



# 廣東工業大學

## 課程設計

課程名稱 单片机原理及应用技术

題目名稱 单片机课程论文

學生學院                     

專業班級                     

學 號                             

學生姓名                             

指導教師                             

2012 年 07 月 01 日

## 摘要

随着计算机技术的迅速发展，单片微机的应用的不断深入，各种各样的单片机层出不穷，单片机技术的更新换代带来了新的技术革命。《单片机原理及应用技术》这本书系统的讲解了 MCS-51 单片机的原理及应用，从其组成原理、指令系统、汇编语言程序设计、中断、扩展等等直到其应用其实例，由浅到深，让初学者可以更好地接受这些原本陌生的知识，同时也让初学者能较快的上手单片机。我们可中掌握了微型计算机的软硬件知识，完成计算机工程的基础训练，同时可有效结合工程实际需要去初步应用单片机开发技术，并提高综合开发能力。

**关键词：**单片机 组成 指令 中断 扩展 程序设计

## Abstract

With the rapid development of computer technology, the application of the single chip microcomputer unceasingly thorough, all kinds of single chip microcomputer emerge in endlessly, the upgrading of microcontroller technology brings new technology revolution. The single-chip microcomputer principle and application technology of this book system explained MCS-51 single-chip microcomputer principle and application, from its principle, instruction system, assembly language program design, interrupt, expand, and so on until its application in fact an example, from shallow to deep, let a beginner can better accept these originally strange knowledge, also let a beginner can quick started microcontroller. We can master the miniature computer hardware and software knowledge, complete computer engineering of basic training, and at the same time effectively combined with the engineering practice need to preliminary application development of single chip microcomputer technology, and improve the comprehensive developing ability.

**Keywords:** single chip microcomputer, instruction, interruption, program design, expansion, component

# 目 录

摘要.....	1
1 MCS-51 单片机的组成原理.....	3
1.1 MCS-51 单片机的组成.....	3
1.2 MCS-51 单片机的结构.....	3
1.3 MCS-51 单片机的引脚.....	4
2 MCS-51 单片机的指令系统.....	5
2.1 MCS-51 单片机指令系统分类.....	5
2.2 指令格式.....	5
2.3 指令系统.....	6
3 汇编语言程序设计.....	7
3.1 汇编语言程序设计过程.....	7
4 中断系统.....	8
4.1 8051 中断系统结构.....	8
4.2 中断控制寄存器.....	8
4.2.1 中断请求标志.....	8
4.2.2 中断允许控制.....	9
4.2.3 中断优先级控制器 IP.....	9
4.3 中断响应条件及响应过程.....	9
4.3.1 中断响应条件.....	9
4.3.2 中断响应过程.....	9
5 存储器的扩展.....	10
5.1 程序存储器的扩展.....	10
5.2 数据存储器的扩展.....	11
6 单片机应用系统扩展设计.....	12
6.1 存储器的扩展.....	12

6.2 I/O 接口扩展电路设计 .....	13
<b>7 键盘, 显示接口技术</b> .....	<b>15</b>
7.1 键盘与计算机接口 .....	16
7.2 独立式按键接口设计 .....	17
<b>8 测控系统常用元器件</b> .....	<b>18</b>
8.1 集成应用放大器应用要点 .....	18
8.2 集成电压比较器 .....	18
<b>9 A/D 和 D/A 接口技术</b> .....	<b>19</b>
9.1 模拟量输入通道 .....	19
9.2 A/D 转换接口技术 .....	20
9.3 D/A 转换接口技术 .....	20
<b>实验一: 交通灯设计</b> .....	<b>21</b>
<b>实验二: 脉冲计数器</b> .....	<b>22</b>
<b>实验三: 方波发生器</b> .....	<b>24</b>
<b>实验四: 同时在 8051 单片机上扩展 8K EPROM 和 8K SPRAM</b> .....	<b>25</b>
<b>实验五: 单片机与矩阵键盘试验</b> .....	<b>27</b>
<b>实验六: 单片机与 LED 数码管接口技术</b> .....	<b>30</b>
<b>实验七: 单片机与 DAC0808 接口技术</b> .....	<b>31</b>
<b>实验八: 单片机之间的串行口实验</b> .....	<b>34</b>
<b>心得体会</b> .....	<b>37</b>
<b>参考文献</b> .....	<b>38</b>

# 1 MCS-51 单片机的组成原理

## 1.1 MCS-51 单片机的组成

MCS-51 单片机在一块芯片上集成了 CPU、RAM、ROM、定时器/计数器和 I/O 等一台计算机所需要的基本功能部件。其内部包含以下几个部件：

- 1) 8 位 CPU，片内振荡器。
- 2) 4K 字节 ROM，128 字节 RAM。
- 3) 21 个特殊功能寄存器。
- 4) 32 根可编程 I/O 口线。
- 5) 各 64K 的可寻址外部程序存储器、外部数据存储器。
- 6) 2 个 16 位的定时器/计数器。
- 7) 中断结构：5 个中断源、2 个优先级。
- 8) 一个可编程全双工串行口。
- 9) 有位寻址功能，适于布尔处理的位处理机。

其内部组成方框图如下：

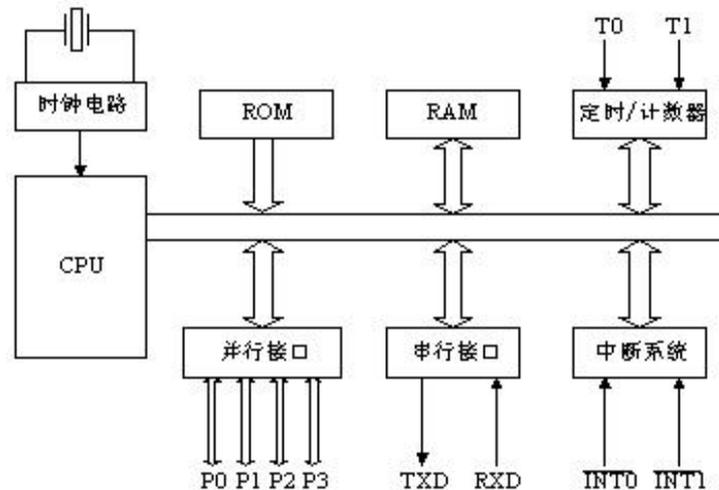


图 1 8051 单片机内部组成方框图

## 1.2 MCS-51 单片机的结构

8051 单片机内部总体结构框图如下：

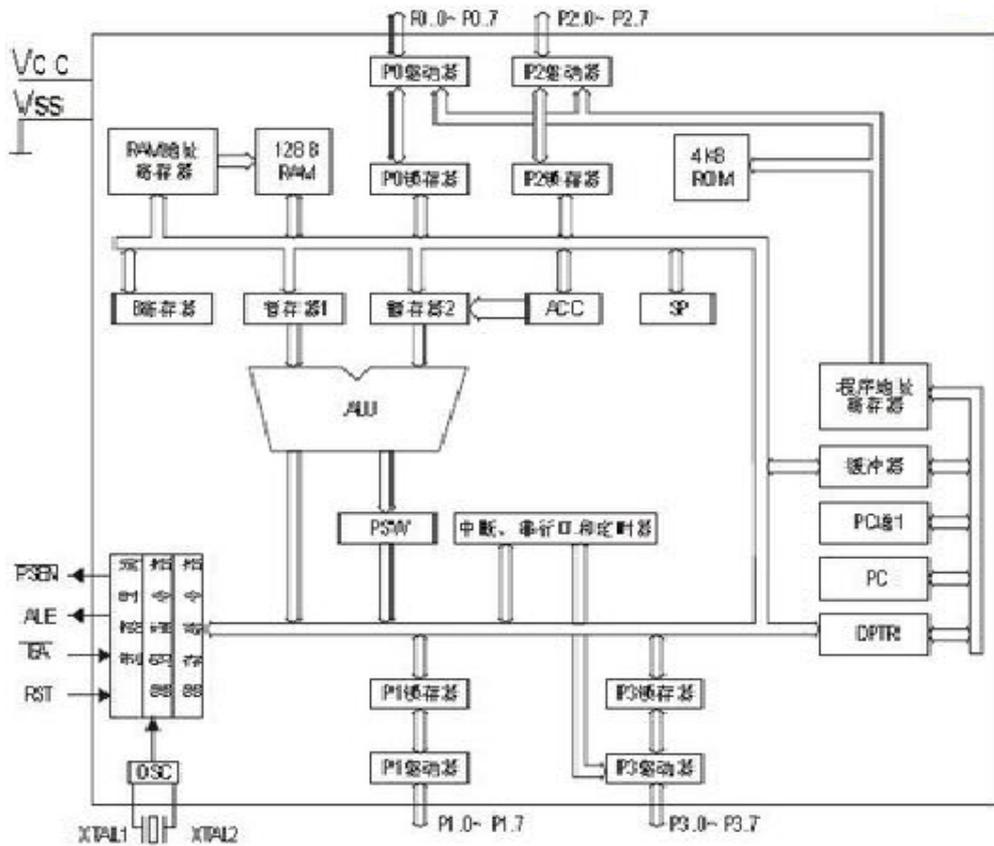


图 2 单片机内部总体结构

### 1.3 MCS-51 单片机的引脚

8051 系列单片机有 40 个引脚，采用双列直插式 (DIP) 封装形式，引脚图如下：

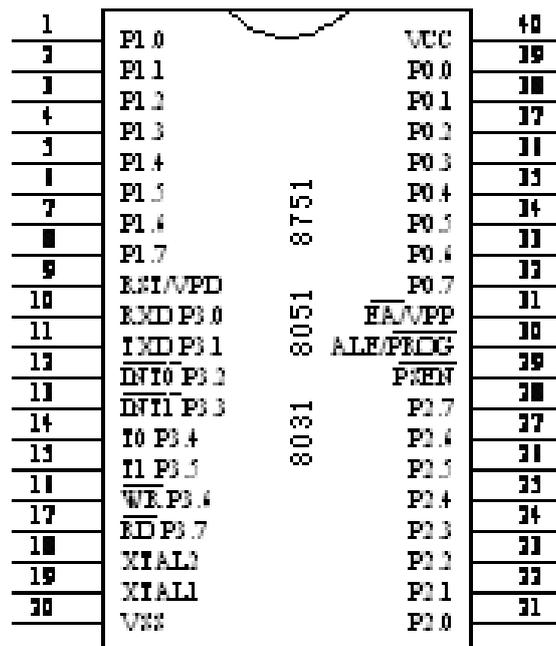


图 3 MCS-51 单片微机引脚图

## 2 MCS-51 单片机的指令系统

### 2.1 MCS-51 单片机指令系统分类

计算机的许多功能都要由程序实现，而程序是由指令编写而成的，熟悉掌握单片机的指令非常重要。

按寻址方式分，可将指令系统分为以下 7 种。

- 1) 立即寻址
- 2) 直接寻址
- 3) 寄存器寻址
- 4) 寄存器间接寻址
- 5) 相对寻址
- 6) 变址寻址
- 7) 位寻址

按功能分，可将指令系统分为以下 4 种。

- 1) 数据传送指令
- 2) 算数、逻辑运算指令
- 3) 位操作指令（布尔指令）
- 4) 控制转移类指令

### 2.2 指令格式

MCS-51 单片机采用汇编语言指令，一条指令最多包括 4 部分：标号、操作码、操作数和注释，机构为：

**标号：操作码 操作数；注释**

要求：

1) 标号位于语句的开始，由字母和数字组成，代表该语句的地址。标号必须由字母开头，冒号结束。字母和数字的总数不超过一定数量，一般标号不能成为助记符。标号不是语句的必要组成部分。

2) 操作码在标号之后，是指令的助记符，表示语句的性质，是语句的核心。没有标号时，它作为语句的开始。

3) 操作数在操作码之后，两者用空格分开。操作数可以是数据，也可以是地址，且必须满足寻址方式的规定。有多个操作数时，操作数之间要用“，”分

开。

4) 注释在语句的最后，以“;”开始，是说明语句的功能和性质的文字。

## 2.3 指令系统

### 2.3.1 数据传送指令

数据传送指令包括数据传送、交换、堆栈数据的压入与弹出，是最基本、使用率最高的一类指令。助记符有 MOV、MOVX、MOVC、XCH、XCHD、SWAP、PUSH、POP 八种，这里就不赘述它们的功能了。

### 2.3.2 算数运算指令

顾名思义，算数运算指令就是实现算数加、减、乘、除等的指令。助记符有 ADD、ADDC、SUBB、MUL、DIV、INC、DEC 和 DA。

### 2.3.3 逻辑运算指令

逻辑运算指令就是实现逻辑运算与、或、非等的指令，。助记符有 ANL、ORL、XRL、RL、RR、RLC、RRC、CPL、CLR。

### 2.3.4 控制转移类指令

因为操作需要，有时候程序不按顺序逐条执行指令，需要改变程序运行方向，即将程序跳转到某个指定的地址再顺序执行下去。这时就需要用到控制转移类指令。常用助记符有 LJMP、AJMP、SJMP、JMP、JZ、JNZ、CJNE、DJNE、LCALL、ACALL、RET、RETI、NOP。

### 2.3.5 位操作指令

又称布尔指令，功能是对内部 RAM 中可进行位操作的区域进行位操作。常用助记符有 MOV、CLR、CPL、SETB、ANL、ORL、JC、JNC、JB、JNB、JBC。

### 2.3.6 伪指令

伪指令是指由汇编程序提供的，在汇编时起作用，在执行时不起作用的一类指令。其特点是没有机器码。常用的有定位伪指令 ORG、汇编结束伪指令 END、定义字节伪指令 DB、定义双字节伪指令 DW、单字节数据赋值伪指令 EQU。

汇编语言的指令有 111 条之多，要记住这 111 条指令有相当大的难度，只能通过不断的记忆和通过实际应用才能很好的掌握。

## 3 汇编语言程序设计

### 3.1 汇编语言程序设计过程

一般情况下，程序设计过程可分为以下几个步骤。

- 1) 分析问题，确定算法或接替思路
- 2) 画流程图
- 3) 编写程序
- 4) 调试和修改

汇编语言程序设计是单片机学习中最复杂的部分，我们必须在熟练掌握各条指令的功能的基础上，通过各种实际程序，慢慢熟悉汇编语言的编写，需要耐心。

## 4 中断系统

### 4.1 8051 中断系统结构

8051 单片机中断系统图如下：

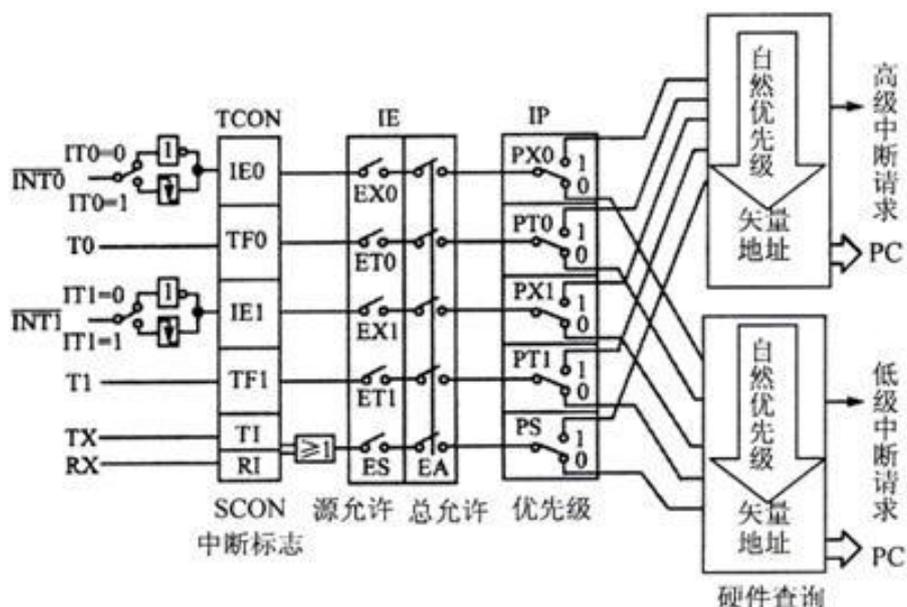


图 4 8051 单片机中断系统结构图

设置中断的原因：由于 CPU 的速度要比外设的速度快得多，在 CPU 与外设减缓信息的时候，若采用查询方式则 CPU 要浪费很多时间等待外设，中断就是为了解决这个矛盾而提出来的。

中断可以实现以下功能：

- 1) 同步工作
- 2) 实时处理
- 3) 故障处理

### 4.2 中断控制寄存器

#### 4.2.1 中断请求标志

##### 1) TCON 的中断标志位

TCON 为定时器/计数器 T0、T1 的控制寄存器，同时也锁存了 T0、T1 的溢出中断源和外部中断请求源，与中断相关的位如下：

TCON	8FH	8DH	8BH	8AH	89H	88H
(88H)	TF1	TF0	IE1	IT1	IE0	IT0

其中各位的功能就不再赘述。

#### 2) SCON 的中断标志位

SCON 为串行口控制寄存器，其低二位 T1 和 R1 为锁存串行口的接收中断和发送中断标志。格式如下：

SCON						99H	98H
(98H)						T1	R1

#### 4.2.2 中断允许控制

8051 对中断源的开放或屏蔽是通过中断允许寄存器 IE 控制的。IE 的格式如下：

IE	AFH		ACH	ABH	AAH	A9H	A8H
(A8H)	EA		ES	ET1	EX1	ET0	EX0

#### 4.2.3 中断优先级控制器 IP

8051 单片机有两个中断优先级，对于每一个中断请求源可编程为高级优先级中断或低级优先级中断，实现二级中断嵌套。8051 内还有一个优先级寄存器 IP，只要用程序改变其内容，就可以对各中断级别进行设置。格式如下：

IP				BCH	BBH	BAH	B9H	B8H
(B8H)				PS	PT1	PX1	PT0	PX0

### 4.3 中断响应条件及响应过程

#### 4.3.1 中断响应条件

基本条件：

- 1) 中断源发出请求，即把中断标志寄存器 TCON、SCON 相应位置 1
- 2) 中断允许 EA=1，CPU 允许响应中断申请
- 3) IE 寄存器中断允许位置 1，允许该中断源发出中断申请
- 4) 没有更高一级中断请求

#### 4.3.2 响应过程

满足中断响应条件后，CPU 就响应中断。

## 5 存储器的扩展

### 5.1 程序存储器的扩展

MCS-51 单片机程序存储器的寻址空间为 64KB，对于 8051 片内程序存储器为 4KB 的 ROM 或 EPROM，在单片机的应用系统中，内存的存储容量往往不够，必须外扩程序存储器。8051 外扩程序存储器结构图如下：

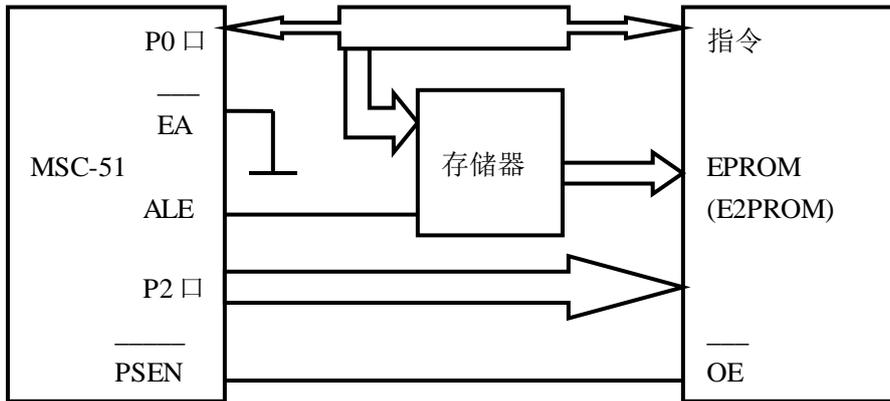


图 5 MCS-51 外扩程序存储器结构图

例如 8031 单片机扩展一片 4KB 的 2732 程序存储器的电路图如下：

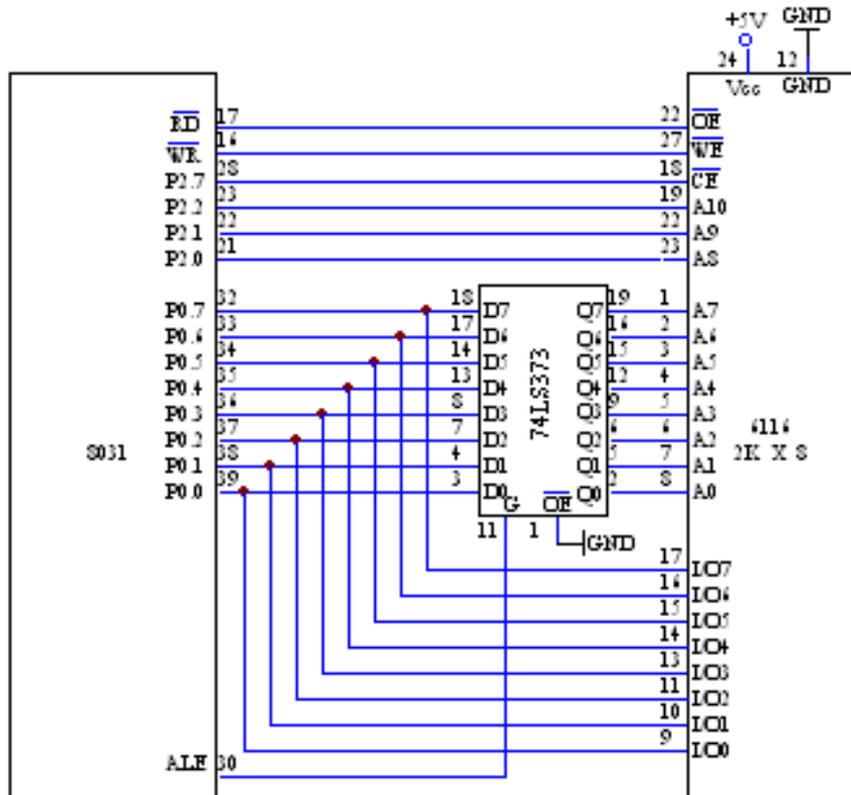


图 6 8031 扩展一片 2732 的电路图

## 5.2 数据存储器的扩展

8051 单片机有 128B RAM，当数据量超过 128B 时，也需要把数据存储区进一步扩展。常用 RAM 芯片分静态和动态两种。静态 RAM 有 6116 (2KB)、6264 (8KB) 等，动态的有 DRAM2164 (8KB) 等，另外还有集成 IRAM 和 E2PROM。使用 E2PROM 作数据寻初期有断电保护数据的优点。

# 6 单片机应该系统扩展设计

## 6.1 存储器的扩展

### 6.1.1、程序存储器的扩展及取指过程

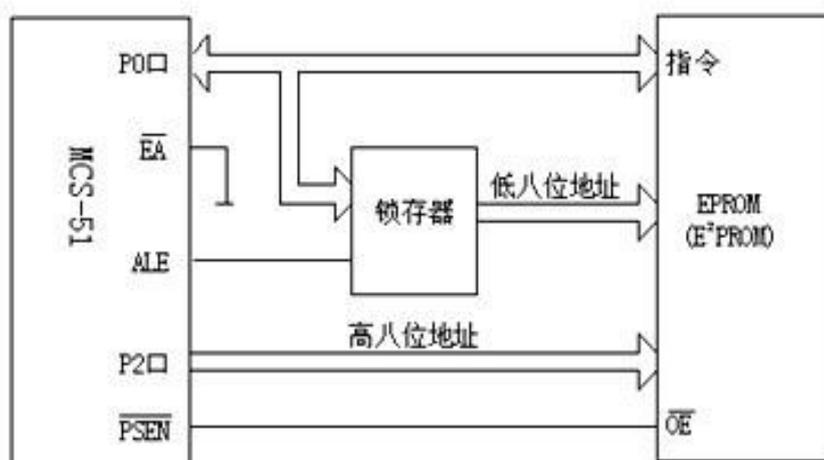


图 6-1 MCS-51 外扩程序存储器结构图

### 6.1.2、数据存储器的扩展

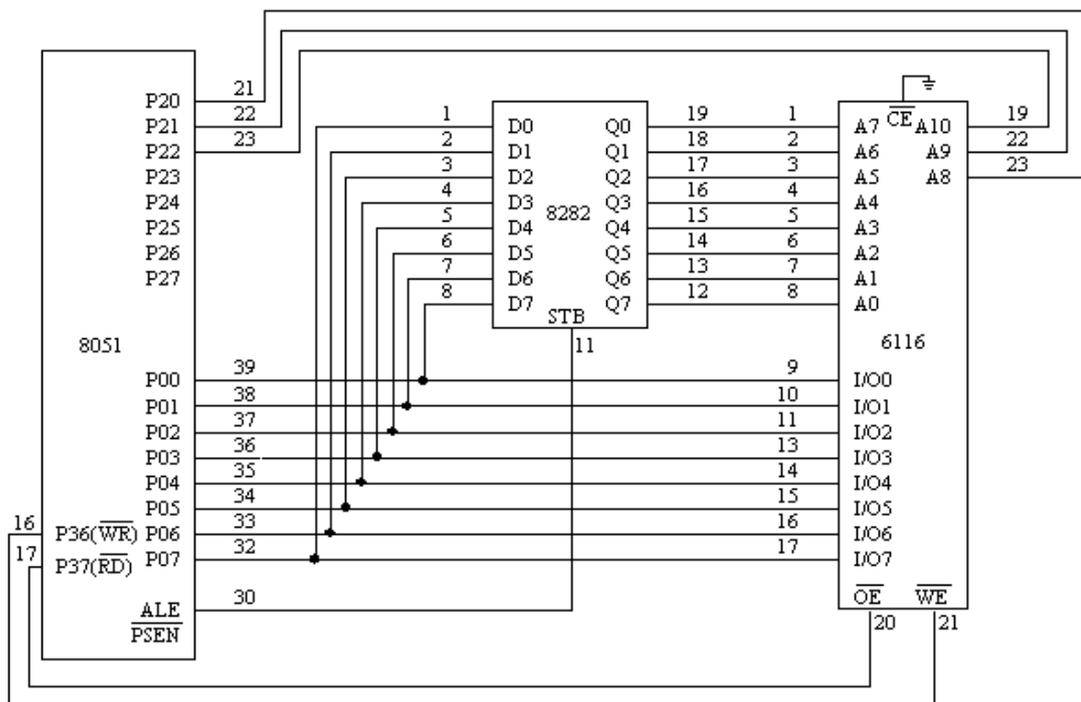


图 6-2 8051 扩展一片 6116 电路图

## 6.2 I/O 接口扩展电路设计

### 6.2.1、简单 I/O 口的扩展

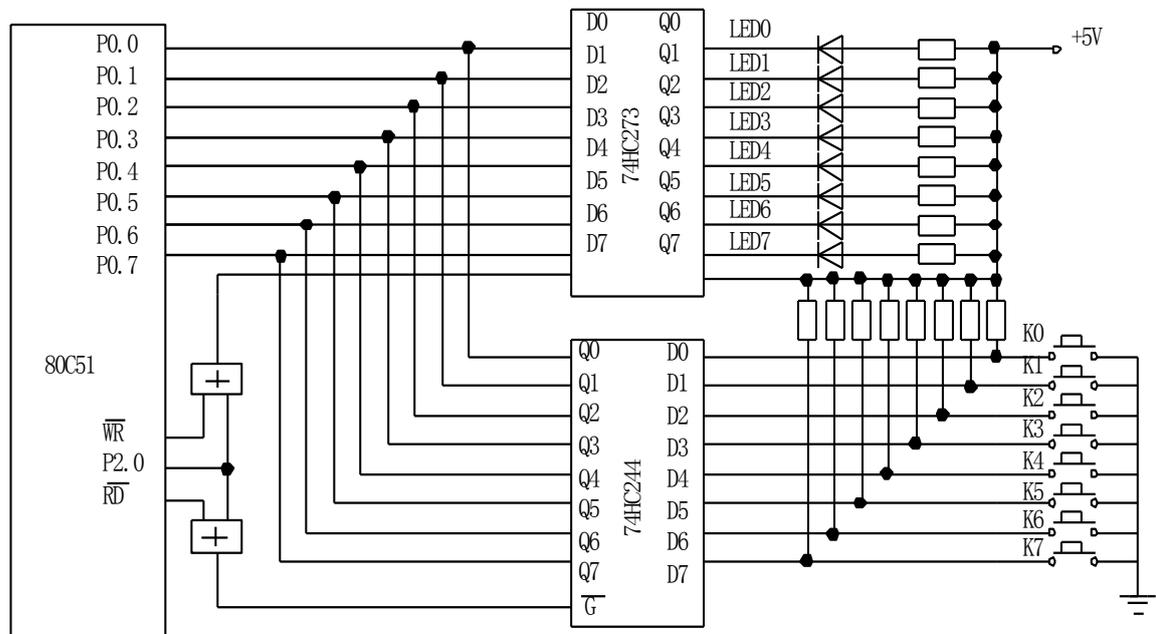
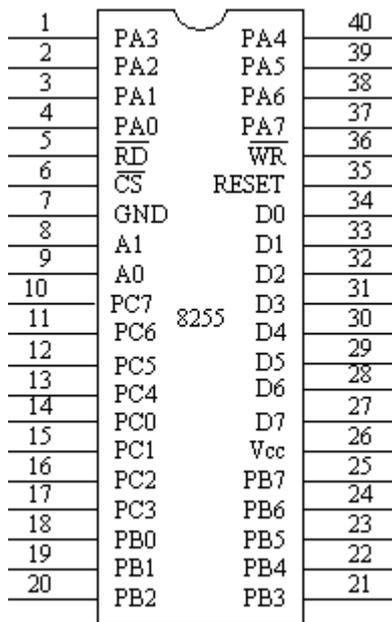
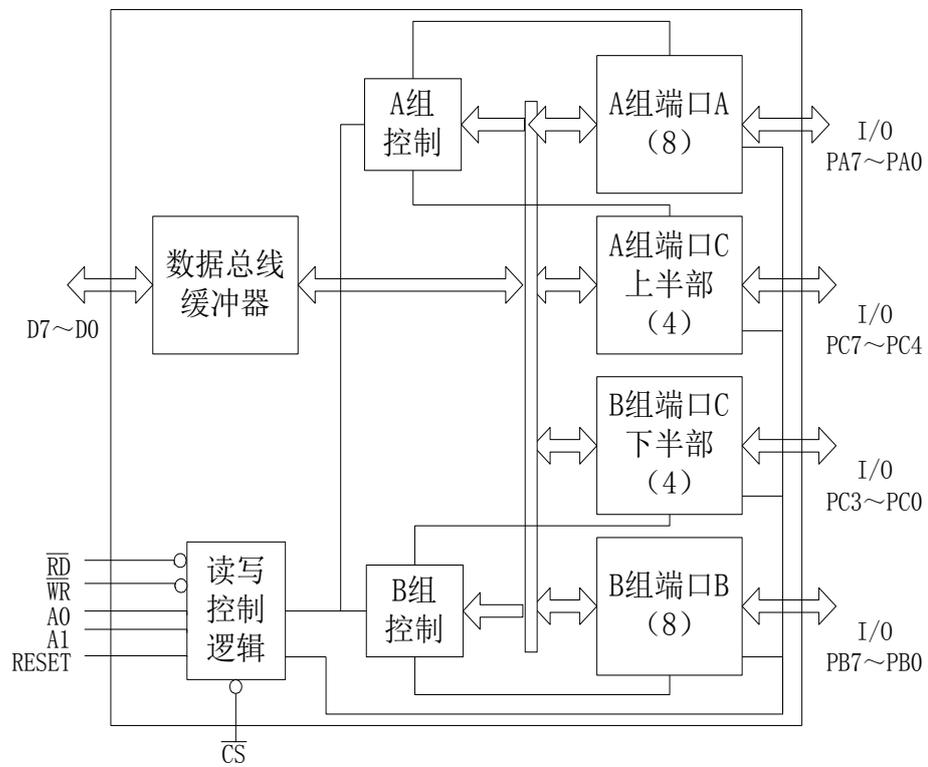


图 6-8 简单 I/O 接口扩展电路

### 6.2.2 8255 可编程并行 I/O 接口扩展与电路设计



(a) 引脚定义



(b) 内部功能结构

## 第 7 章 键盘，显示接口技术

### 7.1 键盘、显示与计算机接口

键盘输入数据和命令，显示用于计算机的状态、命令和计算结果。

#### 一、键盘

单片机键盘有两种：

一种是全编码键盘，其键码全由硬件提供，但是这种方式硬件结构复杂，成本高；另一种是非编码键盘，这种键盘多采用矩阵方式，利用软件识别键码及完成各种键功能处理。

单片机系统中多采用非编码键盘。

#### 二、显示器

##### 1.LED 显示器

内部由发光二极管组成段显示。

数码管结构又分为共阳极型和共阴极型。

##### 2. LCD 液晶显示器

常用的 LCD 可分为字符型和点阵型两大类。

字符型可用来显示字符和数字，

点阵型可用来显示汉字及图形。

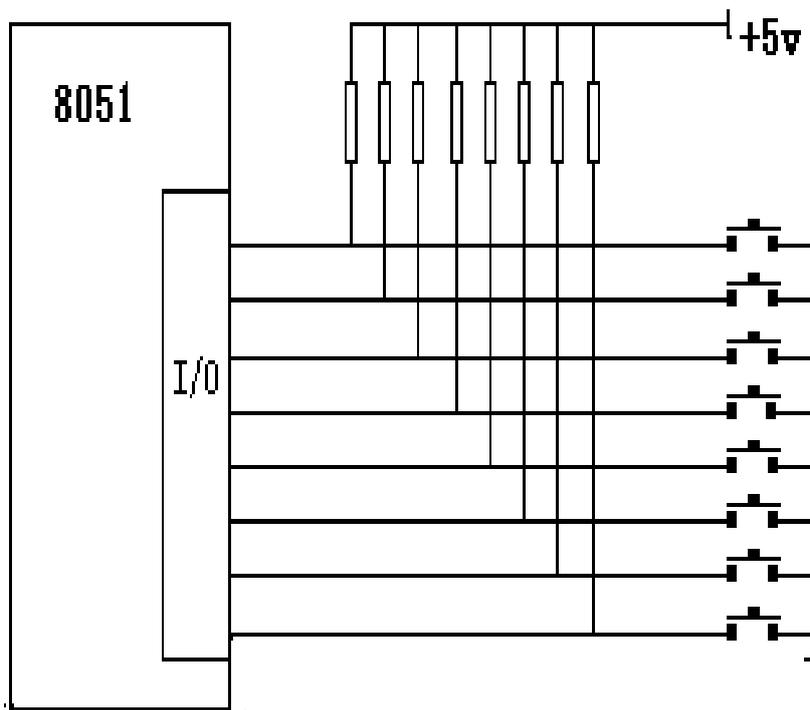


图 7-1 独立连接式非编码键盘

## 7.2 独立式按键接口程序设计

### 键盘处理程序任务

#### 1) 键输入

检查键盘是否有键被按下，消除按键抖动。确定被按键的键号，获取键号。

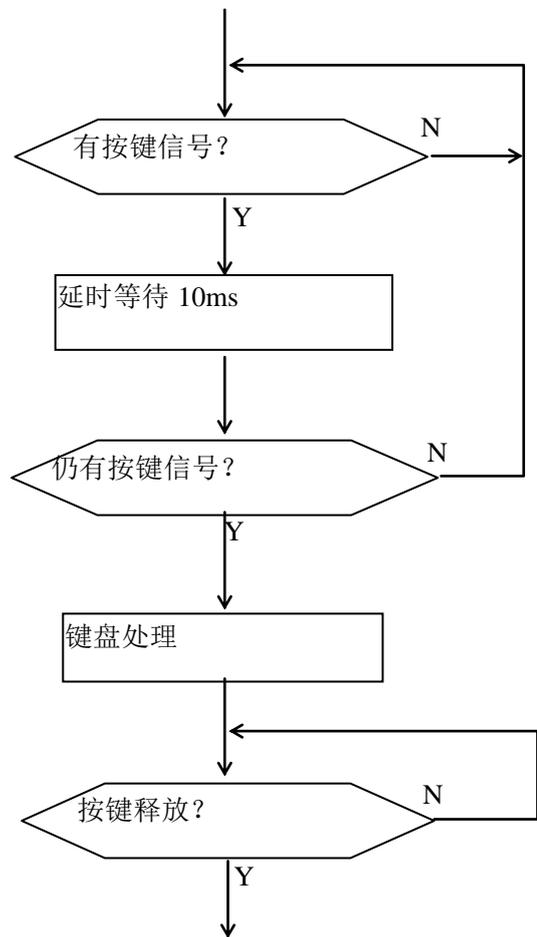
硬件电路消除抖动或软件消除抖动。

#### 2) 键译码

键号为键盘位置码，根据键号查表得出被按键的键值。键值：数字键 0~9、字符键 0AH~0FH、功能键 10H~。

#### 3) 键处理

根据键值转移到不同程序段。若键值属于数字、字符键，则调用显示数字和字符的子程序。若键值属于功能键，则进行多分支转移，执行各个功能程序段。



## 7.3 矩阵式键盘工作方式

### 1. 扫描法

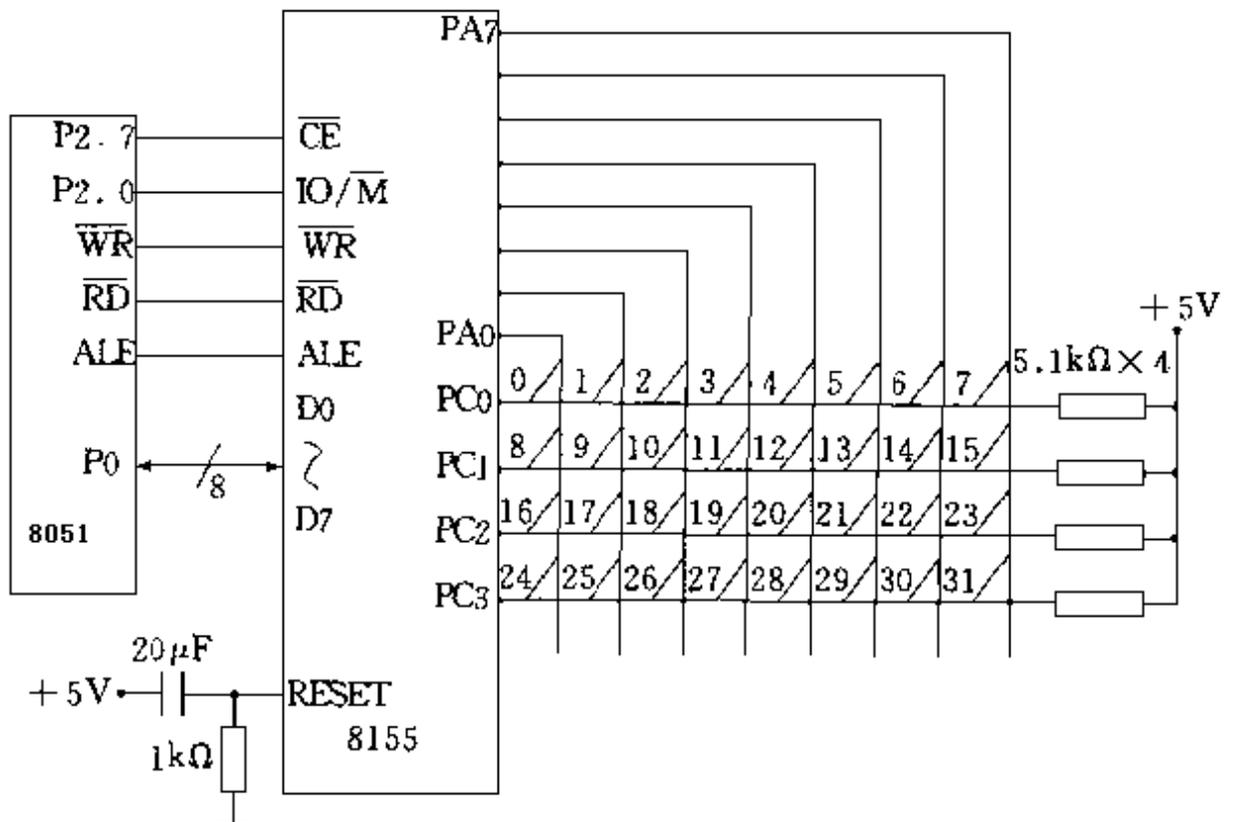


图 7-3 矩阵式键盘电路

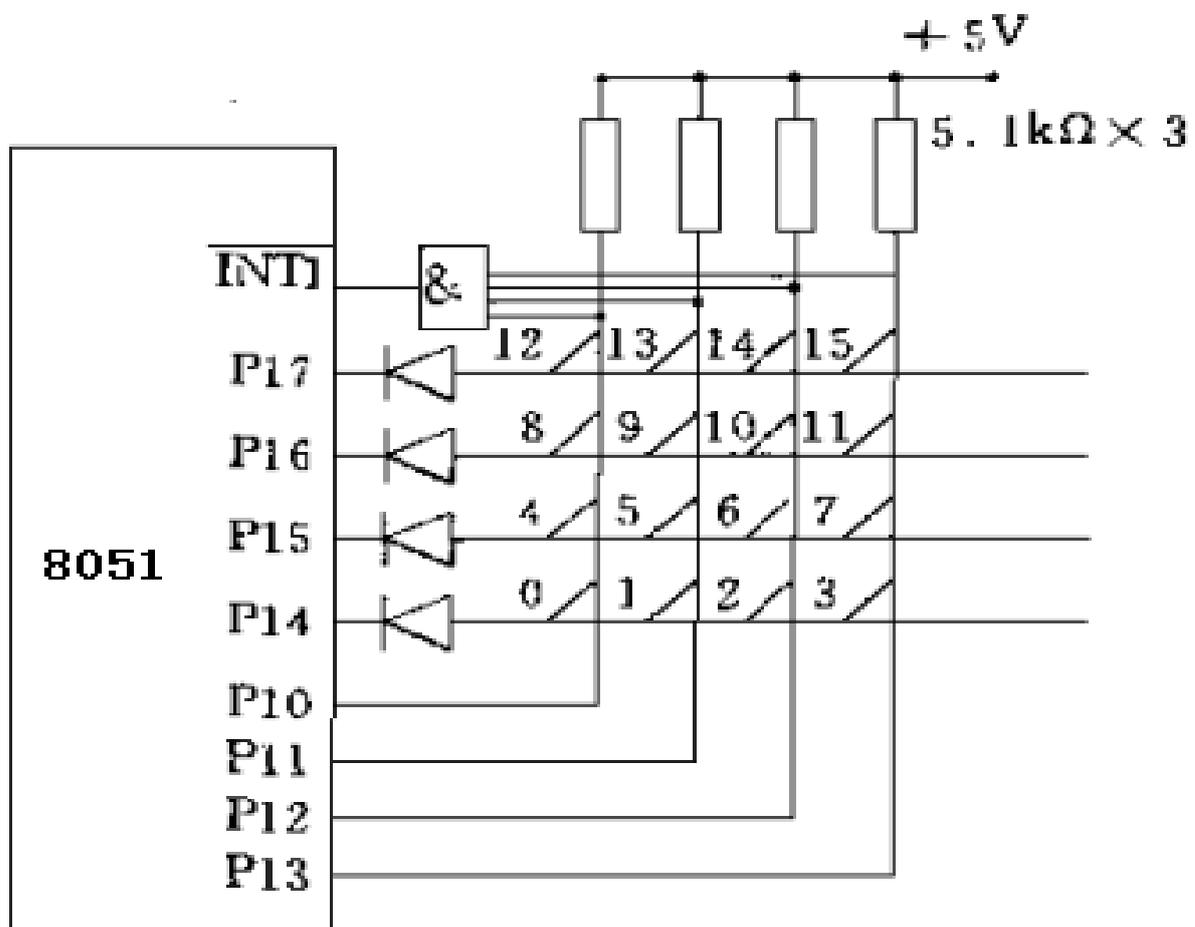


图 7-5 中断式键盘电路

## 第 8 章 测控系统常用元器件

### 8.1 集成运算放大器应用要点

集成运放的应用一般是稳态应用，包括放大或者处理连续正弦波形成的复杂波形，这时应考虑如下两个方面：1 反馈 2. 闭环增益和带宽

#### 1. CF741 通用单运放

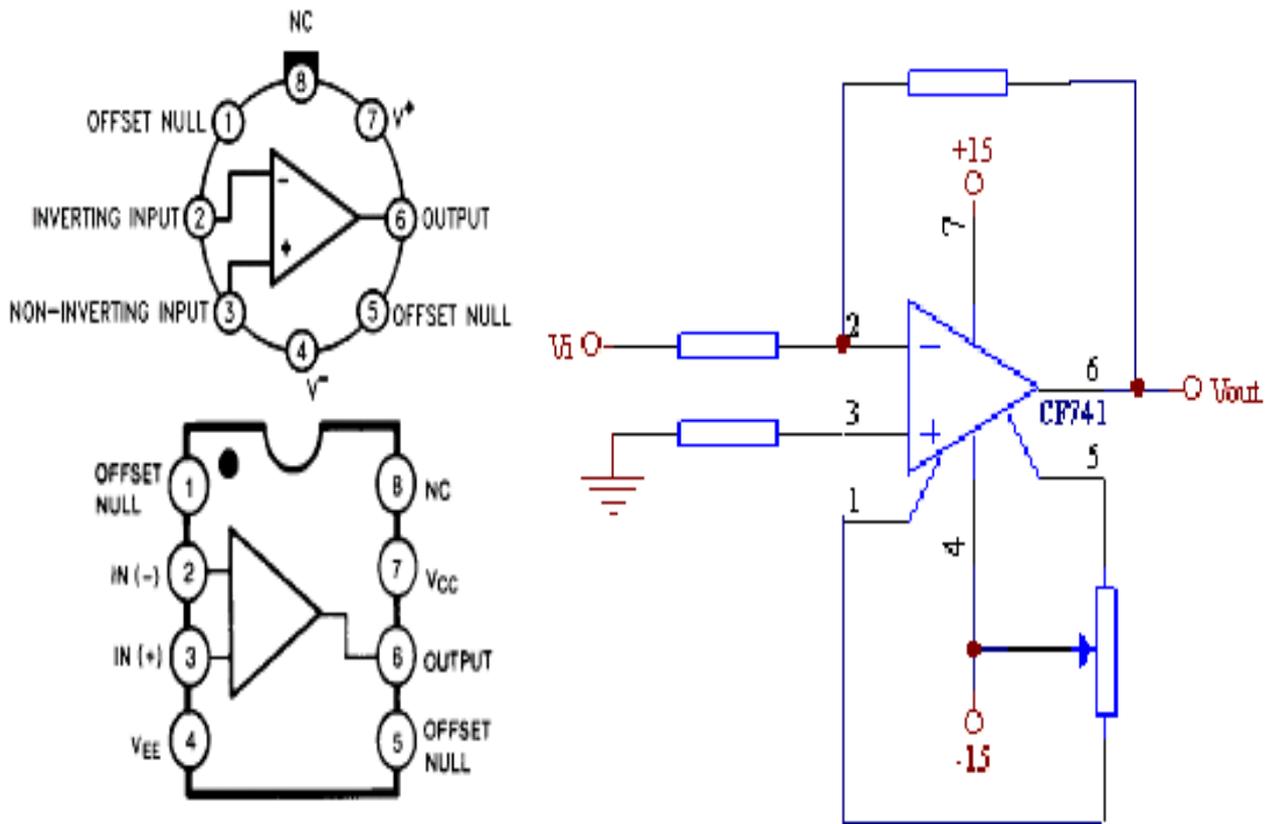


图 8-1 CF741 封装引脚

## 第9章 A/D 和 D/A 接口技术

### 9.1 模拟量输入通道

#### 一、模拟量输入通道的构成特点

##### 1. 测控系统依据采集信号

模拟量输入——采集被测对象信号。

传感器输出电信号——计算机的 TTL 电信号。

##### 2. 传感器模拟信号

输出信号较弱，应用模拟、数字混杂电路，考虑边界区域、线性区域、分辨率等。

##### 3. 模拟量输入通道靠近现场，易受干扰。由于传感器输出信号一般都比较微弱。

### 9.2 A/D 转换器接口

#### 一 A/D 转换器主要有三种类型:

##### 1. 双积分式 A/D 转换器

抗干扰能力强、转换精度高、价格便宜等优点，缺点是转换速度 较慢。

##### 2. 逐次逼近式 A/D 转换器

转换速度较快，其转换时间在几微秒至几百微秒之间，目前品种 最多、应用最广的 A/D 转换器件。

##### 3. 并行式 A/D 转换器

#### 二 D/A 转换器的主要技术指标

1 分辨率 2 转换时间

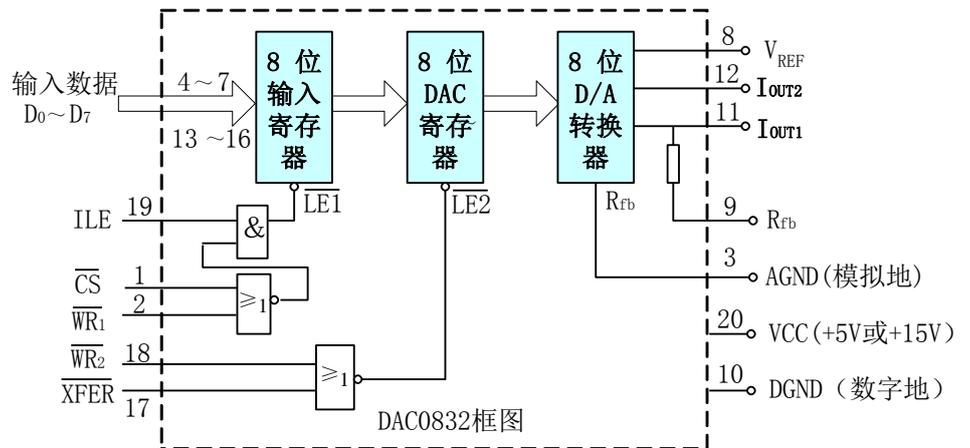
#### 三 典型 D/A 转换器

##### 1 DAC0832 特性: a 8 位电流输出型 D/A 转换器;

bT 型电阻网络;

c 差动输出;

##### 2 内部结构



## 9. D/A 转换器的应用

### 一 应用

#### 1 函数发生器

a 只要往 D/A 转换器写入按规律变化的数据，即可在输出端获得正弦波、三角波、锯齿波、方波、阶梯波、梯形波等函数波形。

#### 2 直流电机的转速控制

a 用不同的数值产生不同的电压，控制电机的转速

b 其他需要用电压/电流来进行控制的场合

### 二 D/A 转换器与单片机的接口

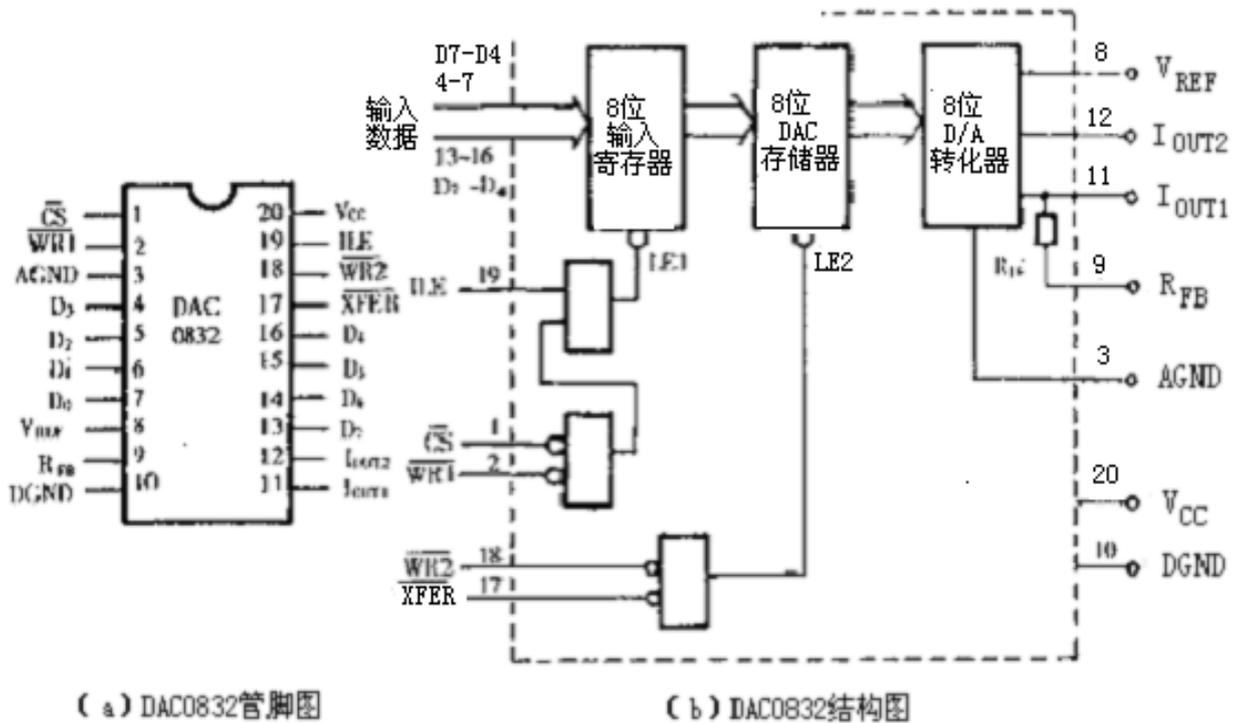


图 9-15 DAC0832引脚及结构图

### 三 D/A 转换的典型应用举例

- (1) 波形发生器
- (2) 程控界限检测器
- (3) 程控稳压源
- (4) 程控恒流源

## 实验一：交通灯的设计

实验目的：

- 1、熟悉 P1 口的使用方法。
- 2、熟悉延时程序的编写

实验要求：在一个十字路口分为东西南北走向，信号等按以下的状态顺序工作：

- (1) 状态 1，南北绿灯亮通车，东西红灯，延时一定时间；
- (2) 状态 2，南北红灯亮，东西绿灯通车
- (3) 循环至状态 1，继续

程序：

```
ORG 0           ;程序初始地址
AJMP STAR      ;跳转到 STAR
ORG 40H        ;STAR 程序的开始地址
STAR:          ;开始
MOV R3,#5AH    ;R3 寄存器赋值
ST:
MOV P1, R3     ;定义 P1 口，初始灯亮的情况 (2457 位=1)
MOV R1, #06H   ;数码管十位的初值 (6)
MOV R0, #00H   ;数码管个位的初值 (0)
MOV DPTR, #TEL ;定义指针 DPTR 的初地址

LOP:  MOV A, R1      ;把 R1 的内容赋值给累加器 A
      MOVC A, @A+DPTR ;取段选
      MOV P2, A     ;数码管十位显示数值
      MOV A, R0     ;把 R0 的内容赋值给累加器 A
      MOVC A, @A+DPTR ;取段选
      MOV P3, A     ;数码管个位显示数值

LOOP:  MOV R7, #8    ;
LOOP1: MOV R6, #50   ;
LOOP2: MOV R5, #100  ; 1s 延时
LOOP3: DJNZ R5, LOOP3 ;
      DJNZ R6, LOOP2 ;
      DJNZ R7, LOOP1 ;

      CJNE R0, #0, TEX ;判断显示管个位数值是否为 0
      CJNE R1, #0, NT  ;判断显示管十位数值是否为 0
      SJMP FC          ;个位十位同时为 0 则跳转 FC
NI:
      DEC R1          ;NT 个位是 0 时，十位减 1
      MOV R0, #9     ;个位跳到数值为 9 的段选地址
```

```

SJMP LOP ;重新显示新赋值(9)的个位

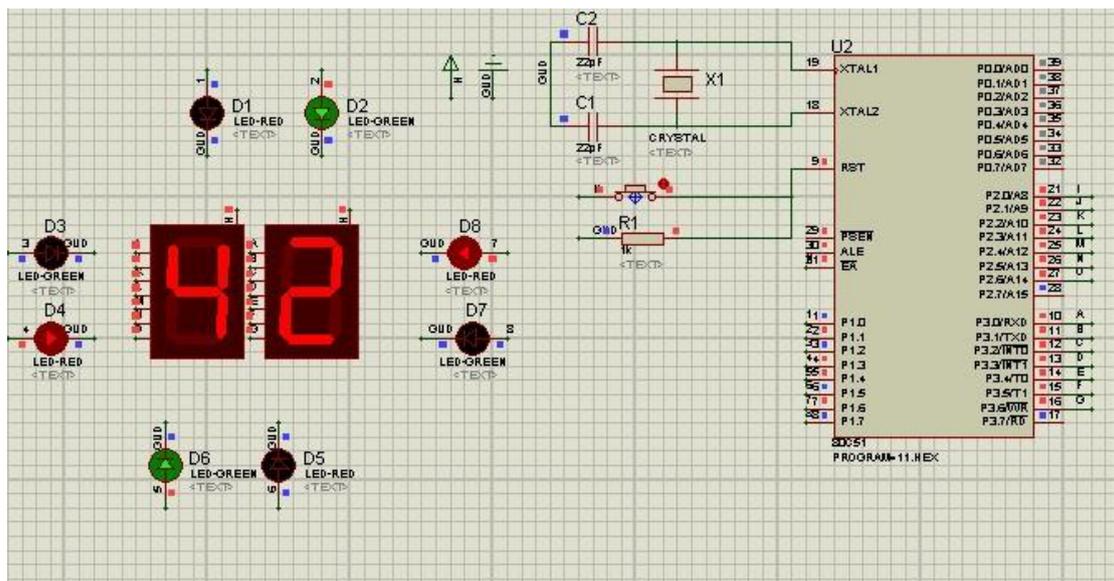
TEX:
DEC R0 ;个位不是0时,个位减1
SJMP LOP ;子程序复位

FC:
CJNE R3,#5AH,FU ;个位十位同时是0,判断P1端灯亮情况
MOV R3,#0A5H ;#
SJMP ST ;#改变P1端为另一种情况(只有2种情况)
FU: MOV R3,#5AH ;#
SJMP ST ;改变灯亮情况后,程序复位

TEL: DB 40H,0F9H,24H,30H,19H
DB 12H,02H,0F8H,00H,10H ;数字0-9的段选
END ;结束

```

仿 真 图 :



## 实验二、脉冲计数器

### 试验目的:

1. AT89C51 有两个定时/计数器，本实验中，定时/计数器 1 (T1) 作定时器用，定时 1s；定时/计数器 0 (T0) 作计数器用。被计数的外部输入脉冲信号从单片机的 P3.4(T0) 接入，单片机将在 1s 内对脉冲计数并送四位数数码管实时显示，最大计数值为 0FFFFH。
2. 用 proteus 设计、仿真基于 AT89C51 单片机的脉冲计数器。
3. 学会使用 VSM 虚拟计数/计时器。

```

程序:          ORG 0

                SJMP STAR
                ORG 1BH
                SJMP T1S
STAR:  MOV TMOD,#015H   ;初始化 TMOD,T1 定时, T0 计数
        MOV R7,#20     ;计数器赋初值
        MOV TH0,#0     ;T0 清零
        MOV TL0,#0
        MOV TH1,#3cH   ;T1 赋初值
        MOV TL1,#0b0H
        SETB PT1       ;初始化 TCON,T1 优先级最高
        MOV IE,#8DH   ;设置 IE,开中断允许
        MOV P2,#0
        MOV P1,#20
        SETB TR0       ;启动 T0、T1
        SETB TR1
        ;MOV R7,#20
ST1:   MOV P1,TL0
        SJMP ST1
T1S:   MOV TH1,#3CH   ;重新赋值
        MOV TL1,#0B0H
        DJNZ R7,T1S0  ;R7 减 1, 不为 0 则转移到 T1S0
        CLR TR0       ;关 T0
        MOV P2,TH0   ;计数器 T0 值送 P2P1 显示
        MOV P1,TL0
        CLR TR1       ;关 T1
T1S0:  RETI
        END

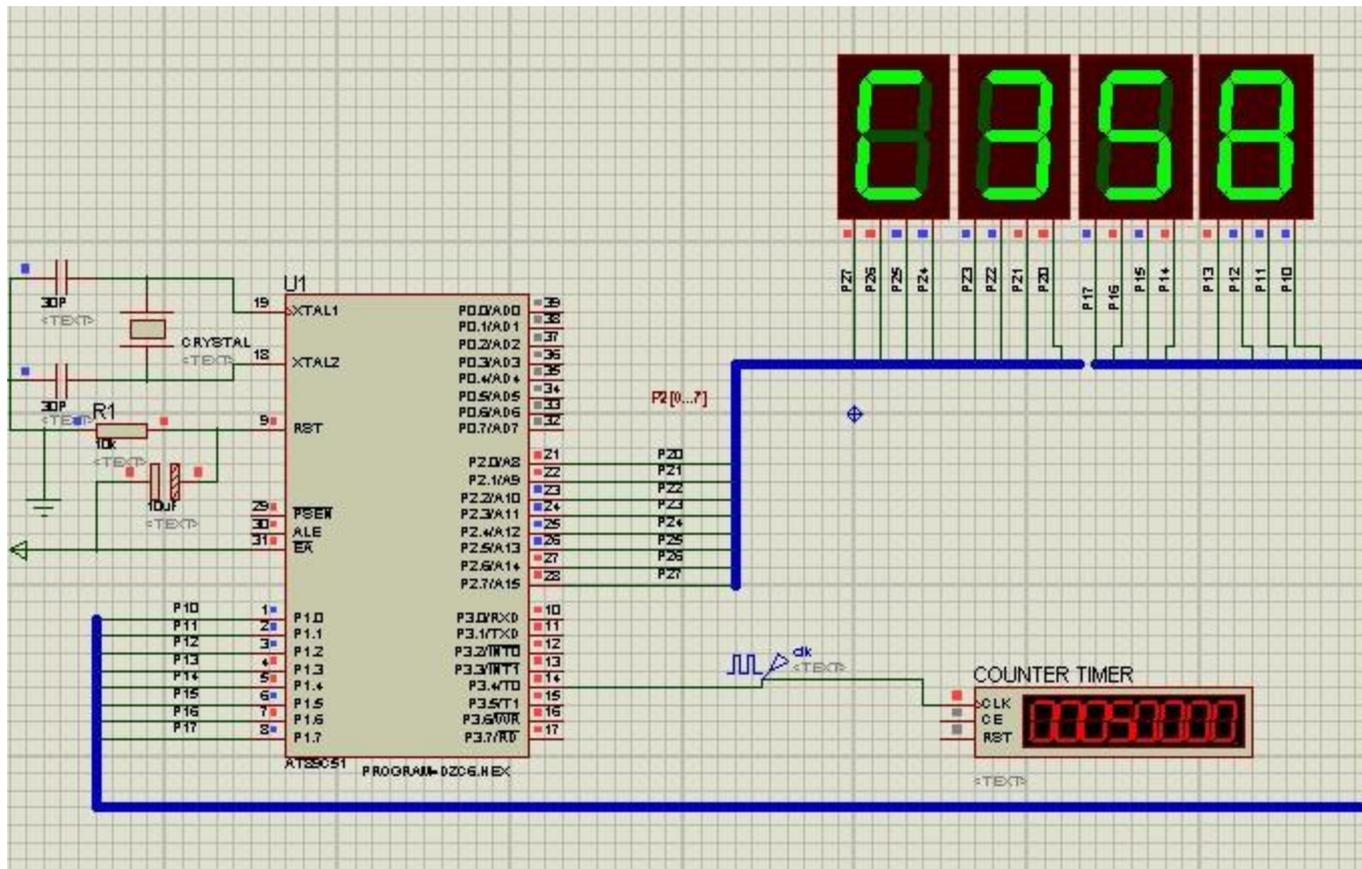
```

仿

真

图

:



### 实验三：方波发生器

实验目的：

1. 定时/计数器是单片机的一个重要的功能部分，可用来实现定时、计数、频率测量、脉冲宽度测量、产生信号、信号检测等。用 AT89C51 单片机定时/计数器 0 的定时功能可构成一简单的方波发生器，实现周期为  $400\ \mu\text{s}$  的方波输出。若改变定时/计数器 0 的初值可得到不同周期的方波输出。
2. 用 PROTEUS 设计、仿真基于 AT89C51 单片机的方波发生器。
3. 学会使用 VSM 虚拟示波器观察波形。

程序：           ORG 0

                  SJMP STAR

                  ORG 0BH

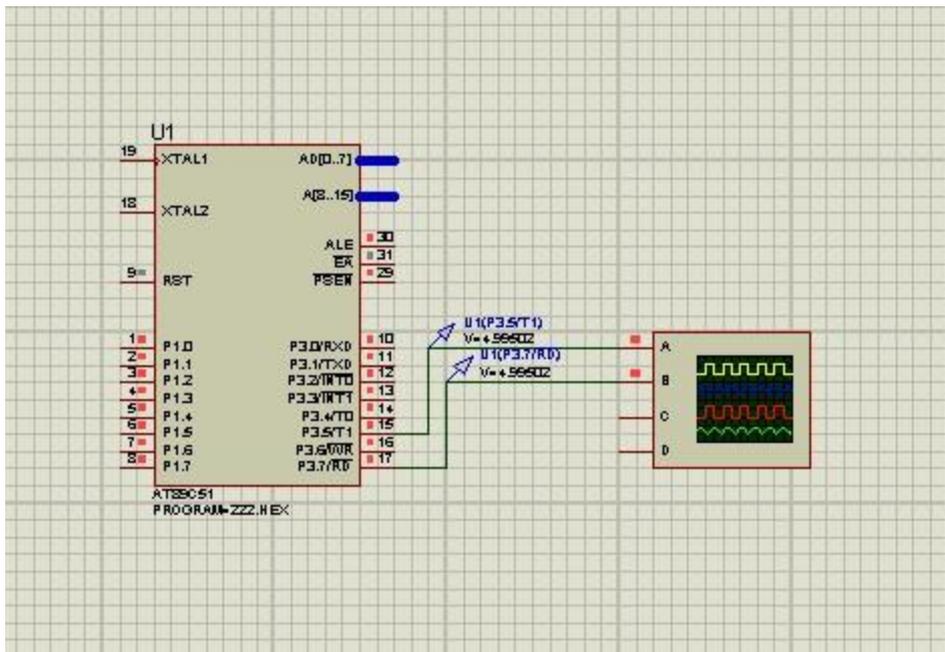
```

    SJMP TOF
STAR: MOV TMOD, #0
      MOV P3, #03FH
      MOV TH0, #0F9H
      MOV TL0, #18H
      MOV IE, #0FFH
      SETB TR0
      SJMP $
TOF: CPL P3.5
     CPL P3.7
     MOV TH0, #0F9H
     MOV TL0, #18H

     RETI
     END

```

仿真图：



#### 实验四：同时在 8051 单片机上扩展 8K EPROM 和 8K SPRAM

实验目的：

- 1、 熟悉程序存储器和数据存储器与单片机扩展的连接方式
- 2、 熟悉用线选法选择某一扩展芯片

### 3、 熟悉用分析仪去分析数据

#### ROM 程序:

```
MAIN:  MOV R2, #0
MOV A, R2
STAR1: MOV P1, A
      INC R2
      MOV A, R2
      ANL A, #0FH
      LCALL DLY
      CJNE A, #10H, STAR1
      SJMP STAR2
STAR2: CLR A
      MOV DPTR, #1456H;
      MOV A, #55H;
      MOVX @DPTR, A      ;对外部 RAM 1456H 单元写入数据 55H
      CLR A;
      MOVX A, @DPTR      ;对外部 RAM 1456H 读出数据到 A
      NOP
      MOV A, #98H;
      INC DPL
      MOVX @DPTR, A      ;将#98H 写入外部 RAM 1457H
      NOP
      MOVX A, @DPTR      ;从外部 RAM 1457H 读出数据到 A
      NOP
      CLR A
      SJMP STAR1
DLY:  MOV R1, #2
D1:  MOV R3, 255
      DJNZ R3, $
      DJNZ R1, D1
      RET
      END
```

#### RAM 程序:

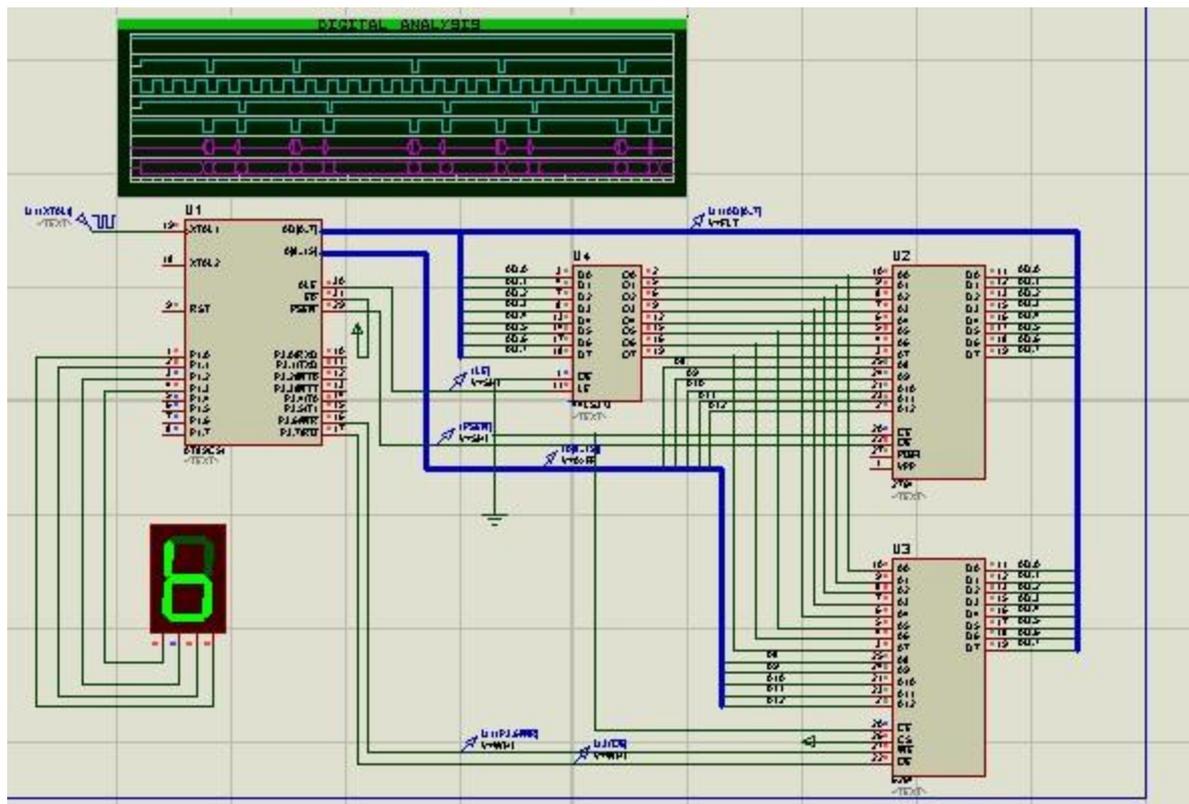
```
ORG 0
SJMP STAR
ORG 10H
STAR:MOV DPTR, #1456H;
      MOV A, #55H;
      MOVX @DPTR, A; ;对外部 RAM 1456H 单元写入数据 55H
      CLR A;
      MOVX A, @DPTR; ;对外部 RAM 1456H 读出数据到 A
      NOP
```

```

MOV A, #98H;
INC DPL
MOVX @DPTR, A; ;将#98H 写入外部 RAM 1457H
NOP
MOVX A, @DPTR; ;从外部 RAM 1457H 读出数据到 A
NOP
SJMP STAR
END

```

仿真图：



## 实验五：单片机与矩阵键盘试验

实验目的：

1. 理解 AT89C51 单片机对 4\*4 矩阵键盘进行动态扫描,当按键按下时,相应的按键值(0~F)将实时显示在数码管上。
2. 用 PROTEUS 设计设计键盘接口的电路设计和程序设计,并进行实时仿真。

程序:

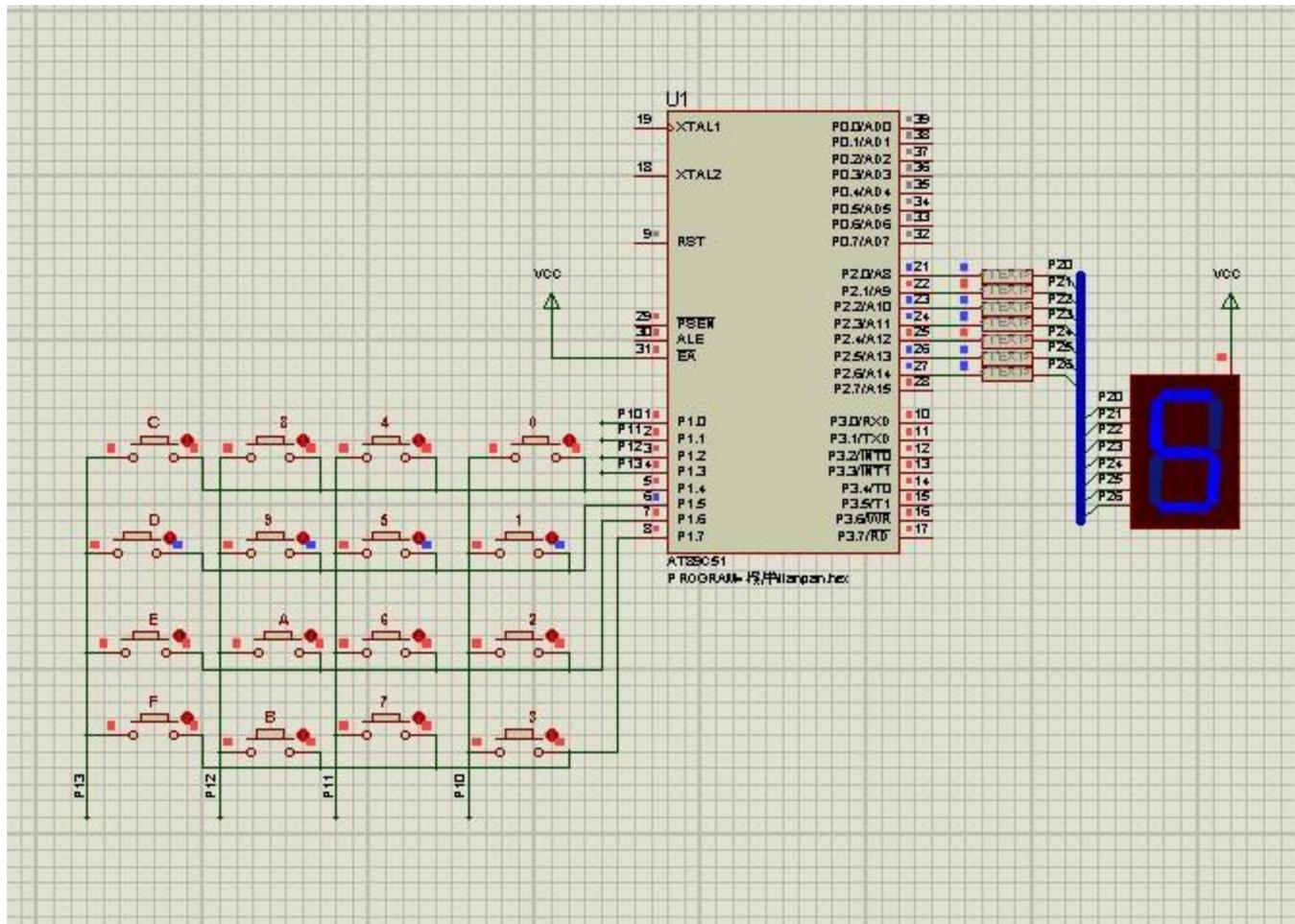
```
    ORG 0
    SJMP STAR
    ORG 30H
STAR:
    MOV SP,#5FH
    MOV P2,#0FFH
LOOP:
    MOV R0,#4
    MOV R1,#0EFH
    MOV R2,#0FFH
NKEY:
    MOV A,R1
    MOV P1,A
    MOV A,P1
    ORL A,#0F0H
    CJNE A,#0FFH,KEYIN
    MOV A,R1
    RL A
    MOV R1,A
    DJNZ R0,NKEY
    LJMP LOOP
KEYIN: MOV R2,A
        ACALL DLY
PRES:  MOV A,P1
        ORL A,#0F0H
        CJNE A,#0FFH,PRES
        LCALL DLY
        LCALL KEYVAL
        LCALL SEG7
        MOV P2,A
        LJMP LOOP
DLY:  MOV R7,#30H
        MOV R6,#0
D1:  DJNZ R6,$
        DJNZ R7,D1
        RET
KEYVAL:MOV B,#0
        MOV A,R2
KEY1:  RRC A
        JNC KEY2
        INC B
        INC B
        INC B
```

```

INC B
LJMP KEY1
KEY2:  MOV A,R1
RR A
RR A
RR A
RR A
KEY3: RRC A
JNC KEY4
INC B
LJMP KEY3
KEY4:  MOV A,B
RET
SEG7: INC A
      MOVC A,@A+PC
      RET
DB 0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H
DB 80H,90H,88H,80H,0C6H,0C0H,86H,8EH
END

```

仿真图：



## 实验六：单片机与 LED 数码管接口技术

### 实验目的：

1. 单片机与 LED 数码管的接口技术多种多样，这里只讲述用单片机控制四位数码管的动态扫描显示技术，包括电路设计和程序设计，四位数码管能稳定显示四位数 3210。
2. 用 proteus 实现该接口的电路设计和程序设计，并进行实时仿真。
3. 熟悉 LED 与单片机的接口方式

### 程序：

```
ORG 0
mov p1,#0feh
mov p2,#0H
sjmp $

SJMP STAR
ORG 30H
STAR:    MOV R2,#0
ST1:    ACALL DISP
        ;INC R2
        SJMP STAR    ;1
DISP:    ;MOV R3,#1    ;10
        ;PUSH 2
DISP1:    ;POP 2
        MOV R5,#0FEH
DISP2:    ;MOV P2,#0FFH

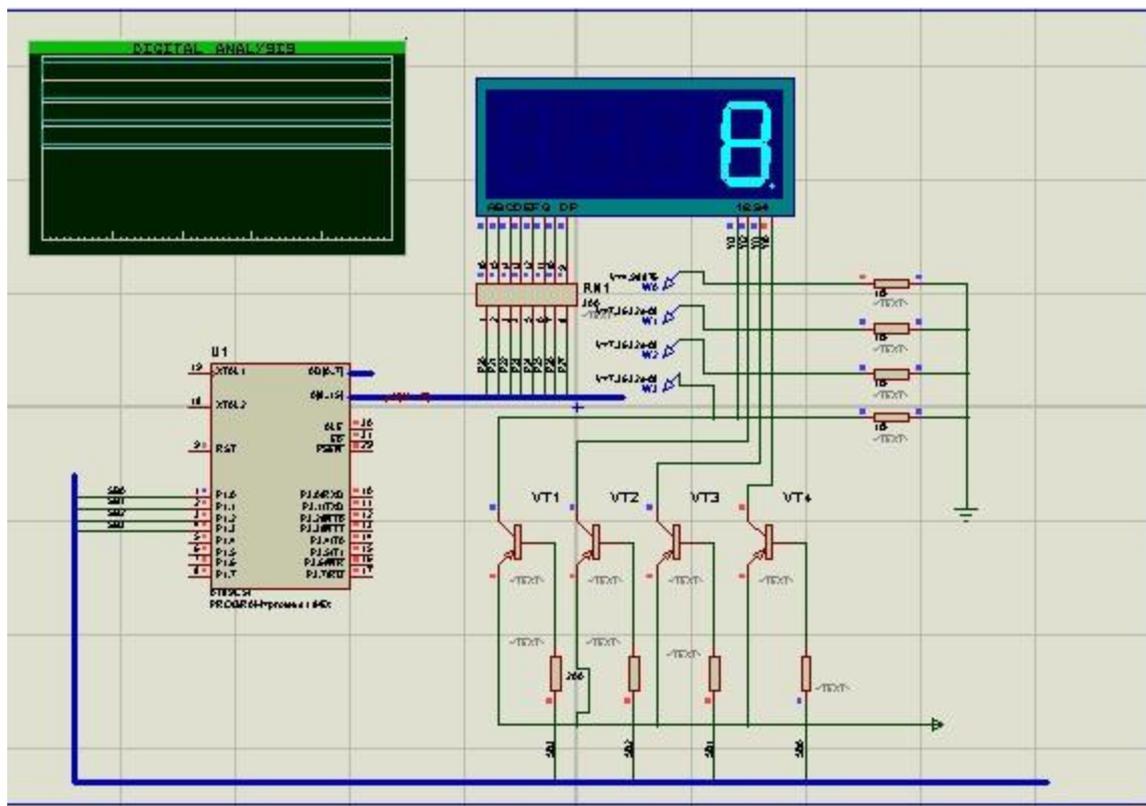
MOV A,R2
ANL A,#0FH
MOV DPTR,#TAB
MOVC A,@A+DPTR
        cpl a
MOV P2,#0 ;FFH ;C0H ;A
        MOV A,R5
MOV P1,A
LCALL DLY4MS
INC R2
MOV A,R5
JNB ACC.3,DISP3
RL A
MOV R5,A
SJMP DISP2
```

```

;PUSH 2
DISP3:    ;DJNZ R3,DISP2    ;1    ;R3 减 1 不为 0 转移
;POP 2
;MOV P1,#0FFH
;MOV P2,#0FFH
RET
TAB:      DB 0C0H,0F9H,0A4H,0B0H
          DB 99H,92H,82H,0F8H
          DB 80H,90H,88H,83H
          DB 0C6H,0A1H,86H,8EH
DLY4MS:   MOV R6,#78H
D1:       MOV R7,#200
          DJNZ R7,$
          DJNZ R6,D1
          RET
          END

```

仿真图：



## 实验七：单片机与 DAC0808 接口技术

### 实验目的：

1. 用 AT89C51 单片机控制 DAC0808 实现数模 (D/A) 转换。从单片机输出数据到 DAC0808，经其转换成模拟量输出。

2.用 PROTEUS 实现该接口的电路设计和程序设计，并进行实时交互仿真。

3.在 ASF 高级仿真图标上观测数字量信号与模拟量信号。

程序:

```
ORG 0
    SJMP STAR
ORG 0BH
    MOV TH0,#HIGH(61536)
    MOV TL0,#LOW(61536)
    MOV A,R2
    MOV DPTR,#TAB
    MOVC A,@A+DPTR
    MOV P2,A
    INC R2
    CJNE R2,#50,RETF
    MOV R2,#0
RETF:  RETI
TAB:DB 100,118,76,142,60,157,46,170
    DB 110,27
    DB 186,17,193,11,197,7,200,3,198,2
    DB 197,0,194,3,188,7,183,16,174,25
    DB 166,37,156,50,145,60,137,67,128,73
    DB 122,75,124,65,129,54,136,46,145,4

STAR:  MOV TH0,#HIGH(61536)
        MOV TL0,#LOW(61536)
        MOV TMOD,#1
        SETB EA
        SETB ET0
        SETB TR0
        SETB 0
    MOV R2,#0
    SJMP$
END
仿真图:
```



## 实验八：单片机之间的串行口实验

### 实验目的：

1. 通信的双方为单片机甲、单片机乙。由甲方的矩阵键盘输入一键码 (0—F)，将这一数据发送给乙方。乙方接收数据，并显示在自己的数码管上，同时将刚收到的数据，再回发给甲方。甲方将收到的数据显示在自己的数码管上。
2. 用 PROTEUS 实现该串口通信接口的电路设计和程序设计，并进行实时交互仿真。
3. PROTEUS VSM 虚拟终端的应用。

### 程序：

```
ORG 0
SJMP STAR
ORG 30H
STAR: MOV TMOD,#20H
MOV PCON,#0
MOV TH1,#0FDH
MOV TL1,#0FDH

SETB TR1
CLR ES

MOV SP,#5FH

MOV P2,#0FFH

LOOP: MOV SCON,#50H
MOV R0,#4
MOV R1,#0EFH
MOV R2,#0FFH
NKEY: MOV A,R1
MOV P1,A
MOV A,P1
ORL A,#0F0H
CJNE A,#0FFH,KEYIN
MOV A,R1
RL A
MOV R1, A
```

```

DJNZ R0,NKEY
LJMP LOOP
KEYIN: MOV R2,A
ACALL DLY

NPRES:MOV A,P1

ORL A,#0F0H

CJNE A,#0FFH,NPRES
LCALL DLY
LCALL KEYVAL
MOV SBUF,A
JNB TI,$
CLR TI
ACALL DLY
MOV A,SBUF
JNB RI,$
CLR RI
LCALL SEG7

CPL A

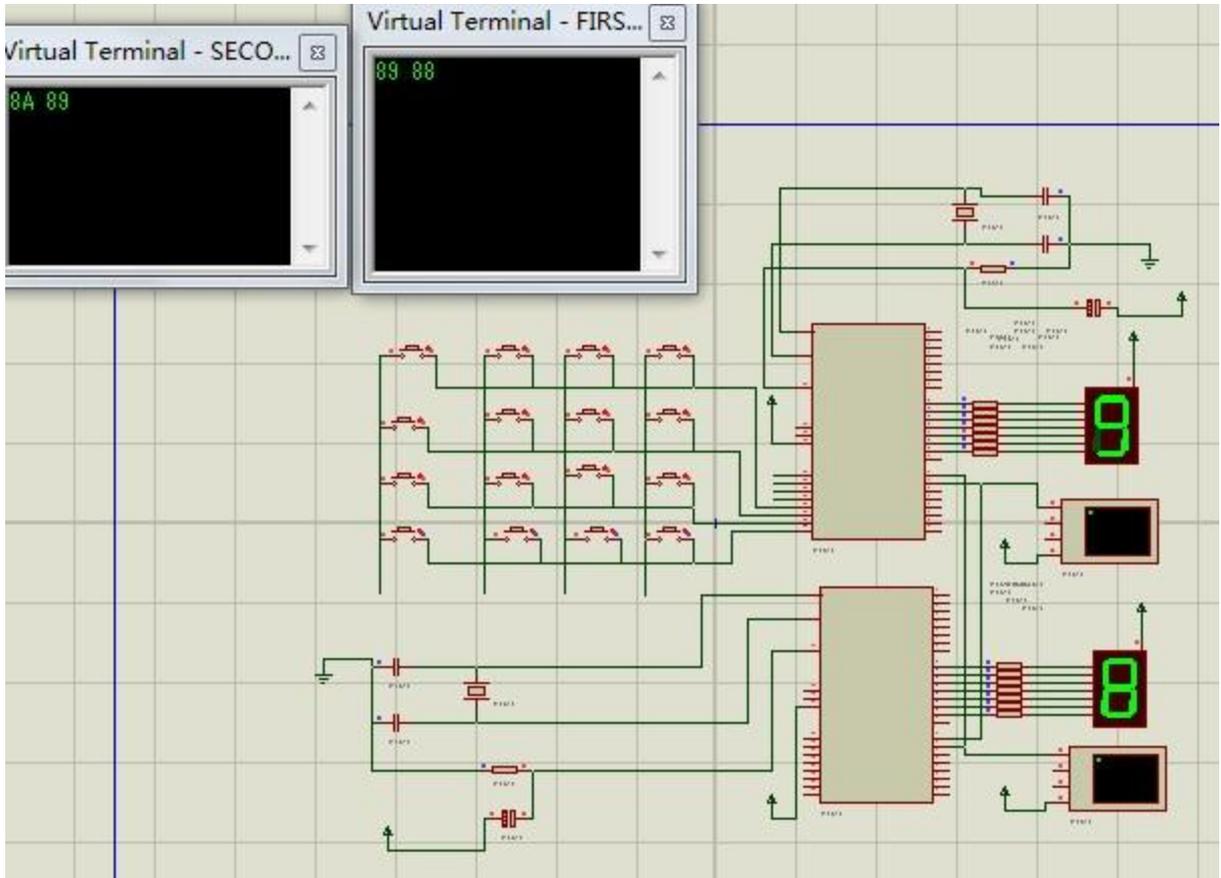
MOV P2,A
LJMP LOOP
DLY:MOV R7,#30H
MOV R6,#0
D1:DJNZ R6,$
DJNZ R7,D1
RET
KEYVAL: MOV B,#0
MOV A,R2
KEY1:RRC A
JNC KEY2
INC B
INC B
INC B
INC B
LJMP KEY1
KEY2:MOV A,R1
RR A
RR A
RR A
RR A

```

```

KEY3:RRC A
JNC KEY4
INC B
LJMP KEY3
KEY4: MOV A,B
RET
SEG7: INC A
MOVC A,@A+PC
RET
DB 03FH,06,5BH,4FH,66H,6DH,7DH,7
DB 7FH,6FH,77H,7CH
DB 39H,5EH,79H,71H,3FH
END
仿真图:

```



## 心得体会

时间真如白驹过隙，一眨眼，一个学期就这么匆匆过去了。回想起往日对单片机的陌生，以及对掌握一门专业技术的兴奋，与如今能够初步地运用一些单片机知识，心中确实颇有感悟，知识的掌握过程不仅需要兴趣，更需要务实、脚踏实地的学习态度。

在过去的一个学期中，我们在王桂棠教授的指导下，以 MCS-51 单片机为学习对象，由浅到深地了解和学习单片机，以下是对本学期学习单片机以来的一些感受：

1、书本的知识要有初步的认知，具体的知识点在需要用到的时候在具体进行了解，由于我们课程的容量很大，而单片机又是一门注重实践的学科，平时老师课上讲的知识，课后自己没有花大量的时间去复习的话基本是很难掌握的。而事实上加之在其他各科的课程安排下，我们在单片机上所花的时间与实际完全掌握所需要的时间是不匹配的，因此我们再平时的学习中对相关知识不需要面面俱到地去掌握，而是在实践中把这些知识一点一点地找回。

2、课中的认真听讲可以为课后省下很多工作，课上老师讲的内容都是经过浓缩提炼并且符合实际情况的，老师的讲授更清晰易懂，课上的实践把握好了可以为我们课后的学习省下很多时间

3、多看别人的程序，学习别人解决问题的思路，除了个别同学之外，我们大多数人都是初步接触单片机，因此大家的水平都有所限制，而在这样的基础上，要想一时半刻就能编写出质量高的程序是比较困难的，而一些掌握的比较好的同学编写的程序，还有一些参考书上的程序都是十分优秀的，把这些程序细细品味分解，我们可以掌握很多单纯课堂上课所了解不到的知识。

4、要充分利用身边的资源，遇到不懂的问题多向老师同学请教常常会让自己恍然大悟，另外网络也是一项很好的资源，很多问题的解决方法还有一些具体的知识内容我们都可以在网上找到解答。

5、不能纸上谈兵，要注重实践：单片机是一门很注重实践与应用的学科课程，倘若缺乏一定的实践编程训练时很难学好的。在刚接触单片机的时候，我就下决心要把它学好，因为高中的时候就觉得编写软件是一件很有趣的事情，而单片机的应用缺少不了程序的辅助。诚然，光有热情而不去实践是远远不足的。为了让大家更好地掌握书本的知识，老师从一开始就向大家讲解了我们本学期的课程安排，以及课程模式。我们作为第一届单片机课程改革的学习班，将会采取基于项目的学习这样一种在国外盛行的教学模式，这意味着我们在学习过程中需要大量地进行编程实验，并且要做答辩。因此，我们班里面的同学也被分为了若干小组，每个小组都需要集体合作对具体的学习项目进行实践。由于缺乏经验，在第

一次实验中我们借鉴了大量别人的方法与代码，并进行了分析，最后制定了我们的实验方案。实验的过程中我们要运用到 proteus 这款软件，在第一次实验之后大家对这款软件的使用都有了一个初步的认识。

6、及时调整。当然随着学习的进度的推进，课程的难度的显现，我们编程过程中需要了解的知识也原来越多，由于每个人的重心不一样，在后期的学习中每个人的进度也会存在差别，在这个时候就要懂得调整自己尽量使自己跟上老师的步伐。

然后针对单片机的课程设置说说自己的一点想法：我们本学期的单片机课程的每次课基本都是四节连堂，在前一二节课大家的听课积极性都比较高，但是后面的一两节课大家就不怎么听得进去了，这里原因个人认为主要有两个，一个是课时容量太大，四节连堂一时大家很难消化，另外根据人的认知规律，连续长时间地学习一门课程，效率也会降低很多，如果以后的单片机课程能够分成两节课一个课时，可能会比较好

最后谢谢老师这一个学期一来的单片机授课，老师的讲课内容很详实，生动，平时我们有什么问题老师总是不惜牺牲自己的休息时间不厌其烦地为我们解答，为了让我们学得更好老师也放弃了很多个人的时间安排为我们进行课程答辩，这一路老师您都辛苦了，谢谢老师。

## 参考文献

- [1] 吴黎明. 单片机原理及应用技术. 北京: 科学出版社, 2005.