto、break、continue语句	58
3.2.7	返回语句	58
3.3	函数	59
3.3.1	函数的定义与调用	59
3.3.2	中断服务函数与寄存器组定义	60
3.4	Keil C51编译器对ANSI C的扩展	61
3.4.1	存储器类型与编译模式	61
3.4.2	关于bit、sbit、sfr、sfr16数据类型	62
3.4.3	一般指针与基于存储器的指针及其转换	65
3.4.4	C51编译器对ANSI C函数定义的扩展	66
3.5	C51编译器的数据调用协议	69
3.5.1	数据在内存中的存储格式	69
3.5.2	目标代码的段管理	71
3.6	与汇编语言程序的接口	73
3.6.1	参数传递规则	73
3.6.2	C51与汇编语言混合编程举例	77
3.7	绝对地址访问	80
3.7.1	采用扩展关键字“_at_”或指针定义变量的绝对地址	80
3.7.2	采用预定义宏指定变量的绝对地址	81
3.8	Keil C51库函数	81
3.8.1	本征库函数	82
3.8.2	字符判断转换库函数	82
3.8.3	输入、输出库函数	83
3.8.4	字符串处理库函数	87
3.8.5	类型转换及内存分配库函数	88
3.8.6	数学计算库函数	89
第4章	单片机片内资源应用	91
4.1	采用Keil C51编写应用程序的基本原则	91
4.2	并行I/O端口	92
4.2.1	典型单片机输入、输出电路	92
4.2.2	单片机I/O端口应用编程	94

4.3	中断系统	100
4.3.1	中断系统结构与中断控制	101
4.3.2	中断响应	104
4.3.3	中断系统应用编程	106
4.4	定时器/计数器	110
4.4.1	定时器/计数器的工作方式与控制	110
4.4.2	定时器方式应用编程	113
4.4.3	计数器方式应用编程	119
4.4.4	利用定时器产生音乐	121
4.5	串行口	124
4.5.1	串行口的工作方式与控制	125
4.5.2	串口/并口转换应用编程	128
4.5.3	单片机与PC机通信应用编程	130
4.5.4	单片机与单片机通信应用编程	132
4.5.5	修改底层函数实现printf()重新定向	139
<b>第5章</b>	<b>系统扩展与低功耗应用</b>	<b>143</b>
5.1	存储器扩展	143
5.1.1	程序存储器扩展	143
5.1.2	数据存储器扩展	144
5.2	并行I/O端口扩展	146
5.2.1	线选法	146
5.2.2	地址译码法	147
5.2.3	8155和8255并行接口扩展芯片应用编程	149
5.3	8051单片机的低功耗应用	158
5.3.1	空闲工作方式	158
5.3.2	掉电工作方式	159
5.3.3	低功耗方式应用编程	159
<b>第6章</b>	<b>键盘与显示器接口应用</b>	<b>163</b>
6.1	LED显示器接口技术	163
6.1.1	七段LED数码管显示器	163
6.1.2	单个74HC595驱动多位LED数码管	168
6.1.3	串行接口8位共阴极LED驱动器MAX7219	171
6.2	键盘接口技术	177
6.2.1	编码键盘接口	178
6.2.2	非编码键盘接口	180
6.3	8279可编程键盘/显示器芯片接口技术	183
6.3.1	8279的引脚排列	183
6.3.2	8279的数据输入、显示输出及命令格式	184



6.3.3	8279接口应用编程 .....	189
6.4	点阵字符型LCD接口技术 .....	191
6.4.1	点阵字符型LCD显示模块 .....	192
6.4.2	直接方式接口应用编程 .....	197
6.4.3	间接方式接口应用编程 .....	200
6.4.4	4位数据总线接口应用编程 .....	203
6.5	12864点阵图型LCD接口技术 .....	206
6.5.1	12864点阵图型LCD显示模块 .....	206
6.5.2	12864 LCD接口应用编程 .....	209
6.6	T6963点阵图型LCD接口技术 .....	213
6.6.1	T6963点阵图型LCD显示模块 .....	213
6.6.2	T6963 LCD接口应用编程 .....	216
<b>第7章</b>	<b>模数与数模转换接口应用 .....</b>	<b>223</b>
7.1	转换器的主要技术指标 .....	223
7.2	数/模转换器DAC接口技术 .....	224
7.2.1	DAC0832接口应用编程 .....	225
7.2.2	DAC1208接口应用编程 .....	229
7.2.3	串行D/A芯片TLC5615接口应用编程 .....	231
7.2.4	利用DAC接口实现波形发生器 .....	233
7.3	模/数转换器ADC接口技术 .....	238
7.3.1	比较式ADC0809接口应用编程 .....	239
7.3.2	积分式ADC ICL7135接口应用编程 .....	243
7.3.3	串行A/D芯片TLC549接口应用编程 .....	248
<b>第8章</b>	<b>I<sup>2</sup>C总线接口应用 .....</b>	<b>253</b>
8.1	I <sup>2</sup> C总线简介 .....	253
8.2	I <sup>2</sup> C总线结构与数据传输 .....	253
8.3	I <sup>2</sup> C总线通用驱动程序 .....	257
8.4	I <sup>2</sup> C接口存储器芯片24C04应用编程 .....	258
8.5	I <sup>2</sup> C接口A/D-D/A转换芯片PCF8591应用编程 .....	263
8.6	I <sup>2</sup> C接口时钟芯片PCF8563应用编程 .....	274
<b>第9章</b>	<b>Proteus仿真设计实例 .....</b>	<b>285</b>
9.1	红外遥控系统 .....	285
9.1.1	功能要求 .....	285
9.1.2	硬件电路设计 .....	285
9.1.3	软件程序设计 .....	285
9.2	点阵LED显示屏 .....	296
9.2.1	功能要求 .....	296

9.2.2	硬件电路设计	296
9.2.3	软件程序设计	297
9.3	电子密码锁	299
9.3.1	功能要求	299
9.3.2	硬件电路设计	299
9.3.3	软件程序设计	300
9.4	DS18B20多点温度监测系统	316
9.4.1	功能要求	316
9.4.2	硬件电路设计	316
9.4.3	软件程序设计	320
9.5	SD卡WAV音频播放器	328
9.5.1	功能要求	328
9.5.2	硬件电路设计	329
9.5.3	软件程序设计	330
附录A	8051指令表	341
附录B	Proteus中的常用元器件	347
参考文献		349

# 8051单片机基础

## 1.1 8051单片机的特点与基本结构

8051系列单片机是在美国Intel公司于20世纪80年代推出的MCS—51系列高性能8位单片机的基础上发展而来的。它在单一芯片内集成了并行I/O口、异步串行口、16位定时器/计数器、中断系统、片内RAM和片内ROM及其他一些功能部件。现在8051系列单片机已经有了很大的发展，除了Intel公司之外，Philips、Siemens、AMD、Fujitsu、OKI、Atmel、SST、Winbond等公司都推出了以8051为核心的新一代8位单片机。这种新型单片机的集成度更高，在片内集成了更多的功能部件，如A/D、PWM、PCA、WDT及高速I/O口等。不同公司推出的8051具有各自的功能特点，但它们的内核都是以Intel公司的MCS—51为基础的，并且指令系统兼容，从而给用户带来了广阔的选择范围，同时又可以采用相同的开发工具。

8051系列单片机在存储器的配置上采用所谓“哈佛”结构，即在物理上具有独立的程序存储器和数据存储器，而在逻辑上则具有相同的地址空间，利用不同的指令和寻址方式可分别访问64kB程序存储器空间和64kB数据存储器，共有111条指令。其中包括乘除指令和位操作指令。有5个中断源，分为2个优先级。8051单片机在片内RAM中开辟了4个工作寄存器区，每个区有8个通用寄存器，可以适用于多种中断或子程序嵌套的情况。在片内RAM中还开辟了1个位寻址区，利用位操作指令可以对位寻址区中每个单元的每一个位进行操作，特别适合于解决各种开关控制和逻辑问题。8051单片机在单芯片应用方式下，其4个并行I/O口P0~P3都可以作为输入、输出，在扩展应用方式下可采用P0和P2口作为片外扩展地址总线。8051单片机内部集成了两个16位定时器/计数器，可进行定时或计数操作，还集成了1个全双工的异步串行接口，为单片机之间相互通信或与上位机通信带来极大的方便。

8051单片机的基本组成如图1.1所示。一个单片机芯片内包括中央处理器CPU。它是单片机的核心，用于产生各种控制信号，并完成对数据的算术逻辑运算和传送。片内数据存储器RAM用于存放可以读/写的数据。片内程序存储器ROM用于存放程序指令或某些常数表格。四个8位的并行I/O接口P0、P1、P2和P3，每个口都可以用作输入或者输出。两个定时器/计数器T0和T1用于外部事件计数器，也可用于定时。内部中断系统具有5个中断源，两个优先级的嵌套中断结构，可实现二级中断服务程序嵌套，每一个中断源都可编程设定为高优先级

或低优先级中断。一个串行接口电路可用于异步接收发送器。内部时钟电路，其晶体和微调电容需要外接，振荡频率可以高达40MHz。以上各部分通过内部总线相连接。

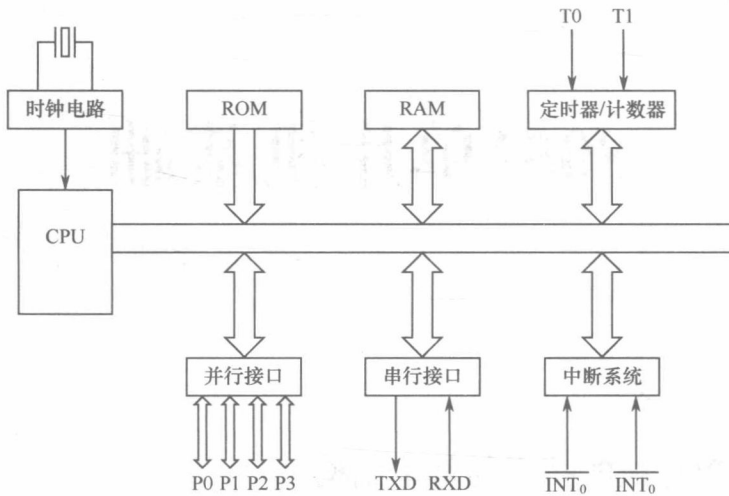


图1.1 8051单片机的基本组成

8051单片机没有单独的片外地址和数据总线，而是通过并行I/O端口的分时复用形成。P0口分时复用为低8位地址线和8位数据线。P2口则作为高8位地址线，从而形成16位地址线和8位数据线。一定要建立一个明确的概念，单片机在进行外部扩展时的地址线和数据线都不是独立的，而是与并行I/O端口公用的。这是8051单片机结构上的一个特点。

图1.2为8051单片机内部结构框图。其中，中央处理器CPU包含运算器和控制器两大部分；运算器完成各种算术和逻辑运算；控制器在单片机内部协调各功能部件之间的数据传送和运算操作，并对单片机外部发出若干控制信息。

### 1. 运算器

运算器以算术逻辑单元ALU为核心，加上累加器ACC、暂存寄存器TMP和程序状态字寄存器PSW等组成。ALU主要用于完成二进制数据的算术和逻辑运算，并通过对运算结果的判断影响程序状态字寄存器PSW中有关位的状态。累加器ACC是一个8位的寄存器（在指令中一般写为A），通过暂存寄存器TMP与ALU相连。ACC的工作最为繁忙，因为在进行算术逻辑运算时，ALU的一个输入多为ACC的输出，而大多数运算结果也需要送到ACC中，在做乘除运算时，B寄存器用来存放一个操作数，也用来存放乘除运算后的一部分结果，若不做乘除操作时，B寄存器可用作通用寄存器。程序状态字寄存器PSW也是一个8位寄存器，用于存放运算结果的一些特征，格式如下：

D7	D6	D5	D4	D3	D2	D1	D0
CY	AC	F0	RS1	RS0	OV	\	P

其中各位的意义如下：

CY：进位标志。在进行加法或减法运算时，若运算结果的最高位有进位或借位，则CY=1，否则CY=0，在执行位操作指令时，CY作为位累加器。

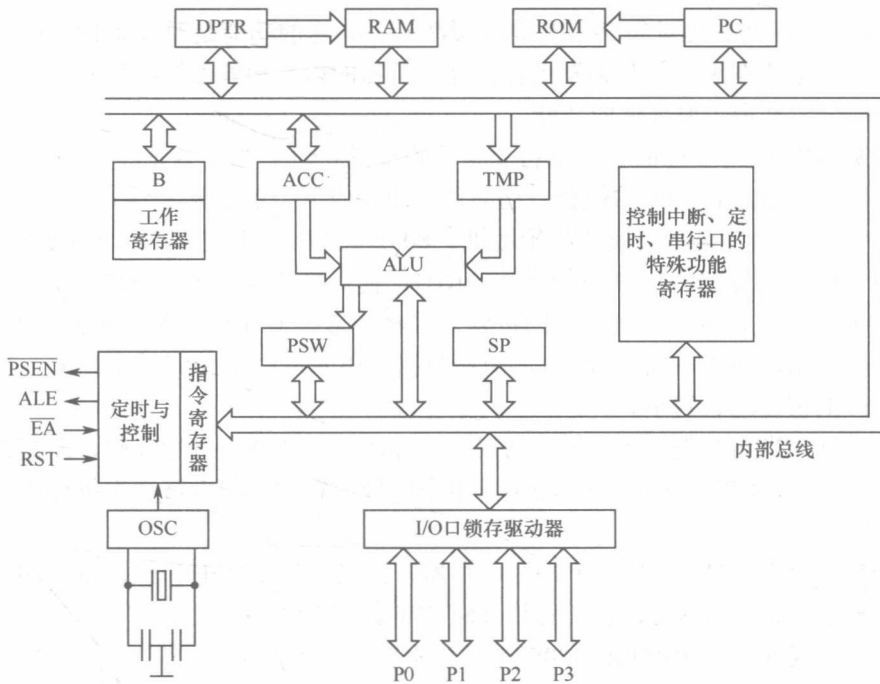


图1.2 8051单片机内部结构框图

**AC:** 辅助进位标志。在进行加法或减法运算时，若低半字节向高半字节有进位或借位，则AC=1，否则AC=0，AC还作为BCD码运算调整时的判别位。

**F0:** 用户标志。用户可根据自己的需要对F0赋以一定的含义，如可以用软件来测试F0的状态以控制程序的流向等。

**RS1和RS0:** 工作寄存器组选择，可以用软件来置位或复位。它们与工作寄存器组的关系见表1.1。

表1.1 RS1和RS0与工作寄存器组的关系

RS1	RS0	工作寄存器组	片内RAM地址
0	0	第0组	00H~07H
0	1	第1组	08H~0FH
1	0	第2组	10H~17H
1	1	第3组	18H~1FH

**OV:** 溢出标志。当两个带符号的单字节数进行运算，结果超出-128~+127的范围时，OV=1，表示有溢出，否则OV=0，表示无溢出。

PSW中的D1位为保留位。

**P:** 奇偶校验标志。每条指令指行完毕后，都按照累加器A中“1”的个数来决定P值，当“1”的个数为奇数时，P=1，否则P=0。

## 2. 控制器

控制器包括定时控制逻辑、指令寄存器、指令译码器、程序计数器PC、数据指针寄存

器DPTR、堆栈指针SP、地址寄存器和地址缓冲器等。它的功能是对逐条指令进行译码，并通过定时和控制电路在规定的时刻发出各种操作所需的内部和外部控制信号协调各部分的工作。下面简单介绍其中主要部件的功能。

**程序计数器PC：**用于存放下一条将要执行指令的地址。当一条指令按PC所指向的地址从程序存储器中取出之后，PC的值会自动增量，即指向下一条指令。

**堆栈指针SP：**用来指示堆栈的起始地址。8051单片机的堆栈位于片内RAM中，而且属于“上长型”堆栈，复位后，SP被初始化为07H，使得堆栈实际上由08H单元开始。必要时可以给SP装入其他值，重新规定栈底的位置。堆栈中数据操作规则是“先进后出”，每往堆栈中压入一个数据，SP的内容自动加1，随着数据的压入，SP的值将越来越大，当数据从堆栈弹出时，SP的值将越来越小。

**指令译码器：**当指令送入指令译码器后，由译码器对该指令进行译码，即把指令转变成所需要的电平信号，CPU根据译码器输出的电平信号使定时控制电路产生执行该指令所需要的各种控制信号。

**数据指针寄存器DRTR：**是一个16位寄存器，由高位字节DPH和低位字节DPL组成，用来存放16位数据存储器的地址，以便对片外64kB的数据RAM区进行读写操作。

采用40引脚双列直插DIP封装的8051单片机引脚分配图如图1.3所示。

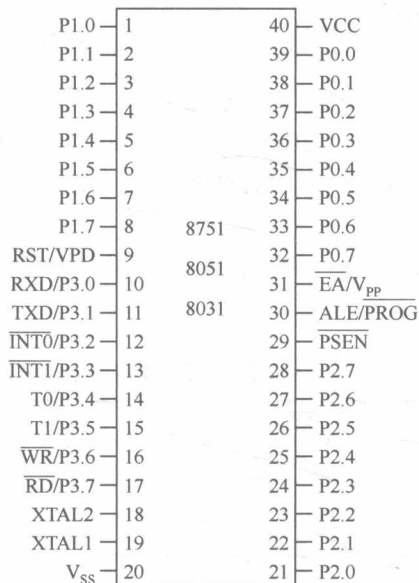


图1.3 8051单片机引脚分配图

各引脚功能如下：

**RST/VPD (9)：**RST是复位信号输入端。当此输入端保持两个机器周期（24个振荡周期）的高电平时，就可以完成复位操作。第二功能是VPD，即备用电源输入端，当主电源发生故障，降低到规定的低电平以下时，VPD将为片内RAM提供备用电源，以保证存储在RAM中的信息不丢失。

**XTAL2 (18)和XTAL1 (19)：**在使用单片机内部振荡电路时，这两个端子用来外接石英晶体和微调电容（图1.4 (a)）。在使用外部时钟时，用来输入时钟脉冲，但对NMOS和



CMOS芯片接法不同,图1.4 (b)为NMOS芯片8051外接时钟,图1.4 (c)为CMOS芯片8051外接时钟。

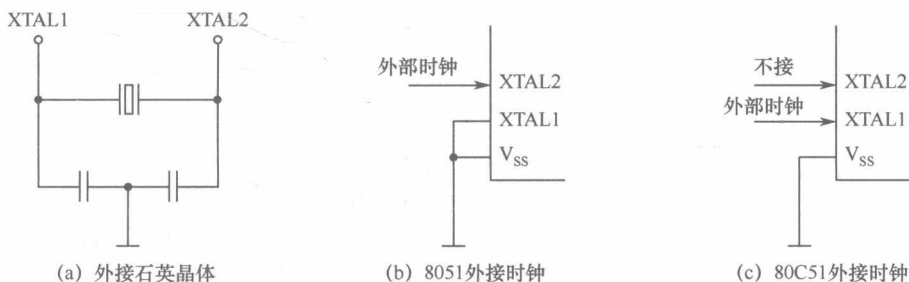


图1.4 8051单片机的时钟接法

$V_{SS}$  (20): 接地。

$V_{CC}$  (40): 接+5V电源。

$\overline{EA}/V_{PP}$  (31): 访问外部存储器的控制信号。当 $\overline{EA}$ 为高电平时,访问单片机片内程序存储器,当程序计数器PC的值超过0FFFH(对8051)或1FFFH(对8052)时,将自动转向执行单片机片外程序存储器内的程序。当 $\overline{EA}$ 保持低电平时,只访问外部程序存储器,不管是否有片内程序存储器。第二功能 $V_{PP}$ 为对8751片内EPROM的21伏编程电源输入。

$ALE/\overline{PROG}$  (30): ALE是地址锁存允许信号,在访问外部存储器时,用来锁存由P0口送出的低8位地址信号。在不访问外部存储器时,ALE以振荡频率1/6的固定速率输出脉冲信号。因此它可用作对外输出的时钟。但要注意,只要外接有存储器,则ALE端输出的就不再是连续的周期脉冲信号了。第二功能 $\overline{PROG}$ 是用于对8751片内EPROM编程的脉冲输入端。

$\overline{PSEN}$  (29): 外部程序存储器ROM的读选通信号。在执行访问外部ROM指令的时候,会自动产生 $\overline{PSEN}$ 信号,而在访问外部数据存储器RAM或访问内部ROM时,不产生 $\overline{PSEN}$ 信号。

P1.0~P1.7 (1~8): 双向I/O口P1。P1口能驱动(吸收或输出电流)4个LS型TTL负载。在对EPROM编程和程序验证时,接收低8位地址。在8052单片机中,P1.0还用作定时器2的计数触发输入端T2,P1.1还用作定时器2的外部控制端T2EX。

P3.0~P3.7 (10~17): 双向I/O口P3,P3口能驱动(吸收或输出电流)4个LS型TTL负载。P3口的每条引脚都有各自的第二功能。

P0.0~P0.7 (39-32): 双向I/O口P0。第二功能是在访问外部存储器时,可分时分作低8位地址和8位数据线,在对8751编程和校验时,用于数据的输入和输出。P0口能以吸收电流的方式驱动8个LS型TTL负载。

P2.0~P2.7 (21~28): 双向I/O口P2。P2口可以驱动(吸收或输出电流)4个LS型TTL负载。第二功能是在访问外部存储器时,输出高8位地址。在对EPROM编程和校验时,它接收高位地址。

## 1.2 8051单片机的存储器结构

图1.5为8051单片机的存储器结构图。在物理上,它有3个存储器空间:程序存储器

(CODE空间)、片内数据存储器 (IDATA和DATA空间)、片外数据存储器 (XDATA空间)。访问不同存储器空间时须采用不同的指令。

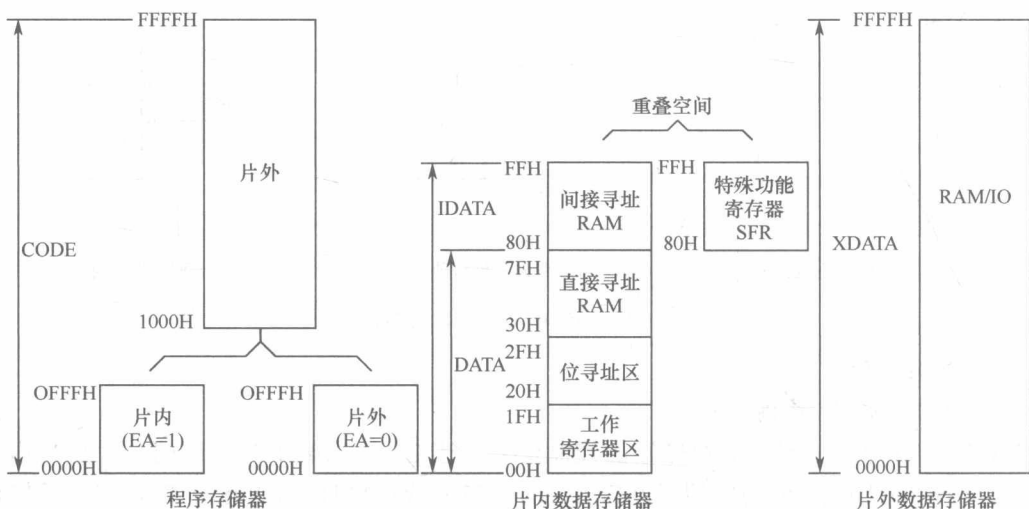


图1.5 8051单片机的存储器结构图

### 程序存储器ROM

8051单片机程序存储器ROM空间大小为64KB，地址范围为0000H~FFFFH，用于存放程序代码和一些表格常数，称为CODE空间。8051单片机专门提供一个引脚 $\overline{EA}$ 来区分片内ROM和片外ROM。 $\overline{EA}$ 引脚接高电平时，单片机从片内ROM中读取指令，当指令地址超过片内ROM空间范围后，就自动转向片外ROM读取指令； $\overline{EA}$ 引脚接低电平时，所有的取指操作均对片外ROM进行。程序存储器的某些地址单元是保留给系统使用的：0000H~0002H单元是所有执行程序的入口地址，复位后，CPU总是从0000H地址开始执行程序；0003H~002BH单元均匀地分为5段，用于5个中断服务程序的入口，产生某个中断时，将自动进入其对应入口地址开始执行中断服务程序，一些新型8051单片机增加了更多的中断源，它们的中断入口地址也相应增加。

### 片外数据存储器RAM

8051单片机片外数据存储器RAM空间大小为64KB，地址范围为0000H~FFFFH，称为XDATA空间。在XDATA空间内进行分页寻址操作时，称为PDATA区。

8051单片机采用所谓“哈佛”结构的存储器配置，即在物理上具有独立的ROM存储器和片外RAM数据存储器，而在逻辑上则采用相同的地址空间，其地址范围都是0000H~FFFFH，但是需要采用不同的指令和寻址方式来访问，从而可分别寻址64KB的ROM程序存储器和64KB的片外RAM数据存储器。

### 片内数据存储器RAM

8051单片机片内数据存储器RAM的空间最大为256B，用于存放程序执行过程的各种变量及临时数据，整个片内RAM的地址范围00H~FFH被称为IDATA空间。片内RAM低128个字节（00H~7FH）被称为DATA空间。它既可用直接寻址访问，也可用间接寻址访问，而片内RAM高128个字节（80H~FFH）则只能采用间接寻址访问。片内RAM中00H~1FH地址范围被称为工作寄存器区，平均分为4组，每组都有8个工作寄存器R0~R7，在某一时刻，

CPU只能使用其中的一组工作寄存器，究竟选择哪一组工作寄存器，则由程序状态字寄存器PSW中RS0和RS1的状态决定，见表1.1。片内RAM中，20H~2FH地址范围被称为位寻址区（又称BDATA区）。其中每个存储器单元的每一位称为一个bit，可以用位处理指令直接操作。片内RAM中的位地址分配如图1.6所示。

RAM 地址	MSB								LSB
7FH									127
2FH	7E	7E	7D	7C	7B	7A	70	78	47
2FH	77	76	75	74	73	72	71	70	46
2DH	6F	6E	6D	6C	6B	6A	69	68	45
2CH	67	66	65	64	63	62	61	60	44
2BH	5F	5E	5D	5C	5B	5A	59	58	43
2AH	57	56	55	54	53	52	51	50	42
29H	4F	4E	4D	4C	4B	4A	49	48	41
28H	47	46	45	44	43	42	41	40	40
27H	3F	3E	3D	3C	3B	3A	39	38	39
26H	37	36	35	34	33	32	34	30	38
25H	2F	2E	2D	2C	2B	2A	29	28	37
24H	27	26	25	24	23	22	21	20	36
23H	1F	1E	1D	1C	1B	1A	19	18	35
22H	17	16	15	14	13	12	11	10	34
21H	0F	0E	0D	0C	0B	0A	09	08	33
20H	07	06	05	04	03	02	01	00	32
1FH	工作寄存器3区								31
18H									24
17H	工作寄存器2区								23
10H									16
0FH	工作寄存器1区								15
08H									8
07H	工作寄存器0区								7
00H									0

图1.6 片内RAM中的位地址分配

8051单片机在与IDATA空间高128个字节（80H~FFH地址范围）安排了一个重叠空间被称为特殊功能寄存器区（又称SFR区），地址也为80H~FFH，但在使用时，可通过指令加以区别。有些特殊功能寄存器是可以位寻址的，其可寻址位被称为sbit。表1.2为8051单片机特殊功能寄存器地址及符号表。表中带\*号的为可位寻址的特殊功能寄存器。片内RAM中的各个单元都可以通过其地址来寻找。对于工作寄存器，一般使用R0~R7表示，对于特殊功能寄存器，也是直接用其符号名较为方便。需要指出的是，8051单片机的堆栈必须使用片内RAM，而片内RAM空间十分有限，因此要仔细安排堆栈指针SP的值，以保证不会发生堆栈溢出而导致系统崩溃。