课后答案网 www.khdaw.com

《单片机原理及应用》思考题与习题参考答案

绪论

0.1 解：

单片微型计算机（Single-Chip Microcomputer），简称单片机。就是将微处理器（CPU）、存储器（存放程序或数据的 ROM 和 RAM）、总线、定时器/计数器、输入/输出接口（I/O 口）和其他多种功能器件集成在一块芯片上的微型计算机。

单片机的主要特点有：（1）可靠性高（2）便于扩展（3）控制功能强（4）低电压、低功耗（5）片内存储容量较小，除此之外，单片机还具有集成度高、体积小、性价比高、应用广泛、易于产品化等特点

0.2 解：

当前单片机的主要产品有：Intel的 8051 系列、Motorola的M68HC系列、Philips（飞利浦）的 80C51 系列、台湾义隆公司（EMC）EM78 系列单片机、美国Microchip公司的PIC单片机系列、Atmel公司的AT90 系列单片机Ubicom公司的Scenix单片机、日本爱普生科技公司的Epson单片机、Zilog公司的Z86 系列、美国国家半导体公司NSCOP8 单片机、台湾Winbond（华邦）的W78 系列等。

MCS-51 为主流产品。

MSP430 的功能较强。是一种特低功耗的 Flash 微控制器。主要用于三表及超低功耗场合。

EM78 系列单片机采用高速 CMOS 工艺制造，低功耗设计为低功耗产品，价格较低。具有三个中断源、R-OPTION 功能、I/O 唤醒功能、多功能 I/O 口等。具有优越的数据处理性能，采用 RISC 结构设计。

Microship 单片机是市场份额增长较快的单片机。它的主要产品是 PIC 系列 8 位单片机。CPU 采用 RISC 结构，运行速度快，价格低适于用量大、档次低、价格敏感的产品。

Motorola 是世界上最大的单片机生产厂家之一，品种全、选择余地大、新产品多。其特点是噪声低，抗干扰能力强，比较适合于工控领域及恶劣的环境。

AVR 是增强 RISC 内载 Flash 的单片机，单片机内部 32 个寄存器全部与 ALU 直接连接，突破瓶颈限制，每 1MHz 可实现 1MIPS 的处理能力，为高速、低功耗产品。端口有较强的负载能力，可以直接驱动 LED。支持 ISP、IAP，I/O 口驱动能力较强。

Scenix单片机除传统的I/O功能模块如并行I/O、UART、SPI、I^2C、A/D、PWM、PLL、DTMF等，增加了新的I/O模块（如USB、CAN、J1850、虚拟I/O等）。其特点是双时钟设置，指令运行速度较快，具有虚拟外设功能，柔性化I/O端口，所有的I/O端口都可单独编程设定。

Epson 单片机主要为日本爱普生科技公司生产的 LCD 配套。其单片机的特点是 LCD 驱动部分性能较好，低电压、低功耗。

Z8 单片机是 Zilog 公司的主要产品，采用多累加器结构，有较强的中断处理能力。价格低。

COP8 单片机片内集成了 16 位 A/D，内部使用了抗电磁干扰 EMI（Electro Magnetic Interference）电路，在看门狗电路及单片机的唤醒方式上都有独到之处。程序加密控制功能也比较好。

课后答案网 www.khdaw.com

W78 系列与标准的 8051 兼容，W77 系列为增强型 51 系列，对 8051 的时序作了改进，在同样时钟频率下，速度提高 2.5 倍。Flash ROM 容量从 4KB 到 64KB，有 ISP 功能。

0.3 解：

（1）第一阶段（1974—1976 年）：制造工艺落后，集成度低，而且采用了双片形式。典型的代表产品有 Fairchild 公司的 F8 系列。其特点是：片内只包括了 8 位 CPU，64B 的 RAM 和两个并行口，需要外加一块 3851 芯片（内部具有 1KB 的 ROM、定时器/计数器和两个并行口）才能组成一台完整的单片机。

（2）第二阶段（1977—1978 年）：在单片芯片内集成 CPU、并行口、定时器/计数器、RAM 和 ROM 等功能部件，但性能低，品种少，应用范围也不是很广。典型的产品有 Intel 公司的 MCS-48 系列。其特点是，片内集成有 8 位的 CPU，1KB 或 2KB 的 ROM，64B 或 128B 的 RAM，只有并行接口，无串行接口，有 1 个 8 位的定时器/计数器，中断源有 2 个。片外寻址范围为 4KB，芯片引脚为 40 个。

（3）第三阶段（1979—1982 年）： 8 位单片机成熟的阶段。其存储容量和寻址范围增大，而且中断源、并行 I/O 口和定时器/计数器个数都有了不同程度的增加，并且集成有全双工串行通信接口。在指令系统方面增设了乘除法、位操作和比较指令。其特点是，片内包括了 8 位的 CPU，4KB 或 8KB 的 ROM，128B 或 256B 的 RAM，具有串/并行接口，2 个或 3 个 16 位的定时器/计数器，有 5～7 个中断源。片外寻址范围可达 64KB，芯片引脚为 40 个。代表产品有 Intel 公司的 MCS-51 系列，Motorola 公司的 MC6805 系列，TI 公司的 TMS7000 系列，Zilog 公司的 Z8 系列等。

（4）第四阶段（1983 年至今）： 16 位单片机和 8 位高性能单片机并行发展的时代。16 位机的工艺先进，集成度高，内部功能强，运算速度快，而且允许用户采用面向工业控制的专用语言，其特点是，片内包括了 16 位的 CPU，8KB 的 ROM，232B 的 RAM，具有串/并行接口，4 个 16 位的定时器/计数器，有 8 个中断源，具有看门狗（Watchdog），总线控制部件，增加了 D/A 和 A/D 转换电路，片外寻址范围可达 64KB。代表产品有 Intel 公司的 MCS-96 系列，Motorola 公司的 MC68HC16 系列，TI 公司的 TMS9900 系列，NEC 公司的 783××系列和 NS 公司的 HPC16040 等。然而，由于 16 位单片机价格比较贵，销售量不大，大量应用领域需要的是高性能、大容量和多功能的新型 8 位单片机。

近年来出现的 32 位单片机，是单片机的顶级产品，具有较高的运算速度。代表产品有 Motorola 公司的 M68300 系列和 Hitachi（日立）公司的 SH 系列、ARM 等。

0.4 解：

面对工控领域对象，嵌入到工控应用系统中，实现嵌入式应用的计算机称之为嵌入式计算机系统，简称嵌入式系统。单片机从体系结构到指令系统都是按照嵌入式应用特点专门设计的，能最好的满足对控制对象、应用系统的嵌入、现场的可靠运行以及非凡的控制品质要求。

0.5 解：

选择原则：主要从指令结构、运行速度、程序存储方式和功能等几个方面选择单片机。

MCS-51 为主流产品。

Motorola 是世界上最大的单片机厂商。品种全、选择余地大、新产品多。其特点是噪声低，抗干扰能力强，比较适合于工控领域及恶劣的环境。

课后答案网 www.khdaw.com

Microship 单片机是市场份额增长较快的单片机。它的主要产品是 PIC 系列 8 位单片机。其特点是运行速度快，低价位，适用于量大、档次低、价格敏感的产品。

美国德州仪器（TI）公司生产的 MSP430 系列单片机是一种特低功耗的 Flash 微控制器。主要用于三表及超低功耗场合。

在嵌入式系统低端的单片机领域，Intel 公司的 MCS-51 形成了既具有经典性又不乏生命力的一个单片机系列。许多半导体厂家、电器公司以 MCS-51 系列中的 8051 为基核，推出了许多兼容性的 CHMOS 单片机----80C51 系列。

此外，还有 Zilog、EM78、Senix、NEC、Epson、NS、三星、富士通、华邦、Philips、ARM 等单片机。

第 1 章

1.1 解：

MCS-51 为主流产品。

MSP430 的功能较强。是一种特低功耗的 Flash 微控制器。主要用于三表及超低功耗场合。

EM78 系列单片机采用高速 CMOS 工艺制造，低功耗设计为低功耗产品，价格较低。具有三个中断源、R-OPTION 功能、I/O 唤醒功能、多功能 I/O 口等。具有优越的数据处理性能，采用 RISC 结构设计。

PIC 系列 8 位单片机是 Microship 公司的产品。CPU 采用 RISC 结构，运行速度快，价格低适于用量大、档次低、价格敏感的产品。

Motorola 是世界上最大的单片机生产厂家之一，品种全、选择余地大、新产品多。其特点是噪声低，抗干扰能力强，比较适合于工控领域及恶劣的环境。

AVR 是增强 RISC 内载 Flash 的单片机，单片机内部 32 个寄存器全部与 ALU 直接连接，突破瓶颈限制，每 1MHz 可实现 1MIPS 的处理能力，为高速、低功耗产品。端口有较强的负载能力，可以直接驱动 LED。支持 ISP、IAP，I/O 口驱动能力较强。

1.2 解：

MCS-51 单片机在片内集成了中央处理器（CPU）、程序存储器（ROM）、数据存储器（RAM）、定时器/计数器、并行 I/O 接口、串行 I/O 接口和中断系统等几大单元。

CPU 是整个单片机的核心部件，由运算器和控制器组成。运算器可以完成算术运算和逻辑运算，其操作顺序在控制器控制下进行。控制器是由程序计数器 PC（Program Counter）、指令寄存器 IR（Instruction Register）、指令译码器 ID（Instruction Decoder）、定时控制逻辑和振荡器 OSC 等电路组成。CPU 根据 PC 中的地址将欲执行指令的指令码从存储器中取出，存放在 IR 中，ID 对 IR 中的指令码进行译码，定时控制逻辑在 OSC 配合下对 ID 译码后的信号进行分时，以产生执行本条指令所需的全部信号。

程序存储器（ROM）用于存储程序、常数、表格等。

数据存储器（RAM）用于存储数据。

8051 内部有两个 16 位可编程序的定时器/计数器 T0 和 T1，均为二进制加 1 计数器。可用于定时和对外部输入脉冲的计数。

8051 的中断系统主要由中断允许控制器 IE 和中断优先级控制器 IP 等电路组成。可实现对 5 个中断源的管理。8051 的中断系统主要由中断允许控制器 IE 和中断优先级控制器 IP 等

课后答案网 www.khdaw.com

电路组成。其中，IE 用于控制 5 个中断源中哪些中断请求被允许向 CPU 提出，哪些中断源的中断请求被禁止；IP 用于控制 5 个中断源的中断请求的优先权级别。

I/O 接口是 MCS-51 单片机对外部实现控制和信息交换的必经之路，用于信息传送过程中的速度匹配和增加它的负载能力。可分为串行和并行 I/O 接口。

1.3 解：

80C51 单片机有 4 个 I/O 端口，每个端口都是 8 位双向口，共占 32 根引脚。每个端口都包括一个锁存器（即专用寄存器 P0～P3）、一个输入驱动器和输入缓冲器。通常把 4 个端口称为 P0～P3。在无片外扩展的存储器的系统中，这 4 个端口的每一位都可以作为双向通用 I/O 端口使用。在具有片外扩展存储器的系统中，P2 口作为高 8 位地址线，P0 口分时作为低 8 位地址线和双向数据总线。

1.4 解：

控制线一共有 6 条：

（1）ALE/$\overline{PROG}$：地址锁存允许/编程线，配合P0 口引脚的第二功能使用。在访问片外存储器时，8051 CPU在P0.7～P0.0 引脚上输出片外存储器低 8 位地址的同时在ALE/$\overline{PROG}$ 上输出一个高电位脉冲，用于把这个片外存储器低 8 位地址锁存到外部专用地址锁存器，以便空出P0.7～P0.0 引脚线去传送随后而来的片外存储器读写数据。在不访问片外存储器时，8051 自动在ALE/$\overline{PROG}$ 上输出频率为fosc/6 的脉冲序列。该脉冲序列可用作外部时钟源或作为定时脉冲源使用。

（2）$\overline{EA}$/Vpp：允许访问片外存储器 / 编程电源线，可以控制 8051 使用片内 ROM还是使用片外 ROM。若 $\overline{EA}$ = 0，则允许使用片内 ROM；若 $\overline{EA}$ = 1 则允许使用片外 ROM。

（3）$\overline{PSEN}$：片外 ROM选通线，在执行访问片外 ROM的指令MOVC时，8051 自动在 $\overline{PSEN}$ 上产生一个负脉冲，用于为片外 ROM芯片的选通。其他情况下 $\overline{PSEN}$ 线均为高电平封锁状态。

（4）RST/V_{PD}：复位 / 备用电源线，可以使 8051 处于复位工作状态。

1. 5 解：

指令：人为输入计算机，由计算机识别并执行一步步操作的命令的形式称为指令。

程序：一系列指令的有序集合称为程序。

程序在计算机中是按序执行的，CPU 通过程序计数器 PC 控制程序的执行顺序，一般情况下程序是按序执行的，当执行转移、调用、返回等指令时，程序转移到相应的目的地址处执行。CPU 根据程序计数器 PC 中的地址将欲执行指令的指令码从存储器中取出，存放在 IR 中，ID 对 IR 中的指令码进行译码，定时控制逻辑在 OSC 配合下对 ID 译码后的信号进行分时，产生执行本条指令所需的全部信号，完成本条指令的执行。

1.6 解：

（1）80C51 在结构上的特点

80C51 采用将程序存储器和数据存储器截然分开，分别寻址的结构，称为 Harvard

课后答案网 www.khdaw.com

结构。

（2）在物理和逻辑上的地址空间

存储器上设有 4 个存储器空间：片内程序存储器、片外程序存储器、片内数据存储器、片外数据存储器。

在逻辑上有 3 个存储器地址空间：片内、片外统一的 64KB 程序存储器地址空间，片内 256B 数据存储器地址空间，片外 64KB 的数据存储器地址空间。

（3）访问格式

访问片内 RAM 采用 MOV 格式

访问片外 RAM 采用 MOVX 格式

1.7 解：

堆栈是一个特殊的存储区。主要功能是暂时存放数据和地址，通常用来保护断点和现场。它的特点是按照“先进后出”的存取数据。堆栈指针 SP 是一个 8 位寄存器，是用于指示堆栈的栈顶地址的寄存器，它决定了堆栈在内部 RAM 中的物理位置。

1.8 解：

8051 有 21 个特殊功能寄存器（PC除外），它们被离散地分布在内部RAM的 80H～FFH地址单元中，共占据了 128 个存储单元，其中有 83 位可以位寻址。特殊功能寄存器SFR的分布和功能见教材P_{18}表 1.2.2。

1.9 解：

8051 单片机的 4 个 I／O 口在结构上是基本相同的，但又各具特点。这四个端口都是 8 位双向口，每个端口都包括一个锁存器、一个输出驱动器和输入缓冲器。在无片外扩展存储器的系统中，这四个端口的每一位都可以作为双向通用 I／O 端口使用。在作为一般的通用 I／O 输入时，都必须先向锁存器写入“1”，使输出驱动场效应管 FET 截止，以免误读数据。各自特点如下：

（1）P0 口为双向 8 位三态 I/O 口，它既可作为通用 I/O 口，又可作为外部扩展时的数据总线及低 8 位地址总线的分时复用口。作为通用 I/O 口时，输出数据可以得到锁存，不需外接专用锁存器；输入数据可以得到缓冲，增加了数据输入的可靠性。每个引脚可驱动 8 个 TTL 负载。

（2）P1 口为 8 位准双向 I/O 口，内部具有上拉电阻，一般作通用 I/O 口使用，它的每一位都可以分别定义为输入线或输出线，作为输入时，锁存器必须置 1。每个引脚可驱动 4 个 TTL 负载。

（3）P2 口为 8 位准双向 I/O 口，内部具有上拉电阻，可直接连接外部 I/O 设备。它与地址总线高 8 位复用，可驱动 4 个 TTL 负载。一般作为外部扩展时的高 8 位地址总线使用。

（4）P3 口为 8 位准双向I/O口，内部具有上拉电阻，它是双功能复用口，每个引脚可驱动 4 个TTL负载。作为通用I/O口时，功能与P1 口相同，常用第二功能。作为第二功能使用时，各位的作用见教材P_{23}表 1.2.5 所示。

1.10 解：

数据指针 DPTR 是 16 位的专用寄存器，它由两个 8 位的寄存器 DPH（高 8 位）和 DPL（低 8 位）组成。专门用来寄存片外 RAM 及扩展 I/O 口进行数据存取时的地址。编程时，既可以按 16 位寄存器来使用，也可以按两个 8 位寄存器来使用（即高位字节寄存器 DPH 和

课后答案网 www.khdaw.com

低位字节寄存器 DPL）。

DPTR 主要是用来保存 16 位地址，当对 64KB 外部数据存储器寻址时，可作为间址寄存器使用，

1.11 解：

程序状态字PSW是 8 位寄存器，用于存放程序运行的状态信息，PSW中各位状态通常是在指令执行的过程中自动形成的，但也可以由用户根据需要采用传送指令加以改变。各个标志位的意义如下：

PSW.7（C_y）：进位标志位。

PSW.6（AC）：辅助进位标志位，又称为半进位标志位。

PSW.5（F0）：用户标志位。

PSW.4、PSW.3（RS1 和 RS0）：寄存器组选择位。

PSW.2（OV）：溢出标志位。

PSW.1（空缺位）：此位未定义。

PSW.0（P）：奇偶校验位。

1.12 解：

开机复位后，ＣＰＵ使用的是第 0 组工作寄存器。它们的地址是００Ｈ－０７Ｈ。ＣＰＵ通过对程序状态字ＰＳＷ中ＲＳ１和ＲＳ０的设置来确定和改变当前工作寄存器组。

1 .13 解：

8051 片内数据存储器有２５６Ｂ寻址空间。存储器空间的地址范围为：00H——FFH

在这个存储器空间又可分为：

基本的数据存储区：００Ｈ——７ＦＨ，可划分为工作寄存器、位寻址区、堆栈与数据缓冲区。

ＳＦＲ空间：８０Ｈ——ＦＦH

1.14 解：

片内ＲＡＭ低１２８单元划分及主要功能如下：

（1）工作寄存器组（００Ｈ——１ＦＨ）

这是一个用寄存器直接寻址的区域，内部数据ＲＡＭ区的 0—３１，共３２个单元。它是 4 个通用工作寄存器组，每个组包含 8 个 8 位寄存器，编号为Ｒ０——Ｒ７。

（2）位寻址区（２０Ｈ——２ＦＨ）

１６个字节单元，共包含１２８位，这１６个字节单元既可以进行字节寻址，又可以实现位寻址。主要用于位寻址。

（3）堆栈与数据缓冲区（３０Ｈ——７ＦＨ）

用于设置堆栈、存储数据。

1.15 解：

程序存储器通过$\overline{EA}$/Vpp引脚选择。若$\overline{EA}$＝０，则允许使用片内ＲＯＭ；若$\overline{EA}$＝１则允许使用片外ＲＯＭ。

数据存储器通过指令区分：访问片内ＲＡＭ采用ＭＯＶ指令，访问片外ＲＡＭ采用ＭＯＶＸ指令。

1.16 解：

课后答案网 www.khdaw.com

时钟周期又称为振荡周期，由单片机内部振荡电路 OSC 产生，定义为 OSC 时钟频率的倒数。时钟周期又称为节拍（用 P 表示）。时钟周期是时序中的最小单位。一个状态有两个节拍，

机器周期定义为实现特定功能所需的时间。MCS-51 的机器周期由 12 个时钟周期构成。

执行一条指令所需要的时间称为指令周期，指令周期是时序中的最大单位。由于机器执行不同指令所需的时间不同，因此不同指令所包含的机器周期数也不尽相同。MCS-51 的指令可能包括 1～4 个不等的机器周期。

当 MCS-51 的主频为 12MHz 时，一个机器周期为 1μs。执行一条指令需要的最长时间为 4μs。

1.17 解：

8051 单片机复位后机器的初始状态，即各寄存器的状态：P C之外，复位操作还对其它一些特殊功能寄存器有影响，它们的复位状态如下：

寄存器	复位时内容	寄存器	复位时内容
PC	0000 H	TCON	0×000000B
ACC	00H	TL0	00H
B	00H	TH0	00H
PSW	00H	TH1	00H
SP	07H	TH1	00H
DPTR	0000H	SCON	00H
P0—P3	FFH	SBUF	不确定
TMOD	××000000B	PCON	0×××0000B

单片机复位方法有：上电自动复位，按键电平复位和外部脉冲三种方式。

第 2 章

2.1 解：

8051 的指令系统由 111 条指令组成。如果按字节数分类，有 49 条单字节指令 46 条双字节指令和 16 条三字节指令，以单字节指令为主；如果按照指令执行时间分类，有 64 条单周期指令、45 条双周期指令和 2 条四周期指令，以单周期指令为主。

8051 的指令系统具有以下特点：

（1）存储效率高、执行速度快，可以进行直接地址到直接地址的数据传送，能把一个并行 I/O 口中的内容传送到内部 RAM 单元中而不必经过累加器 A 或工作寄存器 Rn。这样可以大大提高传送速度和缓解累加器 A 的瓶颈效应。

（2）用变址寻址方式访问程序存储器中的表格，将程序存储器单元中的固定常数或表格字节内容传送到累加器 A 中。这为编成翻译算法提供了方便。

（3）在算术运算指令中设有乘法和除法指令

（4）指令系统中一些对 I/O 口进行操作的指令具有“读——修改——写”的功能。这一功能指:在执行读锁存器的指令时，CPU 首先完成将锁存器的值通过缓冲器 BUF2 度入内部，进行修改、改变，然后重新写到锁存器中去。这种类型指令包含所有的逻辑操作和位操作指令。

课后答案网 www.khdaw.com

（5）8051 单片机内部有一个布尔处理器，对为地址空间具有丰富的位操作指令。布尔操作类指令有 17 条，包括布尔传送指令、布尔状态控制指令、布尔逻辑操作指令、布尔条件转移指令。

2.2 解：

MCS-51 单片机指令系统按功能可分为 5 类：

（1）数据传送指令

（2）算术运算指令

（3）逻辑运算和移位指令

（4）控制转移指令

（5）位操作指令

MCS-51 单片机的指令系统提供了七种寻址方式，其对应的寻址范围如下表：

寻址方式	使用的变量	寻址范围
立即寻址		程序存储器
直接寻址		内部 RAM 低 128 个字节； 特殊功能寄存器 SFR
寄存器寻址	R0~R7；A、B、DPTR、C	
寄存器间接寻址	@R0、@R1、SP	内部 RAM、堆栈指针 SP
	@R0、@R1、@DPTR	外部 RAM
变址寻址	@A+PC、@A+DPTR	程序存储器
相对寻址	PC+偏移量	程序存储器
位寻址		内部 RAM 低 128B 位寻址区 可位寻址的特殊功能寄存器位

2.3 解：

访问特殊功能寄存器，应采用直接寻址、位寻址方式。

访问外部数据存储器，应采用寄存器间接寻址方式。

在 0～255B 范围内，可用寄存器 R0、R1 间接寻址：

MOVX A，@R0 或 MOVX A，@R1

MOVX @R0，A 或 MOVX @R1，A

在 0～64KB 范围内，可用 16 位寄存器 DPTR 间接寻址：

MOVX A，@DPTR

MOVX @DPTR，A

2.4 解：

这条指令是在进行 BCD 码加法运算时，跟在“ADD”和“ADDC”指令之后，用来对 BCD 码的加法运算结果自动进行修正的，使其仍为 BCD 码表达形式。

在计算机中，遇到十进制调整指令时，中间结果的修正是由 ALU 硬件中的十进制修正电路自动进行的。用户不必考虑何时该加“6”，使用时只需在上述加法指令后面紧跟一条“DA A”指令即可。

2. 5 解：

虽然内部 RAM 位寻址区的位地址范围 00H～7FH 与低 128 个单元的单元地址范围 00H～7FH 形式完全相同，但是在应用中可以通过指令的类型区分单元地址和位地址。

位寻址的操作只适用于下列位指令，而直接寻址操作对这些指令是无效的。

MOV C，bit　　MOV bit，C　　CLR bit

课后答案网 www.khdaw.com

SETB bit　　ANLC，/bit　　JNB bit，rel
CPL bit　　JB bit，rel
ANL C，bit

2.6 解:

在实际应用中，可从寻址范围和指令长度两个方面来选择长跳转指令 LJMP 和短跳转指令 AJMP、长调用指令 LCALL 和短调用指令 ACALL。

长跳转 LJMP 在 64KB 范围内转移，而短跳转 AJMP 只能在 2KB 空间转移。长调用 LCALL 调用位于 64KB 程序空间的子程序，而短调用 ACALL 调用位于 2KB 程序空间范围的子程序。AJMP、ACALL 指令代码长度为 2 个字节；LJMP、LCALL 指令代码长度为 3 个字节。

2.7 解:

指令的转移范围不同。

SJMP 是 256B 范围内的相对转移指令，AJMP 是 2KB 范围内的无条件短跳转指令，LJMP 是 64KB 范围内的无条件长跳转指令。

2.8 解:

MOVC　A，　@A+DPTR：访问外部程序存储器指令，功能为程序存储器内容送累加器。

MOVX A，@DPTR：访问外部数据存储器指令，功能为外部数据存储器内容送累加器指令.

2.9 解:

使用@A+DPTR 基址变址寻址时，DPTR 为常数且是表格的首地址，A 为从表格首址到被访问字节地址的偏移量。

使用@A+PC 基址变址寻址时，PC 仍是下条指令首地址，而 A 则是从下条指令首地址到常数表格中的被访问字节的偏移量。

2.10 解:

结果为：（A）＝30H　（R0）=50H　（50H）＝00H　（51H）＝30H

2.11 解:

结果为：(61H)=24H　(62H)=10H　(30H)=00H
(31H)=0FFH　DPTR=2410H　SP=60H

2.12 解:

指令	源操作数的寻址方式	执行指令后的结果
MOV A,40H	直接寻址	(A)=50H
MOV R0,A	寄存器寻址	(R0)=50H
MOV P1,#80H	立即寻址	(P1)=80H
MOV @R0,20H	寄存器间接寻址	(50H)=30H
MOV DPTR,#2000H	立即寻址	(DPTR)=2000H
MOV 40H,30H	直接寻址	(40H)=40H
MOV R0,30H	直接寻址	(R0)=40H
MOV D0H,R0	直接寻址	(D0H)=40H
MOV 10H,#10H	立即寻址	(10H)=10H
MOV A,@R0	寄存器间接寻址	(A)=50H
MOV P2,P1	寄存器寻址	(P2)=80H

2.13 解:

课后答案网 www.khdaw.com

（1）R1 的内容传送到 R0 ；

```
MOV     A,   R1
MOV     R0,      A
```

（2）片外 RAM 20H 单元内容送 R0 ；

```
MOV      DPTR,     #0020H
MOVX     A,        @DPTR
MOV      R0,       A
```

（3）片外 RAM 20H 单元的内容送片内 RAM 20H 单元；

```
MOV      DPTR,     #0020H
MOVX     A,        @DPTR
MOV      20H,      A
```

（4）片外 RAM 1000H 单元的内容送片内 RAM 20H 单元；

```
MOV      DPTR,     #1000H
MOVX     A,        @DPTR
MOV      20H,      A
```

（5）ROM 2000H 单元的内容送 R0 单元；

```
MOV      A,        #00H
MOV      DPTR,     #2000H
MOVC     A,@A+DPTR
MOV      R0H,      A
```

（6）ROM 2000H 单元的内容送片内 RAM 20H 单元；

```
MOV      A,        #00H
MOV      DPTR,     #2000H
MOVC     A,@A+DPTR
MOV      20H,      A
```

（7）ROM 2000H 单元的内容送片外 RAM 20H 单元。

```
MOV      A,        #00H
MOV      DPTR,     #2000H
MOVC     A,@A+DPTR
MOV      DPTR,     #0020H
MOVX     @DPTR,    A
```

2.14 解:

```
        ORG   0000H
              LJMP START
              ORG 0030H
START:        MOV A，22H
              ADD A，32H
              DA A
              MOV 42H，A
              MOV A，23H
```

课后答案网 www.khdaw.com

```
        ADDC A，33H
        DA A
        MOV 43H，A
        SJMP $
        END
```

2.15 解:

```
        ORG 0000H
        LJMP MAIN
        ORG 0030H
MAIN:   MOV R7，#20H
        MOV R1，#40H
        MOV DPTR，#3000H
LOOP:   MOV A，@R1
        MOVX @DPTR，A
        INC R1
        INC DPTR
        DJNZ R7，LOOP
        SJMP $
        END
```

2.16 解:

```
        ORG 0000H
        LJMP MAIN
        ORG 0030H
MAIN:   MOV R0，#30H
        MOV R1，#00H
        MOV R2，#00H
        MOV R3，#07H
LP2:    MOV A，@R0
        ADD A，R2
        MOV R2，A
        JNC LP1
        INC R1
LP1:    INC R0
        DJNZ R3，LP2
        MOV R3，#03H
LP3:    CLR C
        MOV A，R1
        RRC A
        MOV R1，A
        MOV A，R2
        RRC A
        MOV R2，A
        DJNZ R3，LP3
        MOV 3AH，R2
        SJMP $
        END
```

2.17 解:

```
        ORG 0000H
        LJMP START
        ORG 0030H
START:  MOV DPTR，#2001H
        MOVX A，@DPTR
        MOV 30H，A
        MOV DPTR，#2002H
        MOVX A，@DPTR
        ADD A，30H
        MOV B，A
        MUL  AB
        MOV R1，A
        CJNE A，#10，NET1
        MOV DPTR，#2000H
        MOVX @DPTR，A
        SJMP NET3
NET1:   JNC NET2
        CLR C
        MOV A，R1
        SUBB A，#10
        MOV DPTR，#2000H
        MOVX @DPTR，A
        SJMP NET3
NET2:   MOV A，R1
        ADD A，#10
        MOV DPTR，#2000H
        MOVX @DPTR，A
NET3:   SJMP $
        END
```

2.18 解:

```
        ORG     0000H
        LJMP    MAIN
        ORG     0030h
MAIN:   MOV     DPTR,#2000H
        MOV     A,#OFFH
        MOVX    @DPTR,A
        MOV     DPTR,#2100H
        MOV     A,#34H
```

课后答案网 www.khdaw.com

```
        MOVX    @DPTR,A
        MOV     DPTR,#2008H
        MOV     A,#33H
        MOVX    @DPTR,A
        MOV     DPTR,#2108H
        MOV     A,#44H
        MOVX    @DPTR,A
        MOV     DPTR,#200EH
        MOV     A,#0EEH
        MOVX    @DPTR,A
        MOV     DPTR,#210EH
        MOV     A,#32H
        MOVX    @DPTR,A
        MOV     DPTR,#2000H
        MOV     R1,#30H
        MOV     R2,#15
LOOP:   MOVX    A,@DPTR
        MOV     @R1,A
    INC     DPTR
    INC     R1
    DJNZ    R2,LOOP
    MOV     R1,#30H
    MOV     DPTR,#2100H
    MOV     R2,#15
    MOV     R0,#40H
LOOP2:
        MOVX    A,@DPTR
        CLR     C
        ADDC    A,@R1
        INC     LOOP1
        MOV     @R0,@01
        SJMP    LOOP4
LOOP1:
        MOV     @R0,#00
LOOP4:
        INC     R0
        MOV     @R0,A
        INC     R1
        INC     DPTR
        INC     R0
        DJNZ    R2,LOOP2
        MOV     R0,#40H
        MOV     DPTR,#2200H
        MOV     R2,#30
LOOP3:
            MOV     A,@R0
            MOVX    @DPTR,A
            INC     R0
    INC     DPTR
    DJNZ    R2,LOOP3
    SJMP    $
    END
```

2.19 解：

```
        ORG 0000H
        LJMP MAIN
        ORG 0030H
MAIN：  MOV R2，#100
        MOV R3，#00H
        MOV R4，#00H
        MOV R5，#00H
        MOV DPTR，#2000H
LOOP：  MOVX A，@DPTR
        CJNE A，#00H，NET1
        INC R3
        INC DPTR
        DJNZ R2，LOOP
        SJMP NET3
NET1：  JC NET2
        INC R4
        INC DPTR
        DJNZ R2，LOOP
        SJMP NET3
NET2：  INC R5
        INC DPTR
        DJNZ R2，LOOP
NET3：  SJMP $
        END
```

2.20 解：

课后答案网 www.khdaw.com

```
        ORG     0000H
        LJMP    MAIN
        ORG     0030H
MAIN:
        MOV     DPTR,#1000H
        MOV     A,#22H
        MOVX    @DPTR,A
        MOV     DPTR,#1030H
        MOV     A,#33H
        MOVX    @DPTR,A
        MOV     DPTR,#1000H
        MOV     R2,#31H
        MOV     R0,#30H
LOOP:
        MOVX    A,@DPTR
        MOV     @R0,A
        INC     DPTR
        INC     R0
        DJNZ    R2,LOOP
        MOV     DPTR,#1000H
        MOV     A,#00
        MOV     R2,#31H
LOOP1:
        MOVX    @DPTR,A
        INC     DPTR
        DJNZ    R2,LOOP1
        SJMP    $
        END
```

2.21 解:

```
        ORG 0000H
        LJMP MAIN
        ORG 0100H
MAIN:   MOV DPTR，#2040H
        MOV R2，#50
        CLR 7FH
        DEC R2
LS:     MOVX A，@DPTR
        MOV 20H，A
        INC DPTR
        MOVX A，@DPTR
        MOV 21H，A
        MOV A，20H
        CJNE A，21H，LOOP
LOOP:   JNC LOOP1
        MOV A，20H
        MOVX @DPTR，A
        DEC DPTR
        MOV A，21H
        MOVX @DPTR，A
        INC DPTR
        SETB 7FH
LOOP1:  DJNZ R2，LS
        JB 7FH，MAIN
        SJMP $
        END
```

2.22 解:

```
        ORG     0000H
        LJMP    MAIN
        ORG     0030H
MAIN:
        LCALL   SUBONE
        AJMP    $
; SUBONE use dptr,a,30h,31h
SUBONE:
        MOV     DPTR,#2000H
LOOP:
        MOVX    A,@DPTR
        XRL     A,#41H
        JNZ     TT
        MOV     A,30H
        MOV     DPTR,#20A0H
        MOVX    @DPTR,A
        MOV     DPTR,#20A1H
        MOV     A,31H
        MOVX    @DPTR,A
        RET
TT:
        INC     DPTR
        MOV     30H,DPH
        MOV     31H,DPL
```

课后答案网 www.khdaw.com

```
        MOV     A,31H
        CJNE    A,@00H,LOOP
        RET
        END
```

2.23 解：

```
        ORG 0000H
        LJMP MAIN
        ORG 0030H
MAIN:   MOV R2，#20
        MOV R1，#30H
        MOV DPTR，#2000H
TT:     MOVX A，@DPTR
        CLR C
        SUBB A，#30H
        SWAP A
        MOV 41H，A
        INC DPTR
        MOVX A，@DPTR
        CLR C
        SUBB A，#30H
        MOV @R1，A
        MOV A，41H
        XCHD A，@R1
        MOV @R1，A
        INC R1
        INC DPTR
        DJNZ R2，TT
        MOV DPTR，#3000H
        MOV R1，#30H
        MOV R2，#0AH
TT1:    MOV A，@R1
        MOVX @DPTR，A
        INC R1
        INC DPTR
        DJNZ R2，TT1
        SJMP $
        END
```

2.24 解：

```
        ORG     0000H
        LJMP    MAIN
        ORG     0030H
MAIN:
        MOV     DPTR,#2400H
      MOV     A,#07H
      MOVX    @DPTR,A
      MOV     DPTR,@2450H
      MOV     A,#06H
      MOVX    @DPTR,A
        MOV     30H,#24H
        MOV     31H,#00H
        MOV     32H,#25H
        MOV     33H,#00H
        MOV     R2,#51H
LOOP:
        MOV     DPH,30H
        MOV     DPL,31H
        MOVX    A,@DPTR
      MOV     DPH,32H
      MOV     DPL,33H
      MOVX    @DPTR,A
      INC     31H
      INC     33H
      DJNZ    R2,LOOP
      AJMP    $
      END
```

2.25 解:

```
        ORG 0000H
        LJMP MAIN
        ORG 0030H
MAIN:   MOV DPTR,#2030H
        MOV A,#03H
        MOVX @DPTR,A
        MOV DPTR,#2031H
        MOV A,#05H
        MOVX @DPTR,A
        MOV DPTR,#2030H
        MOVX A,@DPTR
        LCALL SQR
        MOV R1,A
        MOV DPTR,#2031H
```

课后答案网 www.khdaw.com

```
        MOVX A,@DPTR
        LCALL SQR
        ADD A,R1
        MOV DPTR,#2040H
        MOVX @DPTR,A
        AJMP $
SQR:    INC A
        MOVC A,@A+PC
        RET
TAB:    DB 0,1,4,9,16,25
        DB 36,49,64,81,100
        DB 121,144,169,196,225
        END
```

课后答案网 www.khdaw.com

课后答案网 www.khdaw.com

第 3 章

3.1 解：

当 CPU 正在处理某件事情的时候，外部发生的某一事件请求 CPU 迅速去处理， CPU 暂时中止当前的工作，转去处理所发生的事件，处理完该事件以后，再回到原来被中止的地方，继续原来的工作。这种过程为中断，实现这种服务的部件称为中断系统。

功能：①实时处理，能对外界异步发生的事件作出及时的处理。②完全消除了 CPU 在查询方式中的等待现象，大大提高了 CPU 的工作效率。③实现实时控制。

3.2 解:

中断优先级是 CPU 响应中断的先后顺序。中断优先处理的原则是:

(1)先响应优先级高的中断请求，再响应优先级低的中断请求。

(2)如果一个中断请求已被响应，同级的其他中断请求将被禁止。

(3)如果同级的多个中断请求同时出现，CPU 则按单片机内部的自然优先级顺序响应各中断请求。

单片机内部自然优先级顺序（由高到低）为:

外部中断 0→定时器 0 中断→外部中断 1→定时器 1 中断→串行接口中断。

3.3 解:

（1）80C51 有以下中断源：

① ① 外部中断 0（$\overline{\text{INT0}}$）请求，低电平有效。

② ② 外部中断 1（$\overline{\text{INT1}}$）请求，低电平有效。

③ ③ T0：定时器/计数器 0 溢出中断请求。

④ ④ T1：定时器/计数器 1 溢出中断请求。

⑤ ⑤ TI/RI：串行接口中断请求。

（2）通过对特殊功能寄存器 TCON、SCON、IE、IP 的各位进行置位或复位等操作，可实现对各种中断的控制功能。

3.4 解:

中断系统的初始化步骤如下：

（1）开相应中断源的中断允许；

（2）设定所用中断源的中断优先级；

（3）若为外部中断，则应规定中断触发方式（低电平或负边沿触发）。

3.5 解:

单片机一旦响应中断请求，就由硬件完成以下功能:

（1） 根据响应的中断源的中断优先级，使相应的优先级状态触发器置 1；

（2） 执行硬件中断服务子程序调用，并把当前程序计数器 PC 的内容压入堆栈，保护断点，寻找中断源；

（3） 清除相应的中断请求标志位（串行口中断请求标志 RI 和 TI 除外）；

（4） 把被响应的中断源所对应的中断服务程序的入口地址（中断矢量）送入 PC，从而转入相应的中断服务程序。

（5）中断返回，程序返回断点处继续执行。

3.6 解:

（1）由中断源提出中断请求，由中断控制允许控制决定是否响应中断，如果允许响应中断，则 CPU 按设定好的优先级的顺序响应中断。如果是同一优先级的中断，则按单片机内部的自然优先级顺序（外部中断 0→定时器 0 中断→外部中断 1→定时器 1 中断→串行接口中断）响应中断。

CPU 响应中断请求后，就立即转入执行中断服务程序。保护断点、寻找中断源、中断处理、中断返回，程序返回断点处继续执行。

(2)由中断允许寄存器 IE 控制开放和禁止中断。欲开放某一中断，则应先开放总中断允许（EA 置 1），然后开放相应中断的中断允许（相应位置 1）；若要要禁止中断，则 EA 置 0

课后答案网 www.khdaw.com

即可。

(3)由中断优先级控制寄存器 IP 控制中断优先级，相应位置 1，则设为高级中断，置 0 则为低级。其中：PS 为串行中断优先级，PT1(0)为定时中断 1(0)优先级，PX1(0)外部中断 1(0)优先级。

3.7 解:

(1)有中断源发出中断请求。

(2)中断总允许控制位 EA=1， CPU 开放总中断。

(3)申请中断的中断源的中断允许位为 1，即该中断没有被屏蔽。

(4)无同级或更高级中断正在服务。

(5)当前指令周期已经结束。

(6)若现行指令为 RETI 或访问 IE 或 IP 指令时，读指令以及紧接着的另一条指令已执行完毕。

满足以上条件，则 CPU 响应响应中断元的中断请求。

3.8 解:

$\overline{\text{INT1}}$ 为低电平触发的中断系统初始化程序如下：

```
        ORG   0000H
        LJMP  MAIN
        ORG   0013H
        LJMP  INTN1
        ORG   0100H
MAIN:   SETB  EA
        SETB  EX1      ;  开 INT1 中断
        CLR   PX1      ;  令 INT1 为低优先级
        CLR   IT1      ;  令 INT1 为电平触发
        SJMP  $
        END
```

3.9 解:

中断服务程序的入口地址如下表：

中断源	中断矢量
外部中断 0（$\overline{\text{INT0}}$）	0003H
定时器 T0 中断	000BH
外部中断 1（$\overline{\text{INT1}}$）	0013H
定时器 T1 中断	001BH
串行口中断	0023H

3.10 解:

(1)符合以下 6 个条件可响应新的中断请求:

① 有中断源发出中断请求。

② 中断总允许控制位 EA=1， CPU 开放总中断。

③ 申请中断的中断源的中断允许位为 1，即中断没有被屏蔽。

④ 无同级或更高级中断正在被服务。

⑤ 当前的指令周期已结束。

⑥ 若现行指令为 RETI 或访问 IE 或 lP 指令时，该指令以及紧接着的另一条指令已执行完。

(2)如果新的中断请求"优先级"低于正在执行的中断请求或与其同级，则不能被响应。

3.11 解:

有两种方式:电平触发和边沿触发。

（1）电平触发方式：CPU 在每个机器周期的 S5P2 期间采样外部中断引脚的输人电平。若为低电平，便置 IE1(IE0)为“1” ，申请中断；若外部中断引脚为高电平，则 IE1(IE0)清零。

课后答案网 www.khdaw.com

（2）边沿触发方式：CPU 在每个机器周期的 S5P2 期间采样外部中断请求引脚的输人电平。如果在相继的两个机器周期采样过程中，一个机器周期采样到外部中断请求为高电平，接着下一个机器周期采样到外部中断请求为低电平，则使 IE1(IE0)置 1，申请中断;否则，IE1(1E0)置 0。

3.12 解:

可以。在相应的中断源的中断程序入口地址处，用一条长跳转指令（LJMP　Add16），转到相应 64K 程序存储器的任意地址（Add16）处，执行相应的中断程序。

3.13 解:

将 3 个中断源的中断请求经过与门连接到MCS-51 的外部中断 0 的输入引脚 $\overline{\text{INT0}}$ 上。3、2、1 中断源的输入引脚分别接到P1.0、P1.1、P1.2 引脚上，以备查询。程序如下：

```
        X1   EQU   2000H      ；定义中断源 1 的入口地址
        X2   EQU   2100HH     ；定义中断源 2 的入口地址
        X3   EQU   2200H      ；定义中断源 3 的入口地址
        ORG 0000H
        LJMP   START
        ORG 0003H
        LJMP   INT00
START:  ……
        ……
INT00: JB    P1.0, LP1          ；查询中断源，若此中断源无中断则转 LP1
       LJMP   2200H             ；转入相应的中断服务
LP1:   JB    P1.1, LP2          ；查询中断源，若此中断源无中断则转 LP2
       LJMP   2100H             ；转入相应的中断服务
LP3:   LJMP   2000H             ；转入相应的中断服务
       ……
X1:    ……
       RETI
X2:    ……
       RETI
X3:    ……
       RETI
```

3.14 解:

80C51 单片机片内设有 2 个定时器 / 计数器：定时器 / 计数器 T0 和定时器 / 计数器 T1，T0 由 TH0、TL0 组成，T1 由 TH1、TLl 组成。T0、T1 由特殊功能寄存器 TMOD、TCON 控制。

3.15 解:

作定时器用时，计数脉冲来自单片机内部，其频率为振荡频率的 1/12。作计数器用时，计数脉冲来自单片机的外部，即 P3.4（T0）和 P3.5（T1）两个引脚的输入脉冲。

3.16 解:

作定时器用时，其定时时间与定时器的工作模式、定时器的定时初值以及单片机的晶振频率有关。作计数器用时，外界计数脉冲的频率不能高于振荡脉冲频率的 1/24。

3.17 解:

（1）工作方式 0：13 位定时器/计数器工作方式。

工作方式 0 由 TH0 的全部 8 位和 TL0 的低 5 位构成 13 位加 1 计数器，此时 TL0 的高 3 位未用。在计数过程中，当 TL0 的低 5 位溢出时，都会向 TH0 进位，而全部 13 位计数器溢出时，则计数器溢出标志位 TF0 置位。

（2）工作方式 1：16 位的定时器/计数器方式。

工作方式 1 由 TH0 作为高 8 位，TL0 为低 8 位，在计数过程中，当全部 16 位计数器溢出时，则计数器溢出标志位 TF0 置位。

（3）工作方式 2：自动重新装入计数初值的 8 位定时器/计数器工作方式。

工作方式 2 的 16 位定时器/计数器被拆成两个 8 位寄存器 TH0 和 TL0，CPU 在对它们初始化时必须装入相同的定时器/计数器初值。定时器/计数器启动后，TL0 按 8 位加 1 计数

课后答案网 www.khdaw.com

器计数，当 TL0 计数溢出时，置位 TF0 的同时又从预置寄存器 TH0 中重新获得计数初值并启动计数。如此反复。适合于需要重复计数的应用场合，也可以当做串行数据通信的波特率发生器使用。

（4）工作方式 3：两个 8 位定时器/计数器（仅适用于 T0）。

在工作方式 3 时，定时器/计数器 0 被拆成两个独立的 8 位计数器 TL0 和 TH0。其中，TL0 既可以作计数器使用，也可以作为定时器使用，定时器/计数器 0 的各控制位和引脚信号全归它使用。其功能和操作与方式 0 或方式 1 完全相同。TH0 只能作为简单的定时器使用，只能借用定时器/计数器 1 的控制位 TR1 和 TF1，也就是以计数溢出去置位 TF1，TR1 则负责控制 TH0 定时的启动和停止。

一般情况下，只有在 T1 以工作方式 2 运行（当波特率发生器用）时，才允许 T0 工作于方式 3。

TMOD 用于控制定时器/计数器 T0 和 T1 的工作方式，M1M0 为工作方式选择位。

M1M0=00　　方式 0，13 位定时器/计数器；

M1M0=01　　方式 1，16 位定时器/计数器；

M1M0=10　　方式 2，自动重新装入计数初值的 8 位定时器/计数器；

M1M0=11　　方式 3，两个 8 位定时器/计数器（仅适用于 T0）。

$C/\overline{T}$ 为定时方式/计数方式选择位。若设定 $C/\overline{T}$=0，则选择定时器工作方式；若设定 $C/\overline{T}$=1，则选择计数器工作方式。一个定时器/计数器同一时刻或者作定时用，或者作计数用，不能同时既作定时又作计数用。

GATE：门控位。它的状态决定了定时器/计数器启/停控制取决于TR0 还是取决于TR0 和 $\overline{INT0}$ 引脚两个条件的组合。若GATE=0，则只由TCON中的启/停控制位TR0 控制定时器/计数器的启/停。此时，只要TR0=1，则接通模拟开关，使计数器进行加法计数，定时器/计数器启动工作。而如果TR0=0，则断开模拟开关，定时器/计数器停止工作。若GATE=1，由外部中断请求信号 $\overline{INT0}$ 和TCON中的启/停控制位TR0 组合状态控制定时器/计数器的启/停。只有TR0=1，且 $\overline{INT0}$ 引脚也是高电平，才能启动定时器/计数器工作，否则，定时器/计数器停止工作。

定时器/计数器的定时器/计数器范围为：

工作方式 0：13 位定时器/计数器方式，因此，最多可以计到 2^{13}，也就是 8 192 次。

工作方式 1：16 位定时器/计数器方式，因此，最多可以计到 2^{16}，也就是 65 536 次。

工作方式 2 和工作方式 3：都是 8 位的定时器/计数器方式，因此，最多可以计到 2^8，也说是 256 次。

3.18 解:

设定好定时器的定时时间，采用中断方式用软件设置计数次数，进行溢出次数累计，从而得到较长的时间。

3.19 解：

选用定时器/计数器 T0 作定时器， 输出为 P1.0 引脚， 2 ms 的方波可由 1 ms 的高低电平相间隔而成， 因而只要每隔 1 ms 对 P1.0 取反一次即可得到这个方波。程序如下：

```
        ORG   0000H
        LJMP  START
        ORG   000BH
        LJMP  T0INT          ;  T0 中断入口
        ORG   0030H
START:  MOV   SP,  #60H      ;  初始化程序
        MOV   TH0,  #0FEH    ;  T0 赋初值
        MOV   TL0,  #0BH
        MOV   TMOD,  #01H    ; 定时器/计数器 0 工作于方式 1
        SETB  TR0            ; 启动 T0
        SETB  ET0            ;  开 T0 中断
        SETB  EA             ;  开总允许中断
```

课后答案网 www.khdaw.com

```
                SJMP   $
        T0INT:  CPL    P1.0
                MOV    TL0,  #0BH
                MOV    TH0,  #0FE0H
                RETI
                END
```

3.20 解：

程序如下：

```
                ORG 0000H
                LJMP START
                ORG 0100H
        START:  MOV SP，#60H
                MOV TMOD，#02H
                MOV TH0，#0E7H
                MOV TL0，#0E7H
                CLR P1.2
                SETB TR0
HIGH0:  SETB P1.2
HIGH1:  JBC TF0，LOW0          ；50μs 到清 TF0，转 LOW0
                AJMP HIGH1                  ；50μs 未到，转 HIGH1 等待
LOW0:           MOV R7，#7                  ；350μs=7×50μs
                CLR P1.2                    ；P1.2=0 输出 350μs 低电平
          LOW1: JBC TF0，LOW2               ；50μs 到清 TF0，LOW2
                AJMP LOW1
          LOW2: DJNZ R7，LOW1               ；7 次未到转 LOW1
                AJMP HIGH0                  ；7 次到转 HIGH0
                END
```

3.21 解：

选择 T0 工作于方式 1 定时 500μs

$$\text{机器周期} T=\frac{12}{f_{osc}}=\frac{12}{12\times10^{6}}=1\times10^{-6}\text{s}=1\mu\text{s}$$

fosc=12MHz

$(2^{16}-X)\times1\mu S=500\mu s$　　X=65036=0FE0CH

程序如下：

```
        ORG 1000H
    MOV TMOD，#01H
    MOV TH0，#0FEH
MOV TL0，#0CH
SETB   TR0
   DEL：MOV   R7，#4            ；2ms=4×500μs
 D500：JBC   TF0，D2            ；500μs 到清 TF0，转移
AJMP   D500             ；50μs 未到，等待
    D2：CPL   P1.0
    MOV TH0，#0FEH          ；重装初值
MOV TL0，#0CH
        DJNZ   R7，D500         ；4 次未到，转 D500
     CPL   P1.1
AJMP DEL            ；4 次到，转 DEL
SJMP $
        END
```

3.22 解：

程序如下：

```
                ORG 0000H
                LJMP START
                ORG 0030H
        START:  MOV TMOD，#09H          ；设 T0 为方式 1，GATE=1
                MOV TL0，#00H
                MOV TH0，#00H
```

课后答案网 www.khdaw.com

```
        MOV R0，#4EH
        JB P3.2，$              ；等待 P3.2 变低
        SETB TR0                ；启动 T0 工作
        JNB P3.2，$             ；等待 P3.2 变高
        JB P3.2，$              ；等待 P3.2 再次变低
        CLR TR9                 ；停止计数
        MOV @R0，TL0            ；存放计数的二进制数低字节入 4EH
        INC R0
        MOV @R0，TH0            ；存放计数的二进制数高字节入 4EH
        MOV R1，#50H            ；BCD 码首址
        MOV R5，#3              ；BCD 码字节数
        CLR  A
LOOP1：MOV @R1，A               ；清存 BCD 码单元
        INC R1
        DJNZ R5，LOOP1
        MOV R7，#10H            ；二进制数位数
LOOP2：MOV R0，#4EH             ；二进制数首址
        MOV R6，#2              ；二进制数字节数
        CLR C
LOOP3：MOV A，@R0
        RLC A
        MOV @R0，A
        INC R0
        DJNZ R6，LOOP3          ；2 字节二进制数左移 1 位
        MOV R5，#3              ；BCD 码字节数
        MOV R1，#50H
LOOP4：MOV A，@R1
        ADDC A，@R1             ；BCD 码乘 2 加 C 运算
        DA A
        MOV @R1，A
        INC R1
        DJNZ R5，LOOP4
        DJNZ R7，LOOP2
        SJMP $
        END
```

3.23 解::

为了衡量串行通信的速度，应该有一个测量单位，在数据通信中，描述数据传送速度的方式有 3 种：波特率定义为每秒传送信号的数量，单位为波特（Baud）。比特率定义为每秒传送二进制数的信号数（或每秒传送二进制码元的个数），单位是 bps（bit per second）或写成 b/s（位/秒）。数据传送速率(或字符传送速率)定义为每秒传送多少个字符(或单位时间内平均数据传移速率，单位是字符/秒。

在串行通信中，传送的信号可能是二进制、八进制或十进制等。只有在传送的信号是二进制信号时，波特率才与比特率数值上相等。而在采用调制技术进行串行通信时，波特率是描述载波信号每秒钟变化为信号的数量（又称为调制速率）。在这种情况下，波特率与比特率在数值上可能不相等。

3.24 解:

异步通信中，接收器和发送器有各自的时钟，数据常以字符为单位组成字符帧传送，用一帧来表示一个字符，其字符帧的数据格式为：在一帧格式中，先是一个起始位“0”（低电平），然后是 5～8 个数据位，规定低位在前，高位在后，接下来是 1 位奇偶校验位（可以省略），最后是 1～2 位的停止位“1”（高电平）。

异步通信的优点是不需要传送同步脉冲，可靠性高，所需设备简单；缺点是字符帧中因包含有起始位和停止位而降低了有效数据的传输速率。

3.25 解:

课后答案网 www.khdaw.com

MCS-51 单片机的串行接口由发送缓冲器 SBUF、发送控制器、接收缓冲器 SBUF、输入移位寄存器、接收控制器、波特率发生器等部件组成。

发送缓冲器 SBUF 用于存放将要发送的数据，接收缓冲器 SBUF 用于存放接收的数据，输入位移寄存器用于接收缓冲并实现串/并转换，发送/接收控制寄存器用于控制串行口的工作，波特率发生器用于控制串行口发送/接收数据的速度。

3.26 解：

随着传输距离的增加，误码率增加，出现重传增加，数据传输速率降低。

3.27 解：

串行接口的接收和发送是对同一地址(99H)两个物理空间的特殊功能寄存器 SBUF 进行读或写的。当向 SBUF 发"写"命令时(执行"MOVSBUF，A"指令)，即向发送缓冲器 SBUF 装载并开始由 TXD 引脚向外发送一帧数据，发送完使发送中断标志位 TI=1。在满足串行接口接收中断标志位 RI(SCON.0)=0 的条件下，置允许接收位 REN(SCON.4)=1，就会接收一帧数据进入移位寄存器，并装载到接收 SBUF 中，同时使 R1=1。当发读 SBUF 命令时(执行"MOV A，SBUF"指令)，便从接收缓冲器 SBUF 读取信息通过 80C51 内部总线送 CPU。

3.28 解：

串行口有四种工作方式：方式 0(8 位同步移位寄存器)，方式 1(10 位异步收发)，方式 2(11 位异步收发)，方式 3(11 位异步收发)。

字符帧的数据格式为：在一帧格式中，先是一个起始位“0”（低电平），然后是 5～8 个数据位，规定低位在前，高位在后，接下来是 1 位奇偶校验位（可以省略），最后是 1～2 位的停止位“1”。两个字符帧之间可以有空闲位，也可以无空闲位。

在 8051 串行口的四种工作方式中，方式 0 和 2 的波特率是固定的，而方式 1 和 3 的波特率是可变的，由定时器 T1 的溢出率（T1 溢出信号的频率）控制。各种方式的通信波特率如下：

① 方式 0 的波特率固定为系统晶振频率的 1/12，其值为$f_{osc}/12$。

其中：f_{osc}——系统主机晶振频率

② 方式 2 的波特率由 PCON 中的选择位 SMOD 来决定，可由下式表示：

$$\text{波特率}=(2^{SMOD}/64)\times f_{osc}$$

即：当SMOD=1 时，波特率为$f_{osc}/32$，当SMOD=0 时，波特率为$f_{osc}/64$

③ 方式 1 和方式 3 的波特率由定时器 T1 的溢出率控制。因而波特率是可变的。定时器 T1 作为波特率发生器，相应公式如下：

$$\text{波特率}=(2^{SMOD}/32)\times\text{定时器T1 溢出率}$$

$$\text{T1 溢出率}=\text{T1 计数率}/\text{产生溢出所需的周期数}$$

$$=(f_{osc}/12)/(2^{K}-TC)$$

式中：K——定时器 T1 的位数

TC——定时器 T1 的预置初值

3.29 解：

当一片 80C51(主机)与多片 80C51(从机)通信时，

① 主机的 SM2 位置 0，所有从机的 SM2 位置 1，处于接收地址帧状态。

② 主机发送一地址帧，其中，8 位是地址，第 9 位为地址/数据的区分标志，该位置 1 表示该帧为地址帧。

③ 所有从机收到地址帧后，都将接收的地址与本机的地址比较。对于地址相符的从机，使自己的 SM2 位置 0（以接收主机随后发来的数据帧），并把本站地址发回主机作为应答；对于地址不符的从机，仍保持 SM2=1，对主机随后发来的数据帧不予理睬。

④ 从机发送数据结束后，要发送一帧校验和，并置第 9 位（TB8）为 1，作为从机数据传送结束的标志。

⑤ 主机接收数据时先判断数据接收标志（RB8），若接收帧的 RB8=0，则存储数据到缓冲区，并准备接收下帧信息。若 RB8=1，表示数据传送结束，并比较此帧校验和，若正确则回送正确信号 00H，此信号命令该从机复位（即重新等待地址帧）；若校验和出错，则发送 0FFH，命令该从机重发数据。

课后答案网 www.khdaw.com

⑥ 主机收到从机应答地址后，确认地址是否相符，如果地址不符，发复位信号（数据帧中 TB8=1）；如果地址相符，则清 TB8，开始发送数据。

⑦ 从机收到复位命令后回到监听地址状态（SM2=1）。否则开始接收数据和命令。

3.30 解：

在微机与单片机构成的测控网络信中，对于串行口数据传输接口的抗干扰能力，在不超过接口标准指定的适用范围时，都具有一定的抗干扰能力，以保证信号传输的可靠性。但在一些工业测控系统中，通信环境往往十分恶劣，就必须充分考虑通信的抗干扰能力，以保证通信的可靠性。

（1）选择合适的通信标准。例如：长距离传输采用 RS-485 标准能有效抑制功模干扰，采用 20Ma 电流环可以降低信号对各种电器噪声的敏感程度。

（2）在高噪声环境下使用光纤传输介质在高噪声环境下可以有效减少噪声干扰。

（3）采用光电隔离技术可以提高系统的安全性和可靠性

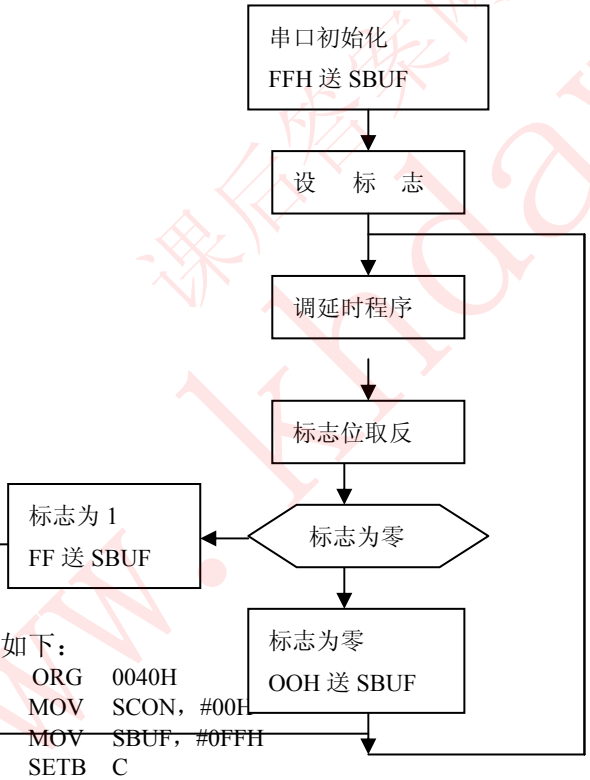
（4）在发送器输出接口采用限流电路或在发送器输出端外接电容器，可以有效抑制数据传输过程中的串扰。

（5）采取降低发送端数据速率的方法可以减少接收端接收数据益处的错误。

3.31 解：11（位）×3600（字符/分钟）=39600b/分钟=660b/s （方式 3 为每个字符 11 位）。

3.32 解：

主程序框图如下：



程序如下：

```
        ORG   0040H
        MOV   SCON，#00H
        MOV   SBUF，#0FFH
        SETB  C
        MOV   00H，C
AA：    ACALL DELAY
        CPL   00H
        MOV   C，00H
        JC          BB
        MOV   SBUF，#00H
        SJMP   CC
BB：    MOV   SBUF，#0FFH
CC：    AJMP  AA
DELAY：  MOV   R7，#8
D1：    MOV   R6，#250
D2：    MOV   R5，#250
```

课后答案网 www.khdaw.com

```
D3：DJNZ  R5，D3
    DJNZ  R6，D2
    DJNZ  R7，D1
    RET
    END
```

3.33 解：

采用查询方式编写发送程序如下：

```
        ORG 0000H
        LJMP START
        ORG 0030H
START：MOV SCON，#80H        ；设定方式 2 发送
       MOV PCON，#80H
       MOV R0，#20H          ；给待发送数据块地址指针 R0 置初值
       MOV R7，#16           ；给数据块长度计数器 R7 置初值
  LS：MOV A，@R0             ；取一字节数据送 A
       MOV C，P
       MOV TB8，C            ；P 随 A 变，P→TB8
       MOV SBUF，A           ；启动发送
       JNB TI，$             ；查询发送标志，等待一字节发送完
       CLR TI    ；清 TI 标志位
       DJNZ R7，LS
       RET
       END
```

3.34 解：

程序如下：

```
        ORG    0000H
        LJMP   START
        ORG    0100H
START：MOV    TMOD，#20H     ；设定 T1 为模式 2 定时
       MOV    TH1，#0F3H     ；送时间常数
       MOV    TL1，#0F3H
       SETB   TR1            ；启动定时器 T1
       MOV    SCON，#0D0H    ；串行接口设定为方式 3，并允许接收
       MOV    R0，#30H
       MOV    R7，#16
       JNB    RI，$          ；查询等待接收
       CLR    RI
       MOV    A，SUBF        ；从串行接口中读取数据
       JN     BP，PN         ；对该字节进行查错处理若 P=RB8 无错，否则有
错
       JNB    RB8，PER       ；若 P=1，RB8=0，有错，转出错处理
       LJMP   RIG            ；若 P=1，RB8=1，无错，转保存数据
PN：   JB     RB8，PER       ；若 P=0，RB8=1，有错，转出错处理
RIG：  MOV    @R0，A         ；若 P=0，RB8=0，无错，保存接收到的数据
       INC    R0
       DJNZ   R7，LOOP       ；16 字节未接收完，则继续
       CLR    7F             ；正确接收完 16 位数据，清出错标志位 7F
       SJMP   $
PER：  SETB   7F             ；校验有错，置位 7F
       SJMP   $
       END
```

第 4 章

4.1 解：

MCS-51 系列单片机具有很强的外部扩展功能。其外部引脚可构成三总线结构，即地址总线、数据总线和控制总线。单片机所有的外部扩展都是通过三总线进行的。

（1）地址总线（AB）

课后答案网 www.khdaw.com

地址总线用于传送单片机输出的地址信号，宽度为 16 位，可寻址的地址范围为 2^{16}=64KB。地址总线是单向的，只能由单片机向外发出。P0 口提供低 8 位地址，P2 口提供高 8 位地址。由于P0 口既做地址线又做数据线，分时复用，所以，P0 口提供的低 8 位地址是由P0 口经锁存器提供的。锁存信号是由CPU的ALE引脚提供的。

（2）数据总线（DB）

数据总线是由 P0 口提供的，宽度为 8 位。P0 口是双向三态口，是单片机应用系统中使用最频繁的通道。P0 口提供的数据总线上要连接多个扩展的外围芯片，而某一时刻只能有一个有效的数据传输通道。具体哪一个芯片的数据通道有效，是由各个芯片的片选信号控制选择的。欲使 CPU 与某个外部芯片交换数据，则 CPU 必须先通过地址总线发出该芯片的地址，使该芯片的片选信号有效，则此时 P0 口数据总线上的数据只能在 CPU 和该芯片之间进行传送。

（3）控制总线（CB）

控制总线实际上是CPU输出的一组控制信号。每条控制信号都是单向的，但是由多条不同的控制信号组合而成的控制总线则是双向的。MCS-51 系列单片机中用于系统扩展的控制信号有 $\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE和 $\overline{EA}$ 。

4.2 解：

接口（也称为 I/O 接口）是指连接 CPU 与外部输入/输出设备之间的部件，这些部件是 CPU 与外设之间进行信息传送的媒介。

I/O 接口芯片都有一个或几个端口，一个端口对应于接口芯片内部的一个寄存器或一组寄存器，计算机系统要为每个端口分配一个地址，各个端口的地址是唯一的，不能重复。在信息传送过程中，接口起着数据锁存、数据缓冲、输入/输出、联络、数据转换、中断管理、时序控制、可编程、电器特征匹配等作用。

4.3 解：

CPU 与外设之间传输数据的控制方式通常有三种：程序方式、中断方式和 DMA 方式。

程序方式：指用输入/输出指令，来控制信息传输的方式，是一种软件控制方式，根据程序控制的方法不同，又可以分为无条件传送方式和条件传送方式。

无条件传送方式接口简单，适用于那些能随时读写的设备。条件传送方式（查询方式）的特点是接口电路简单，CPU 利用率低（程序循环等待），接口需向 CPU 提供查询状态。适用于 CPU 不太忙，传送速度要求不高的场合。要求各种外设不能同时工作，外设处于被动状态。

中断方式：当外设准备好时，由外设通过接口电路向 CPU 发出中断请求信号，CPU 在允许的情况下，暂停执行当前正在执行的程序，响应外设中断，转入执行相应的中断服务子程序，与外设进行一次数据传送，数据传送结束后，CPU 返回继续执行原来被中断的程序。其特点是 CPU 的利用率高，外设具有申请 CPU 中断的主动权， CPU 和外设之间处于并行工作状态。但中断服务需要保护断点和恢复断点（占用存储空间，降低速度）， CPU 和外设之间需要中断控制器。适用于 CPU 的任务较忙、传送速度要求不高的场合，尤其适合实时控制中的紧急事件处理。

存储器直接存取方式（DMA）：外设利用专用的接口（DMA 控制器）直接与存储器进行高速数据传送，并不经过 CPU（CPU 不参与数据传送工作），总线控制权不在 CPU 处，而由 DMA 控制器控制。其特点是接口电路复杂，硬件开销大。大批量数据传送速度极快。适用于存储器与存储器之间、存储器与外设之间的大批量数据传送的场合。

4.4 解：

在计算机系统中，凡需要进行读写操作的部件都存在编址的问题。存储器的每个单元均有自己的地址，对于 I/O 接口，则需要对接口中的每个端口进行编址。通常采取两种编址方法：一种是独立编址，另一种是统一编址。

统一编址又称“存储器映射方式”。在这种编址方式下 ，I/O 端口地址置于存储器空间中，在整个存储空间中划出一部分空间给外设端口 ，端口和存储单元统一编址。其优点是无需专门的 I/O 指令，对端口操作的指令类型多，从而简化了指令系统的设计。缺点是端口占用存储器的地址空间，使存储器容量更加紧张，同时端口指令的长度增加，执行时间较长，端

课后答案网 www.khdaw.com

口地址译码器较复杂。

独立编址又称“I/O 映射方式”。这种方式的端口单独编址构成一个 I/O 空间，不占用存储器地址空间。其优点是端口所需的地址线较少，地址译码器较简单，采用专用的 I/O 指令，端口操作指令执行时间少，指令长度短。缺点是输入输出指令类别少，一般只能进行传送操作。

MCS-51 单片机采用了统一编址方式，即 I/O 端口地址与外部数据存储单元地址共同使用 0000H～FFFFH（64KB）。因此，MCS-51 单片机应用系统扩展较多外部设备和 I/O 接口时，要占去大量的数据存储器的地址。

4.5 解：

所谓的全地址译码法是利用译码器对系统地址总线中未被外扩芯片用到的高位地址线进行译码，以译码器的输出作为外围芯片的片选信号。全地址译码法芯片的地址范围确定方法是：以外部扩展的全部芯片未用到的地址线作为地址译码器的输入，译码器的输出作为片选信号接到外部扩展芯片上。

全地址译码法优点是存储器的每个存储单元只有惟一的一个系统空间地址，不存在地址重叠现象；对存储空间的使用是连续的，能有效地利用系统的存储空间；利用同样的高位地址线，全地址译码法编址产生的选片线比线选法多，为系统扩展提供了更多的冗余条件。其缺点是所需地址译码电路较多，尤其在单片机寻址能力较大和所采用的存储器容量较小时更为严重。全地址译码法是单片机应用系统设计中经常采用的方法。

部分地址译码法是指单片机的未被外扩芯片用到的高位地址线中，只有一部分参与地址译码，其余部分是悬空的。在部分地址译码方式下，无论 CPU 使悬空的高位地址线上的电平如何变化，都不会影响它对外部存储单元的选址，故存储器每个存储单元的地址不是惟一的，存在地址重叠现象。因此，采用部分地址译码法时必须把程序和数据存放在基本地址范围内（即悬空的高位地址线全为低电平时存储芯片的地址范围），以避免因地址重叠引起程序运行的错误。部分地址译码法的优点是可以减少所用地址译码器的数量。

4.6 解：

P2 口用作扩展存储器的高 8 位地址总线以后，即使没有全部占用，空余的几根也不宜用作 I/O 口，否则会给软件编写及使用带来不必要的麻烦。主要是时序上处理比较困难。

4.7 解：

程序存储器和数据存储器虽然共用 16 位地址线和 8 位数据线，但由于数据存储器的读和写由 $\overline{RD}$ 和 $\overline{WR}$ 信号控制，而程序存储器由读选通信号 $\overline{PSEN}$ 控制，这些信号在逻辑上时序上不会产生冲突，因此，两者虽然共处于同一地址空间，但由于控制信号不同，所以不会发生总线冲突。

4.8 解：

硬件连接图如下图所示。

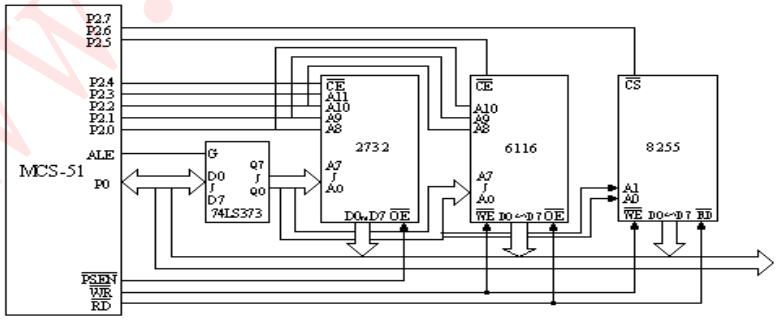


图　4.8 题硬件连接电路图

各芯片的地址范围为：

2732：E000H~EFFFH　　6116：D800H~DFFFH　　8255：BFFCH~BFFFH

4.9 解：

硬件连接电路图如图 4.9 所示。各芯片的地址范围为：

2764（1#）：0000H~1FFFH　　2764（2#）：2000H~3FFFH

6264（1#）：4000H~5FFFH　　6264（2#）：6000H~7FFFH

课后答案网 www.khdaw.com

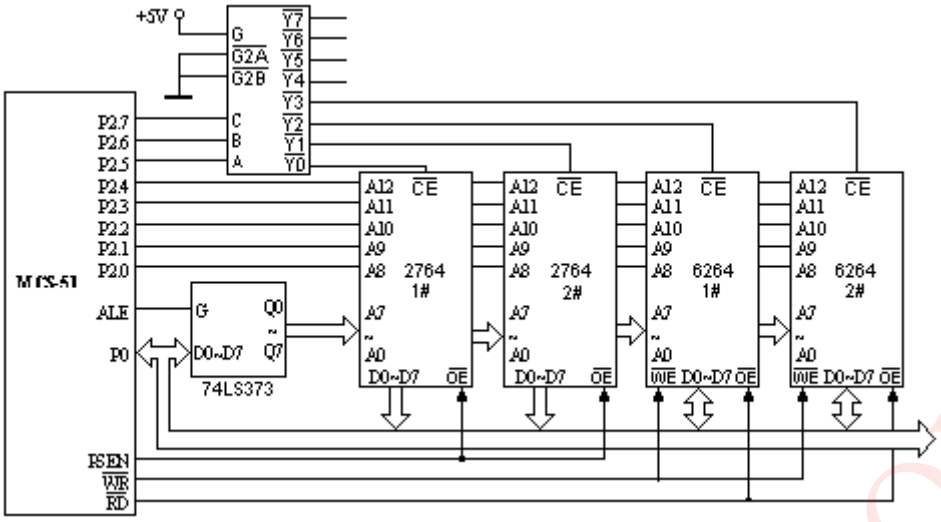


图 4.9　4.9 题硬件连接电路图

4.10 解：

（1）并行总线扩展的方法

（2）串行口扩展方法

（3）I/O 端口模拟串行方法

（4）通过单片机内 I/O 的扩展方法

4.11 解：

8255A 有 3 种工作方式：方式 0、方式 1 和方式 2。

① 方式 0（基本输入/输出方式）。这种方式不需要任何选通信号，适合于无条件传输数据的设备，数据输出有锁存功能，数据输入有缓冲（无锁存）功能。

② 方式 1(选通输入/输出方式)。这种工作方式下，A 组包括 A 口和 C 口的高四位(PC7～PC4)，A 口可由程序设定为输入口或输出口，C 口的高四位则用来作为输入/输出操作的控制和同步信号；B 组包括 B 口和 C 口的低四位（PC3～PC0），功能和 A 组相同。

③ 方式 2（双向 I/O 口方式）。仅 A 口有这种工作方式，B 口无此工作方式。此方式下，A 口为 8 位双向 I/O 口，C 口的 PC7～PC3 用来作为输入输出的控制和同步信号。此时，B 口可以工作在方式 0 或方式 1。

4.12 解：

程序如下：

```
MOV   DPTR，#7F03H
MOV   A，#10010100B
MOV   @DPTR，A
```

4.13 解：

写入控制端口的控制字的最高位为 1 则为方式控制字，否则为 C 口置位/复位控制字。

4.14 解：

8155 输入时钟的周期=1/500×10^3=2×10^{-6} (s)，8155 定时器/计数器的初值=20×10^{-3}/2×10^{-6}=10000=2170H。由于定时器/计数器按方式 1 工作，所以写入定时器/计数器的初值为 6170H。程序如下：

```
        ORG     0000H
        LJMP    START
        ORG     0100H
START:  MOV     DPTR，#7F04H      ；写定时器/计数器的初值
        MOV     A，#70H
        MOVX    @DPTR，A
        INC     DPTR
        MOV     A，#61H
        MOVX    @DPTR，A
        MOV     DPTR，#7F00H      ；写命令字，启动定时器/计数器工作
        MOV     A，#0C5H
```

课后答案网 www.khdaw.com

```
        MOVX    @DPTR，A
        SJMP    $
        END
```

4.15 解：

设 8155 有关寄存器端口地址为：

20H　　命令寄存器

24H　　定时器低字节

25H　　定时器高字节

在 8155 的 TIMEROUT 引脚上输出 10mS 方波的程序如下：

```
        ORG   1000H
MOV   R0，#25H
MOV   A，#58H
MOVX  @R0，A
DEC   R0
MOV   A，#0F0H
MOVX  @R0，A
MOV   R0，#20H
MOVX  @R0，A
SJMP  $
```

4.16 解：

电路连接图如图 4.16 所示。

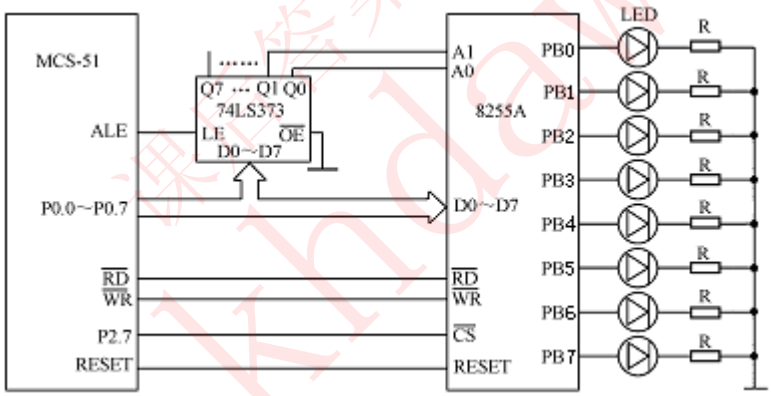


图 4.16　4.16 题硬件连接电路图

其中，PB0～PB3 接红色发光二极管，PB4～PB7 接绿色发光二极管。设 MCS-51 单片机主频为 1MHz。

程序如下：

```
        ORG     1000H
START：MOV    DPTR, #7FFFH        ；数据指针指向 8255A 控制口
        MOV    A, #80H
        MOVX   @DPTR, A            ；工作方式字送 8255A 控制口
        MOV    DPTR, #7FFDH        ；数据指针指向 8255A 的 B 口
  LP1：MOV    A, #0FH             ；置红色发光二极管亮
        MOVX   @DPTR, A            ；置红色发光二极管亮
        LCALL  DELAY               ；调用 1S 延时子程序
        MOV    A, #0F0H
        MOVX   @DPTR, A            ；置绿色发光二极管亮
        LCALL  DELAY
        LJMP   LP1                 ；循环执行
DELAY：  MOV R7，#8                ；1S 延时子程序
    D1：MOV R6，#250
    D2：MOV R5，#250
```

课后答案网 www.khdaw.com

```
D3：DJNZ R5，D3
    DJNZ R6，D2
    DJNZ R7，D1
    RET
    SJMP $
    END
```

4.17 解：

RS-232C 采取不平衡传输方式，是为点对点（即只用一对收、发设备）通信而设计的，采用负逻辑，其驱动器负载为 3kΩ～7kΩ。由于 RS-232C 发送电平与接收电平的差仅为 2～3V，所以其共模抑制能力差，再加上双绞线上的分布电容，因此，RS-232C 适用于传送距离不大于 15m，速度不高于 20kb/s 的本地设备之间通信的场合。

RS-422 由 RS-232 发展而来，RS-422 定义了一种平衡通信接口，将传输速率提高到 10Mb/s，传输距离延长到 1220m（速率低于 100kb/s 时），并允许在一条平衡总线上最多连接 10 个接收器。RS-422 是一种单机发送、多机接收的单向、平衡的通信总线标准。

RS-485 是在 RS-422 的基础上制定的标准，增加了多点、双向通信能力，通常在要求通信距离为几十米至上千米时，广泛采用 RS-485 总线标准。它采用平衡发送和差分接收，即在发送端，驱动器将 TTL 电平信号转换成差分信号输出；在接收端，接收器将差分信号变成 TTL 电平。具有较高的灵敏度，能检测低至 200mV 的电压，具有抑制共模干扰的能力，数据传输可达千米以上。

RS-232 的双机通信接口电路如图 4.17-1 所示。

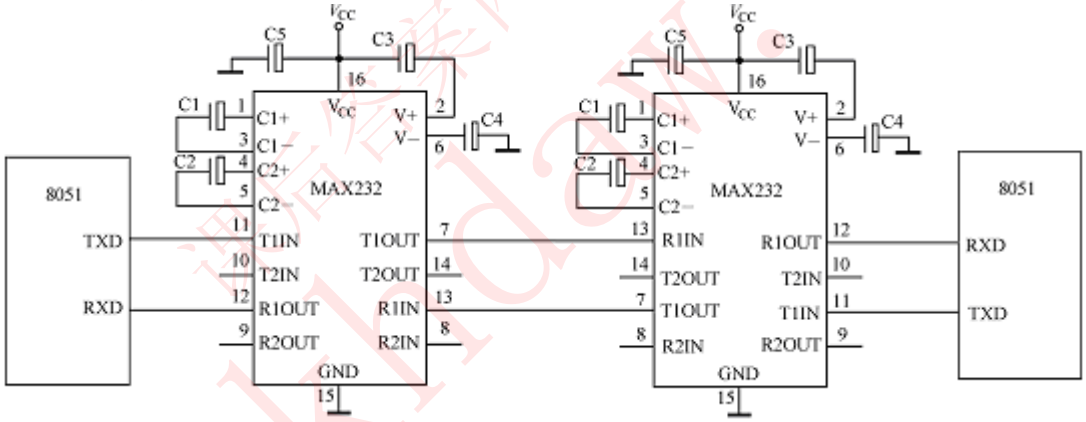


图 4.17-1　4.17-1 题硬件连接电路图

RS-422 和 RS-485 的双机通信接口电路如图 4.17-2 所示。

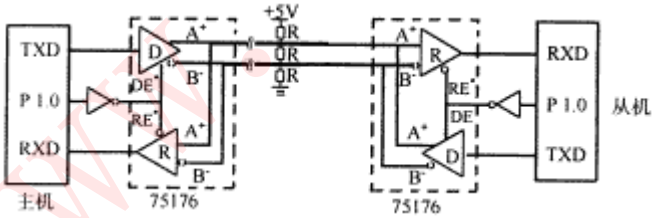


图 4.17-2　4.17-2 题硬件连接电路图

4.18 解：

DS12C887 是美国Dallas公司生产的实时日历时钟芯片，采用CMOS技术，与MC146818B和DS1287 管脚兼容，可直接替换。内含一个锂电池，在断电情况下运行十年以上不丢失数据。具有秒、分、时、星期、日、月、年计数功能，对于一天内的时间记录，有 12 小时制和 24 小时制两种模式。在 12 小时制模式中，用AM和PM区分上午和下午，可实现闰年调整。时间的表示方法有两种，二进制数和BCD码表示方法。内部有 128 字节RAM，其数据具有掉电保护功能。可以选择Motorola和Intel总线时序。用户可对DS12C887 进行编程以实现多种方波输出，并可对其内部的三路中断通过软件进行屏蔽。工作电压为 4.5～5.5 V，工作电流为 7～15mA。DS12C887 具有功耗低、外围接口简单、精度高、工作稳定可靠等优点，可广泛用于各种需要较高精度的实时时钟场合中。

DS12C887 内部有 128B 的存储器，系统占用 15B，用户可用 113B，具体分布情况如下表

课后答案网 www.khdaw.com

所示。

地　址	功　能	取值范围	
		二进制	十进制
00H	秒	00～3BH	00～59
01H	秒报警	00～3BH	00～59
02H	分	00～3BH	00～59
03H	分报警	00～3BH	00～59
04H	时，12 小时模式	0～0CH AM，81～8CH PM	01-12 AM, 81-92 PM
	时，24 小时模式	00～17H	00～23
05H	时报警，12 小时模式	0～0CH AM，81～8CH PM	01-12 AM, 81-92 PM
	时报警，24 小时模式	00～17H	00～23
06H	星期，星期日=1	01～07H	01～07
07H	日	01～1FH	01～31
08H	月	01～0CH	1～12
09H	年	00～63H	00～99
0AH	控制寄存器 A		
0BH	控制寄存器 B		
0CH	控制寄存器 C		
0DH	控制寄存器 D		
32H	世纪	13H，14H	19, 20
0EH～31H, 33H～7FH	用户数据区		

4.19 解：

MAX692A 是美国 Maxim 公司的系统监控芯片产品，具有后备电池切换、电压监视器、 “看门狗”监控等功能。

4.20 解：

“看门狗（WDT）”，也称为程序监视定时器。WDT 的作用是通过不断监视程序每周期的运行事件是否超过正常状态下所需要的时间，从而判断程序是否进入了“死循环”，并对进入“死循环”的程序作出系统复位处理。

在程序中设置适当的指令，清 WDT，就可监视微处理器的工作。例如在主程序开始时，将 WDT 置位，如果主程序执行过程中产生死循环，就无法清 WDT，超过 WDT 的定时时间时，WDT 就会对微处理器发出复位信号。从而实现对系统程序的监控。

4.21 解：

I^2C总线是由串行数据线SDA和串行时钟线SCL构成的，可发送和接收数据。它允许若干兼容器件共享总线。所有挂接在I^2C总线上的器件和接口电路都应具有I^2C总线接口，且所有的SDA/SCL同名端相连。总线上所有器件要依靠SDA发送的地址信号寻址，不需要片选线。

I^2C总线最主要的优点是其简单性和有效性。占用的空间小，降低了互连成本。总线的长度可高达 7.6m，并且能够以 10kbps的最大传输速率支持 40 个组件。支持多主控器件，其中，任何能够进行发送和接收的设备都可以成为主器件。主控能够控制信号的传输和时钟频率。当然，在某时刻只能有一个主控器件。

在单片机控制系统中，广泛使用I^2C器件。如果单片机自带I^2C总线接口，则所有I^2C器件对应连接到该总线上即可；若无I^2C总线接口，则可以使用I/O口模拟I^2C总线。

使用单片机I/O口模拟I^2C总线时，硬件连接非常简单，只需两条I/O口线即可，在软件中分别定义成SCL和SDA。MCS-51 单片机实现I^2C总线接口电路如图 4.21 所示。

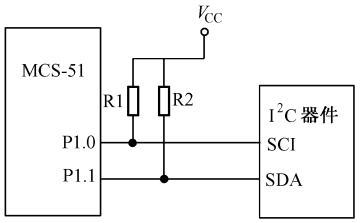


图 4.21　4.21

课后答案网 www.khdaw.com

电路中单片机的P1.0 引脚作为串行时钟线SCL，P1.1 引脚作为串行数据线SDA，通过程序模拟I^2C串行总线的通信方式。I^2C总线适用于通信速度要求不高而体积要求较高的应用系统。

第 5 章

5.1 解：

传感器是将电量或非电量转换为可测量的电量的检测装置，是由敏感元件和转换元件组成的。国际电工委员会 IEC 将传感器定义为：传感器是测量系统中的一种前端部件，它将各种输入变量转换成可供测量的信号。

5.2 解：

（1）输入通道的特点：

① 输入通道要靠近拾取对象采集信息，以减少传输损耗，防止干扰；

② 输入通道工作环境因素严重影响通道的方案设计，没有选择的余地；

③ 传感器的输出往往是模拟信号、微弱信号输出，转换成计算机要求的信号电平时，需要使用一些模拟电路技术，因此输入通道通常是模拟、数字等混杂电路；

④ 传感器、变送器的选择和环境因素决定了输入通道电路设计的繁简，因为在输入通道中必须将传感器、变送器的输出信号转换成能满足计算机输入要求的 TTL 电平，输入通道中传感器、变送器输出信号与计算机逻辑电平的相近程度影响着输入通道的繁简程度；

⑤ 传感器输出信号一般比较微弱，为便于计算机拾取，常需要放大电路，这也是计算机系统中最容易引入干扰的渠道，所以输入通道中的抗干扰设计是非常重要的。

（2）输出通道的特点：

① 小信号输出，大功率控制；

② 输出伺服驱动控制信号，在伺服驱动系统中的状态反馈信号，作为检测信号输入至输入通道；

③ 输出通道接近被控对象，环境复杂恶劣，电磁和机械干扰较为严重。

5.3 解：

D/A 转换器（Digit to Analog Converter）是将数字量转换成模拟量的器件，通常用 DAC 表示，它将数字量转换成与之成正比的电量，广泛应用于过程控制中。

5.4 解：

D/A 转换器的主要性能指标有：

（1）分辨率：单位数字量所对应模拟量增量，即相邻两个二进制码对应的输出电压之差称为 D/A 转换器的分辨率。它确定了 D/A 产生的最小模拟量变化，也可用最低位（LSB）表示。

（2）精度：精度是指D/A转换器的实际输出与理论值之间的误差，它是以满量程V_{FS}的百分数或最低有效位（LSB）的分数形式表示。

（3）线性误差：D/A 转换器的实际转换特性（各数字输入值所对应的各模拟输出值之间的连线）与理想的转换特性（始、终点连线）之间是有偏差的，这个偏差就是 D/A 的线性误差。即两个相邻的数字码所对应的模拟输出值（之差）与一个 LSB 所对应的模拟值之差。常以 LSB 的分数形式表示。

（4）转换时间T_S（建立时间）：从D/A转换器输入的数字量发生变化开始，到其输出模拟量达到相应的稳定值所需要的时间称为转换时间。

5.5 解：

D/A 转换芯片的主要结构特性为：

（1）数字输入特性

数字输入特性包括接收数的码制、数据格式以及逻辑电平等。目前批量生产的 D / A 转换芯片一般都只能接收自然二进制数字代码。

（2）模拟输出特性

课后答案网 www.khdaw.com

目前多数 D/A 转换器件均属电流输出器件，手册上通常给出的输入参考电压及参考电阻之下的满码（全 1）输出电流 I0。另外还给出最大输出短路电流以及输出电压允许范围。

（3）锁存特性及转换控制

D/A 转换器对数字量输出是否具有锁存功能将直接影响与 CPU 的接口设计。如果 D/A 转换器没有输入锁存器，通过 CPU 数据总线传送数字量时，必须外加锁存器，否则只能通过具有输出锁存功能的 I / O 给 D/A 送入数字量

（4）参考电源

D/A 转换中，参考电压源是唯一影响输出结果的模拟参量，是 D/A 转换接口中的重要电路，对接口电路的工作性能、电路的结构有很大影响。

5.6 解：

① 8 位并行 D/A 转换；

② 片内二级数据锁存，提供数据输入双缓冲、单缓冲、直通三种工作方式；

③ 电流输出型芯片，电流稳定时间 1us。通过外接一个运算放大器，可以很方便地提供电压输出；

④ DIP20 封装、单电源（+5 V～+15 V，典型值+5 V）；

⑤ 只需在满量程下调整其线性度；

⑥ 单一电源供电（十 5V～＋15V），

⑦ 低功耗，200mW。

⑧ 与 MCS-51 连接方便。

5.7 解：

$\overline{CS}$：片选输入线，低电平有效。

ILE：数据锁存允许输入，高电平有效。

$\overline{WR1}$：写 1 信号输入，低电平有效。当$\overline{CS}$，ILE，$\overline{WR1}$=010 时，数据写入DAC0832 的第一级锁存。

$\overline{WR2}$：写 2 信号输入，低电平有效。

$\overline{XFER}$：数据传输信号输入，当$\overline{WR2}$，$\overline{XFER}$=00 时，数据由第一级锁存进入第二级锁存，并开始进行D/A转换。

5.8 解：

（1）D/A 转换器的数字输入是由数据线引入的，而数据线上的数据是变动的，为了保持D/A转换器输出的稳定，就必须在微处理器与D/A转换器输入口之间增加锁存数据的功能，即必须有锁存器。

（2）有锁存器和无锁存器的 D/A 转换器与 80C51 接口的电路有显著不同。

① 内部无锁存器的 D/A 转换器，如 DAC80（8 位）、AD7520（10 位）、AD7521（12 位）。它们的结构简单，内部不带锁存器。

对于这一类 D/A 转换器，最适合与单片机 80C51 的 P1、P2 等具有输出锁存功能的 I/O 口直接接口。但是当它们与 P0 口相接口时，则需在器输入端增加锁存器。

② 内部有锁存器的 D/A 转换器，不仅具有数据锁存器，而且还提供地址译码电路，有些包含双重，甚至多重的数据缓冲结构，如 DAC0832、DAC1230、AD7542 以及 AD7549 等。

这种类型的 D/A 转换器以高位的居多。它们以与 80C51 中的 P0 口相接口较为合适，一般只是是需要占用多根口线。

5.9 解：

课后答案网 www.khdaw.com

逐次逼近式 A/D 转换器的转换原理即“逐位比较”，如图 5.9 所市，它由 N 位寄存器、D/A 转换器、比较器和控制逻辑等部分组成，N 位寄存器代表 N 位二进制数码。

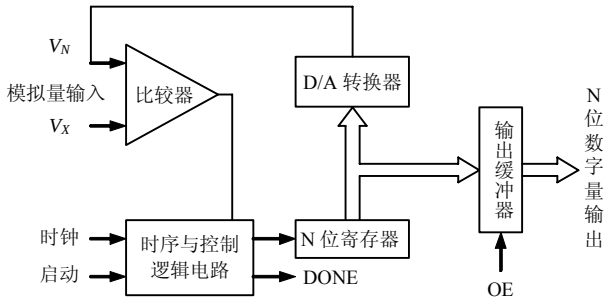


图 5.9 逐次逼近式 A/D 转换器原理图

当模拟量 Vx 送入比较器后，启动信号通过控制逻辑电路启动 A/D 开始转换。首先，置 N 位寄存器最高位（Dn-1）为“1”，其余位清“0”；N 位寄存器的内容经 D/A 转换后得到整个量程一半的模拟电压 VN,与输入电压 Vx 比较。若 Vx≥VN 时，则保留 Dn-1=1；若 Vx<VN 是，则 Dn-1 位清“0”。然后，控制逻辑使寄存器下一位（Dn-2）置“1”，与上次的结构一起经 D/A 转换后与 Vx 比较，重复上述过程，直至判别出 D0 位取“1”还是取“0”为止，此时控制逻辑电路发出转换结束信号 DONE。这样经过 N 次比较后，N 位寄存器的内容就是转换后的数字量数据，整个转换过程就是这样一个逐次比较逼近的过程。

5.10 解：

将模拟量变换成计算机能够直接处理的数字量，便于与计算机连接，通过计算机处理被测的模拟量。

5.11 解：

逐次逼近型的 A/D 转换器 DAC0809 、DAC0816

双积分型 A/D 转换器 MC14433 、ICL7035

5.12 解：

A/D 转换器位数的确定与整个测量控制系统所要测量控制的范围和精度有关，计算时应比总精度要求的最低分辨率高一位。实际选取的 A/D 转换器的位数应与其它环节所能达到的精度相适应。只要不低于它们就行，选得太高既没有意义，而且价格还要高得多。

5.13 解：

（1）积分型、电荷平衡型和跟踪比较型 A/D 转换器转换速度较慢，转换时间从几毫秒到几十毫秒不等，只能构成低速 A/D 转换器，一般运用于对温度、压力、流量等缓变参量的检测和控制。

（2）逐次比较型的 A/D 转换器的转换时间可从几μS 到 100μS 左右，属于中速 A/D 转换器，常用于工业多通道单片机控制系统和声频数字转换系统等。

（3）高速 A/D 转换器适用于雷达、数字通讯、实时光谱分析、实时瞬态记录、视频数字转换系统等。

5.14 解：

课后答案网 www.khdaw.com

在软件编写时，应根据硬件连接电路计算被选择的模拟通道的地址；执行一条输出指令，启动 A/D 转换；转换结束后，执行一条输入指令，读取 A/D 转换结果。

可以采用延时、查询和中断的方法判别 A/D 转换结束 。

5.15 解：

（1）应设置 D/A 转换器的双缓冲方式的情况

有些 D/A 转换器（如 DAC0832）的内部具有两极缓冲结构，即芯片内有一个 8 位输入寄存器和一个 8 位 DAC 寄存器。

这样的双缓冲结构，可以使 DAC 转换输出前一个数据的同时，将下一个数据传送到 8 位输入寄存器，以提高 D/A 转换的速度。更重要的是，能够使多个 D/A 转换器在分时输入数据后，同时输出模拟电压。

（2）D/A 转换器 DAC0832 的双缓冲方式的接口电路如图 5.15 所示。

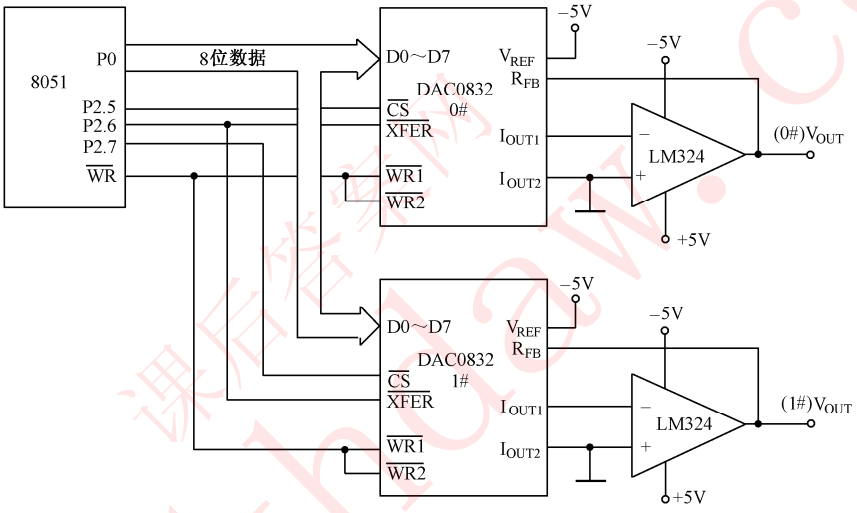


图 5.15　DAC0832 双缓冲连接电路图

5.16 解：

单片机只能处理数字形式的信息，但是在实际工程中大量遇到的是连续变化的物理量，如温度、压力、流量、光通量、位移量以及连续变化的电压、电流等。对于非电信号的物理量，必须先由传感器进行检测，并且转换为电信号，然后经过放大器放大为 0V～5V 电平的模拟量。所以必须加接模拟通道接口，以实现模拟量和数字量之间的转换。A/D(模/数)转换就是把输入的模拟量变为数字量，供单片机处理；而 D/A(数/模)转换就是将单片机处理后的数字量转换为模拟量输出。

5.17 解：

A/D 转换芯片中采样保持电路的作用是把一个时间连续的信号变换为时间离散的信号，并将采样信号保持一段时间。当外接模拟信号的变化速度相对于 A/D 转换速度来说足够慢，在转换期间内可视为直流信号的情况下，可以省略采样保持电路。

5.18 解：

对于 8 位 A/D 转换器，实际满量程电压为 5V，则其量化单位 1LSB=5V/256=0.0196V,考虑到 A/D 转换时会进行四舍五入处理，所以最大量化误差为（1/2）LSB,即 0.0098V。

5.19 解：

硬件电路连接图如图 5.19 所示。

课后答案网 www.khdaw.com

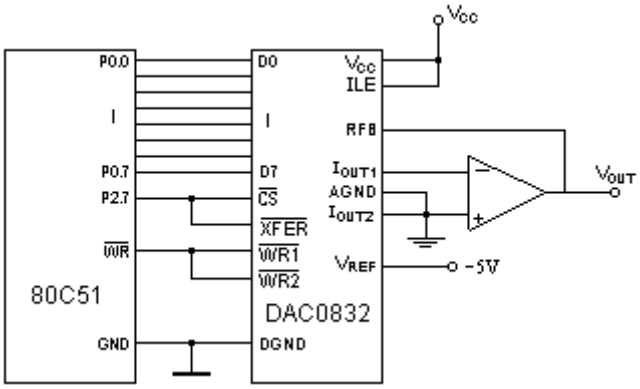


图 5.19　　5.19 题逻辑电路图

当V_O=2.5V时，D=80H；V_O=1.25V时，D=40H。

程序如下：

```
        ORG     0000H
    LJMP    MAIN
        ORG     0100H
MAIN:   MOV     DPTR，#7FFFH
NEXT:   MOV     A，#80H
        MOVX    @DPTR，A
        ACALL   DELAY
        MOV     R4，#04H
        MOV     A，#40H
        MOVX    @DPTR，A
LOOP:   ACALL   DELAY
        DJNZ    R4，LOOP
        AJMP    NEXT
DELAY:  ……
        RET
        END
```

5.20 解：硬件电路连接图如图 5.20 所示。

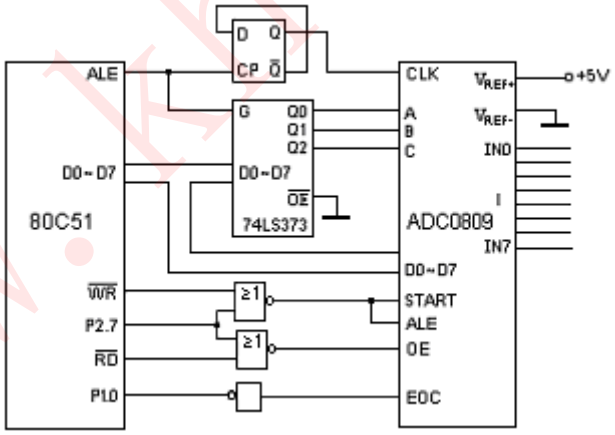


图 5.20　　5.20 题逻辑电路图

设 80C51 的时钟频率为 12MHz，程序如下：

```
        ORG     0000H
        LJMP    MAIN
        ORG     001BH
        LJMP    T1_1
        ORG     0100H
MAIN:   MOV     SP，#60H        ；设堆栈指针
        MOV     R7，#100        ；置采集次数
        MOV     R1，#30H        ；片外 RAM 地址高位
        MOV     R0，#00H        ；片外 RAM 地址低位
        MOV     R2，#20         ；置入初值 20（计 1 s）
        MOV     R3，#60         ；置入初值 60（计 1 min）
```

课后答案网 www.khdaw.com

```
        MOV     TOMD，#10H          ；定时器 T1 工作于模式 1
        MOV     TH1，#3CH           ；计数器初值
        MOV     TL1，#0B0H
        SETB    EA                  ；开中断
        SETB    ET1                 ；定时器 T1 允许中断
        SETB    TR1                 ；启动定时器 T1
LOOP：  SJMP    LOOP                ；等待中断
        DJNZ    R7，LOOP            ；是否到 100 次？
        SJMP    $
        ORG     0200H
T1_1：  MOV     TH1，#3CH           ；中断服务程序，重新赋计数器初值
        MOV     TL1，#0B0H
        DJNZ    R2，RETI_0          ；1 s 未到，返回
        MOV     R2，#20             ；重新置“100 s”计数器初值
        DJNZ    R3，RETI_0          ；1 min 未到，返回
        MOV     R6，#8              ；8 个通道计数器初值
        MOV     DPTR，#7FF8H        ；IN0 地址
LOOP1：MOVX     @DPTR，A            ；启动 A/D 转换
        JB      P1.0，$             ；判转换是否结束
        MOVX    A，@DPTR            ；读取转换结果
        PUSH    DPH                 ；将通道地址压入堆栈
        PUSH    DPL
        MOV     DPH，R1             ；将片外 RAM 地址送 DPTR
        MOV     DPL，R0
        MOVX    @DPTR，A            ；将转换结果存入片外 RAM
        INC     DPTR                ；片外 RAM 地址增 1
        MOV     R1，DPH             ；保存片外 RAM 地址
        MOV     R0，DPL
        POP     DPL                 ；恢复通道地址
        POP     DPH
        INC     DPTR
        DJNZ    R6，LOOP1           ；8 个通道是否采集结束
RETI_0：RETI                        ；中断返回
        END
```

第 6 章

6.1 解：

具有人机对话功能。实现人对应用系统的状态干预和数据输入以及应用系统向人报告运行和运行结果。

6.2 解：

发光二极管显示器,简称 LED(Light Emitting Diode)；

液晶显示器，简称 LCD(Liquid Crystal Display)；

荧光管显示器，简称 CRT。

6.3 解：

要使 LED 显示器显示出字符，必须提供段码和位选码。

段码（即字码）：可以用硬件译码的方法获得，也可以用软件的方法获得。

位选码：静态显示和动态显示。

6.4 解：

（1）建立显示数据缓冲区：存放待显示数字、字符在字型编码表中的序号；

（2）软件译码：利用查表方法获得字型编码(段码)；

课后答案网 www.khdaw.com

（3）位扫描输出：采用移位方法逐位点亮 LED 显示器；

（4）延时子程序：控制点亮时间和时间间隔。

6.5 解：

（1）静态显示方式：静态显示方式是指当显示器显示某一字符时，发光二极管的位选始终被选中。在这种显示方式下，每一个 LED 数码管显示器都需要一个 8 位的输出口进行控制。由于单片机本身提供的 I/O 口有限，实际使用中，通常通过扩展 I/O 口的形式解决输出口数量不足的问题。

静态显示主要的优点是显示稳定，在发光二极管导通电流一定的情况下显示器的亮度大，系统运行过程中，在需要更新显示内容时，CPU 才去执行显示更新子程序，这样既节约了 CPU 的时间，又提高了 CPU 的工作效率。其不足之处是占用硬件资源较多，每个 LED 数码管需要独占 8 条输出线。随着显示器位数的增加，需要的 I/O 口线也将增加。

（2）动态显示方式：动态显示方式是指一位一位地轮流点亮每位显示器（称为扫描），即每个数码管的位选被轮流选中，多个数码管公用一组段选，段选数据仅对位选选中的数码管有效。对于每一位显示器来说，每隔一段时间点亮一次。显示器的亮度既与导通电流有关，也与点亮时间和间隔时间的比例有关。通过调整电流和时间参数，可以既保证亮度，又保证显示。若显示器的位数不大于 8 位，则显示器的公共端只需一个 8 位 I/O 口进行动态扫描（称为扫描口），控制每位显示器所显示的字形也需一个 8 位口（称为段码输出）。

6.6 解：

通常的按键所用开关为机械弹性开关。由于机械触电的弹性作用，按键在闭合及断开的瞬间均伴随有一连串的抖动。键抖动会引起一次按键被误读多次。为了确保 CPU 对键的一次闭合仅作一次处理，必须去除抖动。

消除抖动的方法有硬件和软件两种方法。硬件方法常用 RS 触发器电路。软件方法是当检测出键闭合后执行一个 10ms~20ms 的延时程序，再一次检测键的状态，如仍保持闭合状态，则确认真正有键按下。

6.7 解：

液晶显示器简称 LCD（Liquid Crystal Diodes），是一种被动式的显示器，即液晶本身并不发光，利用液晶经过处理后能够改变光线传输方向的特性，达到显示字符或者图形的目的。

LCD 显示器有笔段式和点阵式两种，点阵式又可分为字符型和图像型。笔段式 LCD 显示器类似于 LED 数码管显示器。每个显示器的段电极包括七个笔划（段）和一个背电极 BP（或 COM）。可以显示数字和简单的字符，每个数字和字符与其字形码（段码）对应。

点阵式 LCD 显示器的段电极与背电极呈正交带状分布，液晶位于正交的带状电极间。点阵式 LCD 的控制一般采用行扫描方式，通过两个移位寄存器控制所扫描的点。

80C51 与液晶显示模块 LCM 的基本接口电路如图 6.7 所示。

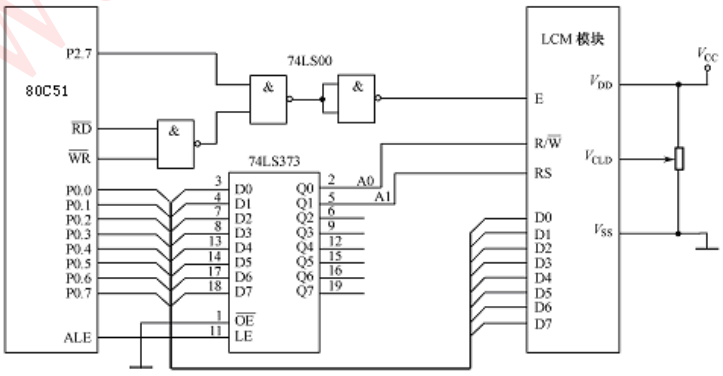


图 6.7 80C51 与液晶显示模块 LCM 的基本接口电路

液晶显示模块初始化子程序（将系统设置成显示 2 行字符，5×7 点阵，开显示，显示光标，字符闪烁，清屏，光标为移动方式，自动地址为增量方式。）：

电源打开后，在电源上升到 4.5V 并维持 15ms 后，写入功能设置控制字，选择数据接口位数等；等待 5ms 后，检查忙标志，在不忙的情况下，再进行其他的功能设置；检查忙

课后答案网 www.khdaw.com

标志，在不忙的情况下，关显示；检查忙标志，在不忙的情况下，清屏；检查忙标志，在不忙的情况下，设定输入方式，初始化结束。程序如下：

```
LCD:    MOV A, #38H                 ; 8 位数据，2 行显示，5×7 点阵
        MOV DPTR, #8000H            ; LCD 的口地址
        MOVX @DPTR, A
        LCALL BUSY
        MOV A, #01H                 ; 清屏
        MOV DPTR, #8000H
        MOVX @DPTR, A
        LCALL BUSY
        MOV A, #07H                 ; AC 自动加 1，整体显示移动
        MOV DPTR, #8000H
        MOVX @DPTR, A
        LCALL BUSY
        MOV A, #0FH                 ; 开显示，开光标，字符闪烁
        MOV DPTR, #8000H
        MOVX @DPTR, A
        RET
BUSY:   PUSH DPH                    ; 保护现场
        PUSH DPL
        PUSH PSW
        PUSH ACC
LOOP:   MOV DPTR, #8001H            ; 读 BH 及 AC
        MOVX A, @DPTR
        JB ACC.7, LOOP              ; 忙，继续等待
        POP ACC                     ; 不忙，恢复现场返回
        POP PSW
        POP DPL
        POP DPH
        RET
```

6.8 解：

（1）判断键盘上有无键按下：列输出全 0，读行输入状态，若状态为全 1，则说明键盘无键按下；若不全为 1，则说明键盘有按下。

（2）消除按键抖动的影响：在判断有键按下后，用软件延时的方法，再判断键盘状态，如果仍为有键按下状态，则认为有一个确定的键按下，否则当作按键抖动处理。

（3）求按键位置，计算键号：用扫描的方法识别闭合键 N 所在的行号 X 和列号 Y，并根据：以下公式计算闭合键的键号 N =X 行首键号+列号 Y。

（4）键闭合一次仅进行一次按键处理：方法是等待按键释放之后，再进行按键功能的处理操作。

6.9 解：

硬件电路连接图如图 6.9 所示。

课后答案网 www.khdaw.com

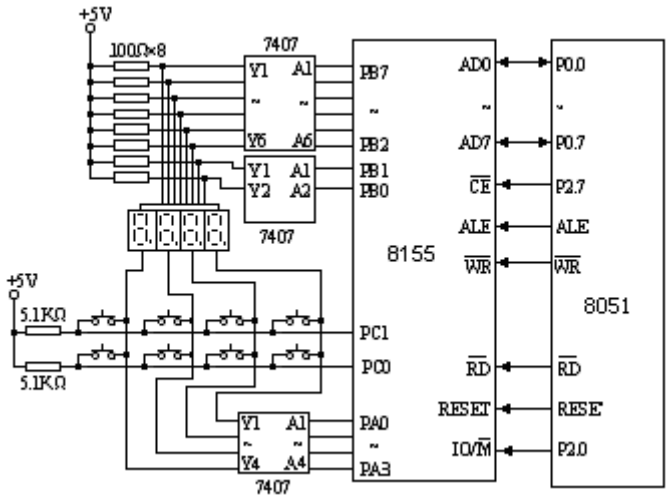


图 6.9　6.9 题 LED 显示器/键盘电路图

6.10 解：

硬件连接电路图如图 6.10 所示。

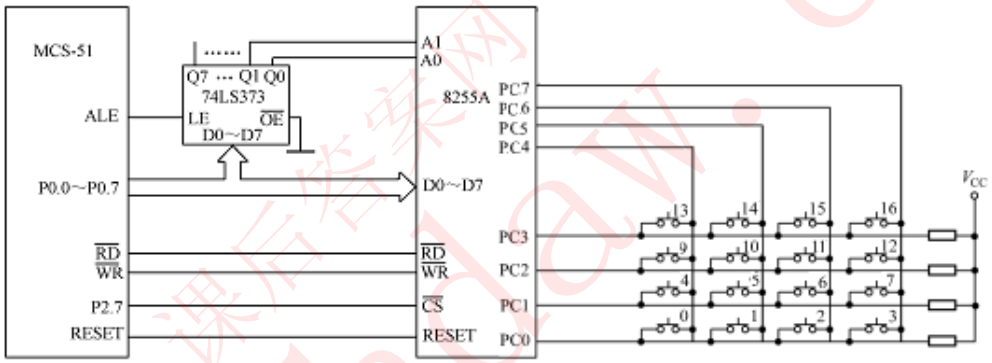


图 6.10　6.10 题扩展键盘电路图

程序如下：

```
        ORG     1000H
START:  MOV     DPTR, #7FFFH    ; 8255 初始化
        MOV     A, #81H
        MOVX    @DPTR, A
KEY:    ACALL   KEY1            ; 检查有键闭合否
        JNZ     LKEY1           ; A 非 0 说明有键按下
        ACALL   DELAY1          ; 执行一次延时子程序（延时 6 ms）
        AJMP    KEY
LKEY1:  ACALL   DELAY1
        ACALL   DELAY1          ; 有键闭合延时 2×6ms=12ms 以去抖动
        ACALL   KEY1            ; 延时以后再检查是否有键闭合
        JNZ     LKEY2           ; 有键闭合，转 LKEY2
        ACALL   DELAY1          ; 无键闭合，说明是干扰信号，不作处理
        AJMP    KEY             ; 延时 6ms 后转 KEY 继续等待键入
LKEY2:  MOV     R2, #0EFH       ; 扫描初值送 R2，设定 PC4 为当前扫描线
        MOV     R4, #00H        ; 回送初值送 R4
LKEY4:  MOV     DPTR, #7FFEH    ; 指向 PC 口
        MOV     A, R2
        MOVX    @DPTR, A        ; 扫描初值送 PC 口
        MOV     A, @DPTR            ; 取回送线状态
        JB      ACC.0, LONE     ; ACC.0=1，第 0 行无键闭合，转 LONE
        MOV     A, #00H         ; 装第 0 行行值
```

课后答案网 www.khdaw.com

```
        AJMP    LKEYP       ; 转计算键码
LONE:   JB      ACC.1, LTWO ;ACC.1=1，第 1 行无键闭合，转 LTWO
        MOV     A, #08H     ; 装第 1 行行值
        AJMP    LKEYP       ; 转计算键码
LTWO:   JB      ACC.2, LTHR ;ACC.2=1，第 2 行无键闭合，转 LTHR
        MOV     A, #10H     ; 装第 2 行行值
        AJMP    LKEYP
LTHR:   JB      ACC.3, NEXT ;ACC.3=1，第 3 行无键闭合，转 NEXT
        MOV     A, #18H     ; 装第 3 行行值
LKEYP:  ADD     A, R4       ; 计算键码
        PUSH    ACC         ; 保存键码
LKEY3:  ACALL   DELAY1      ; 延时 6ms
        ACALL   KEY1        ; 判断键是否继续闭合，若闭合再延时
        JNZ     LKEY3
        POP     ACC         ; 若键释放，则键码送 A
        RET
NEXT:   INC     R4          ; 列号加 1
        MOV     A, R2
        JNB     ACC.7, KND  ; 第 7 位为 0，以扫描到最高列，转 KND
        RL      A           ; 循环左移一位
        MOV     R2, A
        AJMP    LKEY4       ; 进行下一列扫描
KND:    AJMP    KEY         ; 扫描完毕，开始新的一轮
KEY1:   MOV     DPTR, #7FFEH        ; 将 PC 口地址送 DPTR，PC 口高四位作为扫描线
        MOV     A, #00H             ; 所有扫描线均为低电平
        MOVX    @DPTR, A            ; PC 口向列线输出 00H
        MOVX    A, @DPTR            ; 取回送线状态
        CPLA                        ; 行线状态取反
        ANL     A, #0F0H            ; 屏蔽 A 的高四位
        RET                         ; 返回
DELAY1: MOV R7，#4              ; 延时 6mS 延时子程序
        D1:MOV R6，#220
        D2: DJNZ R6，D2
            DJNZ R6，D2
            DJNZ R7，D1
            RET
            END
```

6.11 解：

硬件连接电路图如图 6.11 所示。

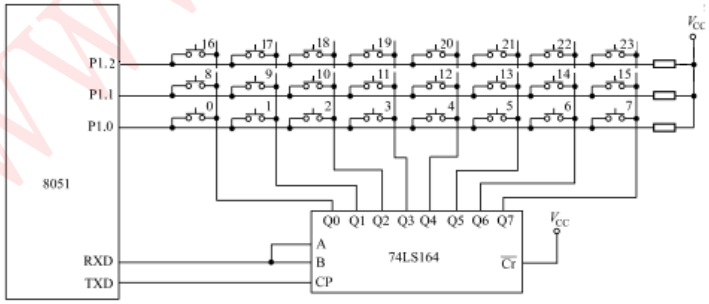


图 6.11　6.11 题扩展键盘电路图

程序如下：

```
        ORG     1000H
SERKEY: MOV     SCON, #00H  ; 设置串行口
        MOV     A, #00H     ; 键盘初始化，送 00H 到列线上
        LCALL   VARTO       ; 发送数据
```

课后答案网 www.khdaw.com

```
CHK:    JNB     P1.0, CHK0      ; 检查是否有键按下
        JNB     P1.1, CHK0      ; 检查是否有键按下
        JNB     P1.2, CHK0      ; 检查是否有键按下
        AJMP    CHK             ; 无键按下，继续查找
CHK0:   LCALL   DLY1            ; 调用 10ms 延时子程序，去抖
        JNB     P1.0, CHEN      ; 确实有键按下，转 CHEN
        JNB     P1.1, CHEN
        JNB     P12, CHEN
        AJMP    CHK             ; 无键按下，继续查找
CHEN:   MOV     R2, #0FEH       ; 首列扫描字送 R2，查键号，最低位为 0
        MOV     R4, #00H        ; 首列偏移值送 R4
CHKN:   MOV     A, R2           ; 发送列扫描字
        LCALL   VARTO
        JB      P1.0, CH1       ; 检查 P1.0 有无键按下；若无，转 CH1
        MOV     A, #0           ; 第一行首列值送 A，00H+（R4）
        AJMP    CKEY            ; 转求键号
CH1:    JB      P1.1, CH2       ; 检查 P1.1 有无键按下；若无，转 NEXT
        MOV     A, #8           ; 第二行首列值送 A
CH2:    JB      P1.1, NEXT      ; 检查 P1.1 有无键按下；若无，转 NEXT
        MOV     A, #16          ; 第三行首列值送 A
        AJMP    CKEY            ; 转求键号
CKEY:   ADD     A, R4           ; 求键号，并入栈保护
        RET
NEXT:   INC     R4              ; 指向下一列
        MOV     A, R2           ; 取出原扫描字
        JNB     ACC.7，KEND     ; 是否已检查完 8 列?
        RL      A               ; 8 列未完，指向下一列
        MOV     R2, A           ; 列扫描字送 R2
        AJMP    CHKN            ; 8 列未完，检查下一列
KEND:   AJMP    SERKEY          ; 8 列查完，未查到有键按下，等待
VARTO:  MOV     SBUF, A         ; 发送 A 中数据
        JNB     TI, $           ; 发送等待
        CLR     TI              ; 清除
        RET

DLY1:   ……                      ; 延时 10ms 子程序（略）
        END                     ; 结束
```

6.12 解：

硬件连接电路图如图 6.12 所示。

课后答案网 www.khdaw.com

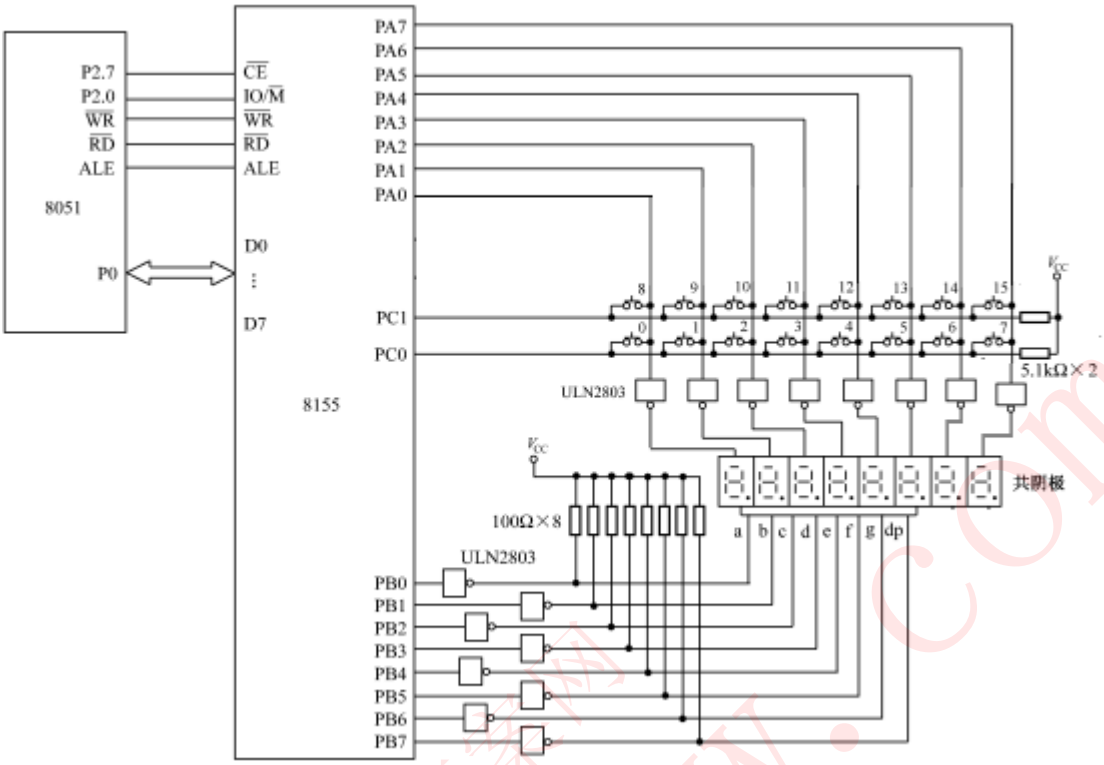


图 6.12　6.12 题扩展键盘显示器电路图

程序如下：

```
        ORG     1000H
KD1:    MOV     A, #0000 0011B ; 8155 初始化，PA，PB 基本输出方式，PC 输入方式
        MOV     DPTR, #7F00H
        MOVX    @DPTR, A
KEY1:   ACALL   KS1            ; 调用判断是否有键闭合子程序
        JNZ     LK1            ; 有键闭合转 LK1
        ACALL   DSP8155        ; 调用 8155 动态显示子程序，延时 6ms
        AJMP    KEY1
LK1:    ACALL   DSP8155
        ACALL   DSP8155        ; 调用两次显示，延时 12ms
        ACALL   KS1
        JNZ     LK2
        ACALL   DSP8155        ; 调用 8155 动态显示子程序，延时 6ms
        AJMP    KEY1
LK2:    MOV     R2, #0FEH
        MOV     R4, #00H
LK3:    MOV     DPTR, #7F01H
        MOV     A, R2
        MOVX    @DPTR, A
        INC     DPTR
        INC     DPTR
        MOVX    A, @DPTR
        JB      ACC.0, LONE
        MOV     A, #00H
        AJMP    LKP
LONE:   JB      ACC.1, LTWO
        MOV     A, #08H
        AJMP    LKP
LKP:    ADD     A, R4
```

课后答案网 www.khdaw.com

```
        PUSH    ACC
LK4:    ACALL   DSP8155
        ACALL   KS1
        JNZ     LK4
        POP     ACC
NEXT:   INC     R4
        MOV     A, R2
        JNB     ACC.7, KND
        RL      A
        MOV     R2, A
        AJMP    LK3
KND:    AJMP    KEY1
KS1:    MOV     DPTR, #7F01H
        MOV     A, #00H
        MOVX    @DPTR, A
        INC     DPTR
        INC     DPTR
        MOVX    A, @DPTR
        CPL     A
        ANL     A, #0FH
        RET
DSP8155: MOV    DPTR, #7F00H        ; 指向 8155 命令寄存器
        MOV     A, #00000011B       ; 设定 PA 口、PB 口为基本输出方式
        MOVX    @DPTR, A            ; 输出命令字
DISP1:  MOV     R0, #7EH            ; 指向缓冲区末地址
        MOV     A, #80H             ; 扫描字，PA7 为 1，从左至右扫描
LOOP:   MOV     R2, A               ; 暂存扫描字
        MOV     DPTR, #7F01H        ; 指向 8155 的 PA
        MOVX    @DPTR, A            ; 输出位选码
        MOV     A, @R0              ; 读显示缓冲区一字符
        MOV     DPTR, #PTRN         ; 指向段数据表首地址
        MOVC    A, @A+DPTR          ; 查表，得段数据
        MOV     DPTR, #7F02H        ; 指向 8155 的 PB
        MOVX    @DPTR, A            ; 输出段数据
        CALL    D1MS                ; 延时 1ms
        DEC     R0                  ; 调整指针
        MOV     A, R2               ; 读回扫描
        CLR     C                   ; 清进位标志
        RRC     A                   ; 扫描字右移
        JC      PASS                ; 结束
        AJMP    LOOP                ; 继续显示
PASS:   RET                         ; 返回
D1MS:   MOV     R7, #02H            ; 延时 1ms 子程序
DMS:    MOV     R6, #0FFH
        DJNZ    R6, $
        DJNZ    R7, DMS
        RET
PTRN:   DB      0C0H, 0F9H, 0A4H, 0B0H, 99H        ; 段数据表
        DB      ……
        DB      ……
        ……

        END
```

6.13 解：

课后答案网 www.khdaw.com

硬件连接电路如图 6.13 所示。8155 控制口的口地址为：7F00H；PA 口地址：7F01H；PB 口地址：7F02H；PC 口地址：7F03H。片内 RAM 地址：7E00H~7EFFH。定时器低位地址：7F04H；定时器高位地址：7F05H。

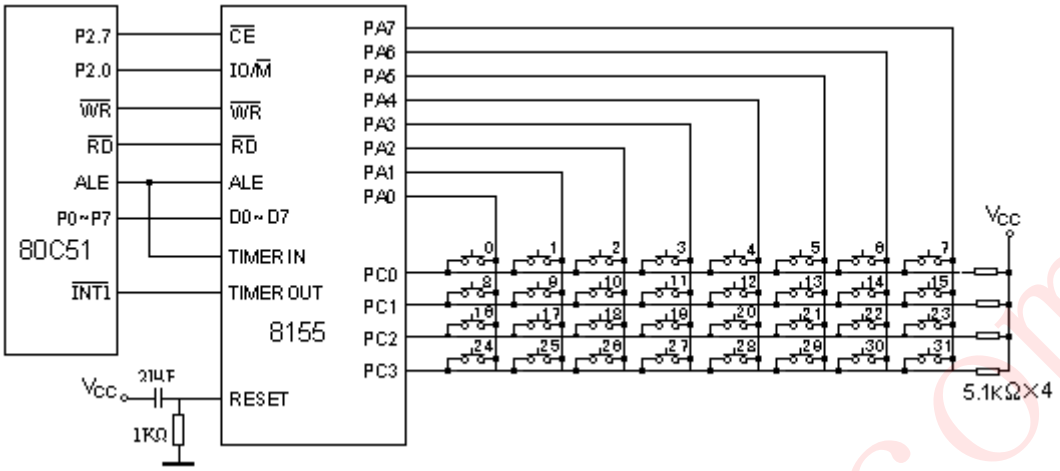


图 6.13　6.13 题扩展键盘电路图

程序如下：

```
          ORG     0000H
          LJMP    MAIN
          ORG     0013H
          LJMP    IEX1
          ORG     0030H
MAIN:  MOV SP，#60H
          MOV     R5，#200              ；中断次数计数值
          MOV     DPTR，#7FF8H
          MOV     A，#11000001          ；8155 初始化
          MOVX    @DPTR，A
          MOV     DPTR，#7FFCH          ；写定时器初值
          MOV     A，#52H
          MOVX    @DPTR，A
          INC     DPTR
          MOV     A，#27H
          MOVX    @DPTR，A
          MOV     DPTR，#7E40H          ；8155 片内 RAM 地址
          SETB    EX1
          SETB    EA
          SJMP    $
          ORG     0200H
IEX1:  DJNZ R5,RETI_0
          LCALL   KEY1                  ；调用键盘扫描及求键值子程序
          MOVX    @DPTR，A              ；存键值
          INC     DPTR                  ；修改地址
RETI_0:  RETI  ；中断返回
          ORG     0100H
KEY1:  ACALL   KS1                      ；调用判断有无键按下子程序
          JNZ     LK1                   ；有键按下时，（A）≠0，转消抖动延时
          AJMP    KEY1                  ；无键按下返回
LK1:  ACALL   T12MS                     ；调延时 12 ms 子程序
          ACALL   KS1                   ；查有无键按下，若有则为键确实按下
          JNZ     LK2                   ；键按下，（A）≠0，转逐列扫描
          AJMP    KEY1                  ；不是键按下返回
LK2:  MOV     R4，#00H                  ；首列号入 R4
          MOV     R2，#0FEH             ；首列扫描字送 R2
LK3:  MOV     A，R2                     ；第一次列扫描
          MOV     DPTR，#7FF9H          ；列扫描字送至 8155PA 口
          MOVX    @DPTR，A              ；使第 0 列线为 0
```

课后答案网 www.khdaw.com

```
                INC     DPTR            ；指向 8155PC 口
                INC     DPTR
                MOVX    A，@DPTR        ；8155PC 口读入行状态
                JB      ACC.0，LONE     ；第 0 行无键按下，转查第 1 行
                MOV     A，#00H         ；第 0 行有键按下，该行首键号#00→A
                AJMP    LKP             ；转求键号
LONE：  JB      ACC.1，LTWO             ；第 1 行无键按下，转查第 2 行
                MOV     A，#08H         ；第 1 行有键按下，该行首键号#08→A
                AJMP    LKP
LTWO：  JB      ACC.2，LTHR             ；第 2 行无键按下，转查第 3 行
                MOV     A，#10H         ；第 2 行有键按下，该行首键号#10H→
A
                LJMP    LKP
LTHR：  JB      ACC.3，LOOP             ；第 3 行无键按下，转查下一列
                MOV     A，#18H         ；第 3 行有键按下，该行首键号#18H→
A
  LKP：  ADD    A，R4                   ；键号=行首建号+列号
         PUSH   ACC                     ；键号进栈保护
  LK4：  ACALL  KS1                     ；等待键释放
         JNZ    LK4                     ；未释放，等待
         POP    ACC                     ；键释放，键号→A
         RET                            ；键扫描结束，出口状态：（A）=键号
LOOP：  INC     R4                      ；指向下一列，列号加 1
         MOV    A，R2                   ；判断 8 列扫描完没有
         JNB    ACC.7，KND              ；8 列扫描完，返回
         RL     A                       ；扫描字左移一位，转变为下一列扫描
字
         MOV    R2，A                   ；扫描字入 R2
         AJMP   LK3                     ；转下一列扫描
    KND：AJMP   KEY1
  KS1：  MOV    DPTR，#7FF9H            ；指向 PA 口
         MOV    A，#00H                 ；全扫描字#00H=00000000B
         MOVX   @DPTR，A                ；全扫描字入 PA 口
         INC    DPTR                    ；指向 PC 口
         INC    DPTR
         MOVX   A，@DPTR                ；读入 PC 口行状态
  CPL    A
         ANL    A，#0FH
         RET                            ；出口状态：（A）≠0 时有键按下
T12MS：  MOV    R7，#18H
T12MS1：  MOV   R6，#0FFH
          DJNZ  R6，$
          DJNZ  R7，T12MS1
     RET
     END
```

6.14 解：打印机接口电路如图 6.14 所示。

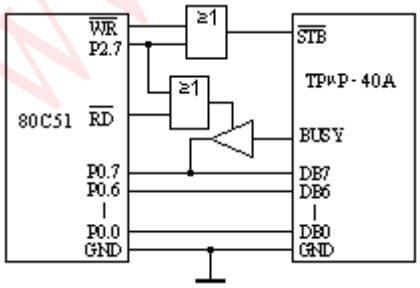


图 6.14　打印机接口电路

程序如下：

```
ORG             0000H
        LJMP    START
        ORG     0100H
START：  MOV    R0，#30H
        MOV     R6，#10
        MOV     DPTR，#7FFFH
LOOP：  MOVX    A，@DPTR
        JB      ACC.7，LOOP
        MOV     A，@R0
        MOVX    @DPTR，A
```

课后答案网 www.khdaw.com

```
INC     R0
DJNZ    R6，LOOP
RET
END
```

第 7 章

7.1 解：

单片机应用系统设计的一般方法及步骤如图 7.1 所示。

（1）明确设计任务

（2）器件选择

（3）总体设计

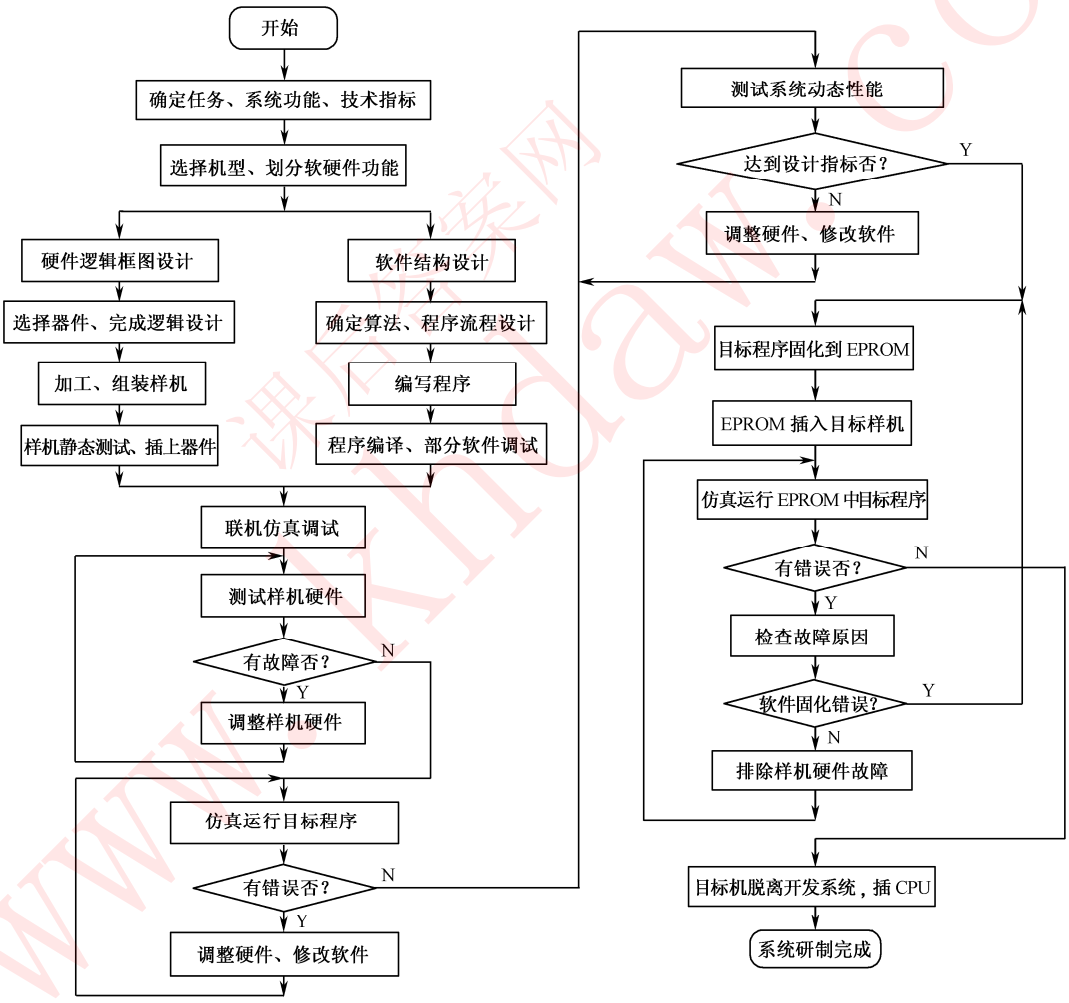


图 7.1　单片机应用系统设计过程流程图

7.2 解：

（1）硬件电路设计

在硬件设计时，要尽量应用最新单片机，采用新技术。要注意通用性的问题，尽可能选择典型电路，并符合单片机常规用法，为硬件系统的标准化、模块化打下良好的基础。系统扩展与外围设备的配置水平应充分满足应用系统的功能要求，并留有适当余地，以便进行二次开发。硬件系统设计应尽量朝“单片”（片上系统 SOC）方向发展，以提高系统的稳定性。

课后答案网 www.khdaw.com

工艺设计时要考虑安装、调试、维修的方便。扩展接口的开发尽可能采用 PSD 等器件开发。

（2）软件设计

软件设计随单片机应用系统的不同而不同。图 7.2 给出了单片机软件设计的流程图。一般可分为以下几个方面。

① 总体规划

② 程序设计技术： 模块程序设计、自顶向下的程序设计。

③ 程序设计：建立数学模型、绘制程序流程图、程序的编制。

④ 软件装配。

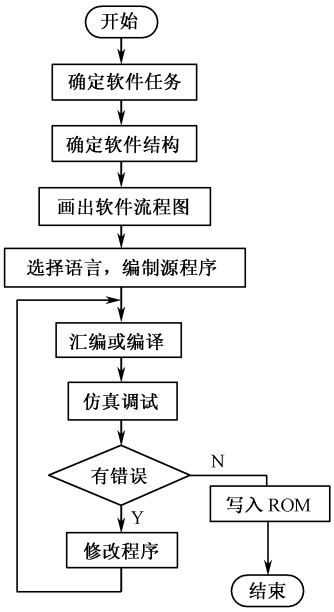


图 7.2 软件设计的流程图

7.3 解：

硬件设计：

（1）存储器扩展：容量需求，在选择单片机时就考虑到单片机的内部存储器资源，如能满足要求就不需要进行扩展，在必须扩展时注意存储器的类型、容量和接口，一般尽量留有余地，并且尽可能减少芯片的数量。选择合适的方法、ROM 和 RAM 的形式，RAM 是否要进行掉电保护等。

（2）I/O 接口的扩展：单片机应用系统在扩展 I/O 接口时应从体积、价格、负载能力、功能等几个方面考虑。应根据外部需要扩展电路的数量和所选单片机的内部资源（空闲地址线的数量）选择合适的地址译码方法。

（3）输入通道的设计：输入通道设计包括开关量和模拟输入通道的设计。开关量要考虑接口形式、电压等级、隔离方式、扩展接口等。模拟量通道的设计要与信号检测环节（传感器、信号处理电路等）结合起来，应根据系统对速度、精度和价格等要求来选择，同时还需要和传感器等设备的性能相匹配，要考虑传感器类型、传输信号的形式（电流还是电压）、线性化、补偿、光电隔离、信号处理方式等，还应考虑 A/D 转换器的选择（转换精度、转换速度、结构、功耗等）及相关电路、扩展接口，有时还涉及软件的设计。高精度的模数转换器价格十分昂贵，因而应尽量降低对 A/D 转换器的要求，能用软件实现的功能尽量用软件来实现。

（4）输出通道的设计：输出通道设计包括开关量和模拟量输出通道的设计。开关量要考虑功率、控制方式（继电器、可控硅、三极管等）。模拟量输出要考虑 D/A 转换器的选择（转换精度、转换速度、结构、功耗等）、输出信号的形式（电流还是电压）、隔离方式、扩展接口等。

（5）人机界面的设计：人机界面的设计包括输入键盘、开关、拨码盘、启/停操作、复位、显示器、打印、指示、报警等。输入键盘、开关、拨码盘应考虑类型、个数、参数及相关处理（如按键的去抖处理）。启/停、复位操作要考虑方式（自动、手动）及其切换。显示器要考虑类型（LED，LCD）、显示信息的种类、倍数等。此外还要考虑各种人机界面的扩展接口。

（6）通信电路的设计：单片机应用系统往往作为现场测控设备，常与上位机或同位机构成测控网络，需要其有数据通信的能力，通常设计为 RS-232C、RS-485、红外收发等通信标准。

（7）印刷电路板的设计与制作：电路原理图和印刷电路板的设计常采用专业设计软件进行设计，如 Protel，OrCAD 等。设计印刷电路板需要有很多的技巧和经验，设计好印刷

课后答案网 www.khdaw.com

电路板图后应送到专业化制作厂家生产，在生产出来的印刷电路板上安装好元件，则完成硬件设计和制作。

（8）负载容限的考虑：单片机总线的负载能力是有限的。如 MCS-51 的 P0 口的负载能力为 4mA，最多驱动 8 个 TTL 电路，P1～P3 口的负载能力为 2mA，最多驱动 4 个 TTL 电路。若外接负载较多，则应采取总线驱动的方法提高系统的负载容限。常用驱动器有：单向驱动器 74LS244，双向驱动器 74LS245 等。

（9）信号逻辑电平兼容性的考虑：在所设计的电路中，可能兼有 TTL 和 CMOS 器件，也有非标准的信号电平，要设计相应的电平兼容和转换电路。当有 RS-232，RS-485 接口时，还要实现电平兼容和转换。常用的集成电路有 MAX232，MAX485 等。

（10）电源系统的配置：单片机应用系统一定需要电源，要考虑电源的组数、输出功率、抗干扰。要熟悉常用三端稳压器（78xx 系列、79xx 系列）、精密电源（AD580，MC1403，CJ313/336/385，W431）的应用。

（11）抗干扰的实施：采取必要的抗干扰措施是保证单片机系统正常工作的重要环节。它包括芯片、器件选择、去耦滤波、印刷电路板布线、通道隔离等。

软件设计：

（1）总体规划：软件所要完成的任务已在总体设计时规定，在具体软件设计时，要结合硬件结构，进一步明确软件所承担的一个个任务细节，确定具体实施的方法，合理分配资源。

（2）程序设计技术：合理的软件结构是设计一个性能优良的单片机应用系统软件的基础。在程序设计中，应培养结构化程序设计风格，各功能程序实行模块化、子程序化。一般有以下两种设计方法：

（a）模块程序设计：模块程序设计是单片机应用中常用的一种程序设计技术。它是把一个较长的程序分解为若干个功能相对独立的较小的程序模块，各个程序模块分别设计、编程和调试，最后由各个调试好的模块组成一个大的程序。其优点是单个功能明确的程序模块的设计和调试比较方便，容易完成，一个模块可以为多个程序所共享。其缺点是各个模块的连接有时有一定难度。

（b）自顶向下的程序设计：自顶向下程序设计时，先从主程序开始设计，从属程序或子程序用符号来代替。主程序编好后再编制各从属程序和子程序，最后完成整个系统软件的设计。其优点是比较符合于人们的日常思维，设计、调试和连接同时按一个线索进行，程序错误可以较早的发现。缺点是上一级的程序错误将对整个程序产生影响，一处修改可能引起对整个程序的全面修改。

（3）程序设计：在选择好软件结构和所采用的程序设计技术后，便可着手进行程序设计，将设计任务转化为具体的程序。

（a）建立数学模型：根据设计任务，描述出各输入变量和各输出变量之间的数学关系，此过程即为建立数学模型。数学模型随系统任务的不同而不同，其正确度是系统性能好坏的决定性因素之一。

（b）绘制程序流程图：通常在编写程序之前先绘制程序流程图，以提高软件设计的总体效率。程序流程图以简明直观的方式对任务进行描述，并很容易由此编写出程序，故对初学者来说尤为适用。

在设计过程中，先画出简单的功能性流程图（粗框图），然后对功能流程图进行细化和具体化，对存储器、寄存器、标志位等工作单元作具体的分配和说明，将功能流程图中每一个粗框的操作转变为具体的存储器单元、工作寄存器或 I/O 口的操作，从而给出详细的程序流程图（细框图）。

（c）程序的编制：在完成程序流程图设计以后，便可以编写程序。程序设计语言对程序设计的影响较大。汇编语言是最为常用的单片机程序语言，用汇编语言编写程序代码精简，

课后答案网 www.khdaw.com

直接面向硬件电路进行设计，速度快，但进行大量数据运算时，编写难度将大大增加，不易阅读和调试。在有大量数据运算时可采用 C 语言（如 MCS-51 的 C51）或 PL/M 语言。
编写程序时，应注意系统硬件资源的合理分配与使用，子程序的入/出口参数的设置与传递。采用合理的数据结构、控制算法，以满足系统要求的精度。在存储空间分配时，应将使用频率最高的数据缓冲器设在内部 RAM；标志应设置在片内 RAM 位操作区（20H～2FH）中；指定用户堆栈区，栈区的大小应留有余量；余下部分作为数据缓冲区。
在编写程序过程中，根据流程图逐条用符号指令来描述，即得汇编语言源程序。应按 MCS-51 汇编语言的标准符号和格式书写，在完成系统功能的同时应注意保证设计的可靠性，如数字滤波、软件陷阱、保护等。必要时可作若干功能性注释，提高程序的可读性。

（4）软件装配：各程序模块编辑之后，需进行汇编或编译、调试，当满足设计要求后，将各程序模块按照软件结构设计的要求连接起来，即为软件装配，从而完成软件设计。在软件装配时，应注意软件接口。

7.4 解：

一个单片机应用系统的硬件设计包含两部分内容：一是系统扩展，即当单片机内部的功能单元不能满足应用系统的要求时必须进行片外扩展，选择适当的芯片，设计相应的电路；二是系统的配置，即按照系统功能要求配置外围设备，如通信接口、键盘、显示器、打印机、A/D、D/A 转换器等，要设计合理的接口电路。

7.5 解：

（1）静态检查

根据硬件电路图核对元器件的型号、规格、极性、集成芯片的插接方向是否正确。用逻辑笔、万用表等工具检查硬件电路连线是否与电路图一致，有无短路、虚焊等现象。严防电源短路和极性接反。检查数据总线、地址线和控制总线是否存在短路的故障。

（2）通电检查

通电检查时，可以模拟各种输入信号分别送入电路的各有关部分，观察 I/O 口的动作情况，查看电路板上有无元件过热、冒烟、异味等现象，各相关设备的动作是否符合要求，整个系统的功能是否符合要求。

7.6 解：

可靠性通常是指在规定的条件下，在规定的时间内完成规定功能的能力。可采用以下的方法提高系统的可靠性。

（1）隔离技术。

（2）屏蔽措施。

（3）双绞线传输。

（4）长线传输的阻抗匹配。

（5）对信号整形。

（6）抑制机械触点，接触器、可控硅的噪声。

（7）提高印刷电路板（PCB）设计中的抗干扰能力。

（8）合理设计地线。

（9）注意各电路之间的电平匹配，总线驱动能力；单片机的空闲端要接地或接电源，或者定义成输出；室外使用的单片机系统或从室外架空引入室内的电源线、信号线，要防止雷击等。

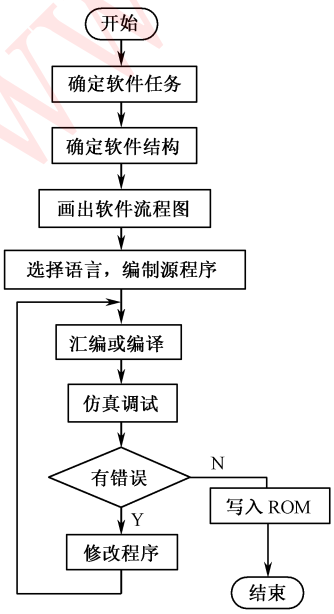


图 7.7　软件设计的流程图

7.7 解：

软件设计随单片机应用系统的不同而不同，一般可分为以下几个方面（图 7.7 画出了单片机软件设计的流程图）。

课后答案网 www.khdaw.com

（1）总体规划
（2）程序设计技术
① 模块程序设计
② 自顶向下的程序设计
（3）程序设计
① 建立数学模型
② 绘制程序流程图
③ 程序的编制
（4）软件装配

7.8 解：

硬件和软件调试完成之后，应进行系统调试。在系统调试时，应将全部硬件电路都接上，应用程序模块、子程序也都组合好，进行全系统软、硬件调试，系统调试的任务是排除软、硬件中的残留错误，使整个系统能够完成预定的工作任务，达到要求的性能指标。

在进行系统调试时，对于有电气控制负载的系统，应先试验空载，空载正常后再试验负载情况。要试验系统的各项功能，避免遗漏。

系统调试成功之后，就可以将程序固化到 ROM 中，程序固化可以在有些仿真系统中进行，最好用专用程序固化器进行固化操作，因为它的功能完善，使用方便、可靠。

将固化好程序的 ROM 插回到应用系统电路板的相应位置，即可脱机运行。系统试运行要连续运行相当长的时间，以考验其稳定性。并要进一步进行修改和完善处理。

一般地，经开发装置调试合格的软、硬件，脱机后应正常运行。但由于开发调试环境与应用系统的实际运行环境不尽相同，也会出现脱机后不能正常运行的情况。当出现脱机运行故障时，应考虑程序固化有无错误；仿真系统与实际系统在运行时，有无某些方面的区别（如驱动能力）；在联机仿真调试时，未涉及的电路部分有无错误。

7.9 解：

设计提示：利用单片机控制函数发生器 ICL8038 芯片，外接少量元器件，制作一台多功能智能函数发生器，要求此函数发生器既能产生各种波形，又能以数字形式显示新产生波形的频率和幅值。

函数发生器一般由以下几部分组成：单片机最小系统，函数发生器 ICL8038，真有效值转换电路，A/D 和 D/A 转换电路，显示、键盘及控制电路等。

ICL8038 是大规模单片函数发生器，只要外接少量元件，就能产生方波、三角波、正弦波等各种波形。

为减少误差，提高函数发生器的准确度，采用真有效值转换电路，以改善 ICL8038 产生的波形畸变。一般采用真有效值转换器芯片 AD636 或 AD537 来实现。

第 8 章

8.1 解：

MCS-96 单片机有以下几种形式：

8096：16 位单片机。由寄存器、算术逻辑单元（RALU）、寄存器阵列、指令寄存器、高速输入/输出单元、串/并行接口、定时器/计数器、PWM 输出、监视定时器、A/D 转换器、控制单元、中断系统、可动态配置的总线、地址译码寄存器等部分组成。

8096BH：与 8096 相比，增加了带有采样和保持电路的 10 位 A/D 转换器，中断源增至 20 个，有专用的波特率发生器，可以动态地重新组合 8 位或 16 位数据总线，EPROM 可在运行时编程并带有各种安全措施，具有灵活可变的 READY 控制等功能。

80C196：功能和指令系统与 8096BH 基本相同，其速度是 8096 的两倍，增加了许多内部 I/O 功能和指令功能，具备 16 位数据总线和 8 位数据总线等功能。

8098：它类似于 8088 微处理器，内部的 CPU 寄存器都为 16 位，对外数据通路为 8 位，这样更便于应用和推广。

80C198：在 8098 基础上，功能与 80C196 相似。

8.2 解：

课后答案网 www.khdaw.com

与 MCS-51 系列单片机相比较，CPU 采用寄存器——寄存器结构，提高了操作速度和数据吞吐能力；具有更快的运算速度，更多的外围子系统，更高效的指令系统，在 89C196KC 以后的芯片中，增加了一个外设事务服务器 PTS，专门用于处理外设中断事务，大大减少了 CPU 的软件开销。

MCS-96 系列单片机的主要特点包括以下几个方面：

① 16 位 CPU，具有高速处理能力，没有累加器，采用寄存器——寄存器结构，具有 232 字节的寄存器阵列；

② 具有高效的指令系统，大大提高了编程效率；

③ 4/8 通道的 10 位 A/D 转换器；

④ 脉宽调制 PWM 输出装置；

⑤ 全双工的串行口，并有专门的波特率发生器；

⑥ 高速的 I/O 系统；

⑦ 5 个 8 位的 I/O 端口；

⑧ 可编程的 8 个优先级中断源；

⑨ 16 位监视定时器；

⑩ 可动态配置的总线；

⑪ ROM/EPROM 的内容可加密；

⑫ 2 个 16 位的定时器/计数器，4 个 16 位的软件定时器。

MCS-96 系列单片机的软、硬件资源远比 MCS-51 的丰富，因而它具有更广阔的应用前景，可以应用于自动控制系统、测试系统、智能仪器、外设控制器、家用电器等。

8.3 解：

MCS-96 单片机由寄存器、算术逻辑单元（RALU）、寄存器阵列、指令寄存器、高速输入/输出单元、串/并行接口、定时器/计数器、PWM 输出、监视定时器、A/D 转换器、控制单元、中断系统、可动态配置的总线、地址译码寄存器等部分组成。

8.4 解：

MCS-96 有 5 个 8 位口，有的只作输入，有的只作输出，也有的是输入/输出双向口且有替换功能。输入口通过输入缓冲器连向内部总线，输出口通过输出缓冲器与内部口寄存器相连，该寄存器用来保持输出口线上的状态，双向口兼有两者的结构。当对双向口进行读取时，数据取自引脚而不是内部寄存器。5 个 8 位口功能如下：

P0 口：仅用作输入，与 A/D 转换器的模拟输入信号复用引脚。

P1 口：准双向 8 位口。

P2 口：多功能双向口。其功能如表 8.1 所示。

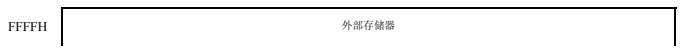
表 8.1　P2 口功能表

口　线	功　能	替 换 功 能	控 制 信 号
P2.0	输出	串行口发送TXD	IOC1.5
P2.1	输入	串行口接收RXD	不受影响
P2.2	输入	外部中断EXTINT	IOC1.1
P2.3	输入	定时器 2 输入T2CLK	IOC0.7
P2.4	输入	定时器 2 复位	IOC0.5
P2.5	输出	PWM输出	IOC1.0
P2.6	准双向口		
P2.7	准双向口		

P3 口与P4 口：P3 口与P4 口具有两种功能。当访问外部存储器时，用作系统的地址/数据分时复用总线。若 $\overline{EA}$ 引脚接高电平，则它们是漏极开路的双向输入/输出口。

8.5 解：

MCS-96 具有一个逻辑上完全统一的存储器空间，寻址范围为 64KB。存储器空间分配如图 8.5 所示。



课后答案网 www.khdaw.com

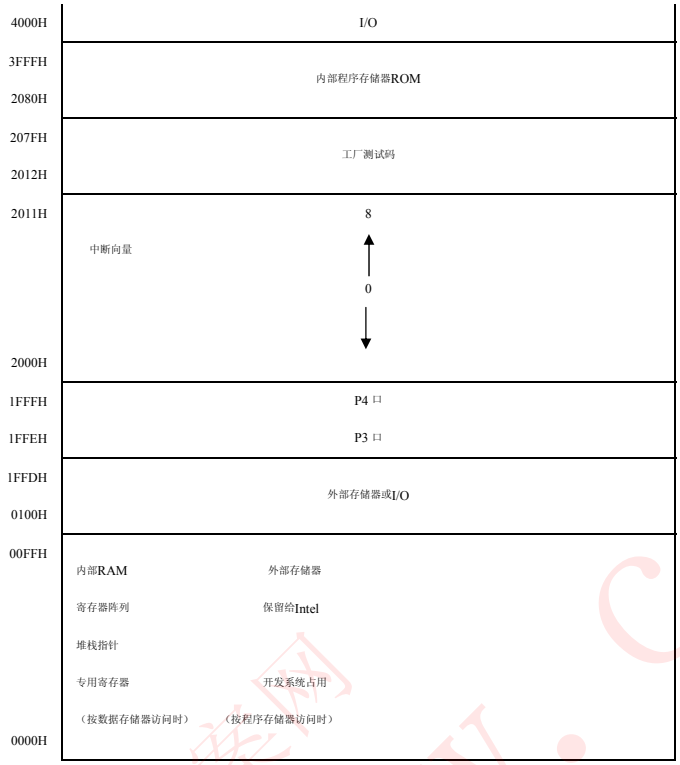


图 8.5　存储器空间分配

8.6 解：

MCS-96 指令系统中具有 6 种基本寻址方式：

（1）寄存器直接寻址

直接访问片内 256 字节的 RAM，一条指令中可以有多至 3 个操作数采用这种寻址方式。

（2）间接寻址

访问整个存储空间，被访问数的地址应放在一个 16 位的寄存器中，一条指令中只能有一个操作数采用间接寻址，其他操作数必须采用寄存器直接寻址。

在这类寻址方式中还包含自动增量间接寻址，但需根据操作数是字节还是字，指令执行后地址寄存器要自动加 1 或 2。

（3）立即寻址

操作数直接放在指令中。一条指令中只能有一个立即数，其他操作数必须采用寄存器直接寻址。

（4）变址寻址

① 短变址寻址：有效地址是由基地址加偏移量组成的，基址寄存器为内部 RAM 中的一个 16 位寄存器，偏移量为一个带符号的 8 位数。可访问整个存储空间。

② 长变址寻址：与短变址类似，只是偏移量是 16 位数。

（5）零寄存器寻址方式

硬件把内部 RAM 的 00H 和 01H 单元固定为零，故称为零寄存器。除了在一些计算和比较运算中用来做常数零外，还可用在长变址寻址中，从而可以直接访问整个存储空间。

（6）堆栈指针寄存器寻址

堆栈指针也可以当做一般的寄存器用，因此可做短变址寻址中的地址寄存器.

8.7 解：

MCS-96 有 2 个中断控制寄存器，它们分别为中断悬挂寄存器和中断屏蔽寄存器，两者的定义是类似的，相应的位对应于同一种中断源，如图 8.7 所示。

课后答案网 www.khdaw.com

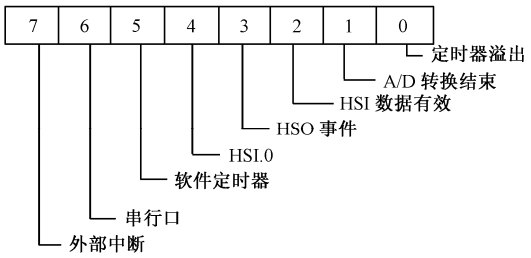


图 8.7　中断悬挂和中段屏蔽寄存器

09H 为中断悬挂寄存器，当 CPU 检测到上述 8 种中断源有由低到高跳变时，就把该寄存器的相应位置位，即把此中断悬挂（悬挂是指有中断申请而尚未得到响应的状态）起来。一旦被悬挂的中断得到响应，相应位被清除。中断悬挂寄存器可以用指令写入，故可以靠对该寄存器的某位置 1 来产生软件中断，也可以靠清除相应位来撤除已悬挂的中断。

08H 为中断屏蔽寄存器，用来控制是否允许某个中断源的中断申请。当某位置 1 时，相应的中断被允许，否则被禁止。对 8 种中断源总的控制要用 EI 和 DI 指令。El 和 DI 指令还可对 PSW.9 置位和复位。8 个中断源是有优先权的，当有多种中断同时挂起时，CPU 先响应优先权最高的一种中断。一旦进入中断服务程序，至少执行一条指令后，才能再响应另一个优先权更高的中断。通常中断服务程序的第一条指令是 DI 或 PUSHF，DI 禁止所有的中断，PUSHF 把 PSW 内容压入堆栈，同时把 PSW 清零。中断屏蔽寄存器是 PSW 的低位字节，因此和 DI 一样，PUSHF 也起到禁止所有中断的作用。如果该服务程序允许被打断，则可对中断屏蔽寄存器置位或用 EI 指令。

中断悬挂寄存器可以用指令写入，即靠对该寄存器的某位置 1 来产生软件中断，也可以靠清除相应位来撤除已悬挂的中断。

中断屏蔽寄存器可以用指令写入，当某位置 1 时，相应的中断被允许，否则被禁止。

8.8 解：

HIS：高速输入单元 HSI 是用来记录外部事件发生时刻及转入相应的处理程序的，它包括 HSI 方式寄存器和 HSI 状态寄存器。

HSO：高速输出部件用来在预定的时间内触发外部事件并申请中断。由 HSO 命令寄存器、HSO 时间寄存器、HSO 保持寄存器、按内容访问的存储器（CAM）阵列、比较器、多路开关、控制逻辑、输出部件和两个定时器组成。

串行口：MCS-96 有一个全双工的串行口，能同时收发信息。与 MCS-51 串行口是兼容的，也有 4 种工作模式（与 MCS-51 串行口类似），通过对串行口控制寄存器的编程设定其工作模式。MCS-96 有一个单独的波特率发生器专供串行口使用。

A/D转换器：MCS-96 系列的部分产品中有一个逐次逼近型的 10 位A/D转换器，共有 8 个通道。当采用 12MHz晶振时，每次转换时间为 42μs。A/D转换器的模拟参考电压为V_{REF}，输入电压应在 0～V_{REF}范围内，正常工作时，V_{REF}必须在V_{CC}±0.3V下保持为 5.0±0.5V。A/D 转换的结果N_{AD}=（2^{10}－1）×（输入电压V_{IN}－ANGND）/（V_{REF}－ANGND）。

PWM：MCS-96 用脉冲宽度调制 PWM 的方法来实现 D/A 转换的功能。其中主要部件有脉宽控制寄存器、比较器和计数器。计数器的计数值为零时，输出高电平。每一状态周期计数值加 1，当计数值与脉宽控制寄存器的值相等时，输出低电平。当计数值溢出再次使计数器复位为零时，输出又变高电平。这样通过设定脉宽寄存器的值可以获得 256 种不同脉宽的波形。

8.9 解：

高速 I/O 部件是由高速输入单元（HSI）、高速输出单元（HSO），一个定时器（定时器 1），一个事件计数器（定时器 2）组成。

高速输入单元 HIS：用来记录外部事件发生时刻及转入相应的处理程序的，它包括 HSI 方式寄存器和 HSI 状态寄存器。

高速输出部件：用来在预定的时间内触发外部事件并申请中断。由 HSO 命令寄存器、HSO 时间寄存器、HSO 保持寄存器、按内容访问的存储器（CAM）阵列、比较器、多路开关、控制逻辑、输出部件和两个定时器组成。

定时器 1：是一个 16 位的计数器，其时钟信号来自内部时钟发生电路，每 8 个状态周

课后答案网 www.khdaw.com

期计数器加 1。当计满时，能触发一个中断，对 I/O 状态寄存器（IOSl）的位 5 置位。作为系统实时时钟的定时器 1，它一直在循环计数，任何时候都可以读它，只有用系统复位操作才能使它停止计数。

定时器 2：定时器 2 也是一个 16 位的计数器，其时钟来自引脚 HSIl/T2CLK，所以实际上是个事件计数器。每当引脚上有正和负跳变时，计数器加 1。当计满时，也触发一个中断，并对 IOS1.4 置位。定时器 2 可以有多种清零方式。

8.10 解：

A/D 转换 8 个通道的模拟电压输入引脚与 P0 口的 8 根口线复用，通过写命令寄存器可以控制通道选择和启动 A/D 转换。每启动一次 A/D 转换都必须对命令寄存器进行一次写操作。用 HSO 触发转换时也不例外。

命令寄存器是双缓冲的，按第一个命令执行的 A/D 转换正在进行时，第二个命令照样可以写入到该寄存器中，而第二个命令必须是靠 HSO 触发去启动转换的。即当一次转换正在进行时，又启动了另一次新的转换，会把正在进行的转换撤销掉，A/D 转换结果寄存器的内容也会被消除掉。因此在新的转换开始前，应把原转换结果读走。

结果寄存器是 16 位的，读出时应按字节分两次读出。

课后答案网 www.khdaw.com