

计算机原理与应用实验指导书

张芨 高平 编

2016 年

目录

第一章 实验箱使用要求.....	3
第二章 TPC-ZK-II 实验系统硬件环境.....	4
2.1 实验系统结构.....	4
2.2 实验系统说明.....	4
第三章 TPC-ZK-II 集成软件开发环境.....	10
实验一 8255A 可编程并行接口实验.....	16
实验二 使用 8259 中断实现定时/计数器实验.....	22
实验三 使用 8251A 的串行接口应用实验.....	27
实验四 A/D 和 D/A 转换实验.....	31
实验五 存储器读写实验.....	37
实验六 DMA 传送实验.....	40
实验七 步进电机控制实验.....	44
实验八 双色点阵发光二极管显示实验.....	46

第一章实验箱使用要求

在实验箱上实验时，连接实验导线一定要关闭实验箱电源，连接好后，在检查一遍是否有误，连接正确后，再依次打开实验箱电源开关及实验底板上电源开关，进行实验。

- 1) 严禁带电操作；
- 2) 实验箱底板上的所有芯片出厂时经过严格测试和老化，实验时切忌插拔；
- 3) 带上理论课教材、实验指导书及准备的实验程序；
- 4) 调试实验程序，记录实验结果；
- 5) 实验结束后，关闭实验底板上的电源开关及实验箱电源开关，确认关闭后，整理好自己的实验导线，摆放老师交代的地方。

第二章 TPC-ZK-II 实验系统硬件环境

2.1 实验箱系统结构

TPC-ZK-II 实验系统依托 PC 主机 CPU 微处理器，主机与实验系统用 USB 通线相连，通过 USB 口通信完成微机接口硬件实验，具有安全性高，实验方便，通用性强的特点。

- 1) 实验系统结构设计上考虑全面，使其能够自由设计实验，可扩展性强，不仅能够完成所有的基本实验内容，还可以用来进行课程设计，创新型实验；此系统对于实验教学来说是一个及有利的工具。
- 2) 接口实验模块：并行通信接口 8255、定时/计数器接口 8254、串行通信接口 8251、中断控制器接口 8259、DMA 控制器接口 8237、存储器接口 6264、模数转换接口 0809、数模转换接口 0832、简单并行通信接口输入输出 244/273、8 位 8 段数码管显示电路、8 位 LED 显示电路、8 位逻辑电平带灯开关、8X8 双色点阵，COM 口 232 电路、图形液晶 12864LCD 模块、继电器控制模块、步进电机带测速模块、直流电机带测速模块、两路正负单脉冲模块、带计数/脉冲/高低电平/TTL 与 COMS 切换的逻辑笔模块、直流信号输入模块、麦克风模块，蜂鸣器模块、带功放喇叭模块、4X4 键盘模块、时钟源/逻辑门/触发器模块、系统总线驱动/开放/扩展实验接口等。
- 3) 实验系统配有集成软件，具有编辑、编译、链接和调试；可以查看实验说明、实验原理图/接线、实验程序进行演示；可以增加和删除自定义实验项目。
- 4) 实验程序可以是宏汇编也可以是 C 语言，可以对汇编程序和 C 语言程序进行调试。
- 5) 开放式模块化设计，实验台有两个扩展接口（一个 PC104ISA 接口，一个 20 芯接口）及独立一块扩展实验区，非常方便用户进行课程设计和二次扩展模块开发实验，扩展性强，教师学生在实验设计时受限少。
- 6) 提供扩展实验模块：8279 键盘显示控制器实验模块、红外遥控收发实验模块、无线通信收发实验模块、温湿度传感器实验模块、FPGA 实验模块、红外热释/压力/光敏/声控传感器实验模块等。
- 7) 实验系统自备电源，具有电源短路保护确保系统安全。实验时采用自锁紧单股导线及排线，连接可靠便捷。



2.2 实验箱硬件系统结构

TPC-ZK-II 实验系统依托 PC 主机 CPU 微处理器，主机与实验系统用 USB 通线相连，通过 USB 口通信完成微机接口硬件实验。

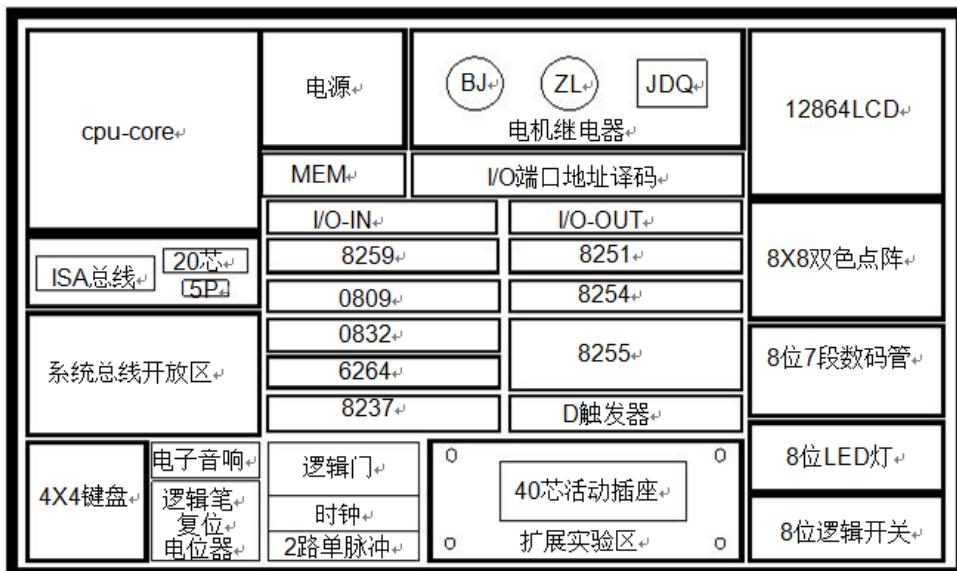
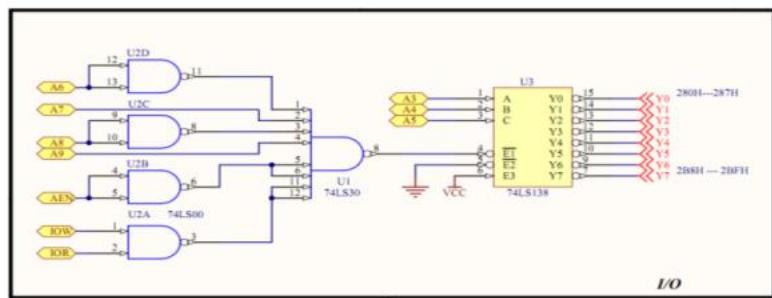


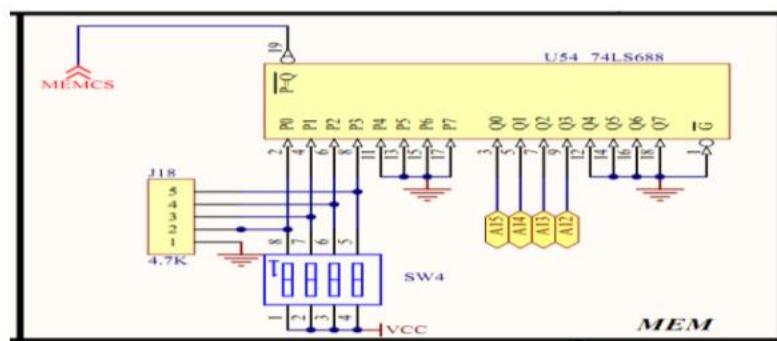
图 2-1 实验系统硬件结构分布图

包括：并行通信接口 8255、定时/计数器接口 8254、串行通信接口 8251、中断控制器接口 8259、DMA 控制器接口 8237、存储器接口 6264、模数转换接口 0809、数模转换接口 0832、简单并行通信接口输入输出 244/273、8 位 8 段数码管显示电路、8 位 LED 显示电路、8 位逻辑电平带灯开关、8X8 双色点阵，COM 口 232 电路、图形液晶 12864LCD 模块、继电

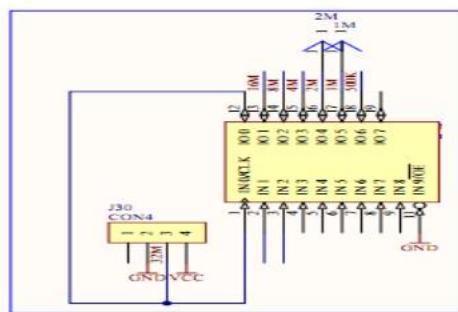
器控制模块、步进电机带测速模块、直流电机带测速模块、两路正负单脉冲模块、带计数/



脉冲/高低电平/TTL 与 COMS 切换的逻辑笔模块、直流信号输入模块、麦克风模块，蜂鸣器



模块、带功放喇叭模块、4X4 键盘模块、时钟源/逻辑门/触发器模块、系统总线驱动/开放/



扩展实验接口等。

图 2-2 实验系统 I/O 端口地址译码电路

图 2-3 实验系统内存地址译码电路

图 2-4 时钟电路

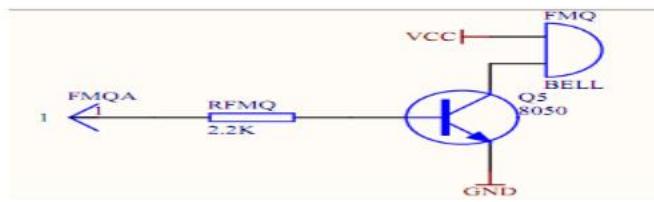


图 2-5 单脉冲电路

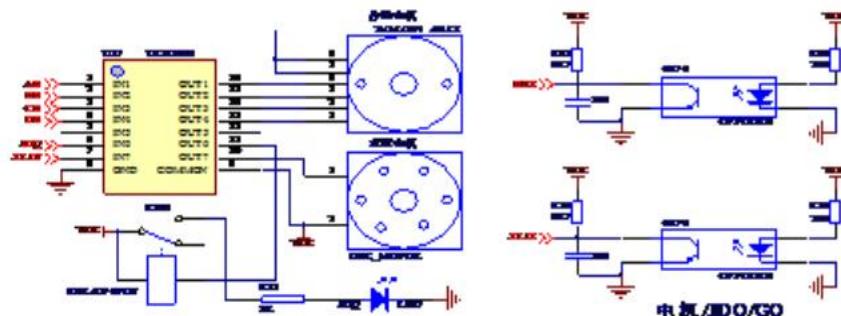
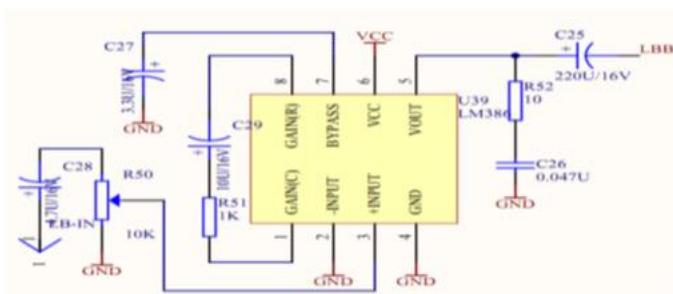
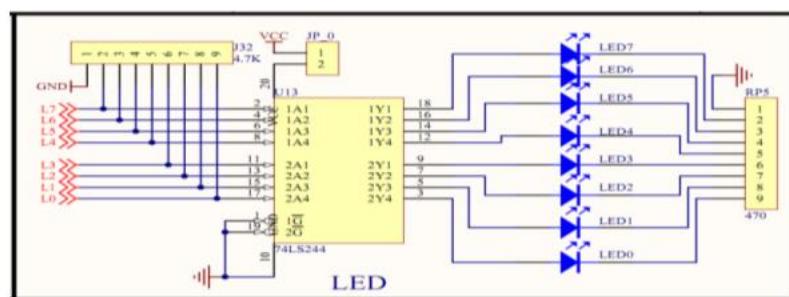
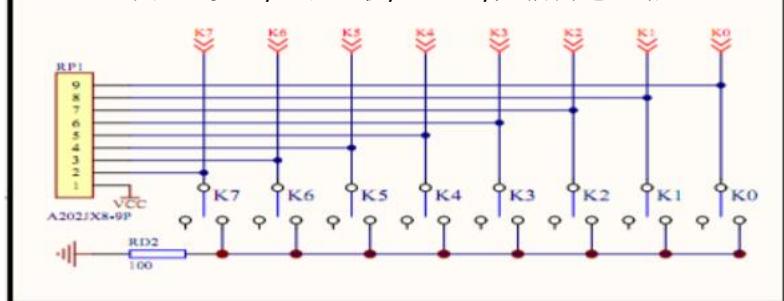


图 2-6 蜂鸣器电路

图 2-7 带功放电子音响电路

图 2-8 步进/直流电机/继电器/光耦测速电路



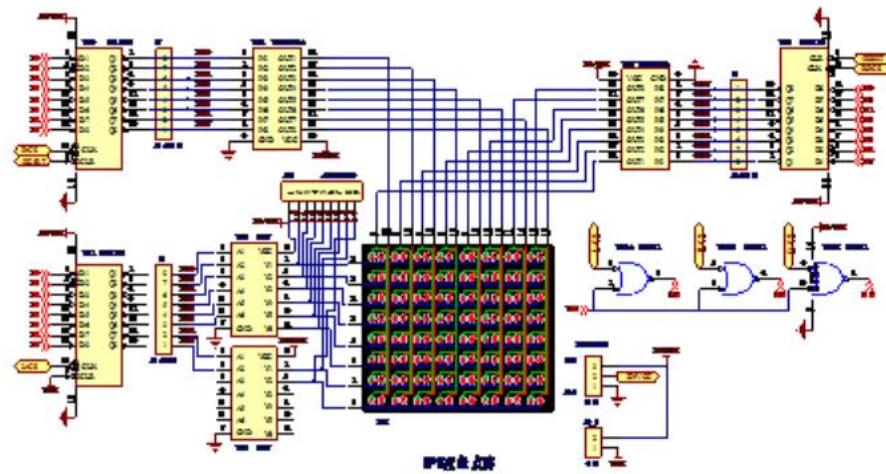


图 2-9 逻辑电平开关电路

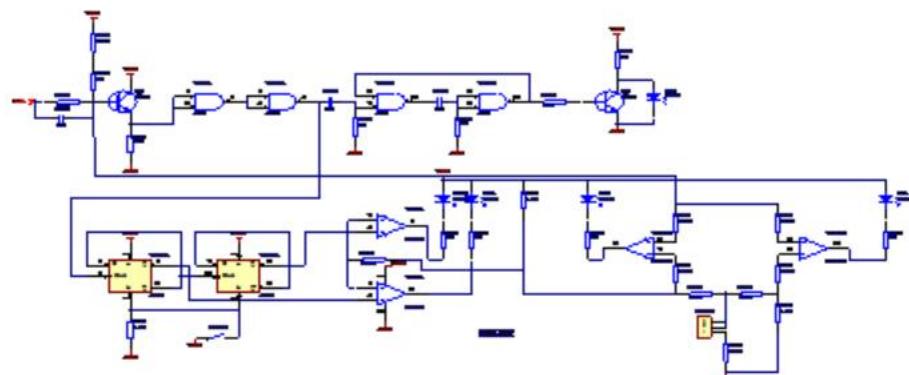


图 2-10 LED 显示电路

图 2-11 双色点阵电路

图 2-12 逻辑电平开关电路

图 2-13 LED 显示电路

实验台上有一个 20 芯双排插座 JX1，用于外接附加的键盘显示实验板和其它用户开发的

实验板。JX1各引脚信号安排如下：

2	4	6	8	10	12	14	16	18	20
GND	GND	1MHz	A1	A0	IOW	IOR	+5V	+5V	RESET
1	3	5	7	9	11	13	15	17	19
CS=2B0H	IRQ	D7	D6	D5	D4	D3	D2	D1	D0

驱动安装完成后，打开TPC-USB集成开发环境（集成开发环境的安装请参看3.2节‘集成开发环境的安装’），选定主菜单“选项”中的“硬件检测”，集成开发环境会检测到设备已连接，如果连接不正确，将会有错误提示。

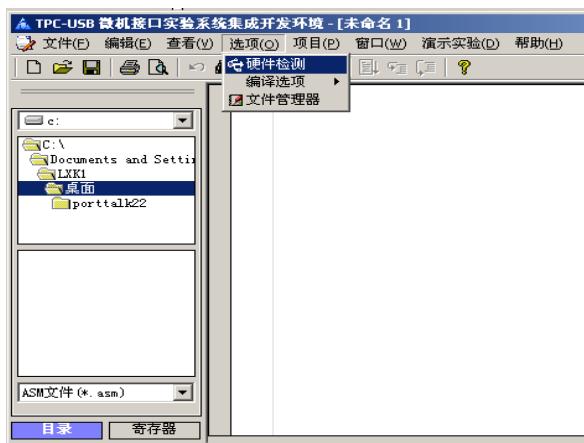


图2-14 硬件检测菜单

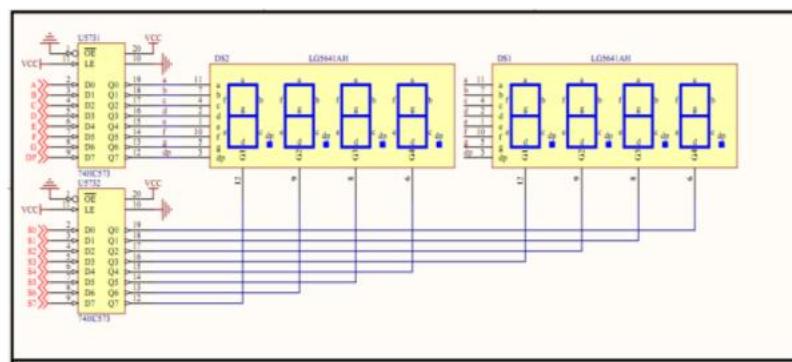


图2-15 正确连接提示

图2-16硬件未连接

The screenshot shows the TPC-ZK-II software interface. The title bar reads "TPC-ZK-II 单片机接口实验系统集成开发环境 - 实验02_单片机静态存储器6264.ASM". The main window displays assembly language code:

```
0001 DATA SEGMENT
0002 MESSAGE DB "PLEASE ENTER A KEY TO SHOW THE CONTENTS!",0DH,0AH,''
0003 DATA ENDS
0004 CODE SEGMENT
0005 ASSUME CS:CODE, DS:DATA, ES:DATA
0006 START: MOV AX, DATA
0007 MOV DS, AX
0008 MOV AX, 0B400H
0009 MOV ES, AX
0010 MOV SI, 0000H
0011 MOV CX, 100H
0012 MOV BX, 40H
0013 REP1: INC BX
0014 MOV ES, BX1, BL
0015 INT 21H
0016 CMPL DL, 5AH
0017 JNE SS1
0018 MOV DL, 40H
0019 SS1: LOOP REP1
0020 MOV BH, OFFSET MESSAGE
0021 MOV AH, 09H
0022 INT 21H
0023 MOV AH, 01H
```

第三章 TPC-ZK-II 集成软件开发环境

TPC-ZK-II 集成开发环境它提供了用户程序的编辑和编译，调试和运行，实验项目的查看和演示，实验项目的添加等功能，方便了学生和老师实验程序的编制和调试。本软件基于 windows2000/XP/2003/WIN7 环境，界面简洁美观，功能齐全。集成开发环境主界面如图。

图 3-1 主键面

图 3-2 打开程序

集成开发环境除了一般的编辑功能外，还支持语法高亮显示，语法错误提示等功能，大大提高了程序的可读性。用户编辑好程序并保存后，即可方便地进行编译。

1. 新建一个源程序

在当前运行环境下，选择菜单栏中的“文件”菜单，菜单下拉后选择“新建”，或是在工具栏中单击“新建”快捷按钮，会出现源程序编辑窗口，建议用“另存为”为文件取名保存后，就新建一个“.asm”文件。

2. 打开一个源程序

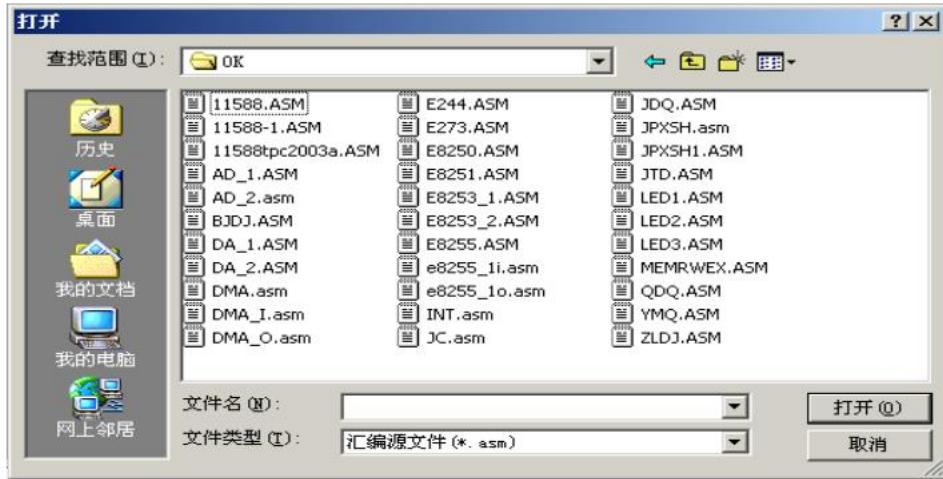
当前运行环境下，选择菜单栏中的“文件”菜单，菜单下拉后选择“打开”，或是在工具栏中单击“打开”，会弹出“打开”文件选择窗口，“打开”窗口如图 3-2 所示。

3. 编辑源程序

本软件提供了基本的编辑功能，并实现了实时的语法高亮，各项操作说明如下：

【撤消】：当前运行环境下，选择菜单栏中的“编辑”菜单，菜单下拉后选择“撤消”，或是在工具栏中单击“撤消”，即可撤消上一步剪切或粘贴操作。

【剪切】：当前运行环境下，选择菜单栏中的“编辑”菜单，菜单下拉后选择“剪切”，



或是在工具栏中单击“剪切”，即可将文档显示区域中选中的内容剪切到剪贴板。

【复制】：当前运行环境下，选择菜单栏中的“编辑”菜单，菜单下拉后选择“复制”，或是在工具栏中单击“复制”，即可将文档显示区域中选中的内容复制到剪贴板。

【粘贴】：当前运行环境下，选择菜单栏中的“编辑”菜单，菜单下拉后选择“粘贴”，或是在工具栏中单击“粘贴”，即可将剪贴板中当前内容粘贴到文档显示区域光标所在处。

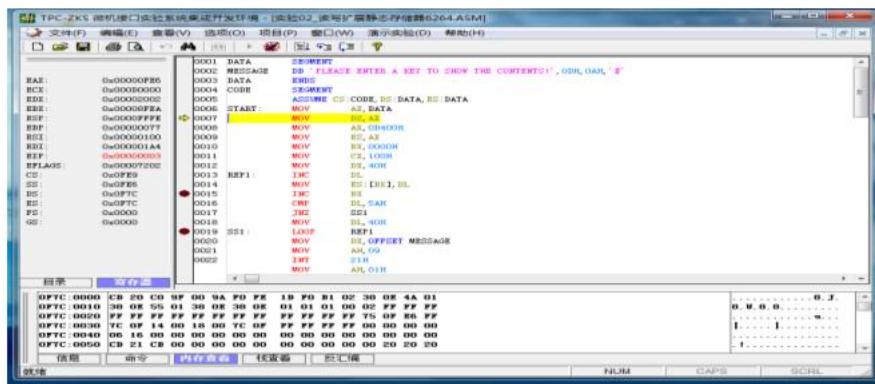
全选当前运行环境下，选择菜单栏中的“编辑”菜单，菜单下拉后选择“全选”，即可将文档区域中所有内容选中。

4. 保存源程序

当前运行环境下，选择菜单栏中的“文件”菜单，菜单下拉后选择“保存”，如果是无标题文档，用户需在提示下输入文档的名称及选择保存的路径，单击确定后保存；否则程序自动保存当前文档显示区域中显示的文档。或者选择菜单栏中的“文件”菜单，菜单下拉后选择“另存为”，并在提示下输入文档的名称及选择保存的路径，单击确定后保存。

5. 源程序编译

在当前运行环境下，选择菜单栏中的“查看”菜单，单击编译调试窗口选项或是单击



工具栏中“输出窗口”按钮则可对输出栏的进行显示。若当前环境显示编译调试窗口，则单击查看输出窗口选项即可隐藏该窗口，编译调试输出窗口选项即消失；若当前隐藏编译调试窗口，则单击输出窗口选项即可显示该窗口，编译调试窗口将显示。

【汇编+链接】：在当前运行环境下，选择菜单栏中的“项目”菜单，选择汇编+链接选项则程序对当前 ASM 源文件进行汇编与链接，编译调试窗口中输出汇编与链接的结果，若程序汇编或链接有错，则详细报告错误信息。双击输出错误，集成开发环境会自动将错误所在行代码高亮显示。

【开始+执行】：在当前运行环境下，选择菜单栏中的“项目”菜单，选择开始+执行选项则程序对当前 ASM 源文件执行，程序自动运行。

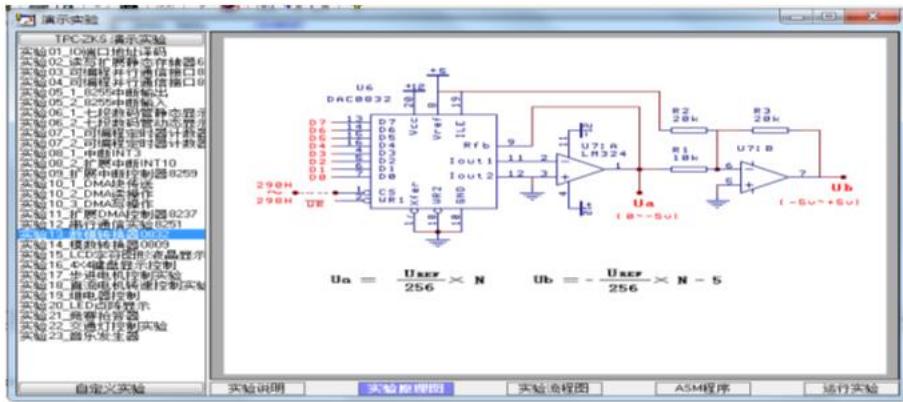
6. 源程序调试与运行

【寄存器窗口】：在当前运行环境下，选择工作区的“寄存器”菜单，寄存器窗口即可显示。寄存器窗口中显示主要的寄存器名称及其在当前程序中的对应值，若值为红色，即表示当前寄存器的值。调试时，单步执行，寄存器会随每次单步运行改变其输出值，同样以红色显示。

【开始调试】：程序的编译和链接成功之后，调试工具将会显示，也可以在“项目”中选择“开始/结束调试”。即可开始进行程序的调试。编译选项选择如图：

图 3-3 调试窗口

在 ASM 程序正常链接之后，选择菜单栏中的“开始/结束调试”菜单，选择开始调试



选项，则对源程序进行反汇编，进入 ASM 的调试状态，并在寄存器窗口中显示主要的寄存器的当前值。

【设置/清除断点】：在 ASM 的调试状态下，对程序代码所在某一行前的灰色列条单击鼠标，即对此行前设置了断点，如果清除断点，只需再在此行前的灰色列条上的断点单击鼠标，此断点标记将被清除。黄色箭头所指的行为当前单步执行到的所在行。

【连续运行】：在 ASM 的调试状态下，选择“项目”菜单栏中的“连续运行”菜单或 F5，则程序连续运行，直至碰到断点或程序运行结束。

【单步】：在 ASM 的调试状态下，选择“项目”菜单栏中的“单步执行”菜单或 F11，则程序往后运行一条语句。

【退出调试】：在 ASM 的调试状态下，选择“项目”菜单栏中的“开始/结束调试”菜单，程序则退出 ASM 的调试状态。

【命令调试】：集成开发环境可以进行命令的调试，调试时，输出窗口可以输出编译信息，命令信息，内存查看信息，栈查看信息等。

7. 实验项目查看与演示

本软件提供了实验项目的查看和演示功能，包括实验说明、实验原理图、实验流程图、ASM 程序，并可以运行实验程序，使用户能方便快捷地了解感兴趣的实验。

图 3-4 帮助功能

8. 常用调试命令

调试指令与 debug 稍有区别，具体调试命令如下：

1) 反汇编 (u)

用法： u [/count] start end;

反汇编给定的线性地址，可选参数'count'是反汇编指令的条数；

例: **u** 反汇编当前 cs:ip 所指向的指令;

u /10 从当前 cs:ip 所指向的指令起, 反汇编 10 条指令;

u /12 0xeff 反汇编线性地址 0xeff 处开始的 12 条指令。

2) 查看内存 (x)

用法: **x/nuf addr;**

查看线性地址'addr'处的内存内容;

nuf 由需要显示的值个数和格式标识[xduot cbhw m]组成, 未指明用何种格式的情况下将使用上一次的格式。

x: 十六进制

d: 十进制

u: 无符号

o: 八进制

t: 二进制

c: 字符

b: 字节

h: 半字

w: 字 (四字节)

m: 使用 memory dump 模式例: **x /10wx 0x234** 以十六进制输出位于线性地址 0x234 处的 10 个双字

x /10bc 0x234 以字符形式输出位于线性地址 0x234 处的 10 个字符

x /h 0x234 以十六进制输出线性地址 0x234 处的 1 个字

3) 查看寄存器 (info reg)

用法: **info reg;**

查看 CPU 整数寄存器的内容。

4) 修改寄存器 (r)

用法: **r reg = expression;**

reg 为通用寄存器;

expression 为算术表达式;

例: **r eax = 0x12345678** 对 eax 赋值 0x12345678;

r ax = 0x1234 对 ax 赋值 0x1234;

r al = 0x12 + 1 对 al 赋值 0x13。

5) 下断点 (lb)

用法: lb addr;

下线性地址断点

例: lb 0xfeff 在 0xfeff 下线性地址断点, 0f00:eff 所处线性地址就是 0xfeff。

6) 查看断点情况 (info b)

用法: info b。

7) 删断点 (del n)

用法: del n;

删除第 n 号断点;

例: del 2 删除 2 号断点, 断点编号可通过前一个命令查看。

8) 连续运行 (c)

用法: c;

在未遇到断点或是 watchpoint 时将连续运行。

9) 单步 (n 和 s)

用法: n;

执行当前指令, 并停在紧接着的下一条指令。如果当前指令是 call、ret, 则相当于

Step Over;

s [count];

执行 count 条指令。

10.退出 (q)

用法: q。