

一、 仿真器（FET）的硬件安装：

1． 仿真器 25—PIN 芯电缆线与计算机并口连接，14—PIN 的扁平电缆与适配器或用户目标板的 JTAG 相连。如果出现仿真器（FET）程序无法下载调试，请核对连接线是否连接好，适配器上是否安装了芯片，PC 并口应为 EPP 或 ECP 模式，若不符，请在 CMOS 中修改

2． JATG 的连接和使用方法：

JATG 的 1，2，3，5，7，9，11 脚分别和芯片的 TDO/TDI，VCC_MSP，TDI/Vpp，TMS，TCK，GND，RES/NMI 相连。

JATG 口的第 8 脚与 11X，12X 系列的第一脚相连。其余系列（13X，14X，41X，43X，44X）的不用连接。

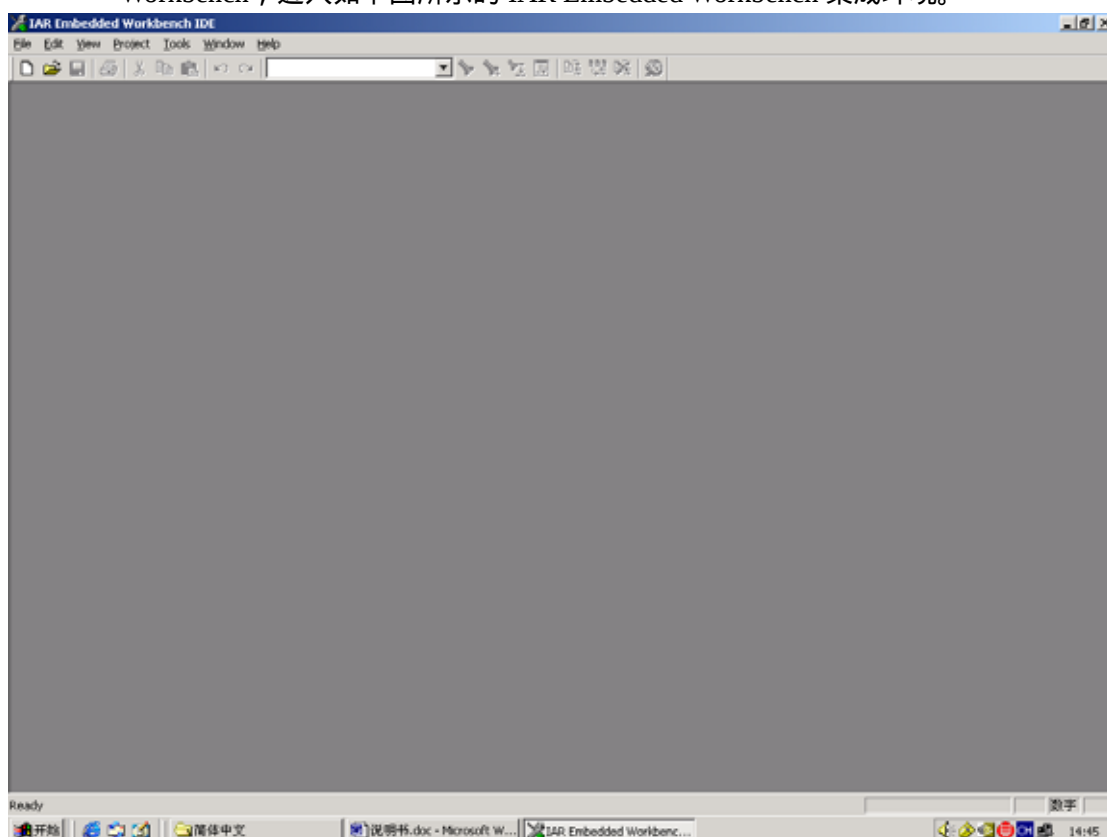
详细情况请查阅 EW430__UsersGuide.pdf

二、 仿真器的软件安装：

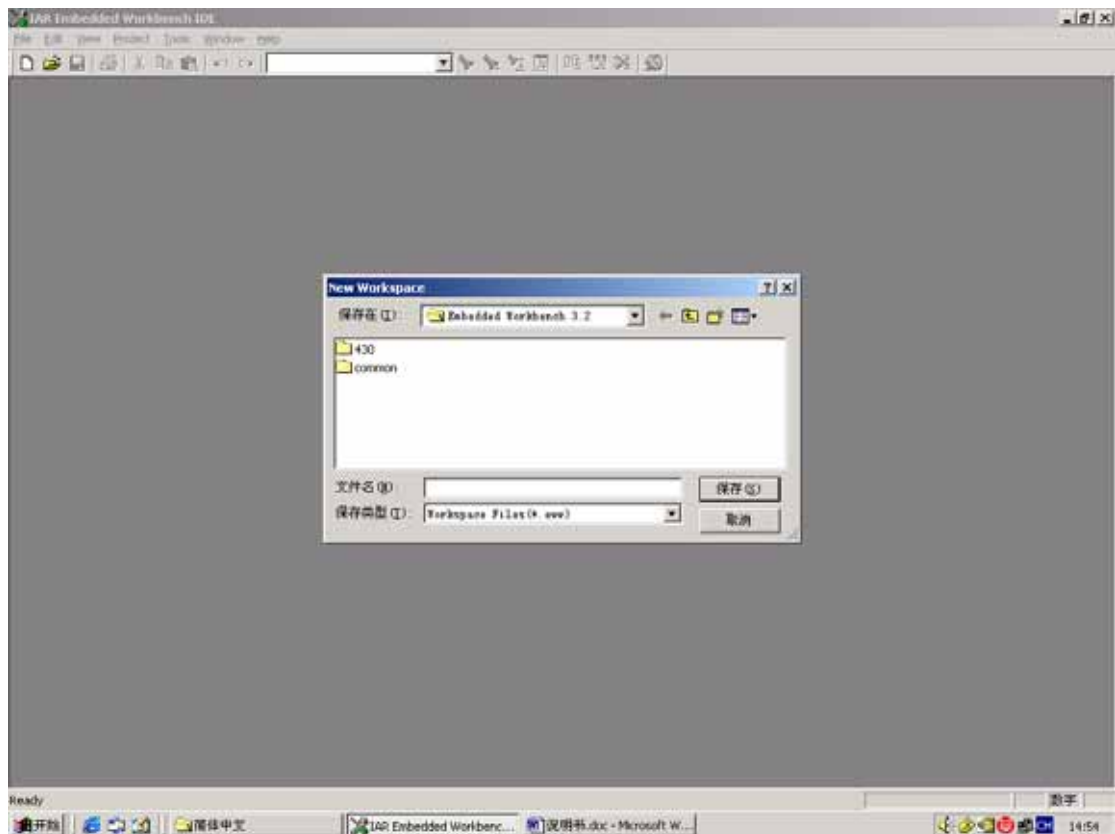
在 Windows 环境下，双击 ew430-ev-310a.exe 目前的最新版本是 3.10a。对机器没有特殊的要求，目前的计算机都能满足对内存，硬盘，机器速度的要求。安装完成后能在开始—程序—IAR Systems—IAR Embedded Workbench Evaluation for MSP430 V2 有 IAR Embedded Workbench 则为安装成功。

三、 Embedded Workbench 使用指南

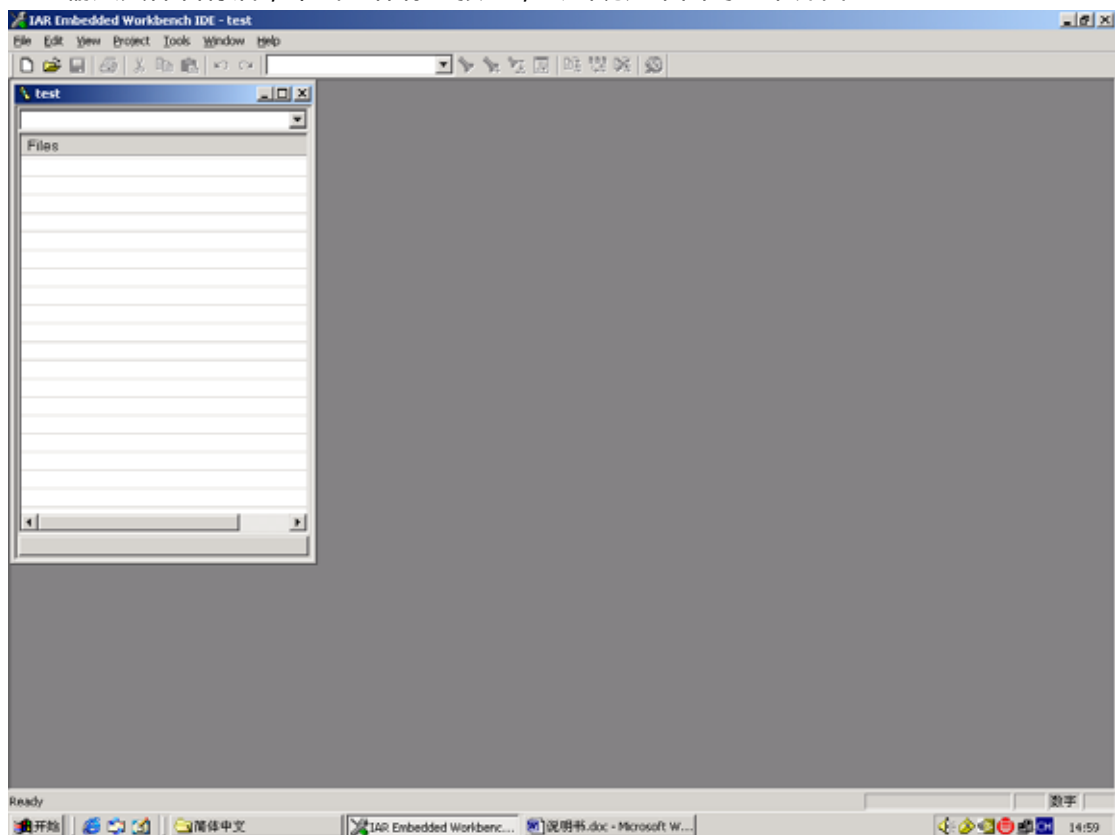
下面举一个具体的例子来说明如何使用 Embedded Workbench。这个例子很简单，就是工具套件中的在 P5.1 上连接发光二极管闪烁的例子。首先打开 IAR Embedded Workbench。在 Windows 环境下依次单击：开始，程序，IAR Systems，IAR Embedded Workbench Evaluation for MSP430 V2，IAR Embedded Workbench，进入如下图所示的 IAR Embedded Workbench 集成环境。



单击 File，New，选择 workspace 后，点击“确定”。则会出现如下图所示界面：

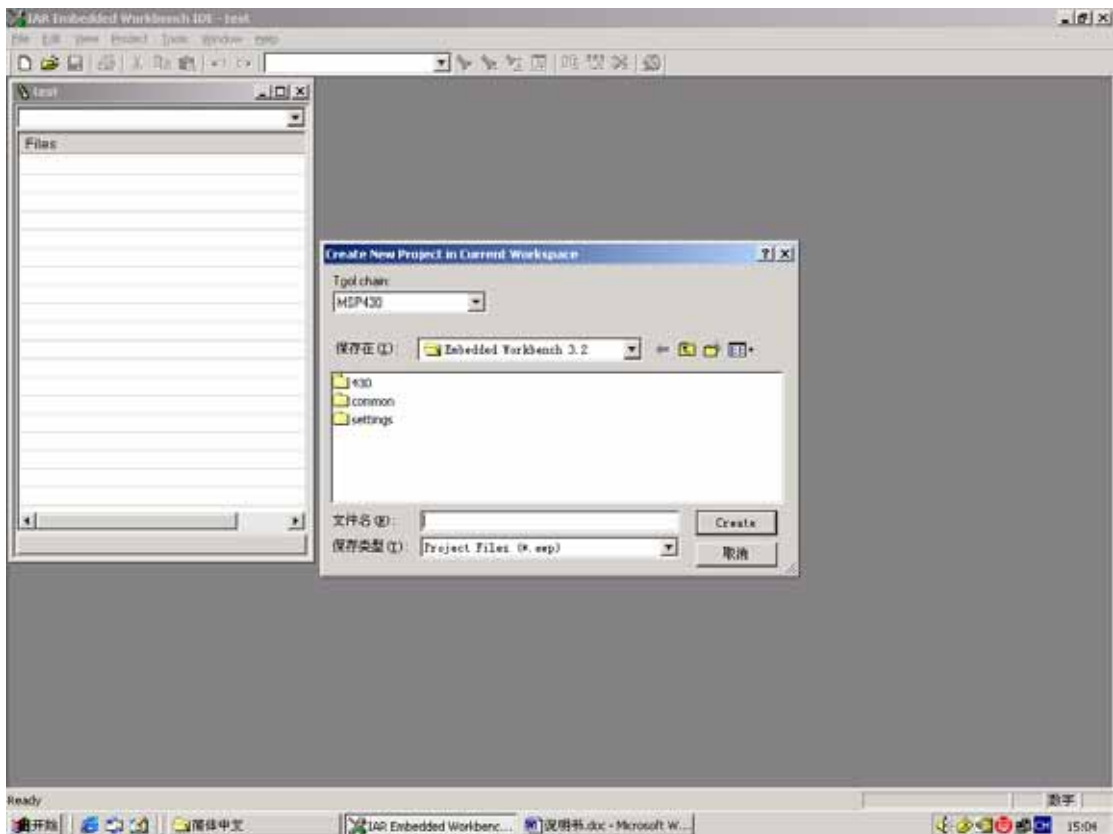


输入文件名称后，单击“保存”按钮，会出现如下图的一个界面：

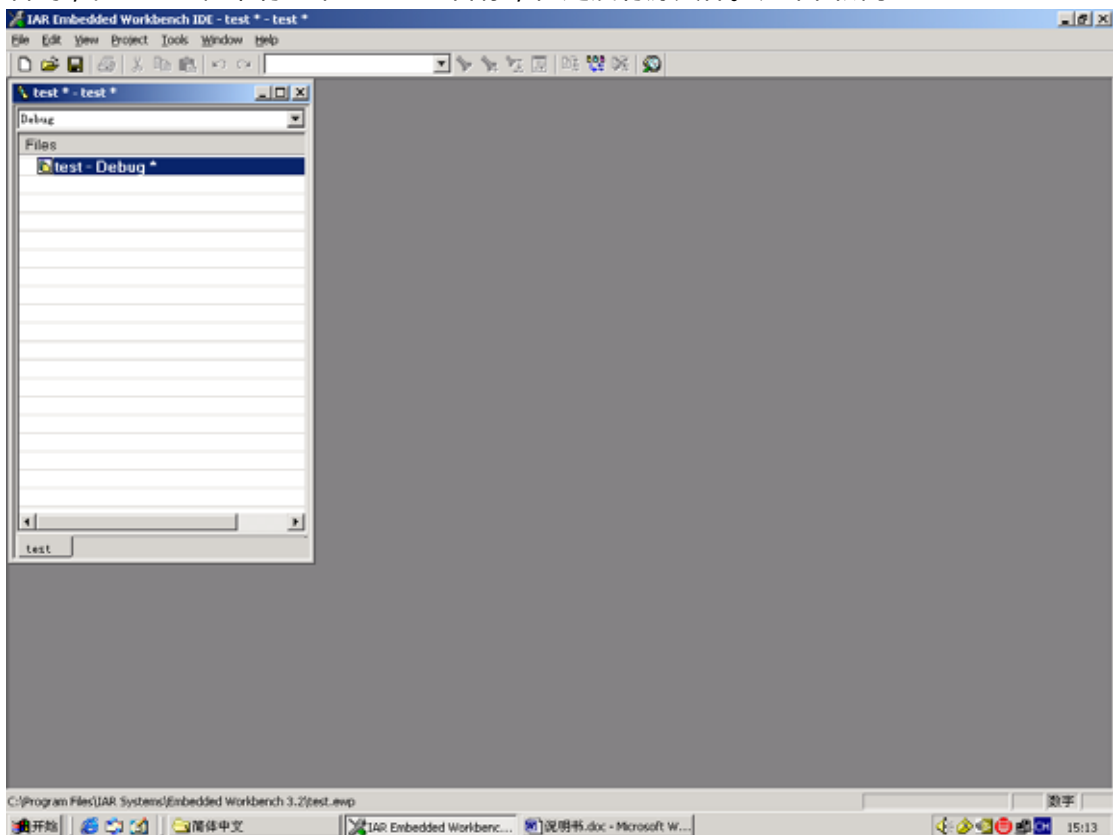


此时表示已经新建了一个 workspace。

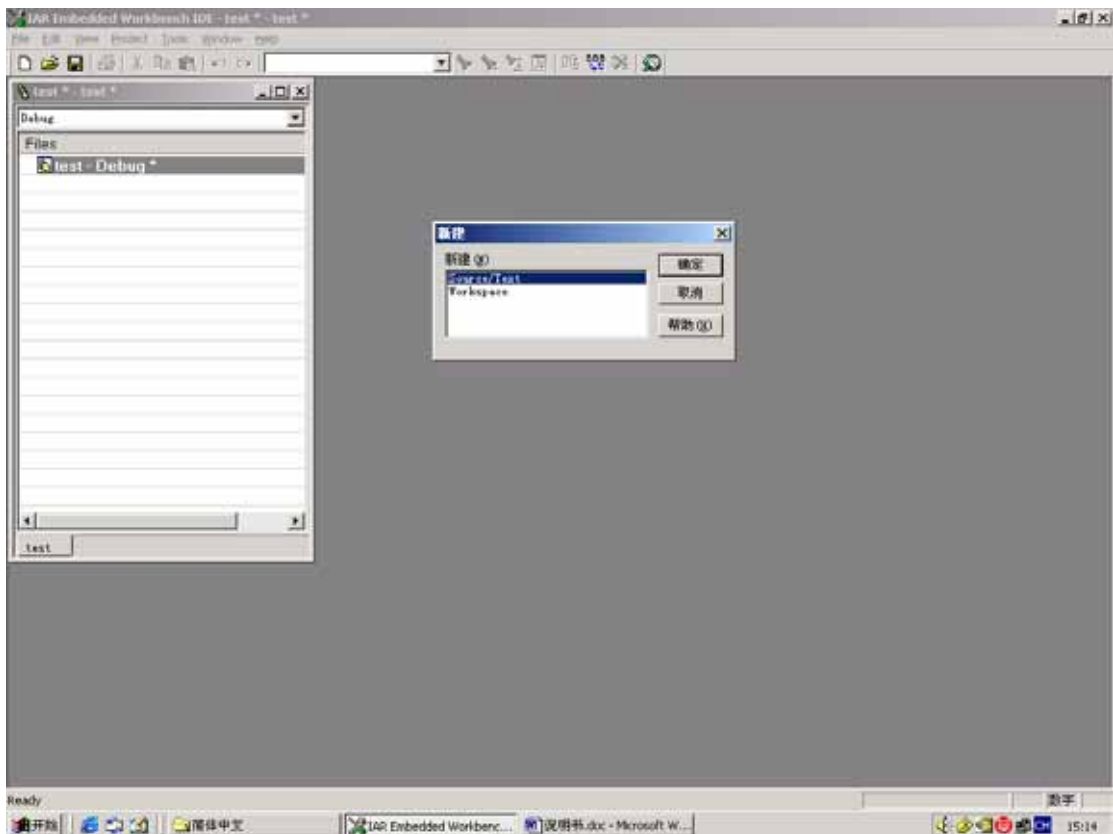
然后在该环境下建立一个工程。单击 Project , Creat new project...后出现下图所示界面。



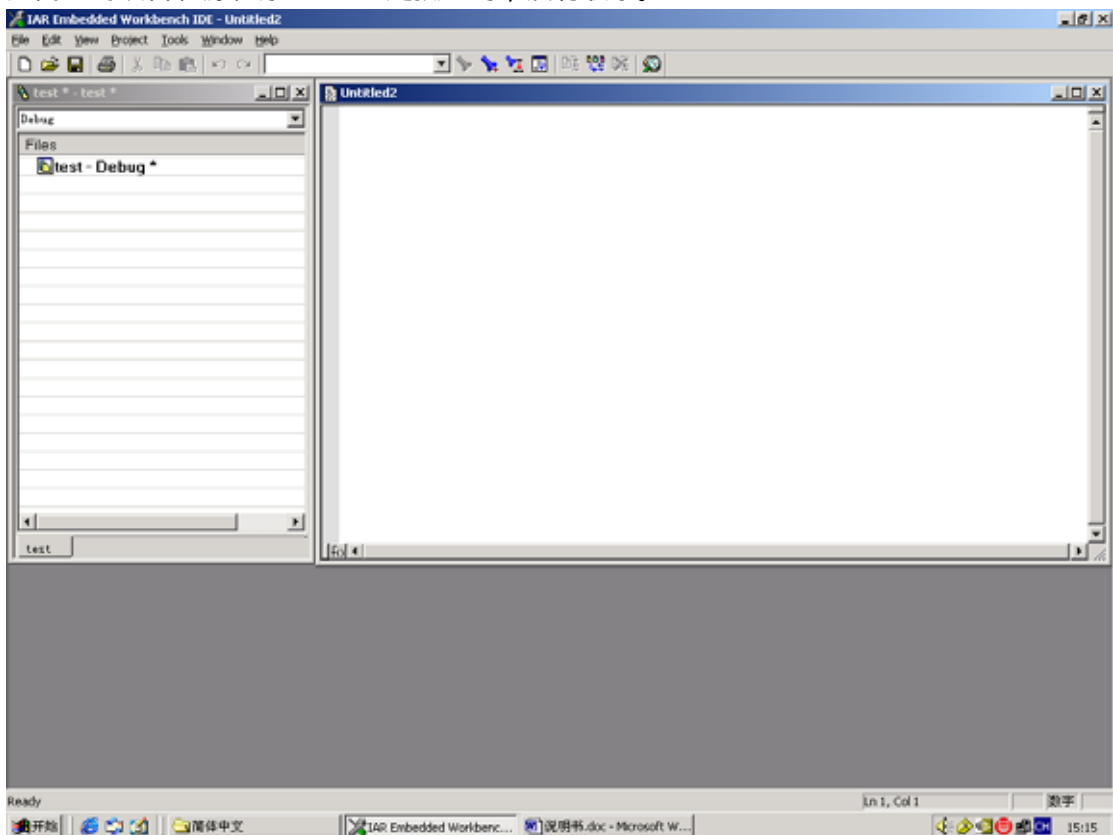
选择目标 mcu，输入工程名称后，单击“Creat”保存到所选择的路径。此时可以看到，在 test 工程中有一个 DEBUG 目标，但是没有源文件。如下图所示



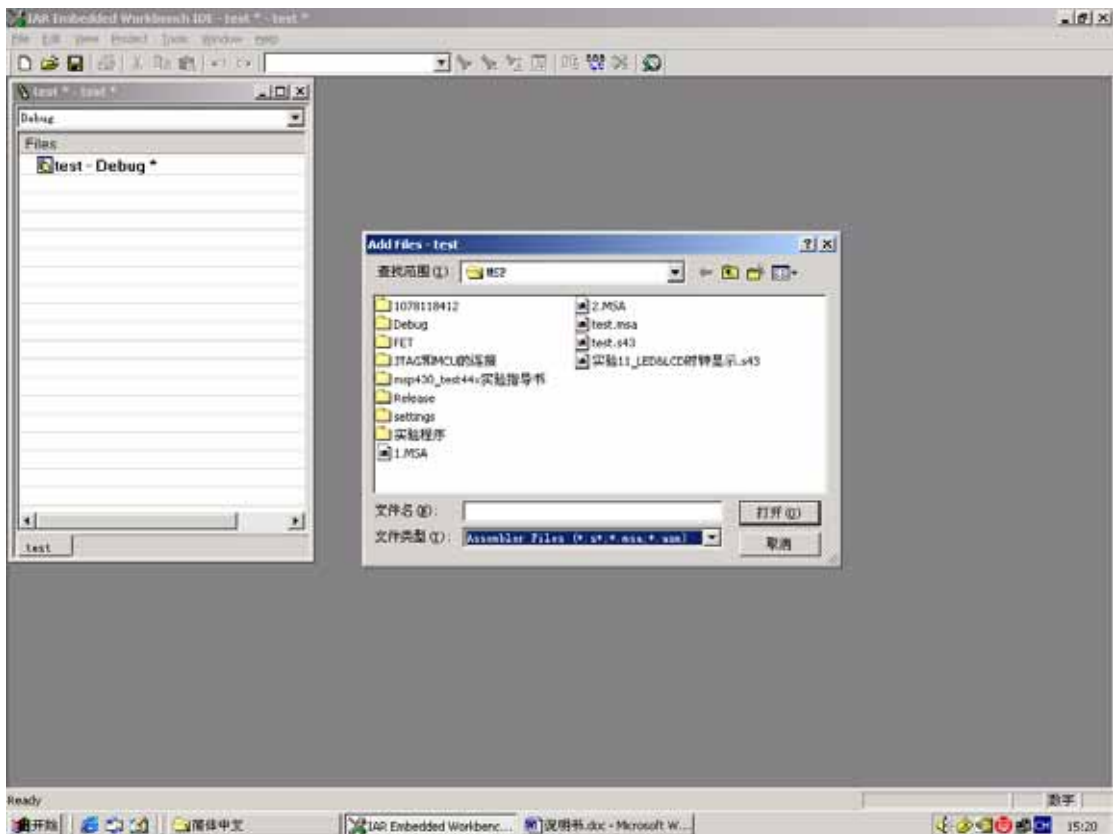
单击 File，New 进入下图所示界面。选择 Source/Text 选项，单击“确定”进入程序编辑界面



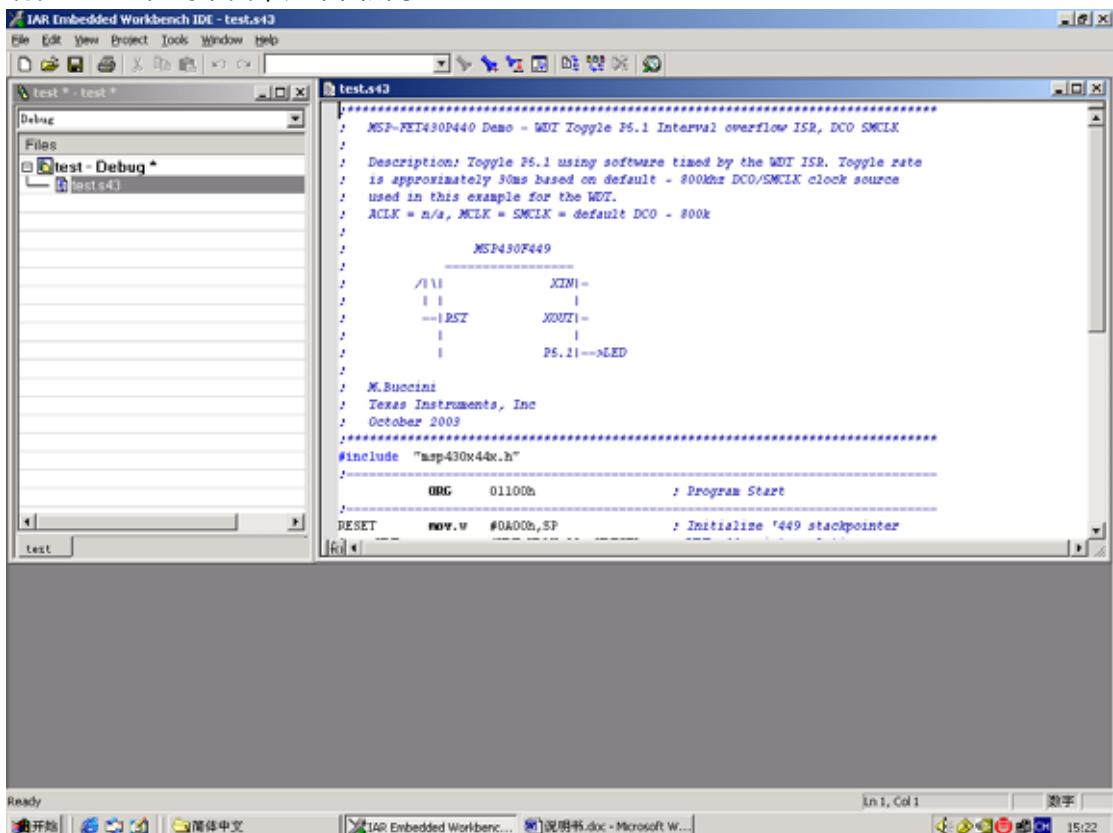
如下图所示，在这里可以进行源程序的编辑。源程序编辑好之后，保存为 test.s43。然而此时项目和源程序 test.s43 是孤立的，没有联系。



单击 Project, Add File 会出现如下图所示的界面。选择所编辑的源程序，单击“打开”按钮。



这样源文件就已经被加入到了项目中了,在左边的文件框中可以看到所选的文件出现在 test 工程的下面,如下图所示



这样项目中就有了 test.s43 程序,编译,连接之后产生的目标代码就是 test.s43 汇编后的代码。

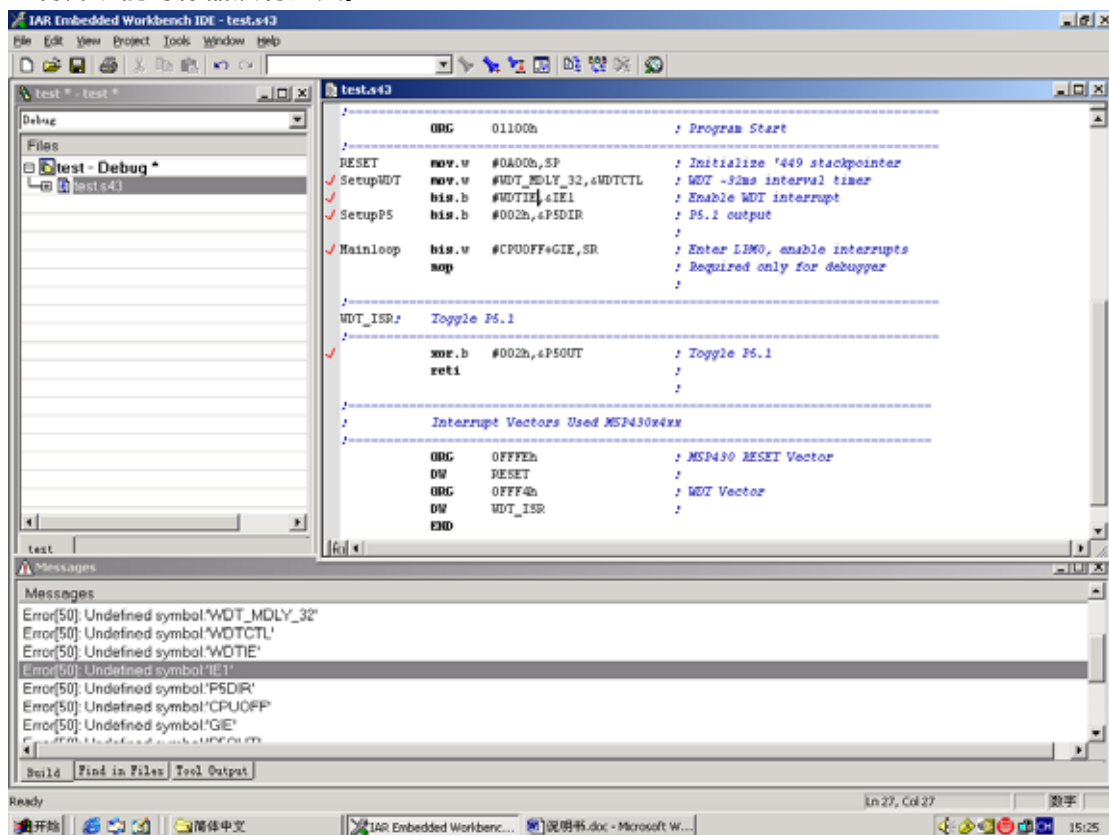
程序清单如下：

```

                                ORG      01100h                      ; Program Start
RESET      mov.w    #0A00h,SP                      ; Initialize '449 stackpointer
SetupWDT   mov.w    #WDT_MDLY_32,&WDTCTL           ; WDT ~32ms interval timer
           bis.b    #WDTIE,&IE1                    ; Enable WDT interrupt
SetupP5    bis.b    #002h,&P5DIR                   ; P5.1 output

Mainloop   bis.w    #CPUOFF+GIE,SR                 ; Enter LPM0, enable interrupts
           nop                                       ; Required only for debugger
           WDT_ISR                                  ; Toggle P5.1
;-----
           xor.b    #002h,&P5OUT                   ; Toggle P5.1
           reti                                         ;
           ORG      0FFFEh                          ; MSP430 RESET Vector
           DW       RESET                          ;
           ORG      0FFF4h                          ; WDT Vector
           DW       WDT_ISR                        ;
           END
```

单击 Project, compile, 进行文件编译或汇编, 出现如下图所示的错误提示。任意单击一个错误提示, 系统会自动给出有错误的语句行, 仔细阅读错误提示, 会发现是一些特殊功能寄存器没有定义。

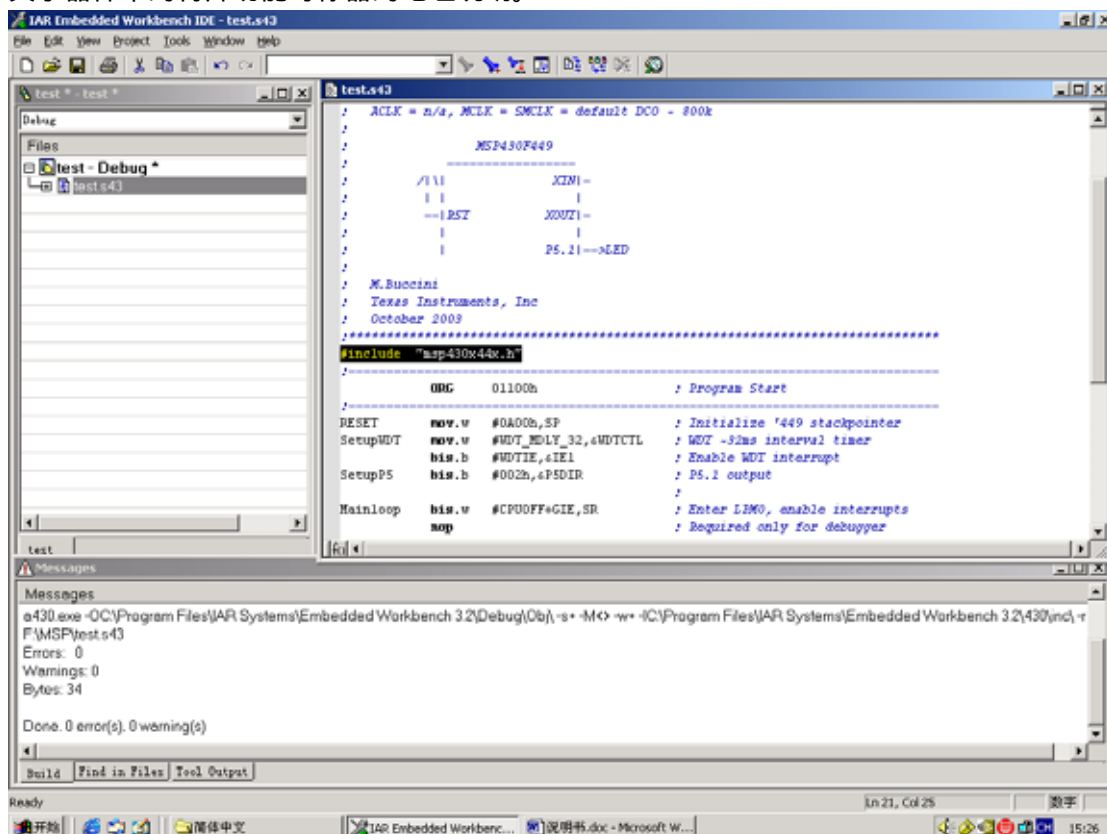


在源文件的开始加一句

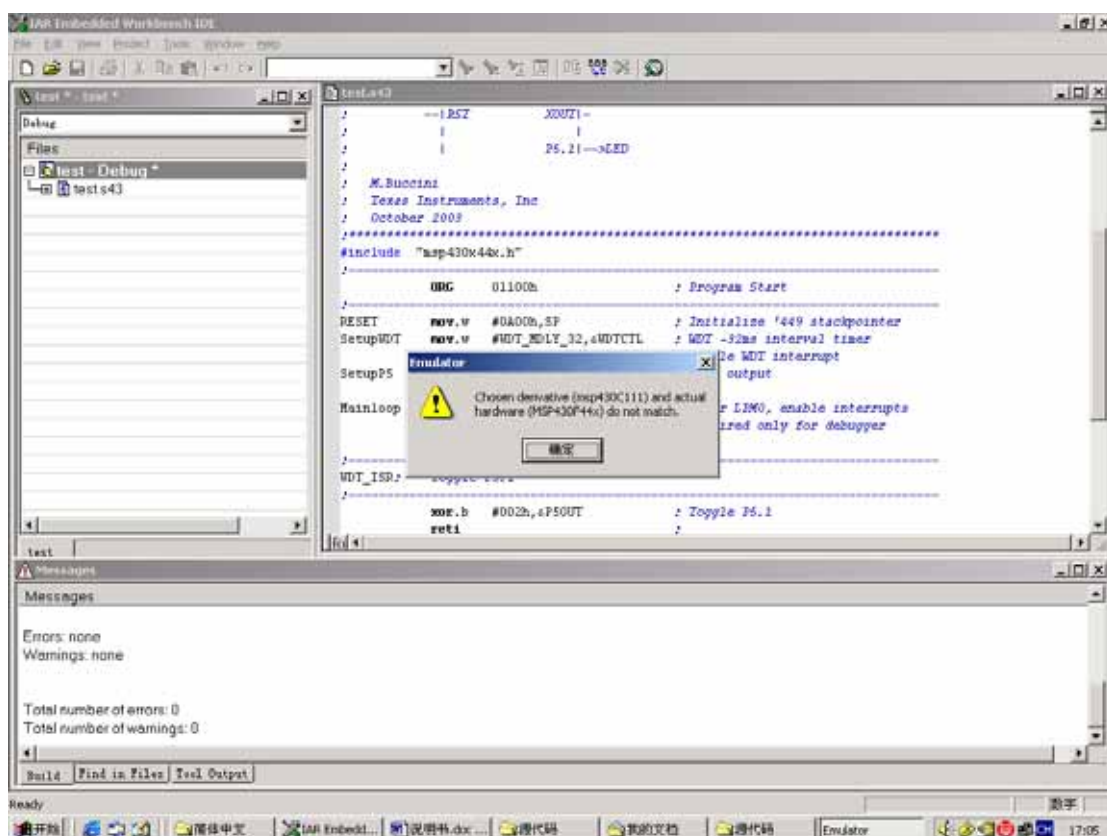
```
#include "msp430x44x.h"
```

如下图所示, 在进行编译则没有错误了, 因为在"msp430x44x.h"等的包含文件中已经有

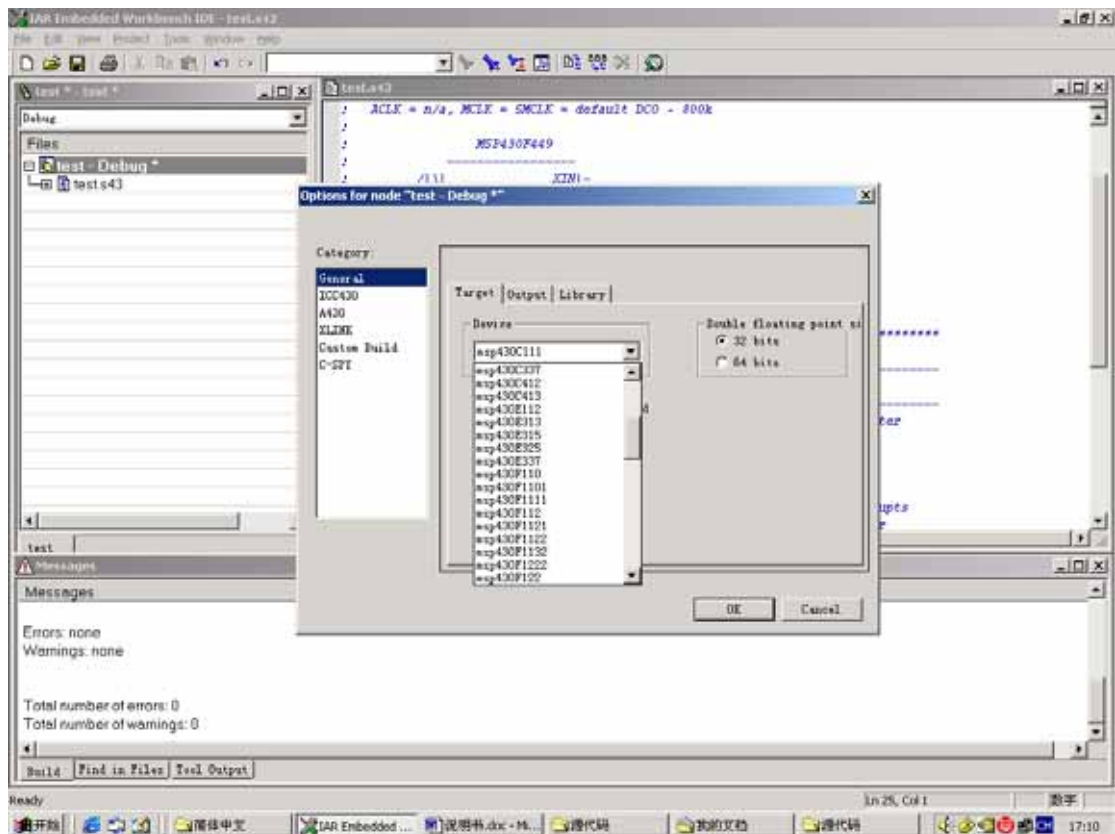
关于器件中的特殊功能寄存器的地址说明。



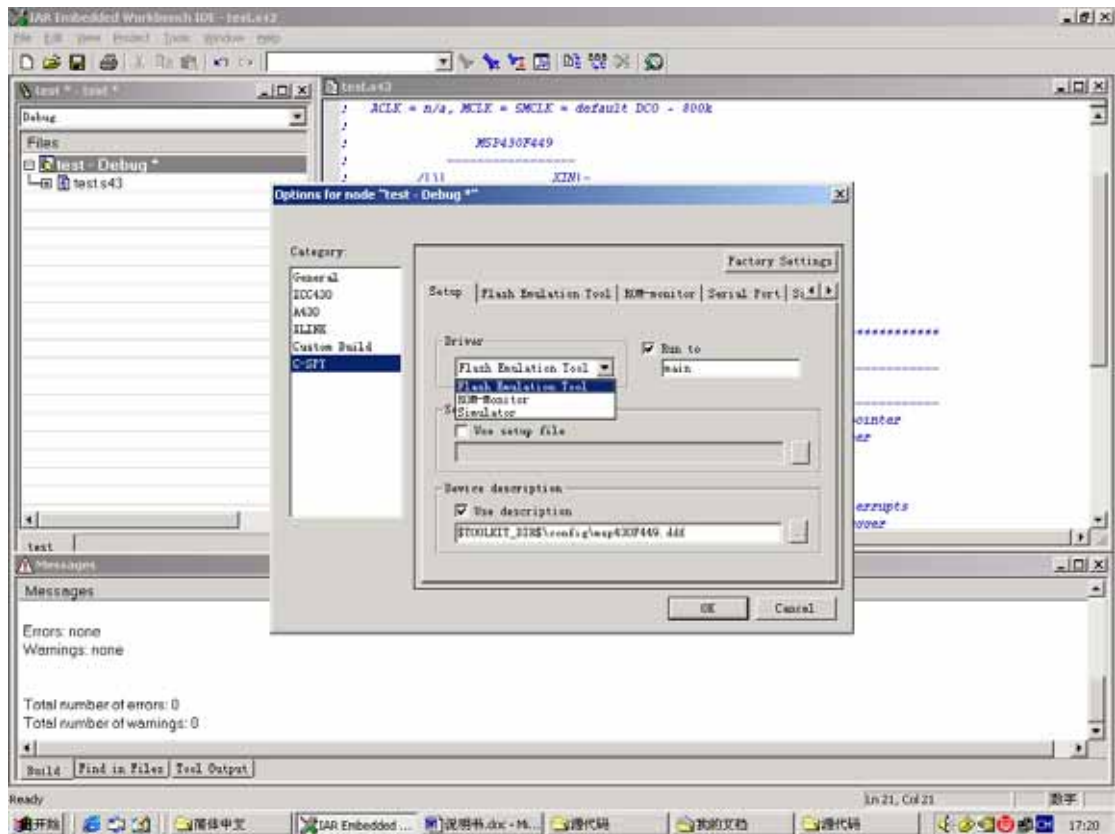
源文件编译通过之后，将生成目标代码。这时单击 Project，Make 进行连接以生成目标代码，出现如下图的错误提示。



这是因为环境的设置不对，单击 Project，Options 进入设置界面后，如下图所示，选择和设计者所使用的 MCU 一致的芯片型号。

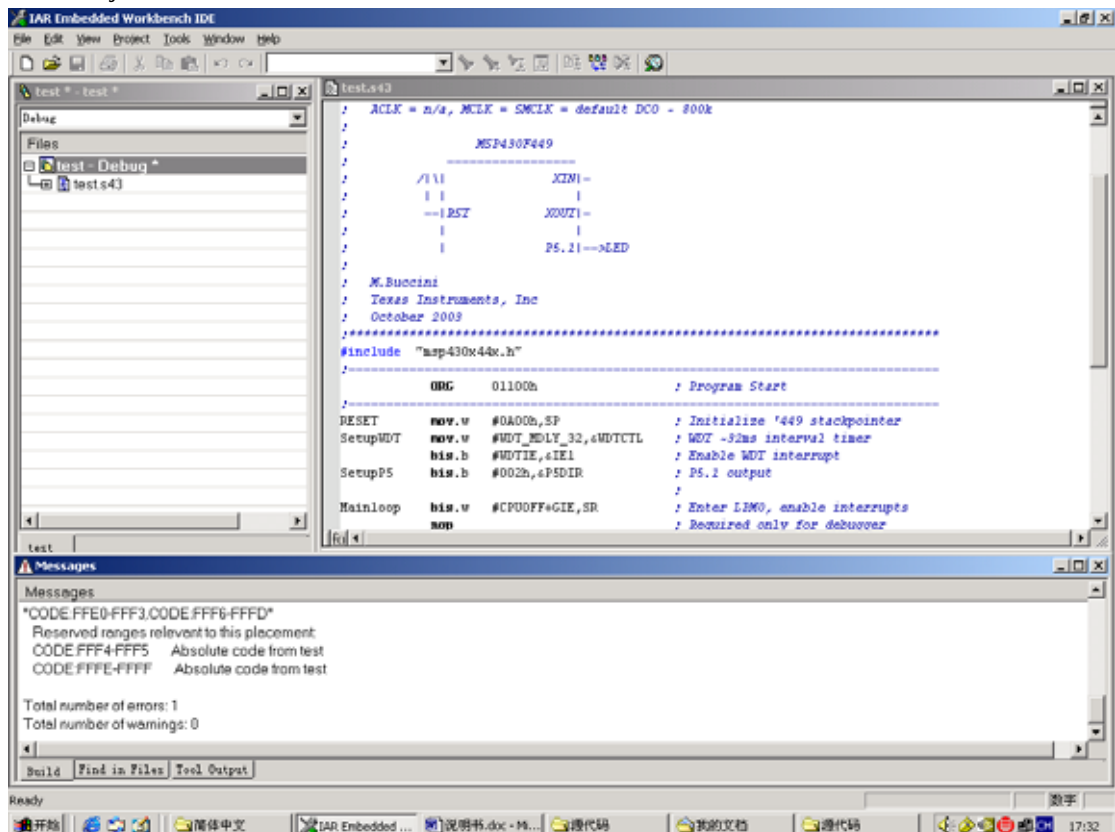


然后再选择 C-SPY，如下图所示：

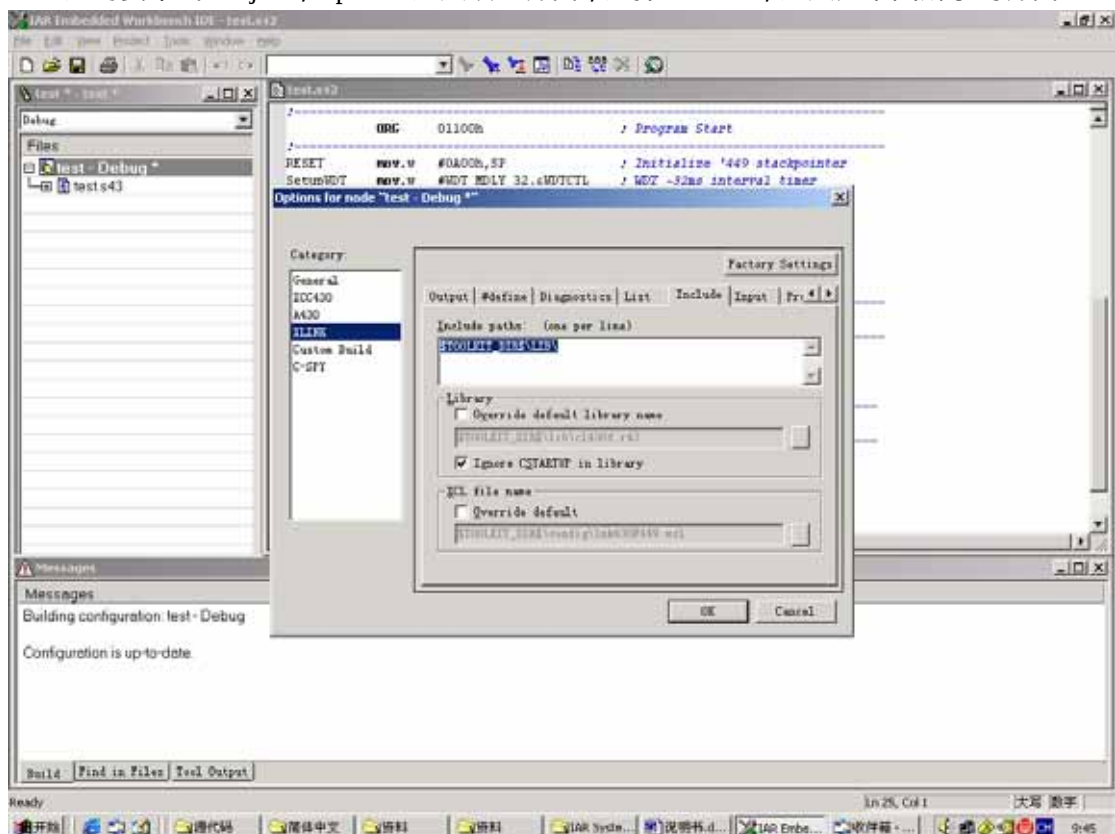


选择要进行仿真的模式，其中有三种模式：Flash Emulation Tool,Rom-Monitor,Simulator

进行硬件仿真时选择 Flash Emulation Tool，进行模拟仿真时选择 Simulator。这时再次单击 Project，Make 进行连接以生成目标代码，出现如下图的错误提示。



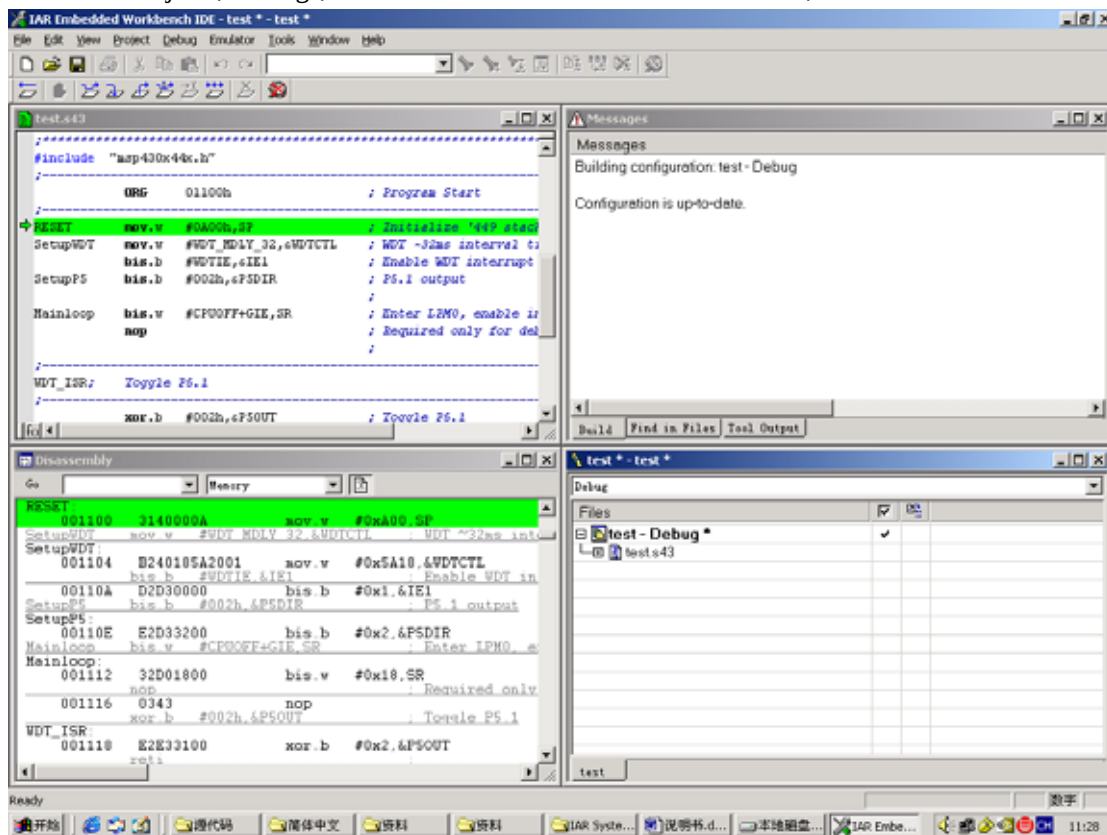
这时要单击 Project，Options 进入设置界面，选择 XLINK，进入如下图所示的界面



在这些设置中，选择 Include 选项，只需在 Ignore CSTARTUP in library 这个选项前

打上勾，其它保持默认。再次单击 Project，Make 进行编译连接，则完全通过。接下来对所编程序进行调试，以验证设计的正确。

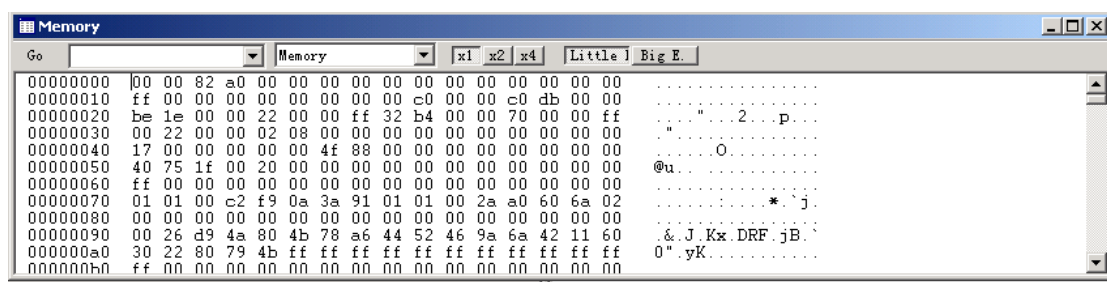
单击 Project，Debug，或按 Ctrl+D 可以进入 CSPY 调试模式，如下图所示：



在 CSPY 调试环境中，可以打开调试程序所需的若干窗口有：源程序窗口，寄存器窗口，观察窗口，存储器窗口，特殊功能寄存器窗口。

1. 存储器窗口

存储器窗口是调试程序常用的窗口，如下图所示。此窗口打开的方法为：单击 View，Memory。MSP430 的存储器为线性统一地址，小模式下地址范围为 0000H~0FFF0H。此窗口可以观察到存储器的全部内容，而且有 3 种数据模式：8 位，16 位，32 位。



2. 源程序窗口

源程序窗口是调试程序必不可少的，一般默认为打开。如果没有打开，可以单击 View，Dedicated Editor 以打开，如图 1 所示。其中的绿色指示为即将执行的指令语句条。随着程序的运行，此绿色指示条随程序流程而改变。单击 View，Disassembly 则可以观察到如图 2 所示的地址—代码—源程序窗口；同时显示格式有所改变：先显示源程序，再另行显示地址—代码—指令。而且可以看到每一条指令都开始于偶地址处。

```

test.s43
; M.Buccini
; Texas Instruments, Inc
; October 2003
;*****
#include "msp430x44x.h"
;-----
;-----
ORG 01100h ; Program Start
;-----
RESET mov.w #0A00h,SP ; Initialize '449 stackpointer
SetupWDT mov.w #WDT_MDLY_32,&WDTCTL ; WDT ~32ms interval timer
bis.b #WDTIE,&IE1 ; Enable WDT interrupt
SetupP5 bis.b #002h,&P5DIR ; P5.1 output
;
Mainloop bis.w #CPUOFF+GIE,SR ; Enter LPM0, enable interrupts
nop ; Required only for debugger
;

```

图 1

```

Disassembly
Go Memory
0010EC FFFFFFFF and.b @R15+,0xFFFF(R15)
0010F0 FFFFFFFF and.b @R15+,0xFFFF(R15)
0010F4 FFFFFFFF and.b @R15+,0xFFFF(R15)
0010F8 FFFFFFFF and.b @R15+,0xFFFF(R15)
0010FC FFFFFFFF and.b @R15+,0xFFFF(R15)
RESET mov.w #0A00h,SP ; Initialize '449 stackpointer
RESET:
001100 3140000A mov.w #0xA00,SP
SetupWDT mov.w #WDT_MDLY_32,&WDTCTL ; WDT ~32ms interval timer
SetupWDT:
001104 B240185A2001 mov.w #0x5A18,&WDTCTL
bis.b #WDTIE,&IE1 ; Enable WDT interrupt
00110A D2D30000 bis.b #0x1,&IE1
SetupP5 bis.b #002h,&P5DIR ; P5.1 output
SetupP5:
00110E E2D33200 bis.b #0x2,&P5DIR
Mainloop bis.w #CPUOFF+GIE,SR ; Enter LPM0, enable interrupts
Mainloop:
001112 32D01800 bis.w #0x18,SR

```

图 2

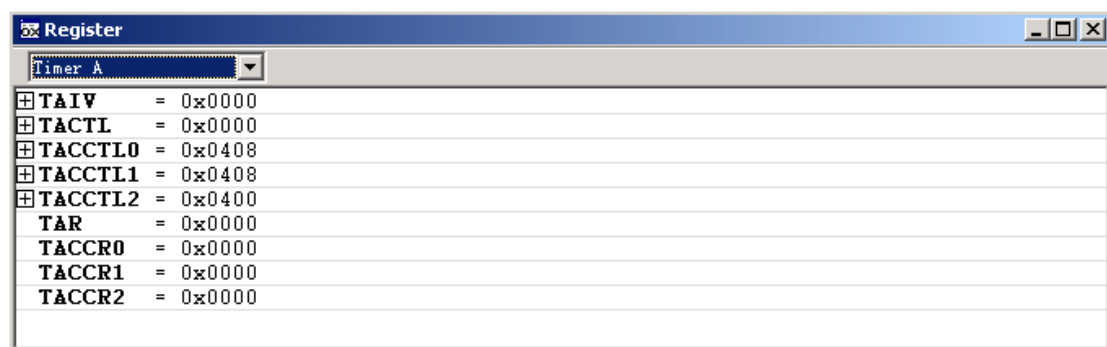
3.寄存器窗口

寄存器窗口是调试程序另一个常用的窗口,如下图所示。其打开方式为:单击 View, Register。其中:R0~R2 分别是前 3 个 PC (程序计数器), SP (堆栈指针), SR (状态寄存器); R3 为常数发生器,模拟指令使用,这里用户看不到; R4~R15 为用户使用,可以查看与修改; CYCLECOUNTER 为程序执行所用的机器周期数,通过它可以方便地知道执行代码的时间。

Register			
CPU Registers			
PC	= 0x1116	R11	= 0x1241
SP	= 0x0A00	R12	= 0xC559
SR	= 0x0018	R13	= 0x1409
R4	= 0x8A41	R14	= 0x0846
R5	= 0x36CE	R15	= 0x7A41
R6	= 0x5F13	CYCLECOUNTER = 0	
R7	= 0x9CCD		
R8	= 0x0000		
R9	= 0xFFFF6		
R10	= 0x0264		

4.特殊功能寄存器窗口

特殊功能寄存器窗口也是调试程序一个常用的窗口，如下图所示。其打开方式为：单击 View , Register 然后点击左上角的下拉按钮。图中选的是 Timer A，所以打开的是与定时器有关的一些特殊功能寄存器，同样可以打开其他一些相关的特殊功能寄存器。



5.观察窗口

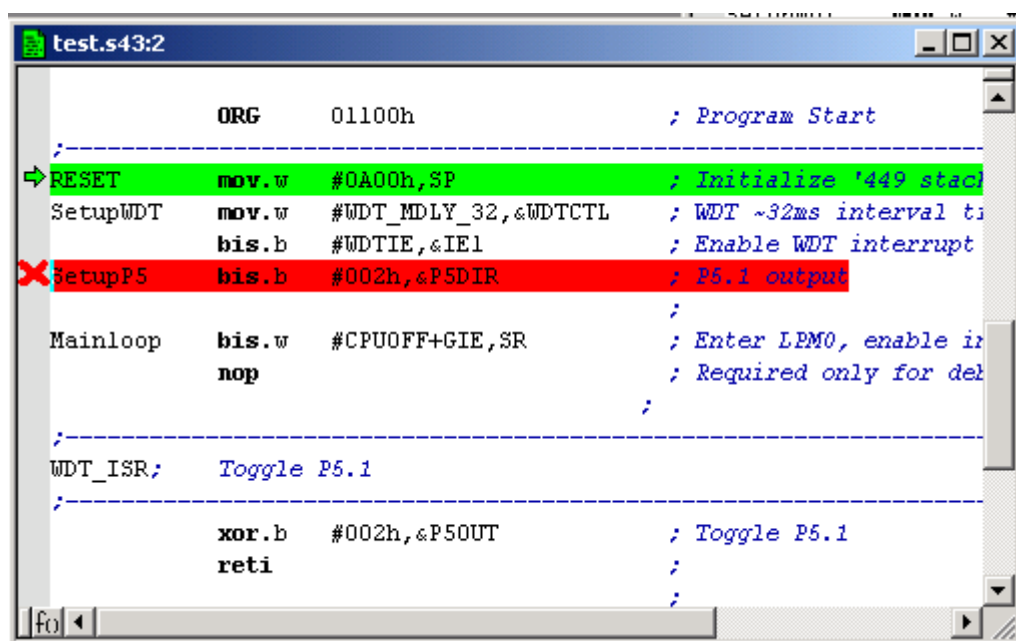
在这里有几种方式调试程序：单步，断点，连续，运行到光标处等。下面进行分别阐述。

1 单步方式

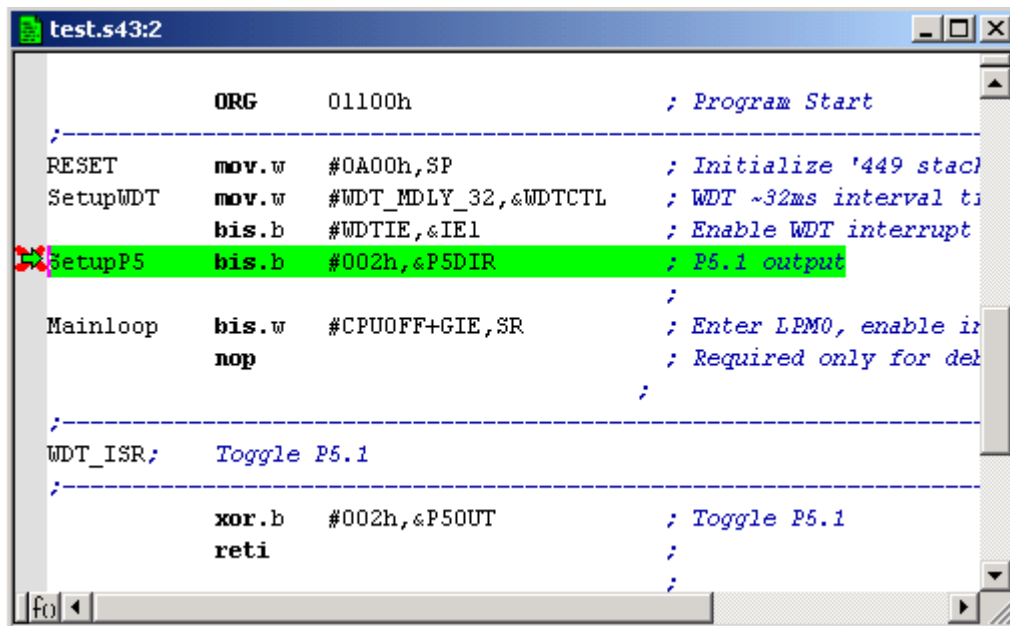
单步执行则要点击 Debug , Step Into 或按 F11 键，这样就能单步运行程序了。

2 断点方式

首先要设置断点 将光标移到要设置断点的语句处。然后按 F9 键或单击 Edit ,Toggle Breakpoint，设置断点的语句条就会用醒目的红色显示，如下图所示。



再单击 Debug , Go 程序就会执行到断点处时停止，如下图所示。



```
test.s43:2
;-----
ORG      01100h                ; Program Start
;-----
RESET    mov.w  #0A00h,SP        ; Initialize '449 stack
SetupWDT  mov.w  #WDT_MDLY_32,&WDTCTL ; WDT ~32ms interval ti
        bis.b  #WDTIE,&IE1      ; Enable WDT interrupt
SetupP5   bis.b  #002h,&P5DIR    ; P5.1 output
;-----
Mainloop  bis.w  #CPUOFF+GIE,SR  ; Enter LPM0, enable in
        nop                    ; Required only for del
;-----
WDT_ISR:  Toggle P5.1
;-----
        xor.b  #002h,&P5OUT     ; Toggle P5.1
        reti
;-----
```

3 连续方式

单击 Debug, Go 或按 F5 键程序就能进行连续的运行, 只有遇到断点才停下来。

4 运行到光标

运行到光标处时先要取消断点: 与设置断点的方式一样。取消断点之后, 将光标移动到想要运行到的语句条, 在单击 Debug, Run To Cursor。

对于更加详细的情况请查阅 MSP-FET430 Users Guide.pdf。