

第 13 章 实验

13.1 基础实验部分

13.1.1 实验一 数据传送

1. 实验目的

通过本次实验掌握 Keil uVision3 开发环境以及单片机汇编语言和 C 语言源程序编辑、编译、调试方式方法，熟悉单片机汇编语言指令，掌握单片机片内外存储器间数据传送的方法，用汇编语言和 C 语言编程实现单片机片内外存储器存取数据。

2. 实验要求

在本次实验前，在自己的电脑安装 Keil uVision3 以上版本的软件，根据实验内容和程序框图编写汇编语言和 C51 程序。

3. 实验器材

仿真实验：装有 Keil uVision3 以上版本软件的笔记本或台式 PC 机。

4. 实验内容

编写软件实现：设置单片机片内存储器存储区首地址为 30H、片外存储器存储区首地址为 3000H，存取数据字节个数 16 个，将片内存储区内容设置为 01H~10H 共 16 个字节，读取片内首地址为 30H 单元内容，将该内容传送到片外数据存储器存储区中保存，将保存在片外数据区数据依次取出送 P1。观察片内、外存储区数据变化、P1 口状态变化。

5. 程序框图

数据传送实验主程序框图如图 13-1 所示，片内数据区数据初始化子程序框图如图 13-2 所示，片内数据传送到片外数据区子程序框图如图 13-3 所示。

6. 实验步骤：

(1) 新建一个文件夹，用自己名字首字母-班级命名。

(2) 按照教材 2.1.1 节介绍，建立一个工程，并在工程里添加用于写代码的文件，该文件名自定，但后缀名为.asm（汇编语言）或后缀名为.C（C51 程序）

(3) 对该文件进行编辑、编译、修改直到编译该源代码无错误

(4) 在 Keil 环境下 Debug 调试该程序，使用单步调试 (Step)，观察寄存器 (R0,R2,A,SP) 内容变换、内部数据区 30H-3FH 单元内容，片外数据区 3000H-300FH 内容以及 P1 口内容变化，或连续运行 (Run) 观察运行结果。

(5) 运行结果：内部数据区 30H-3FH 单元内容为 01H-10H，片外数据区 3000H-300FH 内容为 01H-10H，P1 口循环显示 01H-10H 状态。

调试时注意事项：

- 打开 View，查找寄存器窗口（register windows），查找片内外数据区（memory window/Xemory），打开 Xemory 窗口，在地址处（Addres）输入：c:地址值(代码区)，若输入 d: 地址值(片内数据区)，x:地址值（片外数据区），地址值为十六进制值。例如：输入输入 c:0000h，则窗口内显示是首地址为 0000H 开始代码值；若输入 d:30H，则窗口内显示是片内 30H 单元开始的单元内容；若输入 x:3000H，则窗口内显示是片外数据区 3000H 单元开始单元内容。

- 打开 Peripherals，查找 I/O-Port/Port1，观察 P1 口（8 位）输出状态。

7. 思考题：

将片外数据区首地址为 2000H~203FH 单元内容设置为 3FH~00H 数据，并将片外数据区 2000H~203FH 内容传送到片内 30H 开始连续单元中。

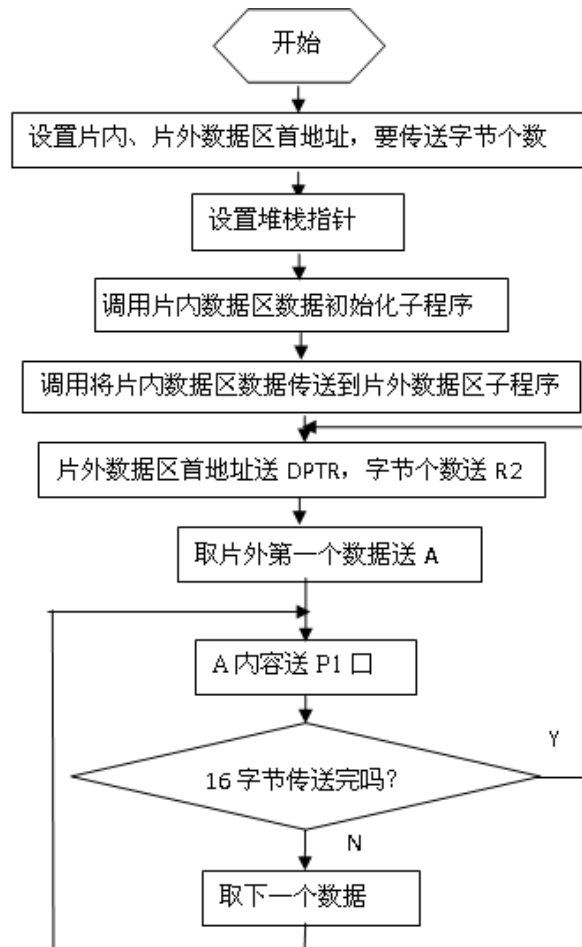


图 13-1 数据传送实验主程序框图

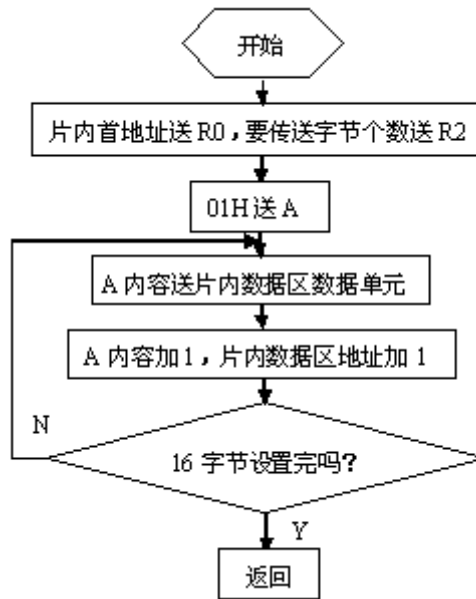


图 13-2 片内数据区数据初始化子程序框图

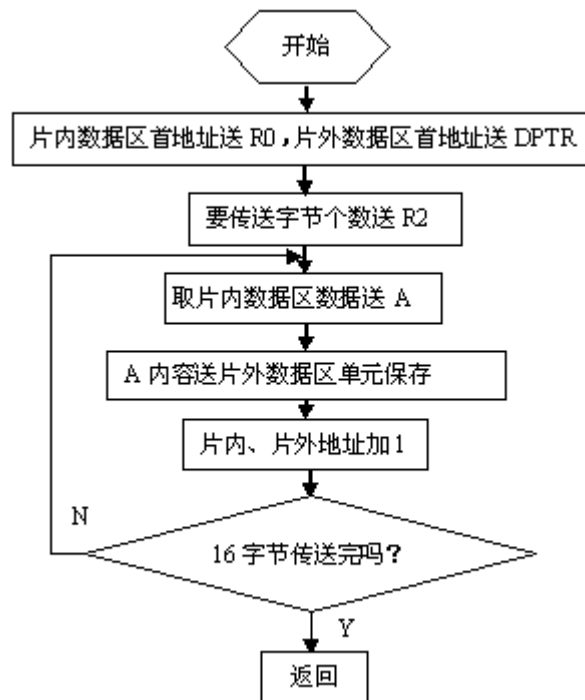


图 13-3 片内数据传送到片外数据区子程序框图

13.1.2 实验二 多分支实验

1. 实验目的

通过本次实验掌握 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C51 源程序方式方法，熟悉单片机汇编语言指令，掌握程序散转的方法，实现程序的多分支转移。

2. 实验要求

在本次实验前，在自己的电脑安装 Keil uVision3 和 Proteus7.0 以上版本的软件，使用 Proteus 软件画参考硬件原理图，如图 13-4。

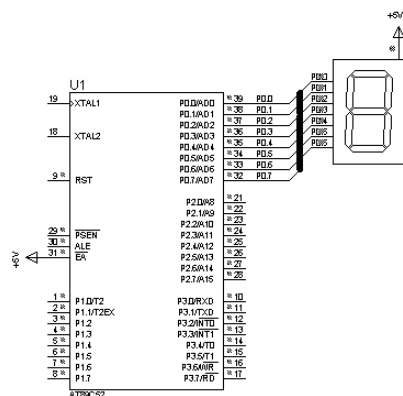


图 13-4 分支实验仿真原理图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个共阳极的七段数码管 7SEG-COM-AN-GRN，一个 +5V 稳压电源。

4. 实验内容

编写软件实现：根据 STC89C52 单片机片内 30H 单元内容（00 或 01 或 02 或 03）进行散转，1 个数码管循环显示对应的数字。（为了便于观察你也可以增加一个循环，循环取分支变量，观察显示结果）

5. 分支部分程序框图

程序框图如错误!未找到引用源。

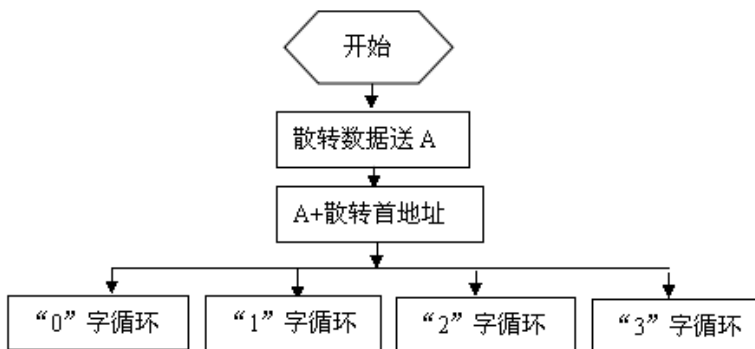


图 13-5 多分支实验框图

6. 实验步骤：

(1) 重复实验一的实验步骤 2 和实验步骤 3，建立本次实验的工程，并在工程加载本次实验文件（实验内容源代码），编辑、编译该文件，直到没有错误。

(2)按照教材 2.2 节介绍知识，画出图 13-4 分支实验仿真原理图。

(3) 在 KEIL 环境下，调试、运行源代码。

(4) 按照教材 2.3 节介绍，进行 keil μ Vision4 与 proteus7 Professional 的联调，在 Keil 环境，改变源代码 A 累加器内容 (0~3 任意值)，编辑、编译、调试、运行，观察 P0 口状态值，在 Proteus 环境下，运行程序，观察数码管显示结果。

(5) 运行结果：Keil 环境下，P0 口显示 A 累加器内数字 (0~3 任意值) 对应的字型码 (0C0H、0F9H、0A4H、0B0H 之一)。Proteus 环境下，数码管显示 A 累加器内容。

7. 思考题：编写分支程序实现，根据 A 累加器内容变化 (4~7 任意值)，数码管显示对应数字。

13.1.3 实验三 外部中断与定时器/计数器中断实验

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C51 源程序方式方法，掌握外部中断和定时/计数器的工作原理和使用方法。

2. 实验要求

在本次实验前，使用 Proteus 软件画硬件仿真原理图如错误!未找到引用源。所示，STC89C52 的 P3.2 连接按键 K，P1.0 口连接示波器。

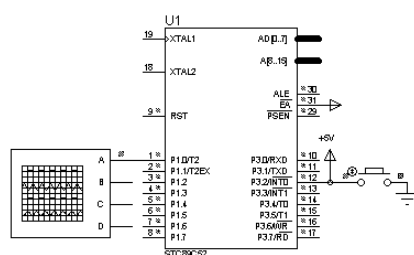


图 13-6 外部中断与定时器/计数器中断原理图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机

元件清单：1 个 STC89C52、1 个按键开关 BUTTON、1 个 +5V 稳压电源、一个示波器。

4. 实验内容

若系统时钟频率为 6MHz，编写软件实现：按一下 K 键，产生一次外部中断 0 中断信号，启动 T1 定时，使 P1.0 输出周期为 1ms 的方波。

5. 程序框图

定时/计数器 T1 中断服务子程序框图如图 13-7 所示，外部中断 0 与定时器/计数器 T1 中断实验主程序框图如图 13-8 所示，外部中断 0 中断服务子程序框图如图 13-9 所示。

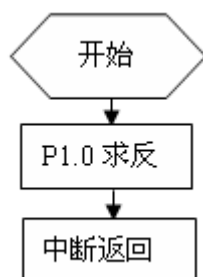


图 13-7 定时/计数器 1 中断服务子程序框图

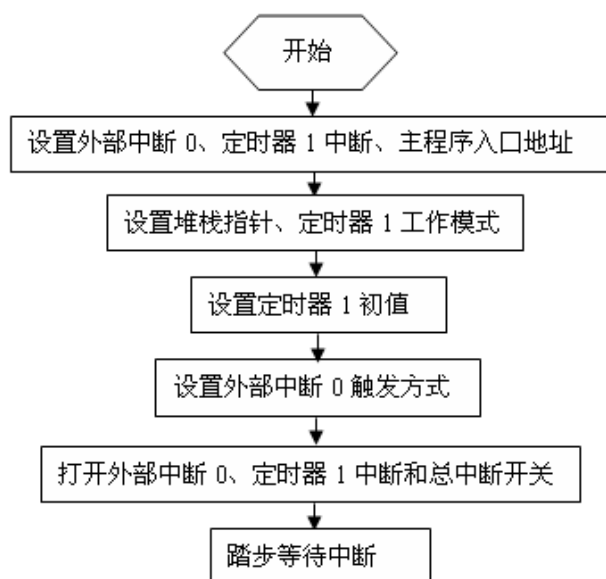


图 13-8 外部中断与定时/计数器中断主程序框图

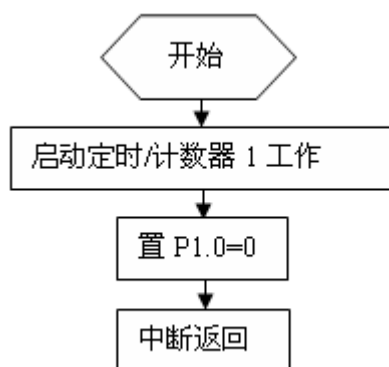


图 13-9 外部中断 0 中断服务子程序框图

13.1.4 实验四 串口双机通讯

1. 实验目的

通过本次实验掌握 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C 语言源程序方式方法，熟悉单片机汇编语言指令，掌握单片机串行口通信方法。

2. 实验要求

在本次实验前，在自己的电脑安装 Keil uVision3 和 Proteus7.0 以上版本的软件，使用 Proteus 软件画参考硬件原理图，如图 13-10。

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：2 个 STC89C52、2 个 74HC245 总线收发器、4 个 7SEG-BCD 数码管，一个 SW-SPDT-MOM 双向开关，2 个 BUTTON 按钮开关，2 个示波器，一个+5V 稳压电源。

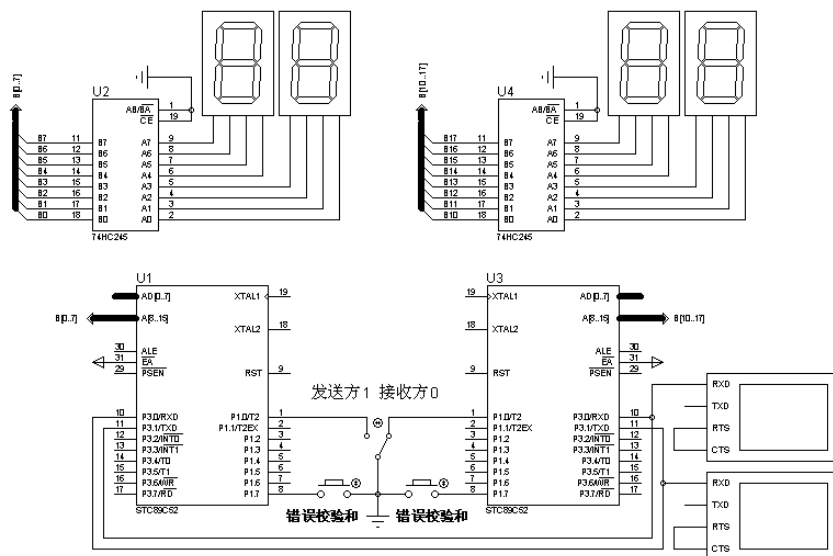


图 13-10 点对点异步通讯

4. 实验内容

按硬件原理图连线，编写软件实现：

假定有 A、B 两机，以方式 1 进行串行口通信，其中 A 机发送信息，B 机接收信息，双方的晶振频率 $f_{osc} = 11.0592\text{MHz}$ ，通信波特率为 9600。

通信协议：通信开始时，A 机首先发送一个启动信号 AA，B 机接收到后发送一个应答信号 BB 表示同意接收。

A 机收到 BB 后，就可以发送数据了。假定该实验发送的数据长度是 32 个字节，发送数据缓冲区为片内数据区，首地址为 30H，该数据缓冲区为 bufS，数据块发送完后要立即发送校验和，进行发送数据准确性校验。

B 机将接收到的数据存储到数据缓冲区 bufR，接收数据存放在片外数据区，接收数据首地址为片内 50H，接收数据块后，再接收 A 机发来的校验和，并将其与 B 机求出的校验和进行比较。若两者相等，说明接收正确，B 机回答 00H；若两者不等，说明不正确，B 机回答 FFH，请求重发。A 机接收到 00H，停止发送，若接收到 FFH 信号重新发送。

设计要求

- (1) A、B 两机点对点的通信，该程序可以在双方机中运行，不同的是在程序运行之前，要判别 P1.0 口的输入，若 $P1.0 = 1$ ，表示该机是发送方；若 $P1.0 = 0$ ，表示该机是接收方。
- (2) 双方的 P1.7 为校验和控制，按钮抬起时 ($P1.7=1$)，发送方送出正确校验和，发送 1 次数据！按钮按下时 ($P1.7=0$)，发送方送出错误校验和，则重发！
- (3) 发送方将发送的数据送数码管显示，接收方将接收到数据也送数码管显示。

5. 程序框图

STC89C52 单片机点对点通讯主程序框图如图 13-11，点对点通讯发送子程序框图如图 13-12。点对点通讯接收子程序框图如图 13-13。

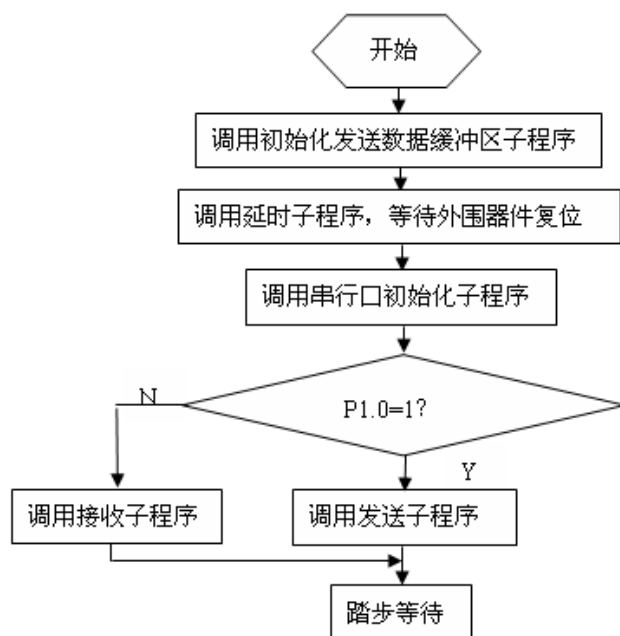


图 13-11 点对点通讯主程序框图

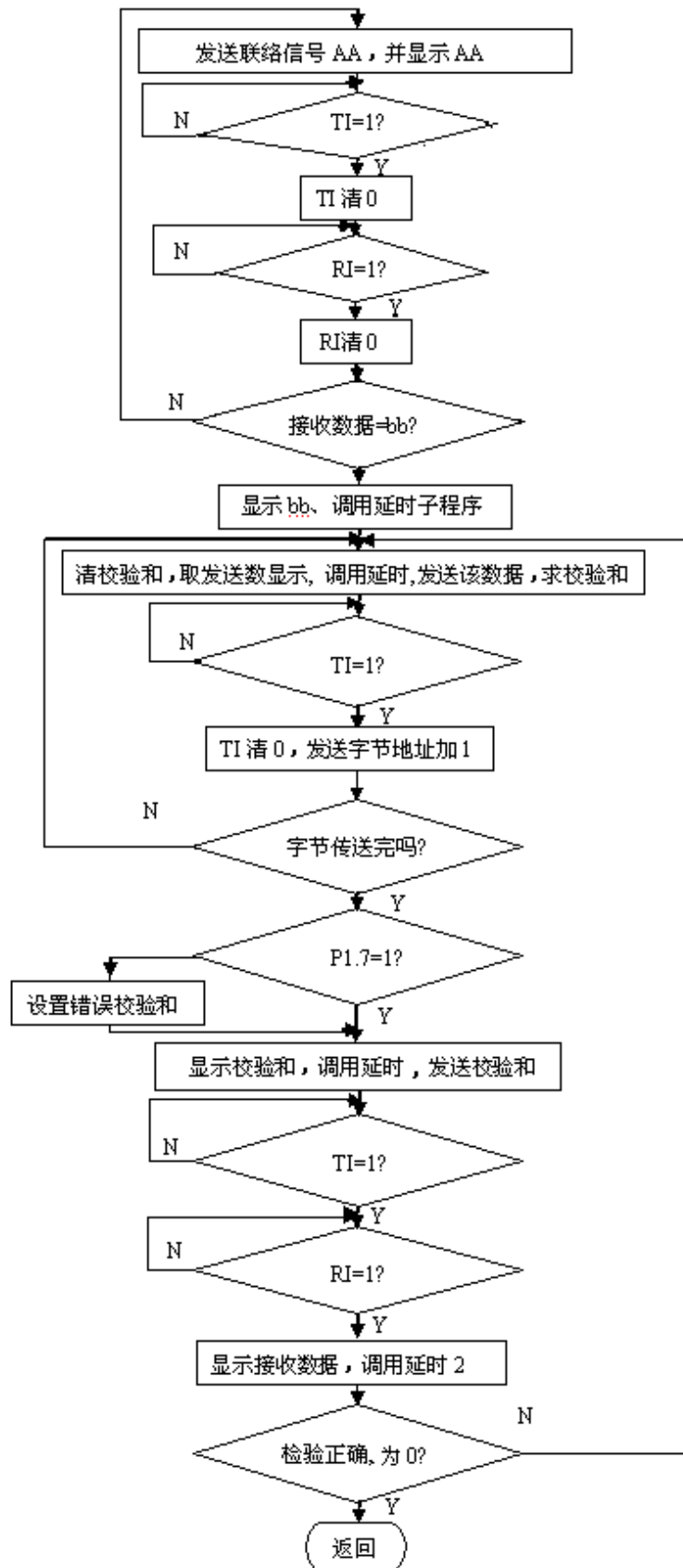


图 13-12 点对点通讯发送子程序框图

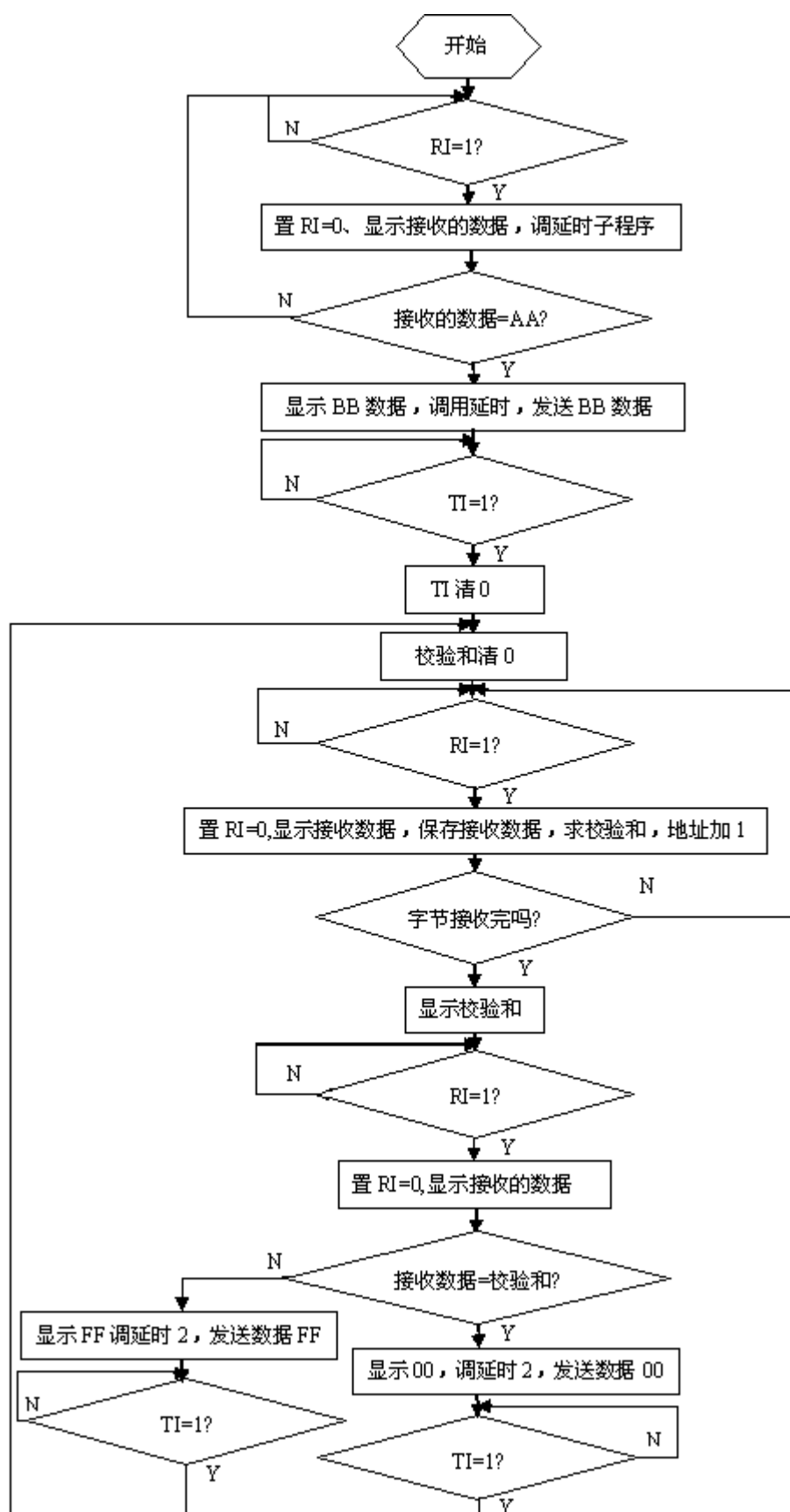


图 13-13 点对点通讯接收程序框图

13.1.5 实验五 存储器扩展实验

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C51 源程序方式方法，掌握 STC89C52 单片机扩展外部存储器方法。

2. 实验要求

本次实验 STC89C52 单片机扩展一片 EPROM27256 和一片 SRAM62256，实现片内外数据传输。在本次实验前，使用 Proteus 软件画硬件原理图，电路如图 13-14 所示。

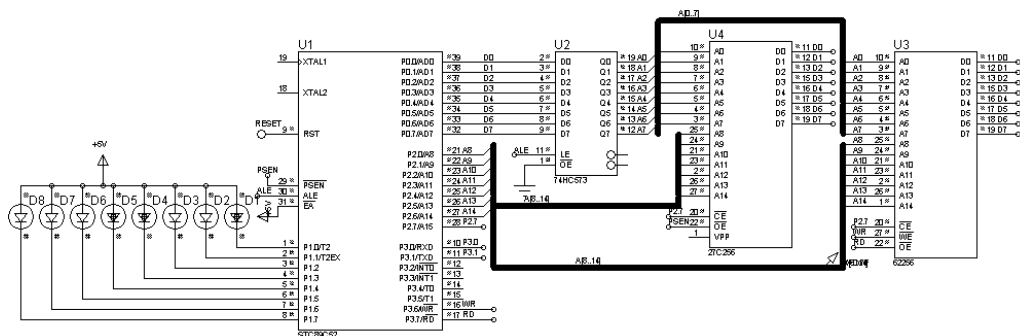


图 13-14 存储器扩展仿真电路图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 HC573 锁存器、1 个 EPROM27C256、8 个发光二极管 LED-YELLOW，一个 +5V 稳压电源。

4. 实验内容

根据硬件原理图编写软件实现：

- (1) 将编写好程序固化在扩展片外程序存储器 EPROM27256
- (2) 将片内数据存储区 30H-4FH 单元内容设置为 00H-1FH。
- (3) 将片内数据存储区 30H-4FH 单元内容传输到片外数据存储区 SRAM62256 开始的 32 个单元中。

(4) 将片外数据存储区 SRAM62256 开始的 32 个单元内容循环取出送到 P1 口驱动发光二极管，二极管亮灭状态映射单元内容，对应位为“0”，该位灯亮，若为“1”灯灭，初始状态单元内容为 00H，8 个灯全亮，为了观察效果请加入适当延时程序。

5. 程序框图

单片机片内外数据传送并送显示程序框图如图 13-15 所示。

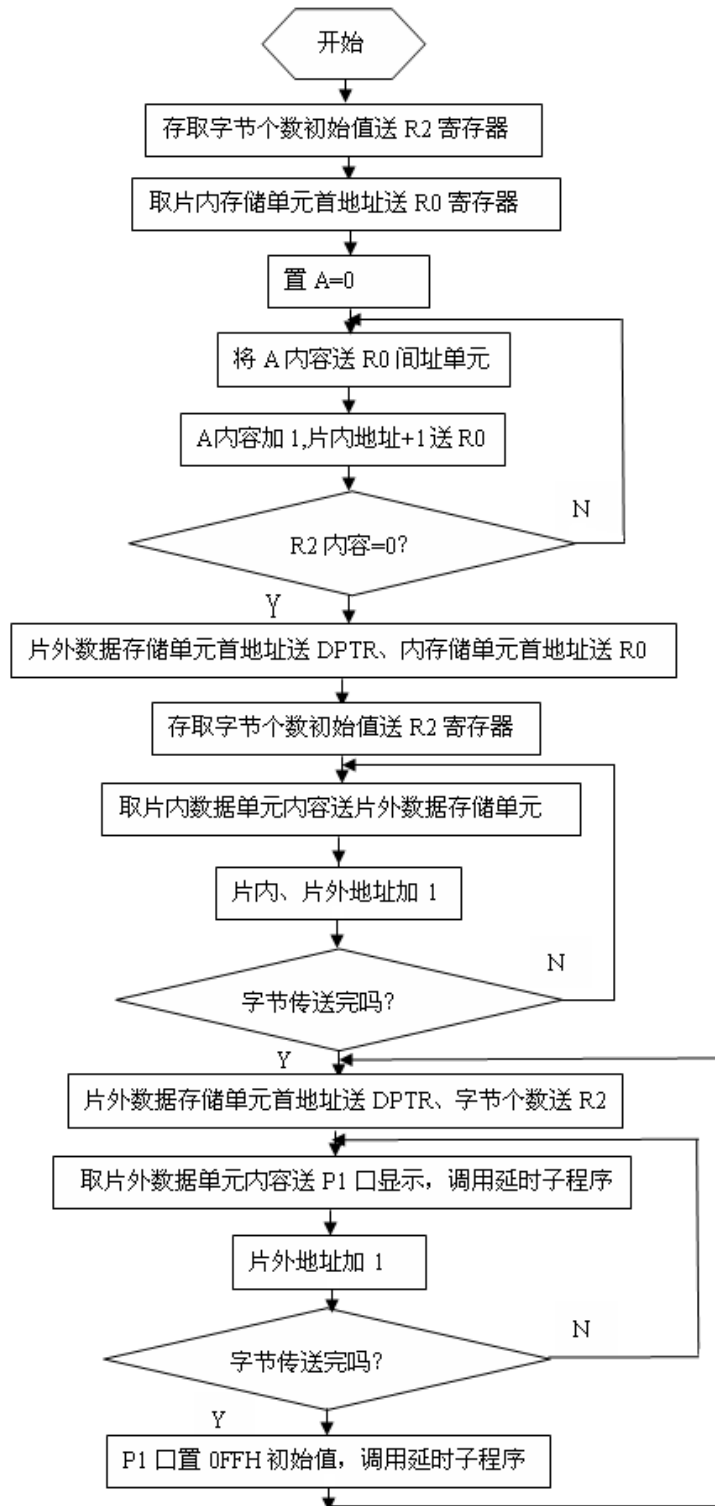


图 13-15 单片机片内外数据传送并送显示程序框图

13.1.6 实验六 82C55 控制交通灯

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C 语言源程序方式方法, 掌握 STC89C52 单片机扩展外部并行 I/O 口方法, 掌握通

过 8255A 并行口传输数据的方法，以控制发光二极管的亮与灭。

2. 实验要求

本次实验 STC89C52 单片机扩展一片可编程并行 I/O 口 82C55，用 8255 做输出口，控制十二个发光管亮灭，模拟交通灯管理，在本次实验前，使用 Proteus 软件画硬件原理图，电路如图 13-16 所示。

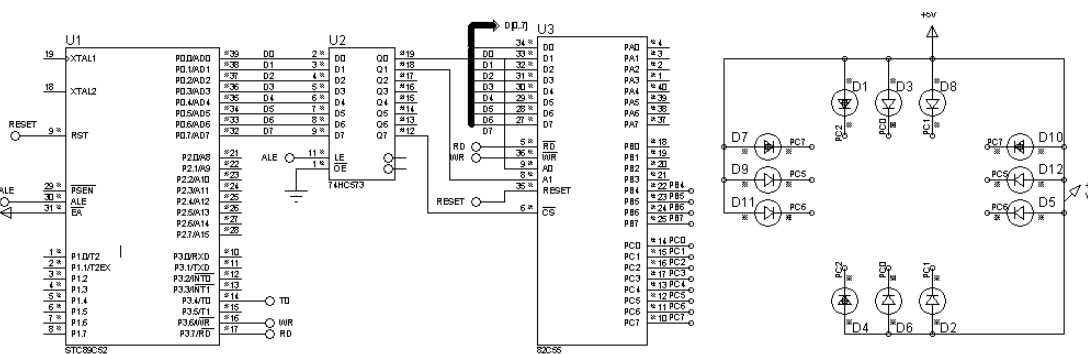


图 13-16 82C55 控制交通灯仿真电路图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 HC573 锁存器、1 个 82C55、12 个发光二极管（4 个 LED-RED，4 个 LED-GREEN，4 个 LED-YELLOW），一个 +5V 稳压电源。

4. 实验内容

根据硬件原理图编写软件实现：

设计 1、3 方向黄绿红用 PC2PC1PC0 控制，2、4 方向黄绿红灯用 PC7PC6PC5 控制来模拟交通路灯的管理。要完成本实验，必须先了解交通路灯的亮灭规律，设有一个十字路口 1、3 为南北方向，2、4 为东西方向，初始状态为四个路口的红灯全亮，之后，1、3 路口的绿灯亮，2、4 路口的红灯亮，1、3 路口方向通车。延时一段时间后，1、3 路口的绿灯熄灭，而 1、3 路口的黄灯开始闪烁，闪烁若干次以后，1、3 路口红灯亮，而同时 2、4 路口的绿灯亮，2、4 路口方向通车，延时一段时间后，2、4 路口的绿灯熄灭，而黄灯开始闪烁，闪烁若干次以后，再切换到 1、3 路口方向，之后，重复上述过程。

根据交通灯亮灭规则，我们将 PC 口控制交通灯亮灭状态字列出：

- 初始状态值：红灯全亮、黄、绿灯灭，则有状态字为：110 00110B=C6H(全亮)，
- 第 1 状态：1、3 路口的绿灯亮，2、4 路口的红灯亮，则有：110 00 101=C5H
- 第 2 状态：1、3 路口的绿灯灭，黄灯开始闪烁，此时 2、4 路口状态不变，则有：110 00 011=C3H 延时 110 00 111=C7H 延时，循环输入 C3H、C7H 共 8 次，闪烁 8 次
- 第 3 状态：1、3 路口红灯亮，2、4 路口的绿灯亮，则状态字为 101 00 110=A6H
- 第 4 状态：2、4 路口的绿灯熄灭，而黄灯开始闪烁，此时 1、3 路口状态不变，则有：011 00 110=66H 延时 111 00 110=E6H 延时，循环输入 66H、E6H 共 8 次，闪烁 8 次

5. 程序框图

82C55 控制交通灯实验程序框图见图 13-17。

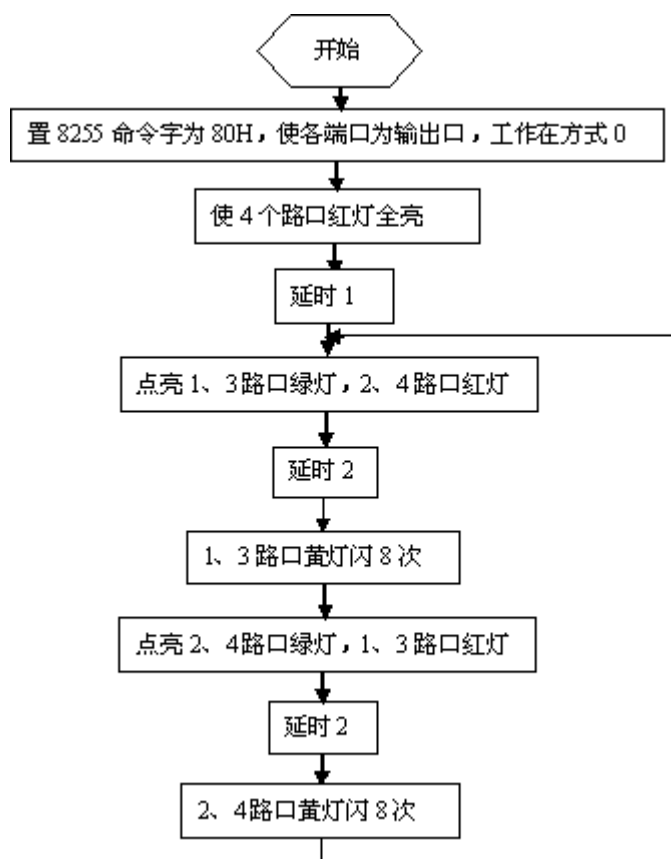


图 13-17 82C55 控制交通灯程序框图

13.1.7 实验七 键盘和显示实验

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C 语言源程序方式方法, 掌握 STC89C52 单片机扩展可编程并行接口 82C55 方法, 82C55 连接键盘和数码管, 掌握扫描键盘和驱动数码管亮灭方法。

2. 实验要求

在本次实验前, 使用 Proteus 软件画电路原理图, 电路如图 13-18 所示。

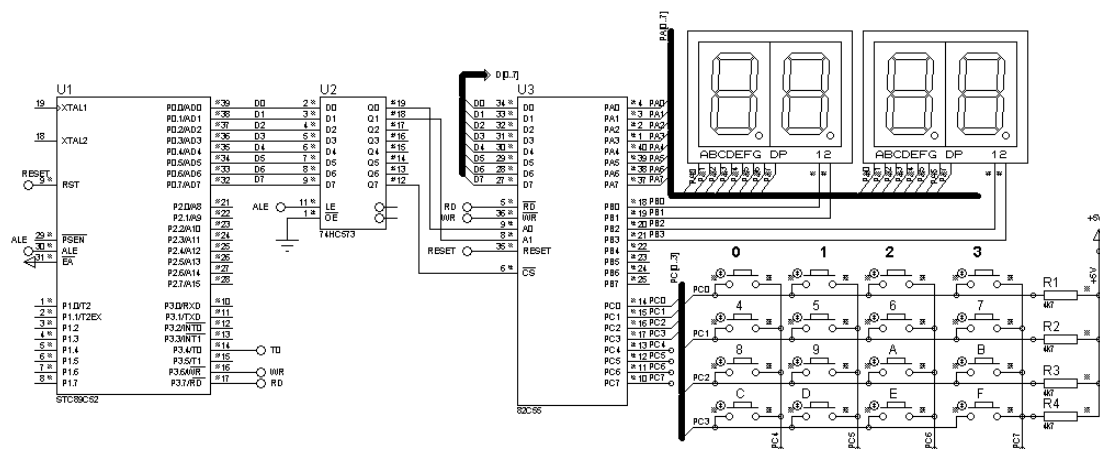


图 13-18 键盘、显示实验仿真电路图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 HC573、1 个 82C55、2 组七段数码管 7SEG-MPX2-CA、16 个按键键盘 BUTTON、4 个 4.7K 电阻 10WATT4K7，一个 +5V 稳压电源。

4. 实验内容

根据电路原理图编写软件实现：

- (1) 按逐行扫描方法或反转法，编写键盘扫描子程序。
- (2) 编写数码管显示子程序。
- (3) 将从键盘输入数据送数码管显示出来。

5. 设计分析

分析图 13-18 电路 82C55 两侧知：

CPU 端：单片机数据线与 82C55 数据线相连，单片机地址线 A7 连接 82C55 片选，单片机地址线 A1A0 连接 82C55 的端口选择线 A1A0，单片机的读写线 RD、WR 与 82C55 的读写线相连。

外设端：82C55 芯片 A 口连接数码管段码端，B 口连接数码管位码端，C 口低 4 位连接键盘行线，C 口高 4 位连接键盘列线。

82C55 的 A、B、C 和控制/命令端口地址为：FF7CH、FF7DH、FF7EH、FF7FH

6. 程序框图

方法一：逐行扫描方法，该键盘、显示实验的主程序框图如图 13-19，键盘扫描子程序框图如图 13-20，数码管显示子程序框图如错误!未找到引用源。。

方法二：线反转法，PC 口低 4 位为行线，PC 口高 4 位为列线，先行线输出，读列值，然后列线输出，读行值，组合列值和行值，求反查表获得键值。

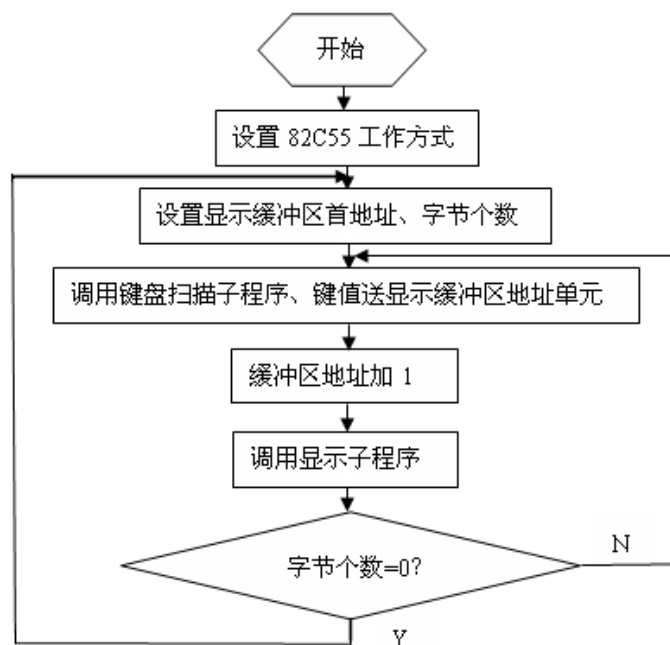


图 13-19 键盘、显示实验主程序框图

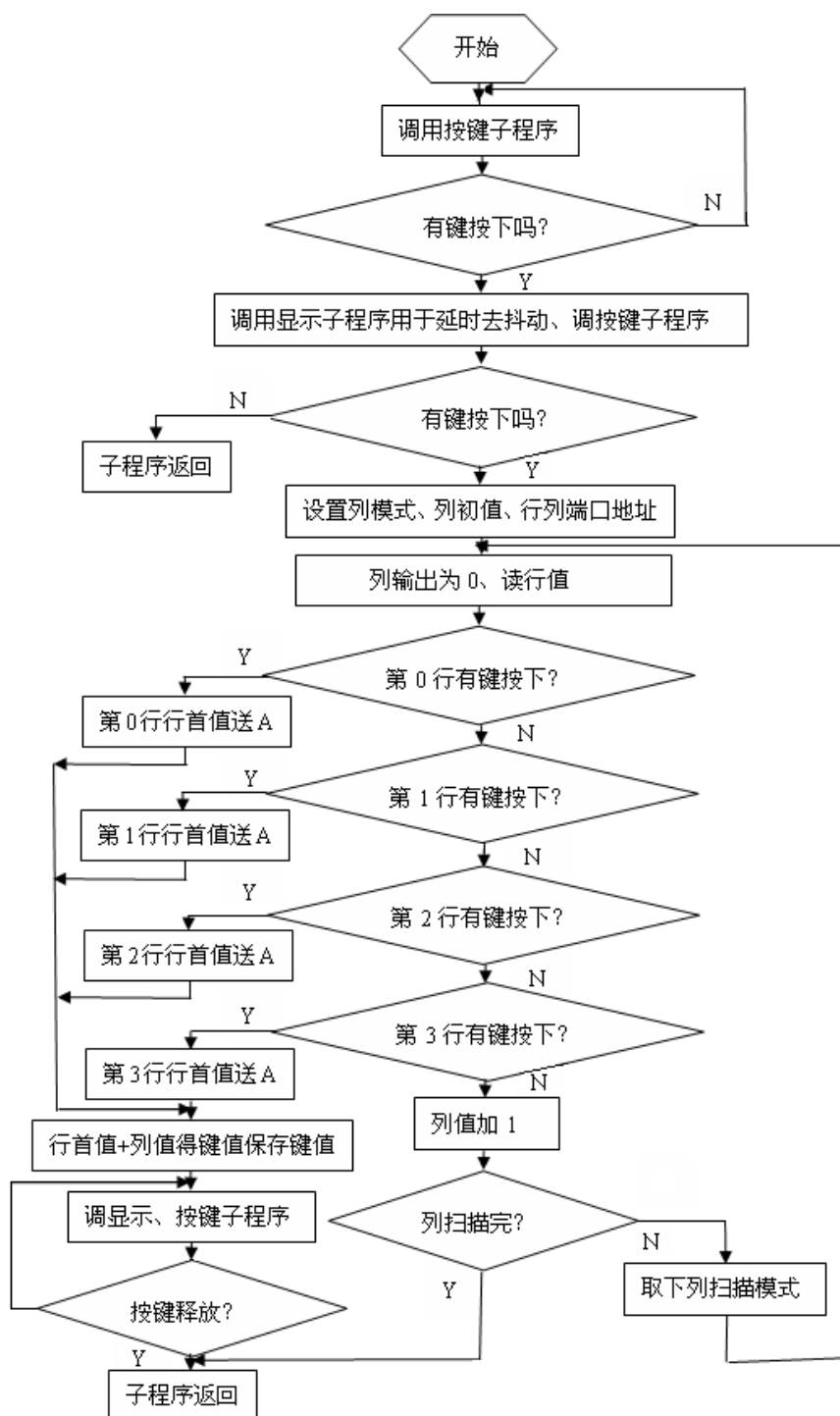


图 13-20 键盘扫描子程序框图

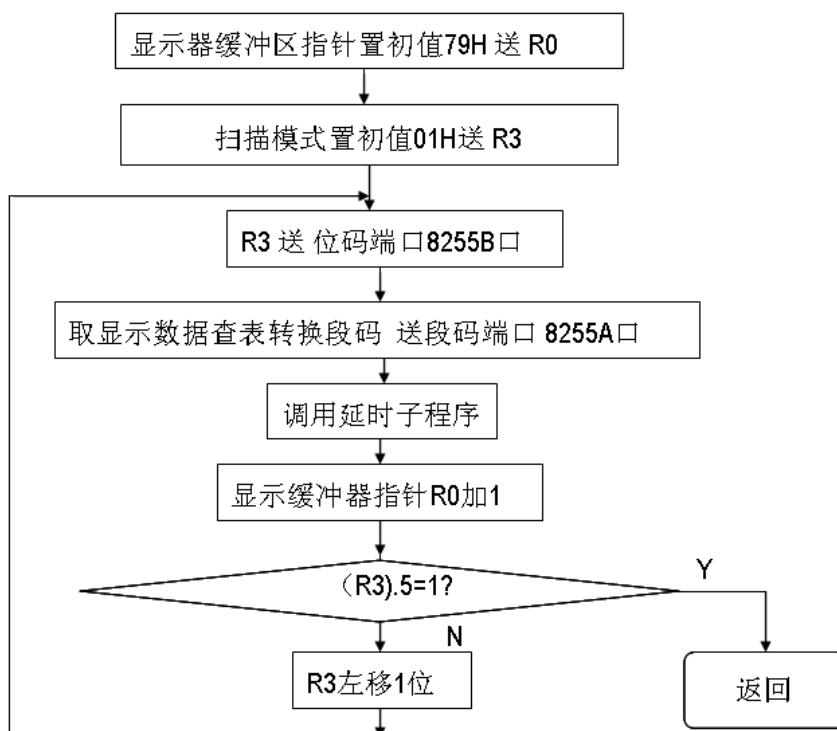


图 13-21 数码管显示子程序框图

13.1.8 实验八 A/D 转换

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C51 源程序方式方法，掌握 STC89C52 单片机与 ADC0809 芯片连接方法，掌握 ADC0809 芯片工作原理和使用方法。

2. 实验要求

在本次实验前，使用 Proteus 软件画 ADC0809 数据采集实验仿真原理图，电路如图 13-22 所示。

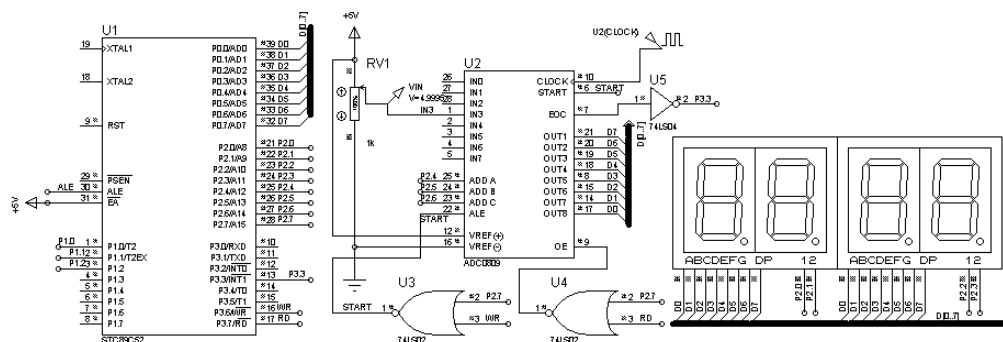


图 13-22 ADC0809 数据采集实验仿真原理图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 ADC0809、2 组七段数码管 7SEG-MPX2-CA、一个电位器 POT-HG、2 个或非门 74LS02，1 个与非门 74LS04。

4. 实验内容

根据图 13-22 电路。编写软件实现如下功能：

- (1) 上电复位启动 ADC0809 的第 3 通道模数转换。
- (2) 每隔 32.56ms(>转换时间) 去采样 ADC0809 的第 3 通道电压值，采样值送数码管显示出来，第 1 位显示通道号，后 2 位显示 A/D 转换后十六进制数据(ADC0809 的工作频率为 500KHz)。

5. 程序框图

A/D 转换实验的单通道数据采集主程序框图如图 13-23，显示格式转换子程序框图如图 13-24。

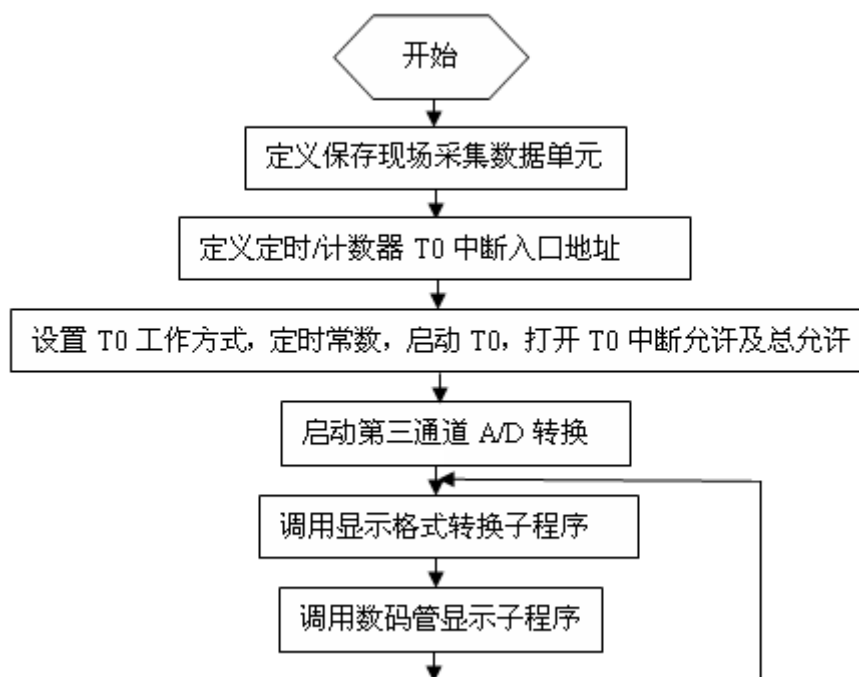


图 13-23 单通道数据采集主程序框图

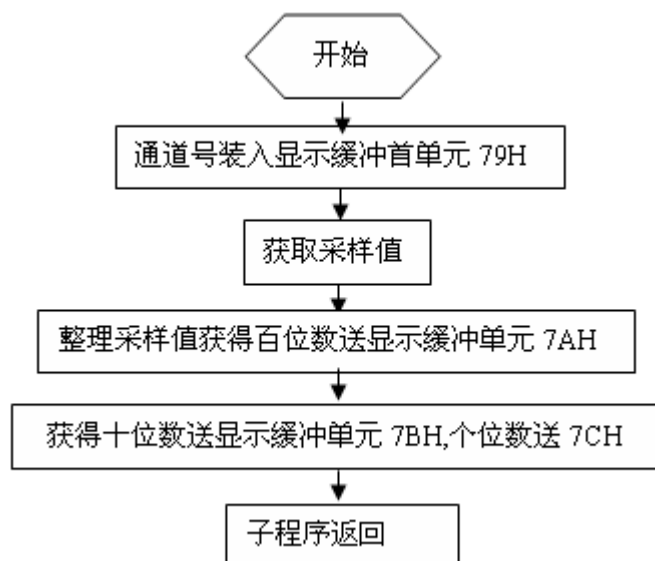


图 13-24 显示格式转换子程序

6. 实验结果

当输入电压 $V_i=0V$ 、 $+1.25V$ 、 $+2.5V$ 、 $+3.75V$ 、 $+5V$ 时，计算 AD 转换数字量理论值以及实际测量值。

13.1.9 实验九 D/A 转换

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C 语言源程序方式方法，掌握 STC89C52 单片机与 DAC0832 芯片连接方法，掌握 DAC0832 芯片工作原理和使用方法。

2. 实验要求

在本次实验前，使用 Proteus 软件画 D/A 转换实验仿真原理图，电路如图 13-25 所示。

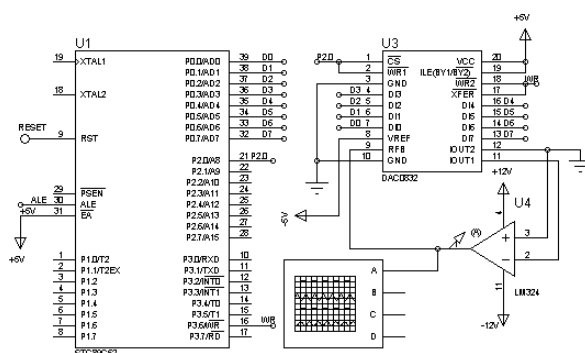


图 13-25 D/A 转换实验原理图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 DAC0832、1 个 LM324、一个示波器、1 个 +5V 稳压电源、1 个 +12V 稳压电源和 1 个 -12 稳压电源。

4. 实验内容

根据实验原理图，编写软件实现：在 DAC0832 工作在单缓冲方式下，用示波器观察 DAC0832 输出矩形波。

设计分析：根据实验原理图知：DAC0832 片选信号 \overline{CS} 和 $\overline{WR1}$ 线与和单片机的 P2.0 相连，DAC0832 的 $\overline{WR2}$ 和 \overline{XFER} 线与和单片机写信号 (P3.6) 相连，DAC0832 的 ILE 直接接 +5V，可知 DAC0832 工作在单缓冲方式，DAC0832 片选地址为 FE00H。

13.1.10 实验十 实时时钟

1. 实验目的

通过本次实验进一步熟悉 Keil uVision3 和 Proteus 开发环境以及它们联机调试单片机汇编语言和 C 语言源程序方式方法，掌握 STC89C52 单片机与数码管连接方法，掌握实时时钟设计方法。

2. 实验要求

在本次实验前，使用 Proteus 软件画 实时时钟实验仿真原理图，电路如图 13-26 所示

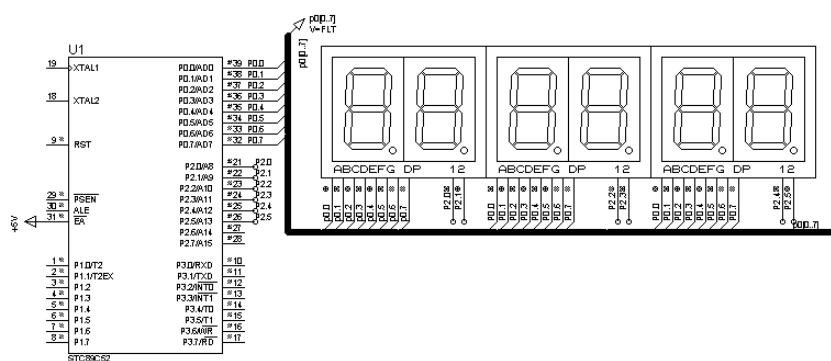


图 13-26 实时时钟仿真电路原理图

3. 实验器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、3 组 7SEG-MPX2-CA、1 个+5V 稳压电源、

4. 实验内容

根据实验原理图，编写软件实现：实现实时时钟计时并在数码管上显示。

5. 程序框图

(1) 汇编编程设计分析：最小计时单位是秒，如何获得 1s 的定时时间呢？从前面介绍知，定时器方式 1，最大定时时间也只能 131ms。可将定时器的定时时间定为 100ms，中断方式进行溢出次数的累计，计满 10 次，即得秒计时。而计数 10 次可用循环程序的方法实现。初值的计算如例题 6-2。片内 RAM 规定 3 个单元为 42H：“秒”单元；41H：“分”单元；40H：“时”单元

主程序框图如图 13-27，实时时钟中断服务子程序框图如图 13-28 所示。

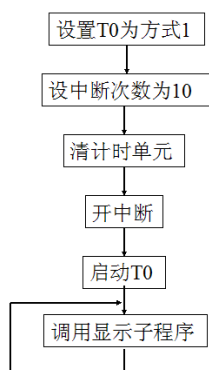


图 13-27 实时时钟主程序框图

(2) C51 编程设计分析：使用定时器 T0，工作方式 2，晶振频率为 12MHz，机器周期为 1μs，定时 250μs，定时初值 N， $(2^8-N) \times 1\mu s = 250\mu s$ ，N=06

计数初值 count，1 秒内有多少个 250μs， $count = 1s / 250\mu s = 1000000 / 250 = 4000$ 即定时 250μs 产生一次定时溢出中断，中断 4000 次，产生 1s，

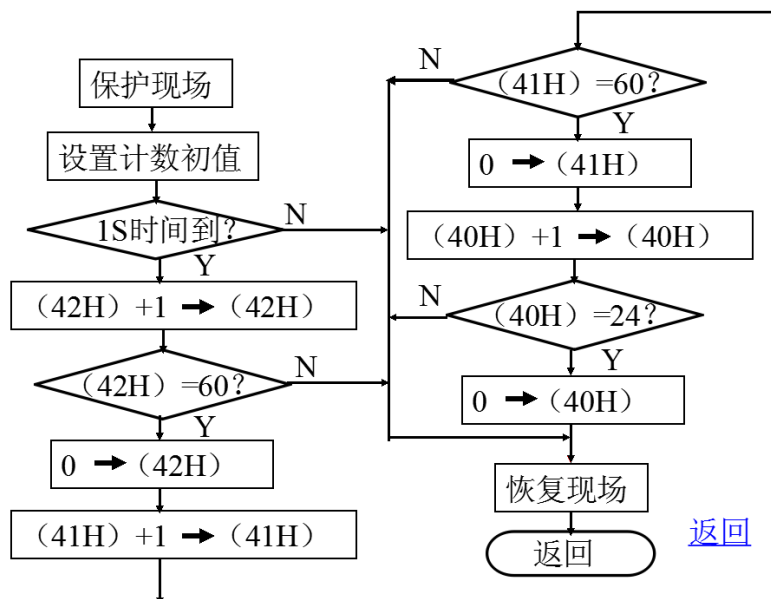


图 13-28 实时时钟中断服务子程序框图

13.2 单片机课程设计

13.2.1 自动交通管理系统

1. 设计目的

利用所学单片机的理论知识进行软硬件整体设计，培养学生分析、解决问题的能力，锻炼学生理论联系实际、综合应用的能力。

2. 设计内容：设计用单片机控制的十字路口交通灯及撞红灯报警控制系统，并实现这些功能。

3. 设备和器材

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 HC573 锁存器、1 个 82C55、12 个发光二极管(4 个 LED-RED, 4 个 LED-GREEN, 4 个 LED-YELLOW)，一个 ADC0809、一个电位器 POT-HG、2 个或非门 74LS02，1 个与非门 74LS04，一个三极管 2N1711，3 个电阻 10WATT，一个喇叭 BUZZER、1 组七段数码管 7SEG-MPX2-CA、一个 +5V 稳压电源，根据需要自选其它元件。

4. 设计要求

用红、绿、黄三支共两组发光二极管表示交通信号灯，利用单片机模拟有时间显示的定时交通信号灯控制管理。信号灯的变化规律可如下：

(1) 放行线：绿灯亮放行 25 秒，黄灯亮警告 5 秒，然后红灯亮禁止。

(2) 禁止线：红灯亮禁止 30 秒，然后绿灯亮放行。

(3) 当某一方向的红灯亮时，若该方向有车通过，则用扬声器声报警。(撞红灯信号可用 3-5V 模拟量表示)，同时用 2 位数码管进行 30 秒钟递减时间显示，(1 秒要用定时器产生)。

13.2.2 基于单片机的函数发生器设计和开发

1. 设计目的

利用所学单片机的理论知识进行软硬件整体设计，培养学生分析、解决问题的能力，锻炼学生理论联系实际、综合应用的能力。

2. 设计内容：

以单片机为基础，设计并开发能输出多种波形（正弦波、三角波、锯齿波、脉冲波、梯形波等）且频率、幅度可调的波形发生器。

3. 设备和器材：

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、1 个 DAC0832、1 个 LM324、1 个+12V 稳压电源、1 个-12 稳压电源、16 个按键键盘 BUTTON、4 个 4.7K 电阻 10WATT4K7、1 个+5V 稳压电源、一个示波器，根据需要自选其它元件。

4. 设计要求：

（1）设计接口电路，将这些外设构成一个简单的单片机应用系统，画出接口的连接图。

（2）编写软件实现下列控制：

- 能输出正弦波、三角波、锯齿波、脉冲波、梯形波。
- 能根据键盘命令或开关进行波形切换。
- 能根据键盘命令或开关对输出波形的频率、幅度进行控制调节。

13.2.3 数字温度计设计

1. 设计目的：利用所学单片机的理论知识进行软硬件整体设计，培养学生分析、解决问题的能力，锻炼学生理论联系实际、综合应用的能力。

2. 设计内容：利用数字温度传感器 DS18B20 或 DS1621 与单片机结合来测量温度。

3. 设备和器材：

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、3 组七段数码管 7SEG-MPX2-CA、1 个数字温度传感器 DS1621、或 DS18B20、一个三极管 2N1711，3 个电阻 10WATT，一个喇叭 BUZZER、一个+5V 稳压电源，根据需要自选其它元件。

4. 设计要求：

（1）利用数字温度传感器 DS18B20 或 DS1621 测量温度信号，测量值在数码管或 LCD 显示屏上显示相应的温度值。其温度测量范围为-55℃~-125℃，精确到 0.5℃。

（2）本温度计属于多功能温度仪，可以设置上下报警温度，当温度不在设置范围内时，可以报警。

（3）数字温度计所测量的温度采用数字显示，还可以用串口发送数据到 PC 机并将实时采集温度值在显示器上显示。

13.2.4 简易家电定时控制仪

1. 实验目的

利用所学单片机的理论知识进行软硬件整体设计，培养学生分析、解决问题的能力，锻炼学生理论联系实际、综合应用的能力。

2. 设计内容：在单片机系统上实现对简易家电（至少 2 个）定时控制功能。

3. 设备和器材：

仿真实验：装有 Keil uVision3 以及 Proteus7.0 以上版本软件的笔记本或台式 PC 机。

元件清单：1 个 STC89C52、3 组七段数码管 7SEG-MPX2-CA、6 个发光二极管(2 个 LED-RED, 2 个 LED-GREEN, 2 个 LED-YELLOW)、一个三极管 2N1711, 3 个电阻 10WATT, 一个喇叭 BUZZER、一个+5V 稳压电源, 根据需要自选其它元件。

4. 设计要求

(1) 对家电的启动进行预先定时设置, 能设置并显示预设时间;

(2) 对家电的工作时间预先定时设置, 能设置并显示工作时间的长短;

(3) 设置好家电开启和工作时间后, 绿色发光二极管点亮, 家电启动后, 黄色发光二极管点亮。

(4) 工作时间到了后, 并能声音报警且红色发光二极管点亮。

(5) 该控制仪还可以作为钟表用。

例如:

电饭煲定时控制: 电饭煲做饭(启动)时间为 11: 30, 工作时间为 30 分钟停止。

空调定时控制: 空调打开(启动)时间为 18: 00, 工作时间为 2 小时 30 分停止。

作为钟表: 六个数码管分别显示时、分、秒。

附录 A 单片机程序 ISP 烧录

单片机开发板是通过 PL2303 与 STC89C52 进行连接,其原理图如图 1-1 所示,通过 TXD 与 RXD 与单片机的 TXD 和 RXD 进行连接.

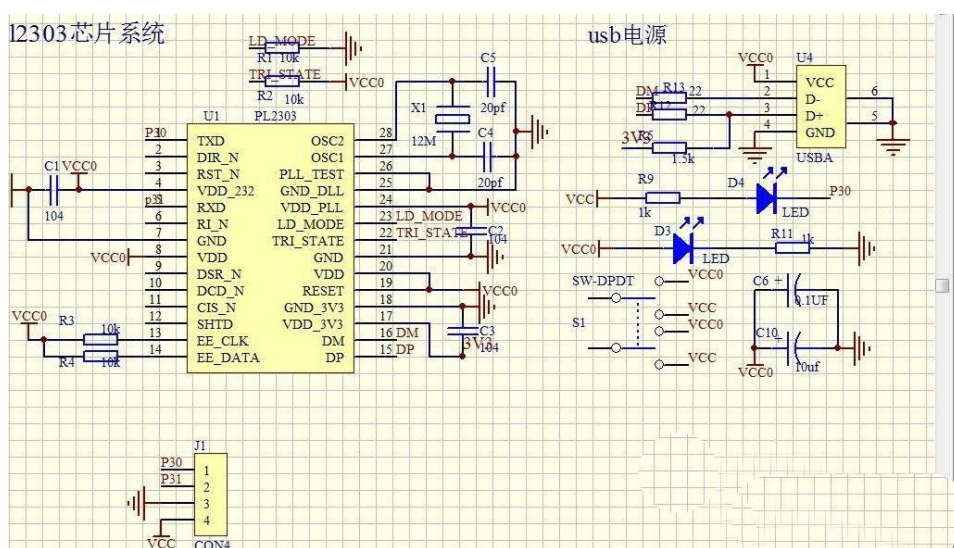


图 1-1 pl2303 与单片机连接原理图

1. 电脑上安装 USB 驱动程序（从宏晶科技官方网站 www.STCMCU.com 查找 PL2303USB 接口驱动程序），如图 1-2 所示。

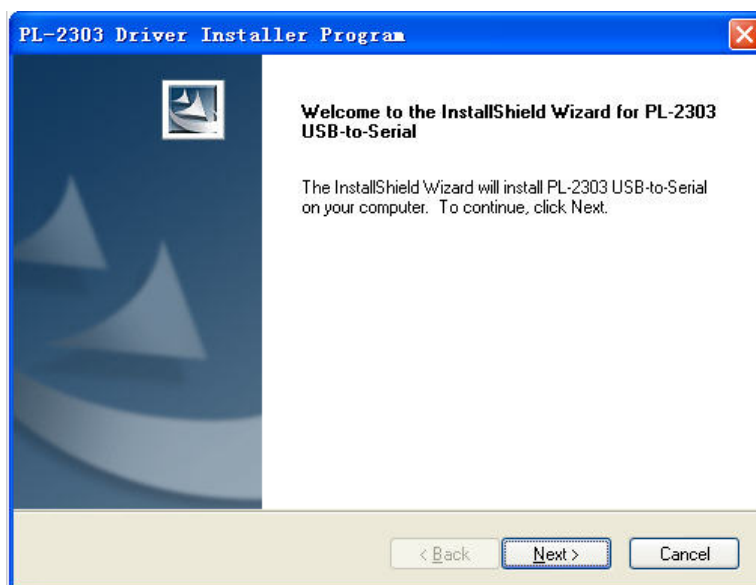


图 1-2 pl2303 驱动软件安装

2. 驱动安装完成后,把开发板连接到电脑上,从电脑设备管理器处检查 USB 的 COM 端口号,如图 1-3 所示。



图 1-3 显示当前端口

3. 打开烧录程序,出现如图 1-4 所示的界面(从宏晶科技官方网站 www.STCMCU.com, 下载 STC-ISP.exe 烧写程序软件)。

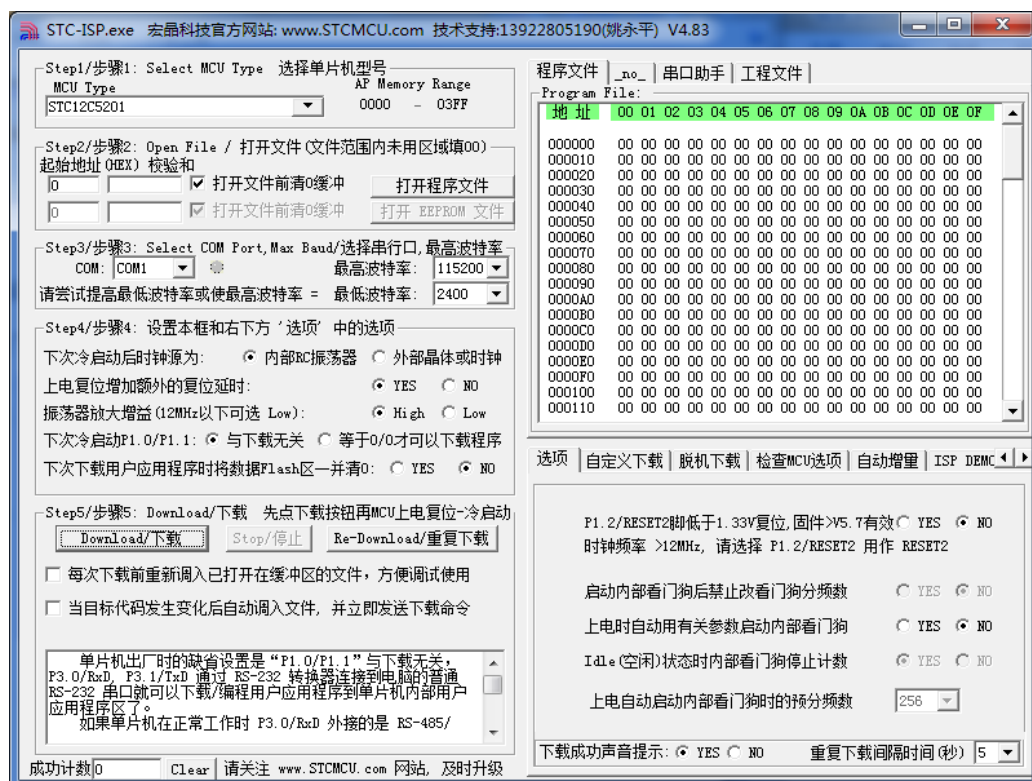


图 1-4 打开程序界面

4. 选择正确的单片机型号(单片机上的第一行字即是型号),左键单击下拉箭头界面如图 1-5,选择 STC89C52。

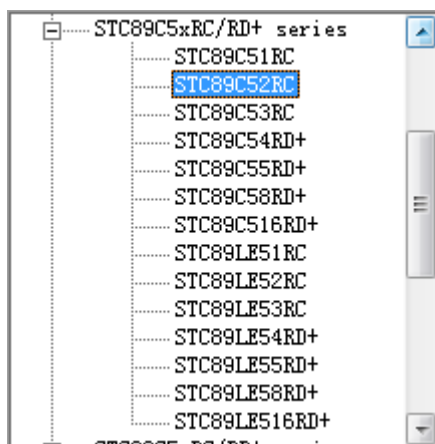


图 1-5 选择端口

5. 选择需要烧到单片机的程序，鼠标左键单击 **Open File**，如图 1-6 所示。

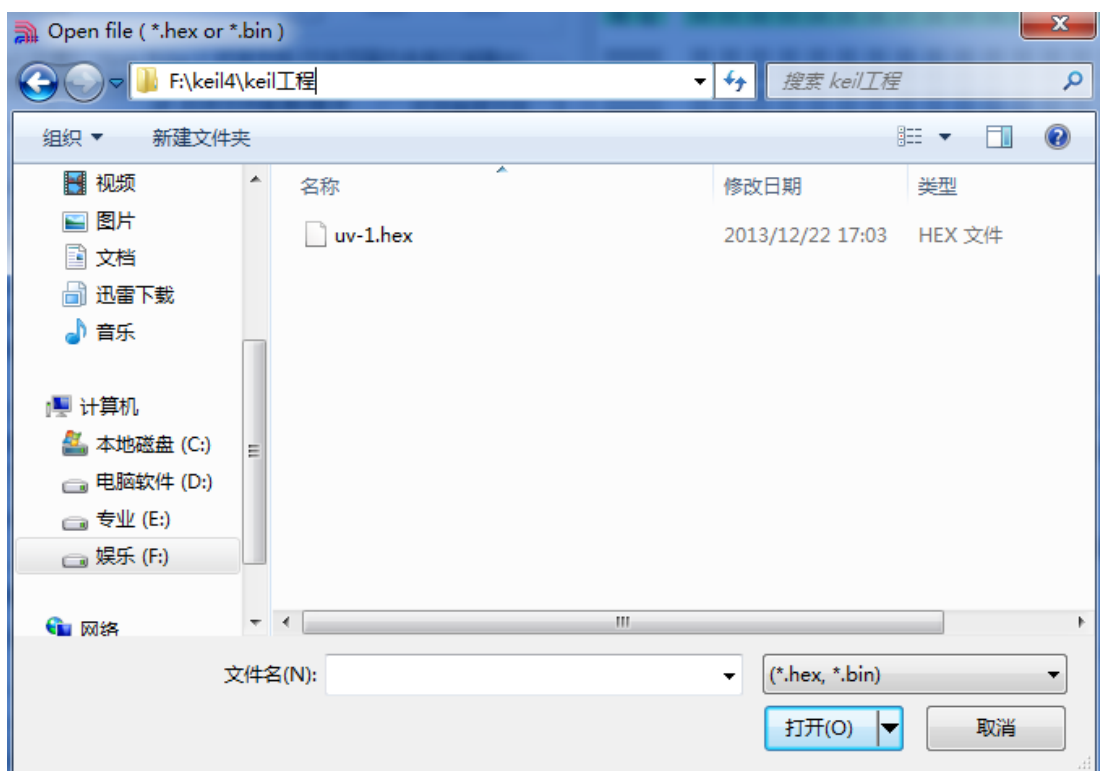


图 1-6 打开烧录文件

6. 选择串口模式，因为开始 USB 的串口为 COM1，顾此时同样选择 COM1 端口，如图 1-7 所示。



图 1-7 选择端口

7. 点击 **Download/下载**，开始下载程序, 要确保此前开发板的电源没有打开，然后给开发板上电, 开发板开始下载程序, 成功下载后出现如图 1-8 所示界面。

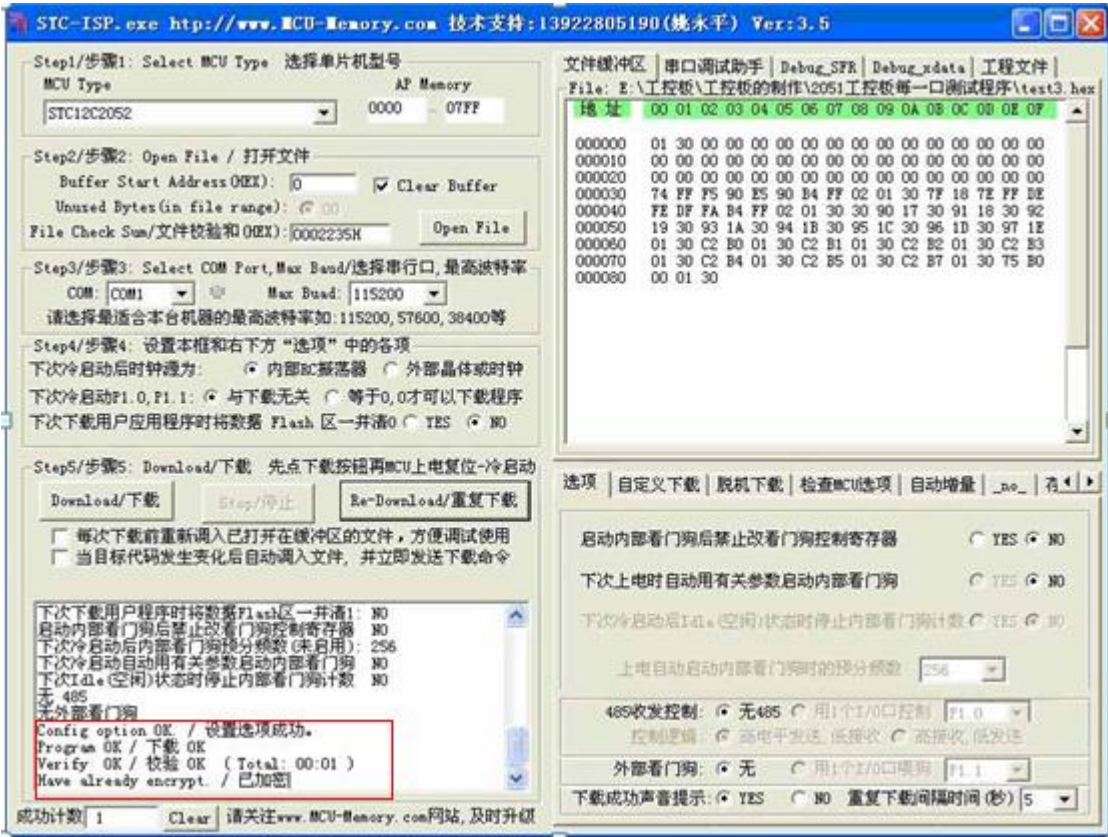


图 1-8 程序下载成功