

SL-MEGA128 高档仿真开发实验器

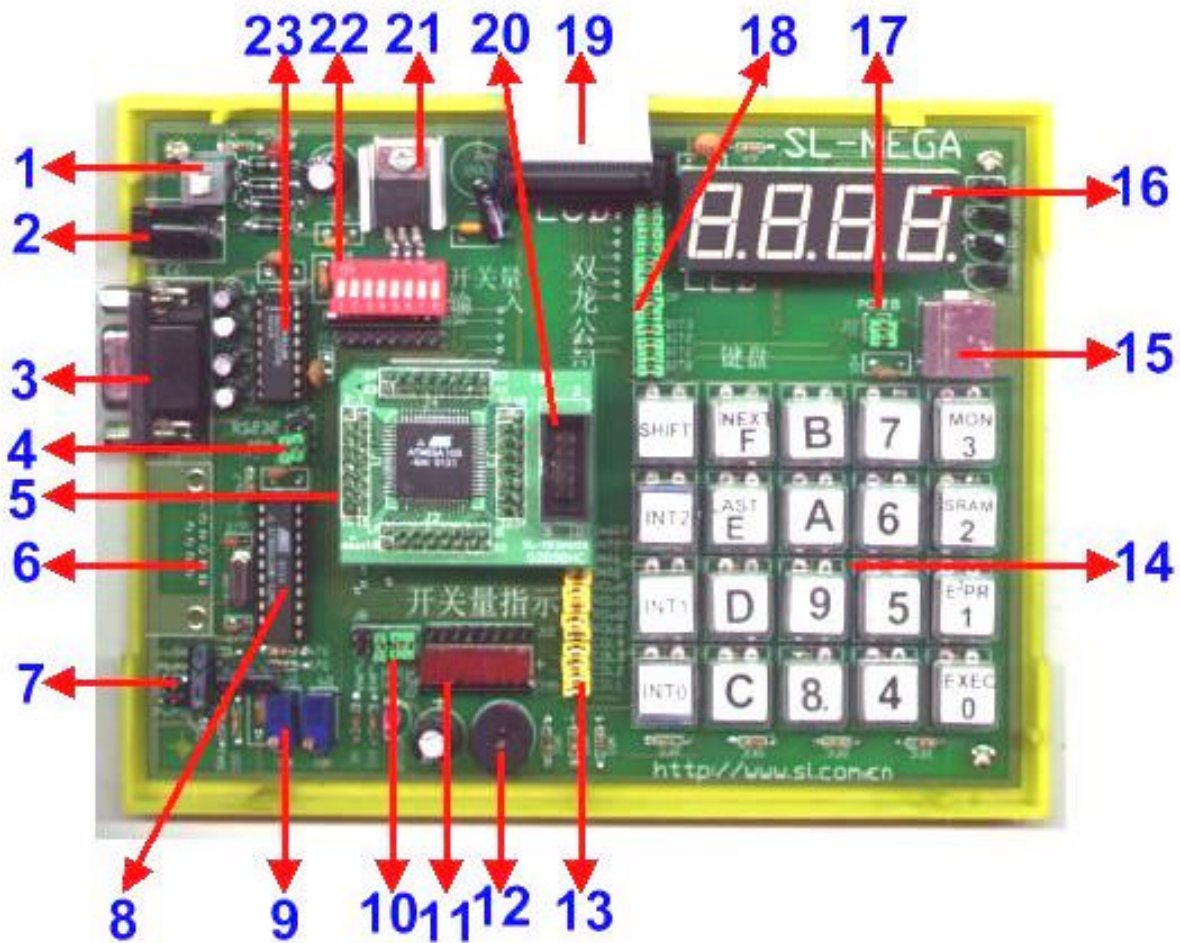
SL-MEGA 高档仿真开发实验器,可对高档 AVR 单片机 AT MEGA 128 进行仿真、ISP/JTAG 下载编程、开发实验,本机也可做科研样机使用。配有主芯片 ATmega128(也可选配 ATmega103)芯片;主芯片的 I/O 口均可外引,供用户做 I/O 口输入/输出实验;有 8 路开关可作输入信号;8 只 LED 发光二极管作输出显示用;有 4 位 LED 数码管;OCMJ4X8 汉字 LCD 液晶显示器,自带 GB2312 16X16 点阵国标一级简体汉字和 ASCII8X8(16X16)英文字库,用户输入区位码或 ASCII 码即可实现文本显示(OCMJ4X8 汉字 LCD 液晶显示器另文详解);有 20 键的键盘;也可接 PC 机通用键盘(购机附通用键盘操作原代码);硬件电路采用模块式结构,用短路块、接插件、专用接线连接;64 脚 TQFP 封装的主芯片(AT mega103/128 单片机),通过转换接口板与主机相连;有双路 RS232 接口,一路为串行 ISP 下载通讯接口,可对单片机实现 ISP 下载编程,另一路为标准 RS232 接口,供用户做通讯实验;ATmega128 配 JTAG 接口(JTAG ICE 仿真器为选购件),解决器件仿真调试、ISP 编程工作;有 2 路 A/D 模拟电压输入接口电路(通过跳线可做全部 8 路的 A/D 实验);2 路 PWM 输出接口电路(通过跳线可做全部 PWM 实验);有音响器;电源模块、电源开关、电源指示、复位电源管理器件,保障复位的可靠性。SL-MEGA128 仿真开发实验器是功能齐全的 AVR 单片机仿真(软件模拟仿真或 JTAG ICE 在线实时仿真)、开发、编程(ISP 下载)、实验工具。使 AVR 单片机开发应用提高到一个崭新水平!

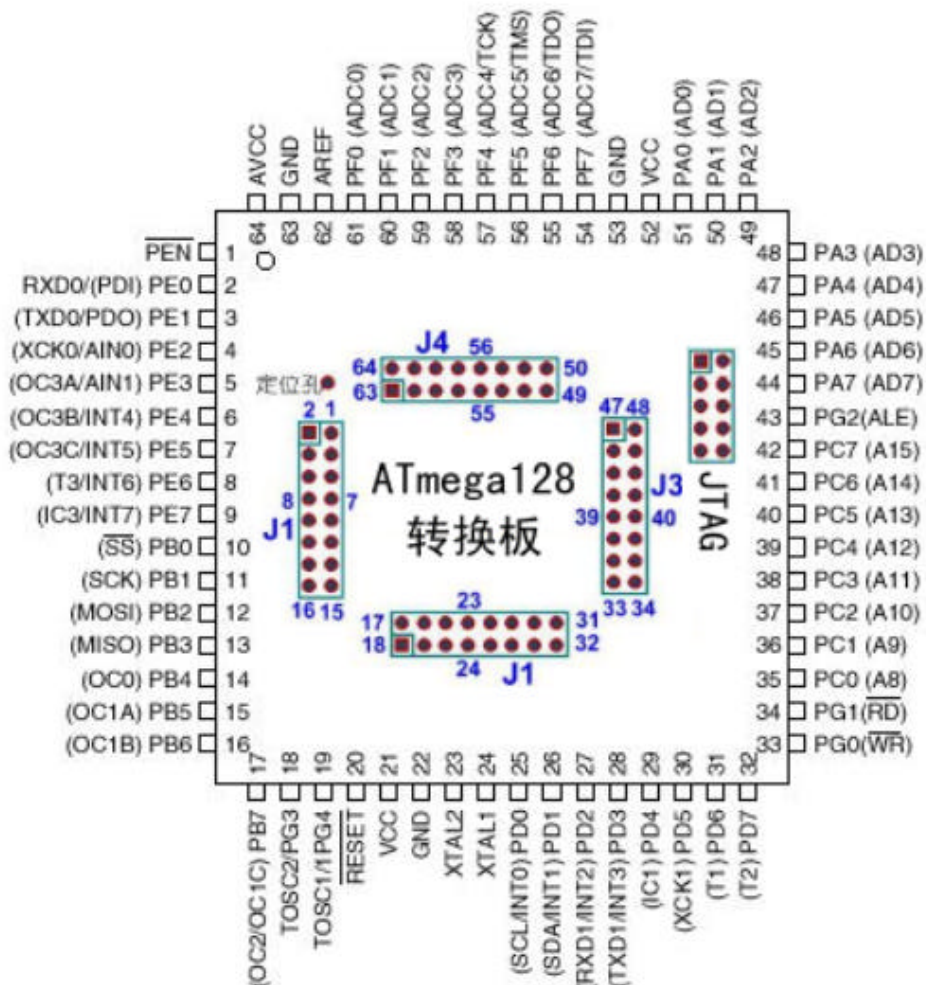
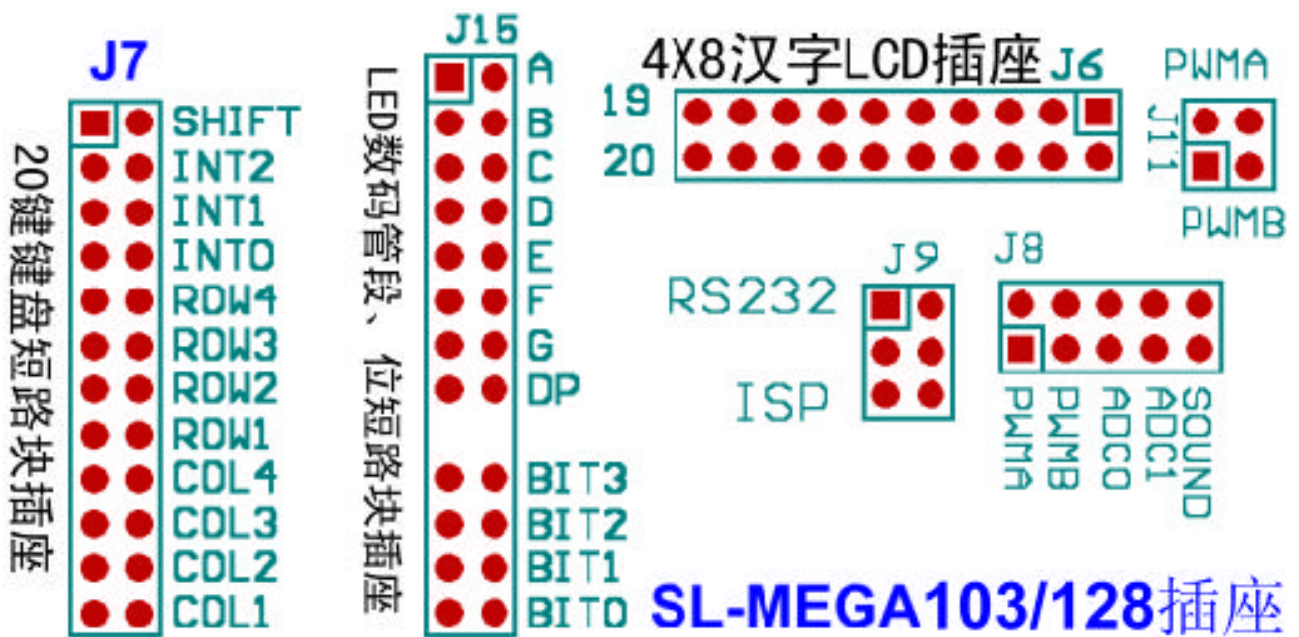


要找
单广
片州
机双
新龙
技电
术子

SL- MEGA 128 开发实验器硬件结构功能部件有:

1. 电源开关(有电源指示)。
 2. POWER 9V 电源插座(附 9V 电源)。
 3. CZ1 ISP 下载通讯插座。
 4. J9 RS232 与 ISP 选择短路块,向上插 RS232 功能,向下插 ISP 功能。
 5. 主芯片 ATmega128 印板座, 器件四周插针可作输入/输出用,用专用接插线连接, 还配有 JTAG ICE 接口插座,解决 ATmega128 器件在线实时仿真、编程工作; 器件四周插针可作输入/输出用,用专用接插线连接。
 6. CZ2 RS232 通讯实验插座。
 7. J11 PWMA/PWMB 输出插座,不插短路块。
 8. ISP 串行下载管理监控。
 9. ADC0/ADC1 模拟电压输入电位器。
 10. J8 PWMA/PWMB/ADC0,ADC1/SOUND(音响器)连接短路块。
 11. 8 位 LED 发二极管,模拟输出用。
 12. 音响器。
 13. J7 连接键盘短路块。
 14. 20 键键盘。
 15. PC 机 PS2 键盘接口。
 16. 4 位 LED 数码管。
 17. J10 PC 机键盘连接短路块(不用不插短路块)。
 18. J15 LED 数码管段、位选短路块。
 19. J6 4X8 汉字 LCD 液晶显示器插座,另一头接液晶显示器。
 20. JTAG ICE 插座。
 21. 7805 三端稳压管。
 22. 8 位 DIP 开关,作模拟输入用。
 23. MAX232 芯片。
- 电源理图见“SLMEGASCH.PDF”文件。





OCMJ4X8:	1. 背光源负 (LED- 0V)	10. DB5
(128*64)	2. 背光源正 (LED+ 5V)	11. DB6
	3. VSS (0V)	12. DB7
	4. VDD (+5V)	13. ASK
	5. DB0	14. ANSWER
	6. DB1	15. RESET
	7. DB2	16. NC
	8. DB3	17. RT1
	9. DB4	18. RT2



MEGALCD.C 测试程序使用说明

通电开机程序下载后:

1. 4X8 汉字 LCD 屏幕显示要求从 SL-MEGA 实验器上小键盘上“输入时间(数据),按 SHIFT 确定”,实时时钟开始运行,实时时钟会正点报时;
2. 接着显示广告语“广州双龙电子有限公司及网址”和 ATMEL 商标图案;
3. 显示菱形、圆动画图案;
4. 显示二路 A/D 转换测量电压,可调多圈电位器,改变输入电压,显示电压随之改变,并且有实时时钟显示;
5. SL-MEGA 实验器上 PC 机键盘接口,应接上 PC 机键盘,要求从 PC 机键盘上输入数字或符号,可用上、下档键操作;

依次 1-5 循环显示!

所有文件档案均在 PC_AT 文件夹中。

SL-AVRAD 及 Studio3.53 的应用

AVR 的开发工具

AVR 的开发工具可分硬件仿真器、软件模拟仿真器(开发实验器),均在 AVR 集成开发环境(IDE)下工作,还有多种 C 高级语言支持。

1 JTGA ICE 仿真器

AVR 高档单片机 ATmega 系列 ATmega16, ATmega32, ATmega323, ATmega64, ATmega128 等具有 JTGA 仿真



图 2 JTGA ICE

Connecting JTAG ICE to Target Board

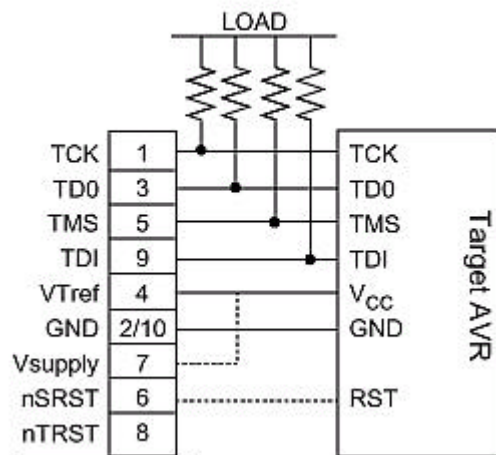


图 3 JTGA ICE 与 AVR 连线

接口, 图 2。可以实现在线实时仿真, 这是在 8 位单片机中属于领先技术。JTGA 仿真, 对不同封装器件可以先焊接, 然后进行实时仿真、编程, 这比用传统的芯片仿真头, 大大降低仿真器价格。

2 AVR 集成开发环境(IDE)

ATMEL AVR Studio3.53 集成开发环境(IDE), 包括: 1.AVR Assembler 编译器; 2.AVR Studio 调试功能; 3.AVR Prog 串行、并行下载功能; 4.JTGA ICE 仿真等功能。AVR IDE 安装后, 双击 AVR Studio 图标, 则出现

AVR Studio3.53集成开发环境窗口, 图 8。

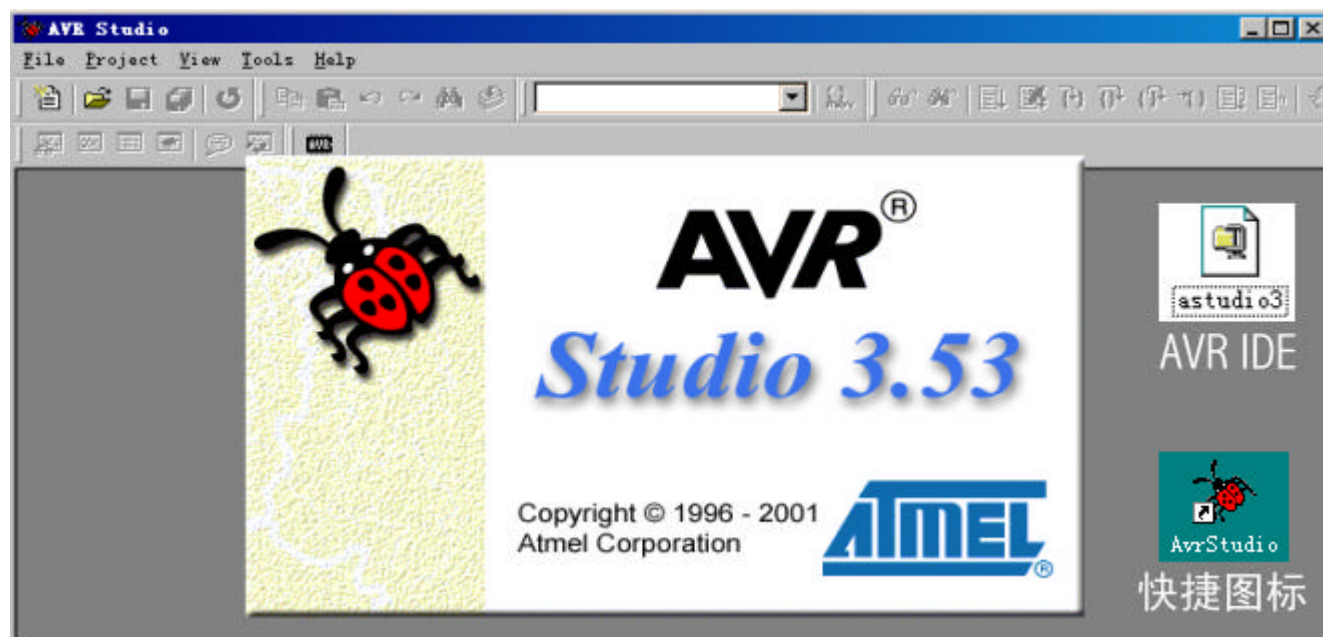


图 8

2.1 AVR Assembler编译器

有源文件编辑、汇编(生成.OBJ/.HEX/.LIS文件)、搜寻、选项(生成汇编文件格式)、窗口、帮助等操作,汇编出错有错误定位、错误指示,便于源文件排错。

1. 建立工程项目:

- ① Project→New→出现Select new project窗口,新建工程项目; ②必须选择工程项目名字和项目的类型

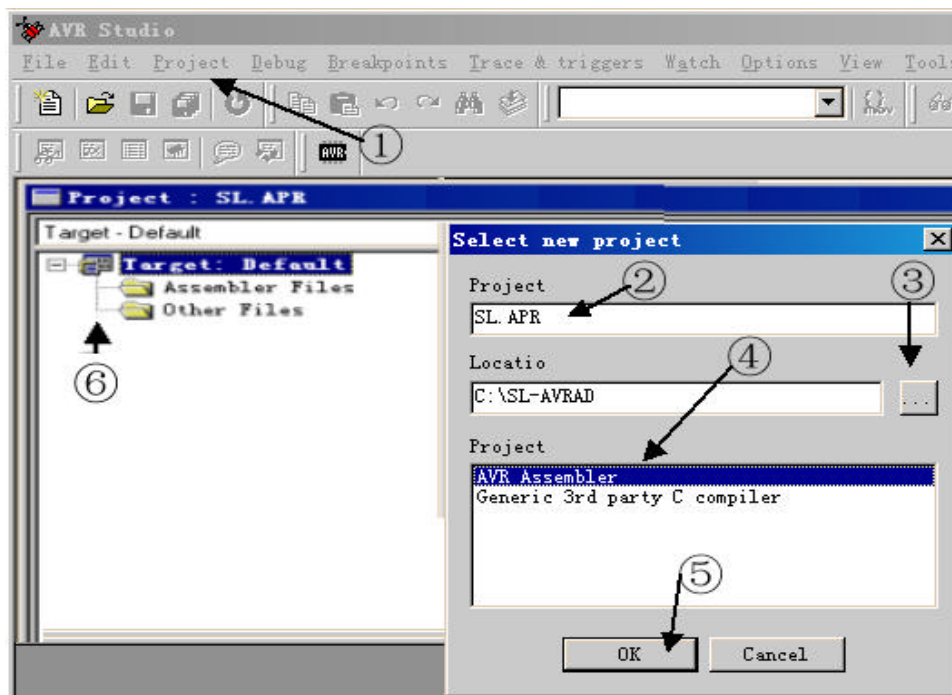


图 9

输入,例:SL.APR工程项目(.APR也可缺省,则默认为.APR); ③选择存放工程项目路径; ④用鼠标选中AVR Assembler汇编; ⑤再键OK; ⑥即自动新建工程项目, 图 9。

2.打开已保存的工程项目

File→Open 选择路径,打开已存在的工程项目。

3.新建汇编文件名:

- ① File→New text file→出现新建文件窗口Create new file; ②输入汇编文件名,例:SL.ASM; ③选择存放路径; ④键OK键; ⑤即源文件添加到工程项目中,并出现新的源文件编辑窗口,可编辑新的源程序, 图 10。

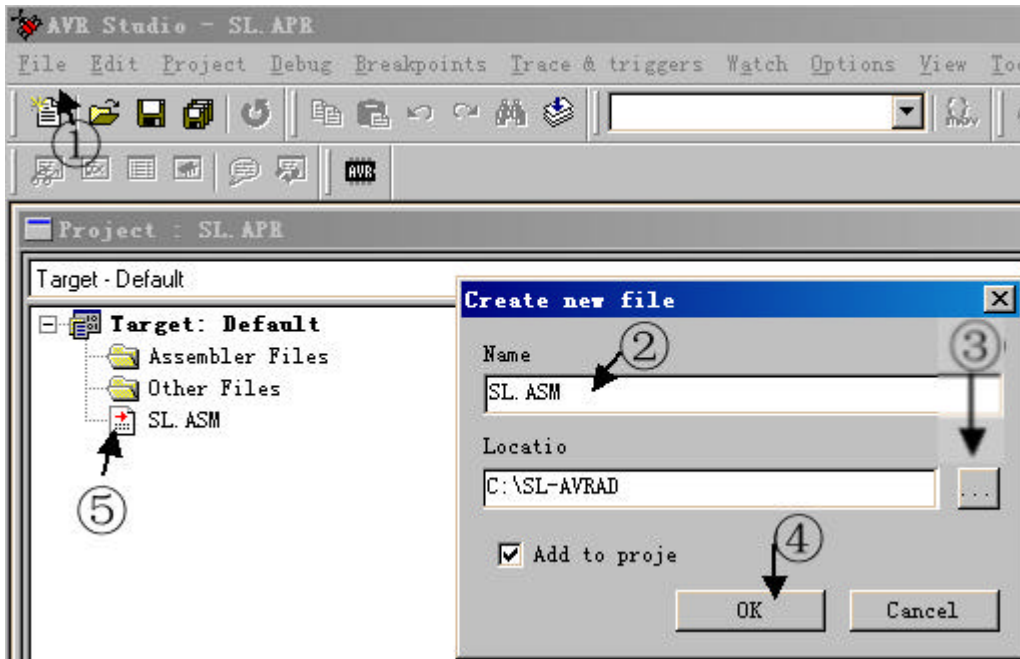


图 10

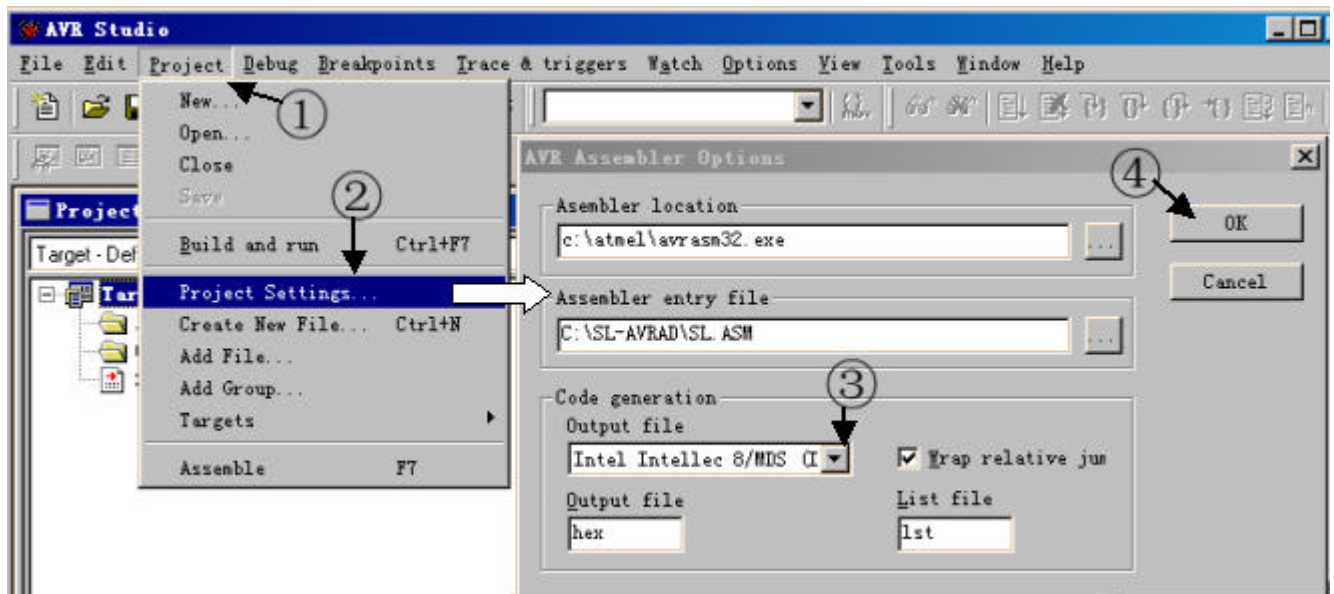
4. 打开已保存的汇编文件

File→Open 选择路径,打开已存在的汇编文件。

5. 源文件编译选项

①选Project菜单; ②选Project Settings编译项目设置;出现AVR Assembler Options选择窗口;③选Output file中Intel Intellec8/MDS.生成Intel格式hex文件; ④键OK键设置完成, 图 11。

图 11



6. 新文件的编辑与编译

① 窗口输入源文件SL.ASM; ②选Project菜单; ③选Project下拉菜单的Assemble编译选项, 进行编译; ④编译通过,显示编译提示, 图 12。

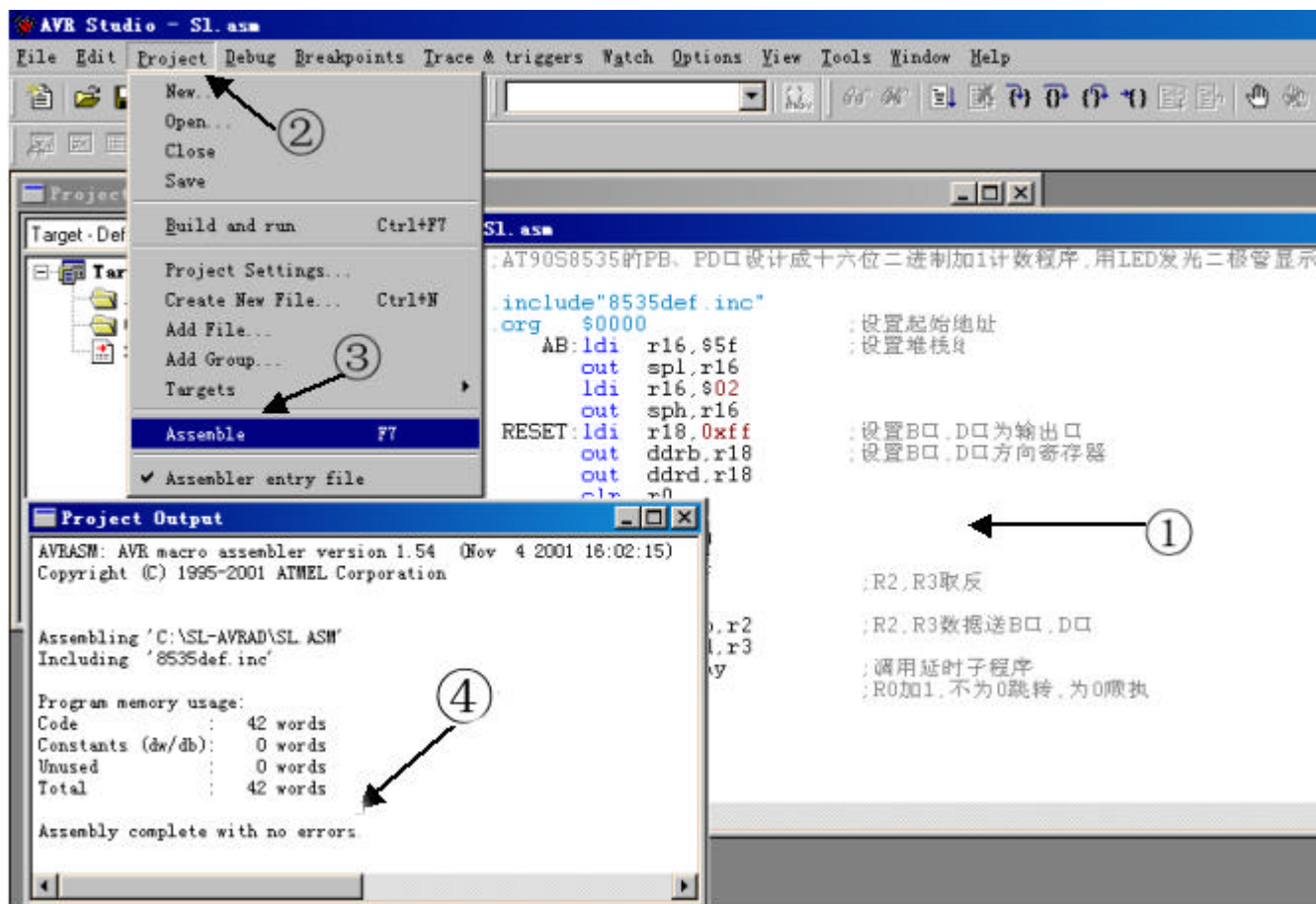


图 12

2.2 AVR Studio

对源文件 DEBUG 调试(装入.OBJ 目标文件,以源文件格式显示调试;如装入.HEX 文件,以反汇编格式显示:地址、机器码、指令等格式显示调试)排错、断点、单步、自动单步、触发、注视、选项、查看、窗口、帮助等操作;调试中可打开多种窗口,有 I/O 窗口,源文件窗口,CPU 窗口,记录窗口,数据窗口等,见“AVR 集成软件调试窗口”图示。

选菜单 Debug→G0 进入 Debug 调试选项窗口;①源文件调试窗口;②打开 Processor 观察窗口,可观察 CPU 的各种参数变化情况;③打开 Standard 观察窗口,可观察 I/O 口电平变化情况;④ Debug 调试快捷按钮。⑤观察窗口快捷按钮,根据源程序调试要求,还可打开更多的观察窗口,图 13。

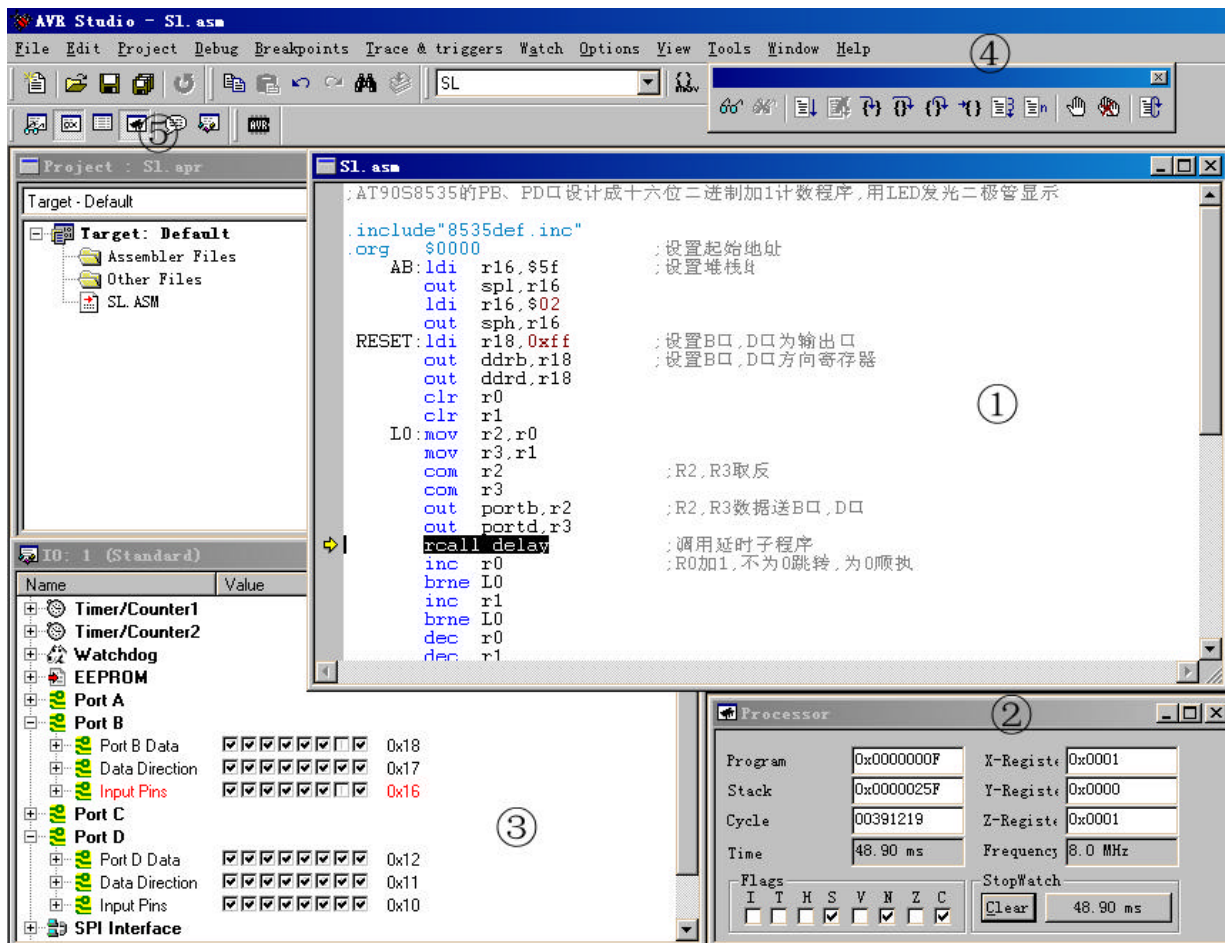


图 13

2. 打开已有文件进入调试

① 选择菜单 File; ② 选择下拉菜单 Open; ③ 在一定路径下找到已有文件,文件类型可以是*.obj; *.d90; *.cof; *.hex;也可以是源文件,工程项目文件等,图 14.14。

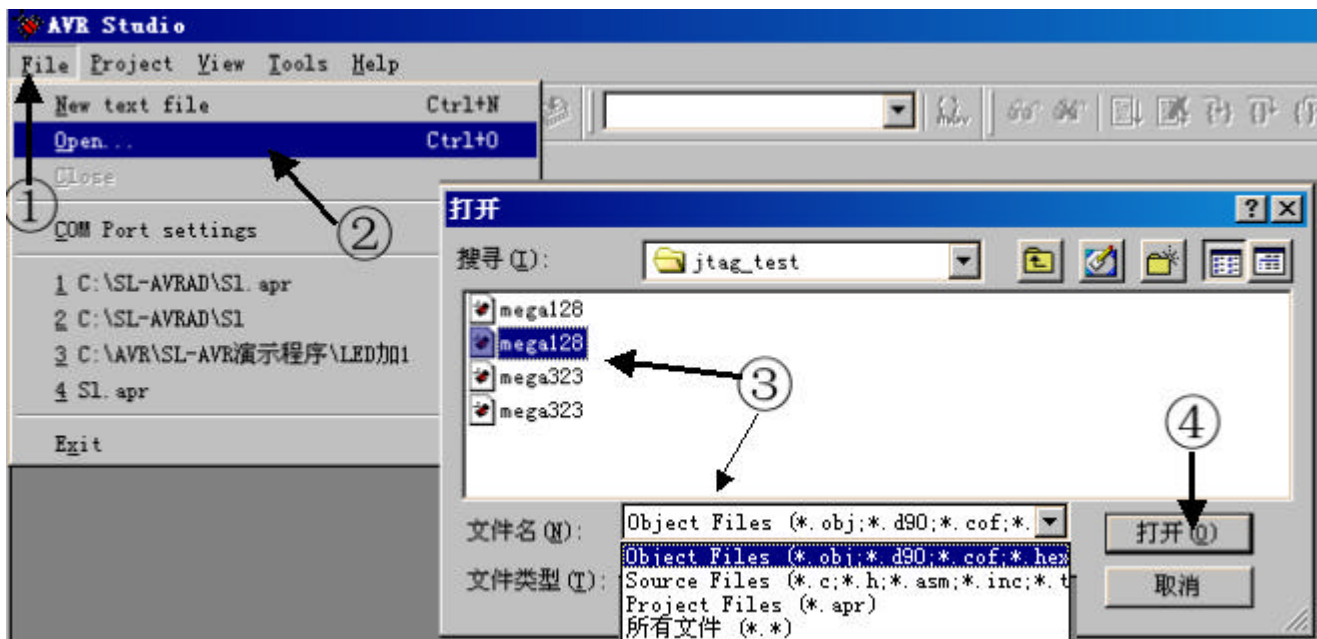


图 14.14

2.3 AVR Prog

1. JTGA ICE 下载编程：一头接 SL-MEGA128 开发实验器 JTGA ICE 下载插座,并接上 9VDC 电源。启动 AVR Studio3.53 程序,如果对芯片编程,可以启动 T00LS 菜单中的 STK500/AVR ISP/JTAG ICE 或(Alt+9)按钮。如图 15,根据窗口提示操作。

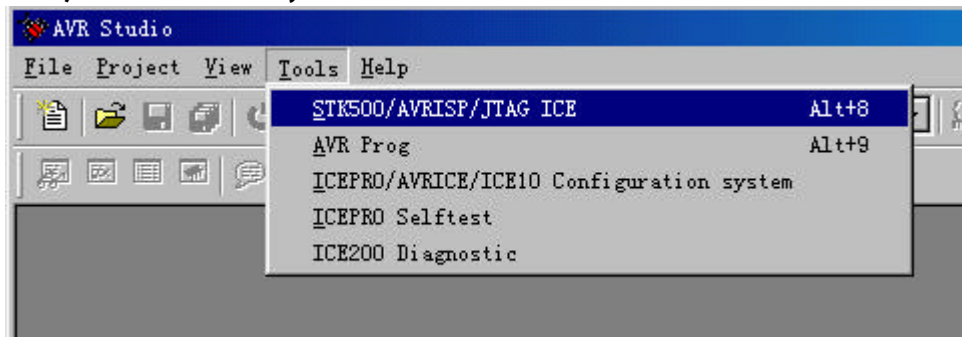


图 15 进入 AVR ISP

2. AVR Prog 串行编程：串行下载软件,当用户接通下载线,一头接 PC 机 RS232 串行口,另一头接 SL-MEGA128 开发实验器 ISP 插座,并接上 9VDC 电源。启动 AVR Studio3.53 程序,如果对芯片编程,可以启动 T00LS 菜单中的 AVR Prog 或(Alt+9)按钮。如图 15,根据窗口提示操作。

2.4. 源文件说明：

大量 AVR 应用实验源程序及编译、调试、下载工作软件可上网下载:WWW.SL.COM.CN
光盘中提供多种源文件：

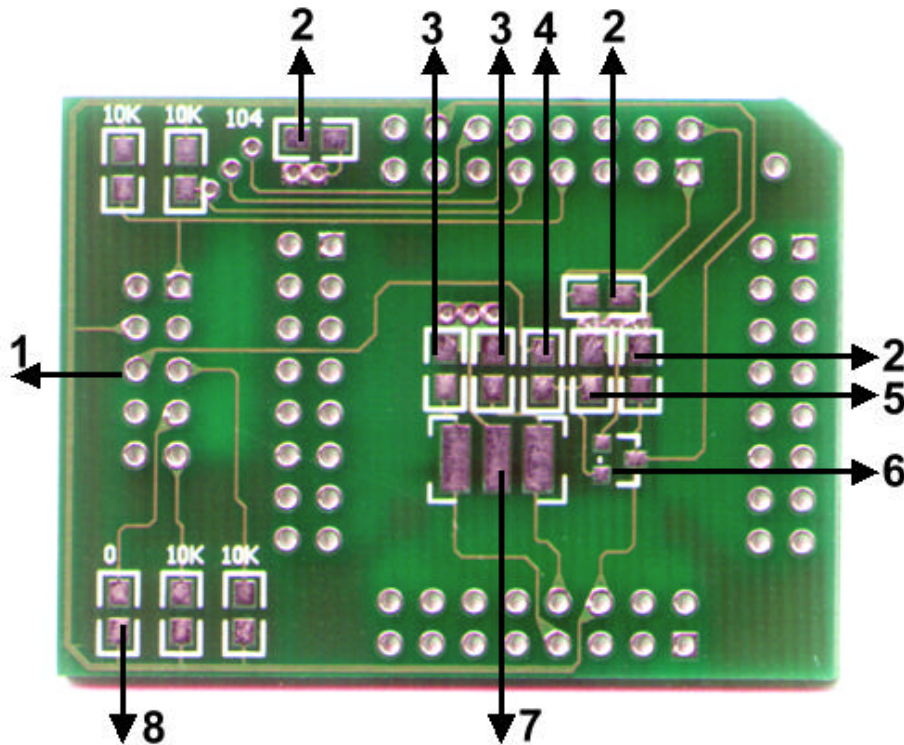
- (1) ATMEL 公司提供的见光盘文件*:\AVR\AVR\asmpack\appnotes\AVR*.ASM(实验程序);
- (2) 双龙公司提供的见光盘文件*:\AVR\AVR\SL-AVR*.ASM(实验应用程序,也适用 SL-AVR+);
- (3) 双龙公司提供的见光盘文件*:\AVR\AVR\指令 ASM*.ASM(指令实验调试程序);
- (4) 双龙公司提供的见光盘文件*:\AVR\AVR\SL-AVRAD\程汇*.ASM(实验程序);
- (5) 双龙公司提供的见光盘文件*:\AVR\AVR\SL-AVRAD*.ASM(实验程序);
- (6) 双龙公司提供的见光盘文件*:\AVR\AVR\SL-AVR+*.ASM(实验程序);

注意:光盘文件拷到你计算机里是只读属性,只有去掉只读属性才能进行修改、运行等操作。

SL-MEGA128 转换座印制板安装说明

SL-MEGA128 转换座印制板是为用户便于 ATmega128 应用系统的引脚转换、开发实验、ISP 下载、JTAG 仿真调试而设计。

现把元件安装面(ATmega128 芯片印制板反面),有关元件安装说明如下:

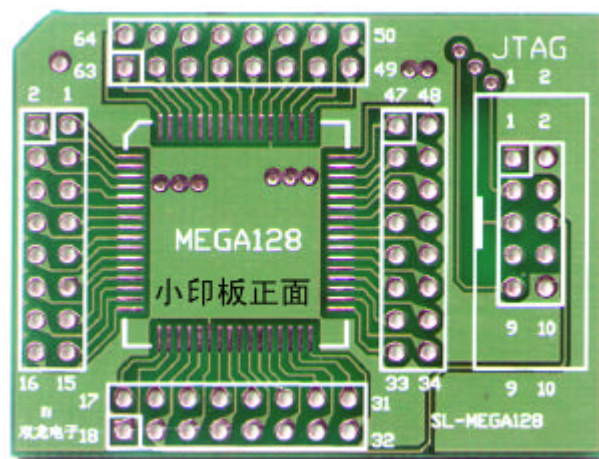


如果 SL-MEGA128 底板已焊晶振, (4)4.7K 表贴电阻、(2)104 表贴电容必须焊接。

当(5)100K 电阻配(6)809 复位电压检测电路时才焊接(本印板也适用于 ATmega103,ATmega128 本身有复位电压检测功能)。

如小印制板要单独工作,或 SL-MEGA128 底板未焊接晶振,则必须(7)焊接表贴 8MHZ 晶振及(3)瓷片 27P 表贴电容,上拉表贴电阻 10KX4、。

如果(1)JTAG ICE 仿真器要用 SL-MEGA128 实验器电源,则(8)0 电阻处应用导线短接。



SL-MEGA 实验测试程序如下:

用 AVR 单片机 ATmega103/128 与 OCMJ4X8 液晶模块做实验

演示源程序 MEGALCD.C

```
/******  
http://www.sl.com.cn  
双龙电子公司----中文 LCD 演示程序(ICCAVR)  
作者: 詹卫前  
1、实验设备: SL-MEGA  
2、数据线 D0~D7 接 PORTA  
ASK=PORTE.6  
ANSWER=PORTE.7  
RESET=PORTC.3  
*****/  
//extern unsigned char kb_buffer[64];  
#include <iom128.h>  
#include <macros.h>  
/* 定义 OCMJ4X8 端口 */  
//DATA_PORT=PORTA  
#define data_port PORTA  
//ASK=PORTE.6  
#define ask (PINE&0x40)  
//ANSWER=PORTE.7  
#define set_answer asm("sbi 0x03,7")  
#define clr_answer asm("cbi 0x03,7")  
//RETE=PORTC.3  
#define set_reset asm("sbi 0x15,3")  
#define clr_reset asm("cbi 0x15,3")  
#define wait asm("nop\n nop")  
/*-- 双龙公司图标数据 --*/  
/*-- 宽度 x 高度=32x32 --*/  
const unsigned char sl_ico[128]={  
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFE,0x7F,0xFF,0xFF,0xF8,0x3F,0xFF,0xFF,0xE0,0x1F,  
0xFF,0xFF,0x80,0x1F,0xFF,0xFF,0x00,0x3F,0xFF,0xFE,0x01,0xFF,0xFF,0xFC,0x07,0xFF,  
0xFF,0xF0,0x1F,0xFF,0xFF,0xF0,0x3F,0xFF,0xFF,0xF0,0x3F,0xFF,0xFF,0xF0,0x7F,0xFF,  
0xFF,0xE0,0x7F,0xFF,0xFF,0xF0,0x3F,0xFF,0xFF,0xF0,0x1F,0xFF,0xFF,0xF8,0x0F,0xFF,  
0xFF,0xFC,0x0F,0xFF,0xFF,0xFC,0x07,0xFF,0xFF,0xFE,0x03,0xFF,0xFF,0xFF,0x03,0xFF,  
0xFF,0xFF,0x03,0xFF,0xFF,0xFF,0x01,0xFF,0xFF,0xFF,0x81,0xFF,0xFF,0xFF,0x01,0xFF,  
0xFF,0xFF,0x03,0xFF,0xFF,0xFE,0x03,0xFF,0xFF,0xF8,0x03,0xFF,0xFF,0xE0,0x07,0xFF,  
0xFE,0x00,0x2F,0xFF,0xFF,0xC0,0x3F,0xFF,0xFF,0xFD,0xFF,0xFF,0xFF,0xFF,0xFF};  
/*-- ATMEL 位图文件数据 --*/
```

```
/*-- 宽度 x 高度=104x49 --*/
const unsigned char atmel_bmp[637]={
0x00,0x00,0x0F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x00,0x0F,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x00,0x00,0x1F,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0x00,0x00,0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x03,0xFC,0x7F,0x9F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0x9F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFE,0x3F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xE7,0xF8,0x00,0x00,0x0F,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,
0x00,0x0F,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x00,0x1F,0xFC,
0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x00,0x1F,0xFC,0x7F,0x9F,0xC7,
0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x00,0x3F,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,
0x00,0x07,0xF8,0x00,0x00,0x3F,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0x00,0x07,0xF8,
0x00,0x00,0x7F,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x00,0x7F,
0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x00,0xFF,0xFC,0x7F,0x9F,
0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x00,0xFF,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,
0x3F,0xFF,0xE7,0xF8,0x00,0x01,0xFF,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,
0xF8,0x00,0x01,0xFF,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x03,
0xFF,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x03,0xFF,0xFC,0x7F,
0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,0x07,0xFB,0xFC,0x7F,0x9F,0xC7,0xF8,
0xFE,0x3F,0x00,0x07,0xF8,0x00,0x07,0xF3,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0x00,
0x07,0xF8,0x00,0x0F,0xF3,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xF8,0x00,
0x1F,0xE3,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xFF,0xF8,0x1F,0xE3,0xFC,
0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xFF,0xF0,0x3F,0xC3,0xFC,0x7F,0x9F,0xC7,
0xF8,0xFE,0x3F,0xFF,0xE7,0xFF,0xF0,0x3F,0xC3,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,
0xFF,0xE7,0xFF,0xE0,0x7F,0xF3,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xFF,
0xE0,0x7F,0xF3,0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xFF,0xC0,0xFF,0xF3,
0xFC,0x7F,0x9F,0xC7,0xF8,0xFE,0x3F,0xFF,0xE7,0xFF,0xC0,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xF8,0x00,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00};
/*-- 文字: 和 --*/
/*-- 楷体_GB231248; 此字体下对应的点阵为: 宽 x 高=64x64 --*/
```

```

const unsigned char hz1[480]={
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x0E,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0F,0x80,0x00,0x00,0x00,
0x00,0x00,0x00,0x1F,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x3F,0xC0,0x00,0x00,0x00,
0x00,0x00,0x00,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0xFC,0x00,0x00,0x00,0x00,
0x00,0x00,0x0F,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x7F,0x00,0x00,0x00,0x00,0x00,
0x00,0x03,0xFF,0x80,0x00,0x00,0x00,0x00,0x00,0x1F,0x87,0xC0,0x00,0x00,0x00,0x00,
0x00,0x18,0x07,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0xC0,0x00,0x00,0x00,0x00,
0x00,0x00,0x03,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x80,0x00,0x00,0x00,0x00,
0x00,0x00,0x03,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x80,0x00,0x00,0x00,0x00,
0x00,0x00,0x03,0x87,0xC0,0x00,0x00,0x00,0x00,0x00,0x03,0xFF,0xE0,0x00,0x06,0x00,
0x00,0x00,0x0F,0xFF,0xC0,0x00,0x1F,0x80,0x00,0x00,0xFF,0xF8,0x00,0x01,0xFF,0xE0,
0x00,0x0F,0xFF,0x80,0x7F,0xFF,0xFF,0xF0,0x07,0xFF,0xE7,0x80,0x7F,0xFF,0xFF,0xF0,
0x1F,0xFE,0x0F,0x80,0x7F,0x80,0x07,0xE0,0x0F,0xF0,0x1F,0x80,0x3E,0x00,0x07,0xC0,
0x07,0x80,0x3F,0x80,0x1E,0x00,0x07,0xC0,0x00,0x00,0x3F,0x80,0x1E,0x00,0x07,0x80,
0x00,0x00,0x7F,0x80,0x1E,0x00,0x07,0x80,0x00,0x00,0xFB,0x80,0x0E,0x00,0x07,0x80,
0x00,0x00,0xF3,0xF8,0x0E,0x00,0x0F,0x00,0x00,0x01,0xE3,0xFE,0x0E,0x00,0x0F,0x00,
0x00,0x03,0xE3,0xBF,0x0E,0x00,0x0F,0x00,0x00,0x07,0xC3,0x9F,0x07,0x00,0x0E,0x00,
0x00,0x07,0x83,0x8F,0x07,0x00,0x0E,0x00,0x00,0x0F,0x03,0x83,0x07,0x00,0x1F,0x00,
0x00,0x1E,0x03,0x80,0x07,0x1F,0xFF,0x80,0x00,0x3C,0x03,0x80,0x07,0xFF,0xFF,0x80,
0x00,0x78,0x03,0x80,0x07,0xFF,0x00,0x00,0x00,0x70,0x03,0x80,0x03,0x00,0x00,0x00,
0x00,0xE0,0x03,0x80,0x03,0x00,0x00,0x00,0x01,0xC0,0x03,0x80,0x03,0x00,0x00,0x00,
0x03,0x80,0x07,0x80,0x03,0x00,0x00,0x00,0x06,0x00,0x07,0x80,0x01,0x00,0x00,0x00,
0x0C,0x00,0x07,0x80,0x00,0x00,0x00,0x00,0x10,0x00,0x07,0x80,0x00,0x00,0x00,0x00,
0x00,0x00,0x07,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00};
const unsigned char hz2[132]={
/*-- 文字：欢 --*/
/*-- 楷体_GB231224; 此字体下对应的点阵为：宽 x 高=32x33 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x00,0x00,0x00,0x70,0x00,
0x00,0x00,0x70,0x00,0x00,0x00,0x60,0x00,0x00,0x00,0x60,0x00,0x00,0x00,0xC0,0x00,
0x00,0x00,0xC0,0x00,0x00,0x60,0x81,0x00,0x03,0xF1,0x8F,0xC0,0x3E,0x71,0xF9,0xE0,
0x10,0x63,0x01,0x80,0x00,0x62,0x03,0x00,0x08,0x64,0xC2,0x00,0x04,0x60,0x60,0x00,
0x06,0xC0,0x60,0x00,0x03,0xC0,0x60,0x00,0x01,0xC0,0x60,0x00,0x01,0xC0,0xD0,0x00,
0x01,0x60,0xD8,0x00,0x03,0x70,0xCC,0x00,0x06,0x31,0x8C,0x00,0x04,0x11,0x86,0x00,
0x08,0x03,0x03,0x80,0x10,0x06,0x03,0xC0,0x20,0x0C,0x01,0xF0,0x00,0x18,0x00,0xFC,
0x00,0x20,0x00,0x70,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00};
const unsigned char hz3[132]={

```



```

/*-- 文字: 迎 --*/
/*-- 楷体_GB231224; 此字体下对应的点阵为: 宽 x 高=32x33 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x60,0x00,0x00,0x00,0x60,0x00,0x06,0x00,0xC0,0x00,0x07,0x01,0x80,0x00,
0x03,0x82,0x01,0xE0,0x01,0x1C,0x3E,0xE0,0x00,0x08,0x10,0x60,0x00,0x0C,0x10,0x60,
0x00,0x0C,0x10,0xC0,0x07,0x0C,0x10,0xC0,0x3F,0x8C,0x50,0xC0,0x23,0x0D,0x93,0xC0,
0x03,0x0F,0x11,0xC0,0x02,0x0E,0x11,0x80,0x02,0x0C,0x10,0x00,0x02,0x08,0x10,0x00,
0x03,0x00,0x10,0x00,0x01,0x00,0x10,0x00,0x01,0x00,0x10,0x00,0x1F,0xC0,0x10,0x00,
0x3C,0x3E,0x10,0x00,0x00,0x07,0xE0,0x00,0x00,0x01,0xFF,0xFE,0x00,0x00,0x3F,0xF8,
0x00,0x00,0x07,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00};
const unsigned char hz4[132]={
/*-- 文字: 您 --*/
/*-- 楷体_GB231224; 此字体下对应的点阵为: 宽 x 高=32x33 --*/
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x00,0x00,0x30,0x60,0x00,
0x00,0x30,0x60,0x00,0x00,0x70,0xE0,0x00,0x00,0x60,0xC1,0x80,0x00,0xC1,0xBF,0xE0,

0x00,0xC3,0xE0,0xE0,0x01,0x82,0x31,0x00,0x03,0xC4,0x30,0x00,0x06,0xC8,0x10,0x00,
0x0C,0xC0,0x10,0x00,0x08,0xC2,0x13,0x00,0x10,0x82,0x11,0xC0,0x00,0x86,0x10,0xC0,
0x00,0x86,0x30,0x40,0x00,0x84,0xB0,0x00,0x00,0x80,0x70,0x00,0x00,0x80,0x30,0x00,
0x00,0x00,0x20,0xE0,0x00,0x07,0x00,0x78,0x02,0x03,0xC0,0x18,0x02,0x20,0xC0,0x00,
0x06,0x30,0x04,0x00,0x06,0x18,0x02,0x00,0x0C,0x0C,0x03,0x80,0x0C,0x07,0x03,0xC0,
0x00,0x03,0xFF,0xC0,0x00,0x00,0x7F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00};
unsigned int adc1;
unsigned int adc2;
unsigned char adcval[6];
unsigned char keytime;
/*      AD 转换程序      */
#pragma interrupt_handler adc_isr:IT_ADC
void adc_isr(void)
{
    unsigned int temp=0;
    while(temp<6)
        temp++;
    temp=ADC;
    switch (ADMUX)
    {
        case 0x00:
        {
            adc1=temp;
            ADMUX=1;
            break;
        }
    }
}

```

```
        case 0x01:
        {
            adc2=temp;
            ADMUX=0;
            break;
        }
        default :
        {
            ADMUX=0;
        }
    }
    ADCSR|=0x40;
}
/*    AD 结果转换成电压    */
#define vref    473
unsigned char * adctoasc(unsigned int adc)
{
    unsigned int temp;
    unsigned char i;
    unsigned char *char_p;
    temp=(unsigned int)((((unsigned long)((unsigned long)adc*vref))/1024);
    for(i=0;i<3;i++)
    {
        adcval[3-i]=temp%10+0x30;
        temp=temp/10;
    }
    adcval[0]=adcval[1];
    adcval[1]='.';
    adcval[4]='V';
    adcval[5]=0;
    char_p=adcval;
    return char_p;
}
/*    OCMJ4X8LCD 初始化    */
void ocmj_init(void)
{
    MCUCR=0;

    DDRA=0xff;
    PORTA=0xff;
    DDRE=0x80;
    PORTE=0xff;
    PORTC=0xff;
    DDRD=0x00;
```

```
    PORTD=0xff;
    clr_reset;//LCD 复位
    delay_ms(10);
    set_reset;
    clr_answer;
    delay_ms(10);
}

/*      写数据到 LCD      */
void ocmj_write(unsigned char data)
{
    while(ask!=0)
        ;
    data_port=data;
    wait;
    set_answer;
    wait;
    while(ask==0)
        ;
    clr_answer;
}

/*      传送 32*32IC0 点阵数据到 LCD      */
/*      x:0~15(字节为单位)  y:0~64      */
/*      ico_p:指向图形数据的指针      */
void ico_tran(unsigned char x,unsigned char y,const unsigned char *ico_p)
{
    unsigned char i,j;
    x+=4;
    for(j=0;j<32;j++)
    {
        for(i=0;i<4;i++)
        {
            ocmj_write(0xf3);
            ocmj_write(x);
            ocmj_write(y);
            ocmj_write(*ico_p);
            ico_p++;
            x++;
        }
        x-=4;
        y++;
    }
}

/*      清除屏幕区域      */
void cls(unsigned x,unsigned y,unsigned char high,unsigned char width)
```

```

{
    unsigned char i,j;
    x+=2;
    for(j=0;j<high;j++)
    {
        for(i=0;i<width;i++)
        {
            ocmj_write(0xf0);
            ocmj_write(x);
            ocmj_write(y);
            ocmj_write(55);
            ocmj_write(90);
            x++;
        }
        x-=width;
        y++;
    }
}

/*      传送 bmp 点阵数据到 LCD      */
/*      x:0~15(字节为单位)  y:0~64      */
/*      bmp_p:指向图形数据的指针      */
/*      high:位图高度(点阵行为单位)      */
/*      width:位图宽度(以字节为单位)** */
void bmp_tran(unsigned char x,unsigned char y,unsigned char high,unsigned char width,const unsigned char
*bmp_p)
{
    unsigned char i,j;
    x+=4;
    for(j=0;j<high;j++)
    {
        for(i=0;i<width;i++)
        {
            ocmj_write(0xf3);
            ocmj_write(x);
            ocmj_write(y);
            ocmj_write(*bmp_p);
            bmp_p++;
            x++;
        }
        x-=width;
        y++;
    }
}

/*      传送 LCD 内部汉字数据到 LCD      */

```



```
/*      x:0x00~0x07 y:0x00~0x03      */
void hz_tran(unsigned char x,unsigned char y,unsigned char *hz_p)
{
    x+=2;
    while((*hz_p)!=0)
    {
        ocmj_write(0xf0);
        ocmj_write(x);
        ocmj_write(y);
        ocmj_write(*hz_p-0xa0);
        hz_p++;
        ocmj_write(*hz_p-0xa0);
        hz_p++;
        if(x<0x09)
            x++;
        else
        {
            x=0x02;
            y++;
        }
    }
}

/*      传送 ASCII 字母到 LCD      */
/*      x:0x00~0x0f y:0~64      */
void asc_tran(unsigned char x,unsigned char y,unsigned char *asc_p)
{
    x+=4;
    while((*asc_p)!=0)
    {
        ocmj_write(0xf1);
        ocmj_write(x);
        ocmj_write(y);
        ocmj_write(*asc_p);
        asc_p++;
        if (x<0x13)
            x++;
        else
        {
            x=0x04;
            y+=8;
        }
    }
}

/*      画点函数      */
```

```
void pset(unsigned char x,unsigned char y)
{
    ocmj_write(0xf2);
    ocmj_write(x+32);
    ocmj_write(y);
}

/*          画线函数          */
void line(unsigned char x1,unsigned char y1,unsigned char x2,unsigned char y2)
{
    unsigned char t;
    signed int xerr=0,yerr=0;
    signed int delta_x,delta_y,distance;
    signed char incx,incy;

/* 计算两个方向的长度 */
    delta_x=x2-x1;
    delta_y=y2-y1;

/* 计算增量的方向,增量为"0"表示为垂直或水平线 */
    if(delta_x>0)
        incx=1;
    else
    {
        if( delta_x==0 )
            incx=0;
        else
        {
            incx=-1;
            delta_x =-delta_x;
        }
    }
    if(delta_y>0)
        incy=1;
    else
    {
        if( delta_y==0 )
            incy=0;
        else
        {
            incy=-1;
            delta_y =-delta_y;
        }
    }
/* 确定画线的范围 */
```

```
if( delta_x > delta_y ) distance=delta_x;
else distance=delta_y;

/* 画线 */
for(t=0;t<= distance+1;t++)
{
    ocmj_write(0xf2);
    ocmj_write(x1+32);
    ocmj_write(y1);
    xerr += delta_x ;
    yerr += delta_y ;
    if( xerr > distance )
    {
        xerr-=distance;
        x1+=incx;
    }
    if( yerr > distance )
    {
        yerr-=distance;
        y1+=incy;
    }
}
}

/*          画圆函数          */
void circle(unsigned char x0,unsigned char y0,unsigned char r)
{
    unsigned char x,y;
    unsigned int xx,rr,xt,yt,rs;
    yt=r;
    rr=r*r+1;          //补偿 1 修正方形
    rs=yt*3/4;          //画 8 分之一圆弧
    for (xt=0;xt<=rs;xt++)
    {
        xx=xt*xt;
        while ((yt*yt)>(rr-xx))
            yt--;
        x=x0+xt;    //第一象限
        y=y0-yt;
        pset(x,y);
        x=x0-xt;    //第二象限
        pset(x,y);
        y=y0+yt;    //第三象限
        pset(x,y);
        x=x0+xt;    //第四象限
```

```
    pset(x,y);
/*****45 度镜像画另外 8 分之一圆弧*****/
    x=x0+yt;    //第一象限
    y=y0-xt;
    pset(x,y);
    x=x0-yt;    //第二象限
    pset(x,y);
    y=y0+xt;    //第三象限
    pset(x,y);
    x=x0+yt;    //第四象限
    pset(x,y);
}
}
void putchar(unsigned char newx,unsigned char newy,unsigned char data)
{
    static unsigned char x,y;
    if(newx<16)
        x=newx+4;
    if(newy<64)
        y=newy;
    ocmj_write(0xf1);
    ocmj_write(x);
    ocmj_write(y);
    ocmj_write(data);
    if (x<0x13)
        x++;
    else
    {
        x=0x04;
        if(y<56)
            y+=8;
        else
        {
            ocmj_write(0xf4);
            y=0;
        }
    }
}
/*      MAIN 入口      */
void main(void)
{
    unsigned char i;
    CLI();
    XDIV = 0x00;
```

```
MCUCR = 0x00;
EICR = 0x00; //external interrupts sources
EIMSK = 0x00;
ocmj_init();//初始化
timer0_init();
timer2_init();
init_kb();
TIMSK = 0x01; //timer interrupt sources
EIMSK=0x10;
adc1=0;
adc2=0;
ADMUX=0x00;
ADCSR=0xdd;
timecal();
SEI();
DDRB=0x80;
while(1)
{
    ocmj_write(0xf4);//清屏
    /*debug
    hz_tran(1,0,"测试扫描码");
    while(1)
    {
        cls(0,1,3,8);
        asc_tran(0,16,kb_buffer);
        delay_ms(100);
    }
    debug*/
    ico_tran(0,0,sl_ico);//传送图标,也可用 bmp_tran() 传送
    hz_tran(3,0,"广州双龙");//传送内部汉字
    hz_tran(2,1,"电子有限公司");
    asc_tran(5,39,"HTTP://");//传送 ASCII 字符
    asc_tran(2,50,"WWW.SL.COM.CN");
    delay_ms(2000);//延时
    ocmj_write(0xf4);//清屏
    bmp_tran(4,2,60,8,hz1);//传送汉字"和"
    delay_ms(3000);
    ocmj_write(0xf4);
    bmp_tran(1,3,49,13,atmel_bmp);//传送 ATMEL 标志图形
    asc_tran(2,54,"WWW.ATMEL.COM");
    delay_ms(3000);
    ocmj_write(0xf4);
    bmp_tran(2,15,33,4,hz2);//传送汉字"欢"
    bmp_tran(6,15,33,4,hz3);//传送汉字"迎"
```



```
    bmp_tran(10,15,33,4,hz4);//传送汉字"您"
    hz_tran(0,0,"恭");
hz_tran(0,1,"祝");
    hz_tran(0,2,"各");
    hz_tran(0,3,"位");
    hz_tran(7,0,"万");
hz_tran(7,1,"事");
    hz_tran(7,2,"如");
    hz_tran(7,3,"意");
    delay_ms(3000);
    for(i=0;i<32;i++)
    {
        ocmj_write(0xf4);
        line(i*2,32,64,i);
        line(64,i,(127-i*2),32);
        line((127-i*2),32,64,(63-i));
        line(64,(63-i),i*2,32);
        delay_ms(100);
    }
    for(i=0;i<32;i++)
    {
        ocmj_write(0xf4);
        circle(63,31,i);
        delay_ms(100);
    }
    ocmj_write(0xf4);
    hz_tran(2,0,"测量电压");
    hz_tran(1,1,"第一路");
    hz_tran(1,2,"第二路");
    for(i=0;i<100;i++)
    {
        cls(4,1,2,3);
        asc_tran(8,22,adctoasc(adcl));
        asc_tran(8,38,adctoasc(adcl));
        timetoasc();
        delay_ms(200);
    }
    ocmj_write(0xf4);
    asc_tran(2,5,"PC");
    hz_tran(2,0,"机键盘输入");
    hz_tran(0,1,"按");
    asc_tran(2,21,"DEL");
    hz_tran(3,1,"键重新启动");
    cls(0,2,2,8);
```

```
putchar(0,32,':');  
putchar(99,99,'>');  
keytime=0;  
do{  
    i=getchar();  
    if(i!=0)  
        putchar(99,99,i);  
}  
while(keytime<20);  
}  
}
```

广州天河双龙电子有限公司
(WWW.SL.COM.CN)

广州双龙:广州天河龙口西路龙苑大厦 A3 座新赛格电子城 331 室(510630)

电话:020-87578852 传真:020-87505012

北京双龙:北京海淀知春路 132 号中发大厦 616 室(100086)

电话:010-82623551、62653785 传真:010-82623550

上海双龙:上海北京东路 668 号赛格市场 1D06 室(200031)

电话:021-53081501 传真:021-53081502