

第 4 章 CCS 应用实验

目录

- 实验 4.1 CCS 操作使用实验
- 实验 4.2 汇编语言 DSP 程序设计实验
- 实验 4.3 C 语言 DSP 程序设计实验
- 实验 4.4 混合语言的DSP 程序设计实验
- 实验 4.5 双精度数据加减运算汇编语言实验
- 实验 4.6 单精度小数乘法运算汇编语言实验
- 实验 4.7 浮点数乘除法运算实验
- 实验 4.8 库函数调用运算实验

实验 4.1 CCS 操作使用实验

(一) 实验目的要求

1. 学习工程创建的方法
2. 了解编译和调试功能
3. 学习使用观察窗口

(二) 主要仪器设备

1. 计算机
2. CCS软件
3. TMS320C64xx实验箱一台

(三) 实验原理和方法

开发 TMS320C6xxx 应用系统一般需要以下几个调试工具来完成：

1. 软件集成开发环境（CCS）：完成程序编译、目标文件产生、下载，进行程序和硬件的联合仿真调试。
2. 仿真器：实现硬件仿真调试时与硬件系统的通信、控制和读取硬件系统的状态和数据。

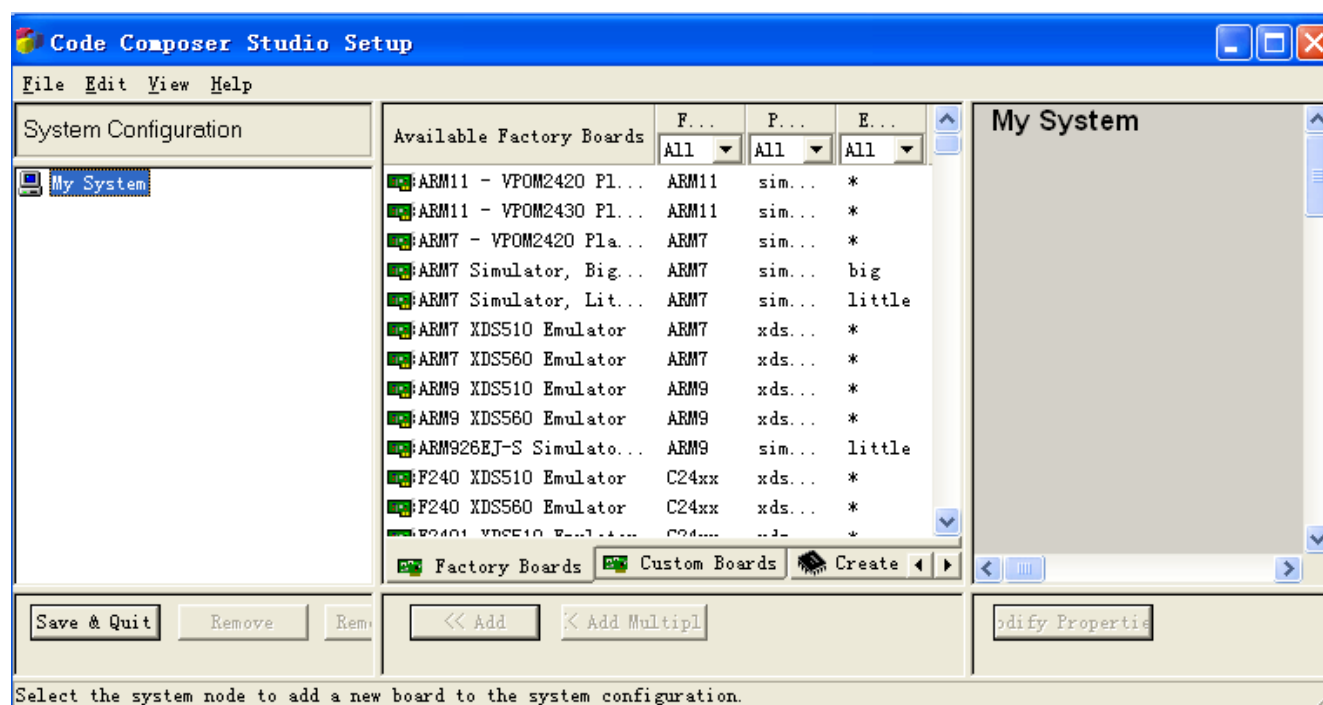
CCS 通过工程来管理文件，一般包括以下几种文件：

- 1) 源程序文件：C 语言或汇编语言文件 (*.c 或*.asm)
- 2) 头文件 (*.h)
- 3) 命令文件 (*.cmd)
- 4) 库文件 (*.lib,*.obj)

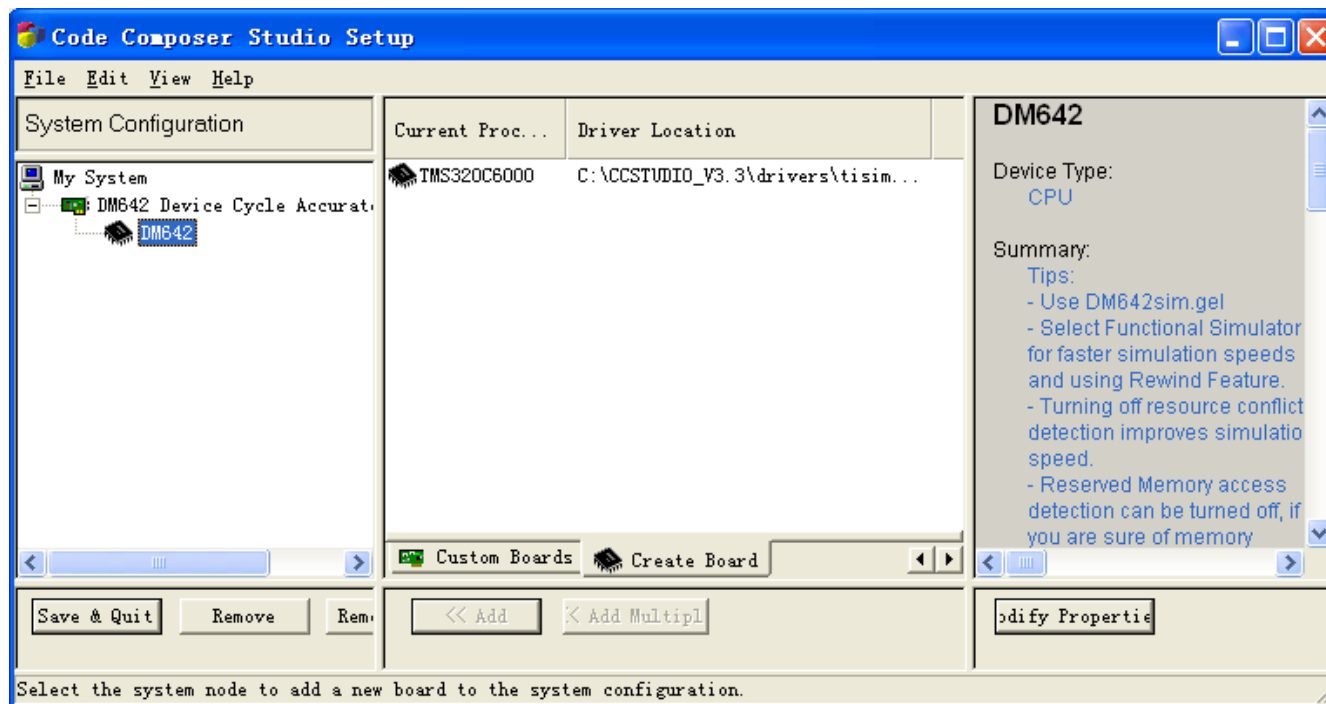
(四)实验内容与步骤

连接实验箱与PC机，打开电源给实验箱供电。

1. 在C:\CCStudio_v3.3\MyProjects下新建一个文件夹：**practice**。
复制C:\CCStudio_v3.3\tutorial\sim64xx\consultant下的所有文件到**practice**文件夹下。
2. 点击setup code composer studio v3.3,运行 CCS Setup,



设置 CCS 在软件仿真（simulator）方式下运行。

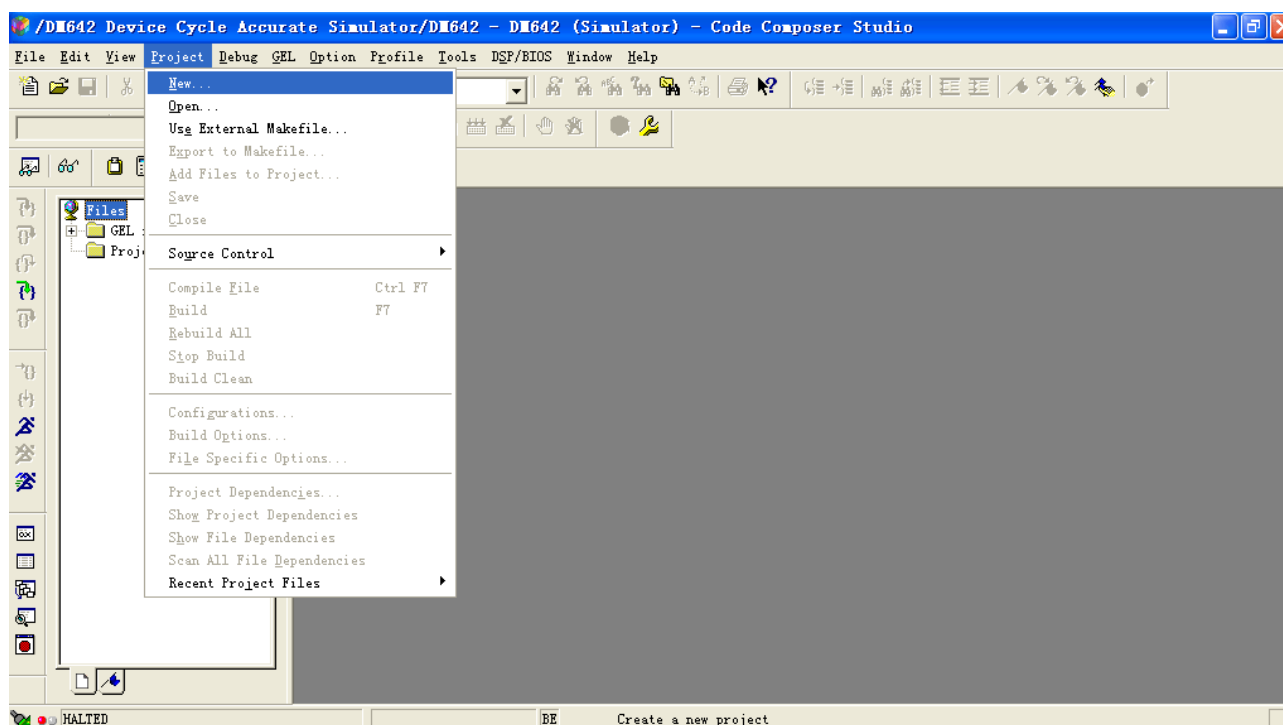


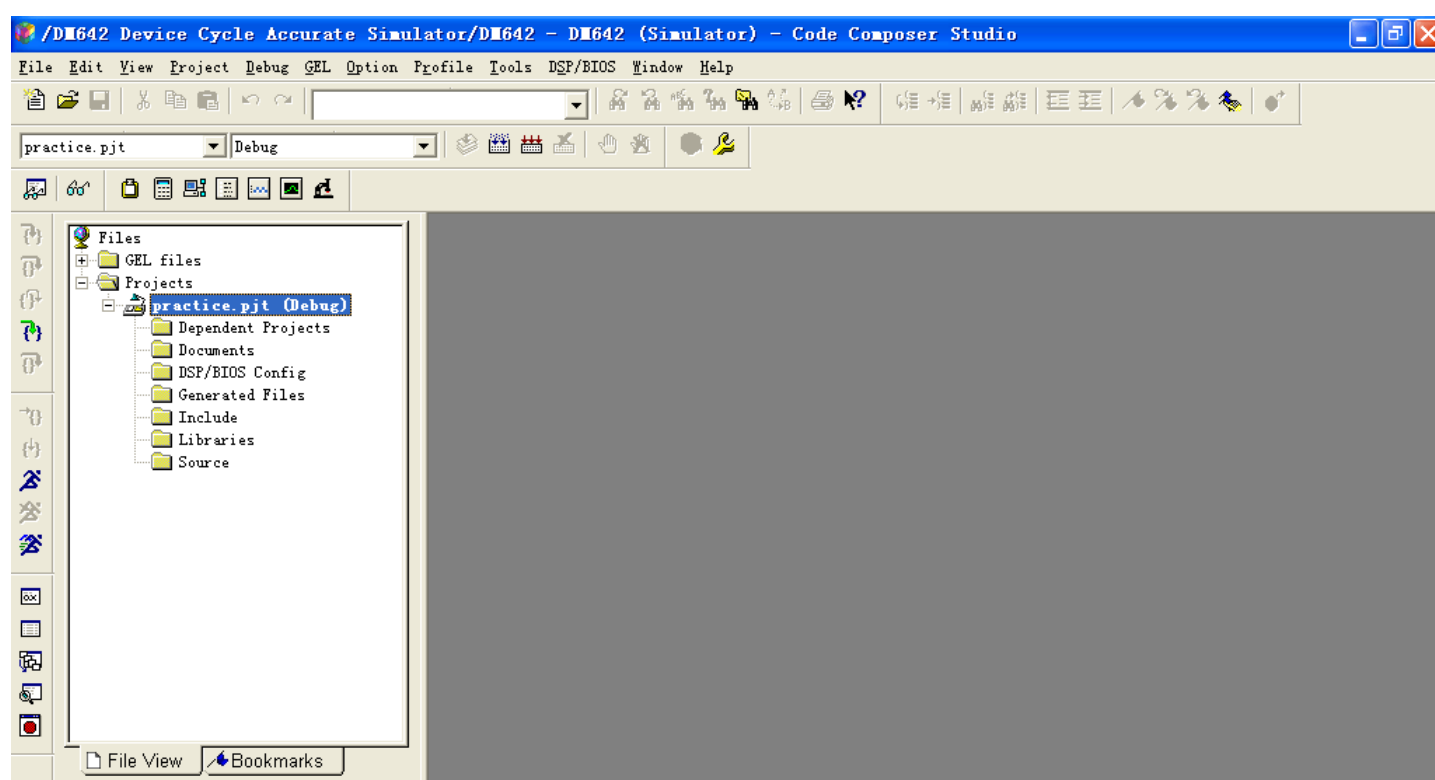
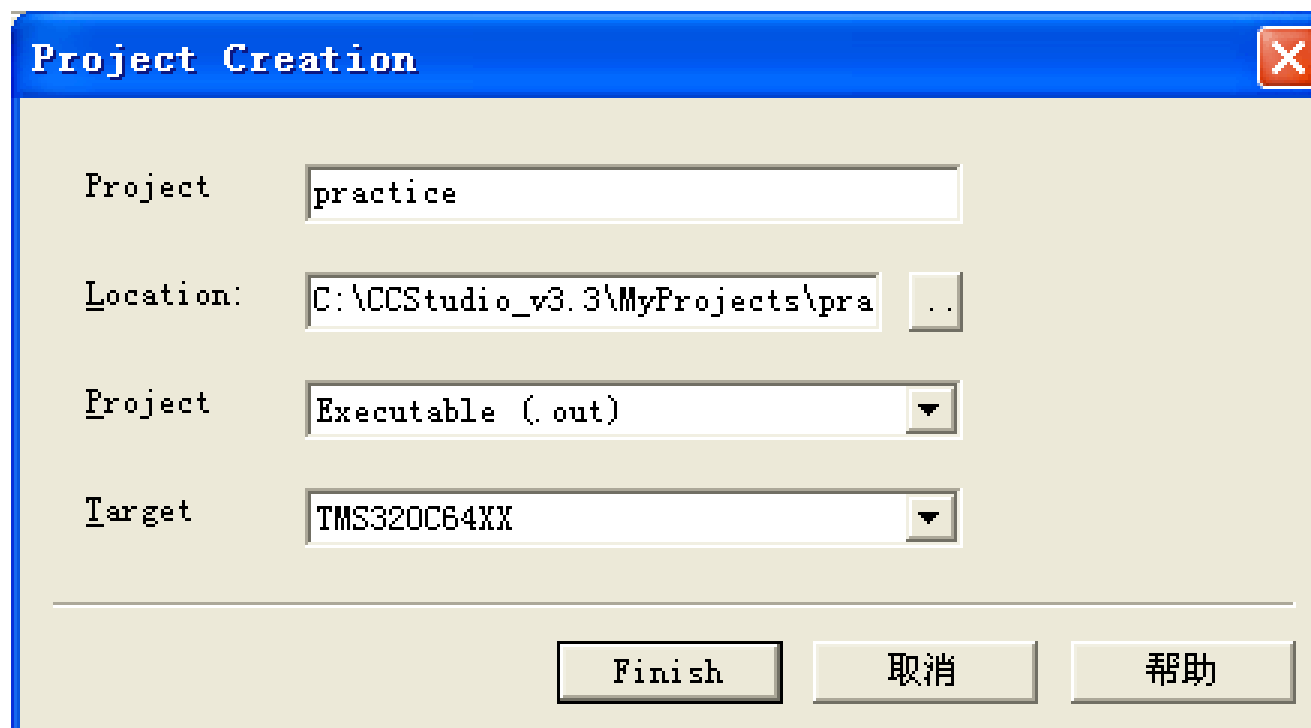
保存配置。

3. 点击 Code Composer Studio, 启动 CCS。

4. 创建工程

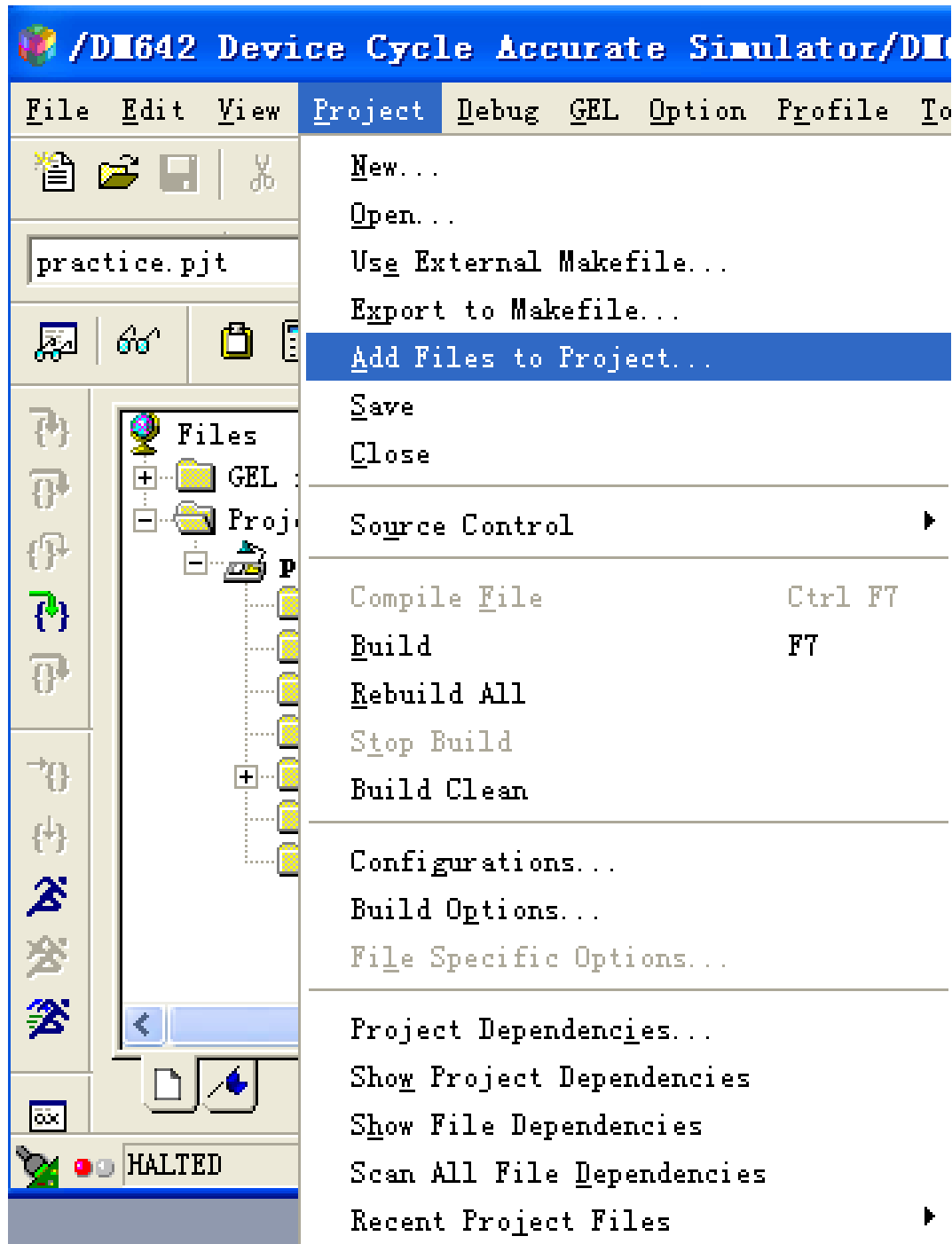
选择菜单“Project”的“New...”项，建立 practice.pjt 工程文件。

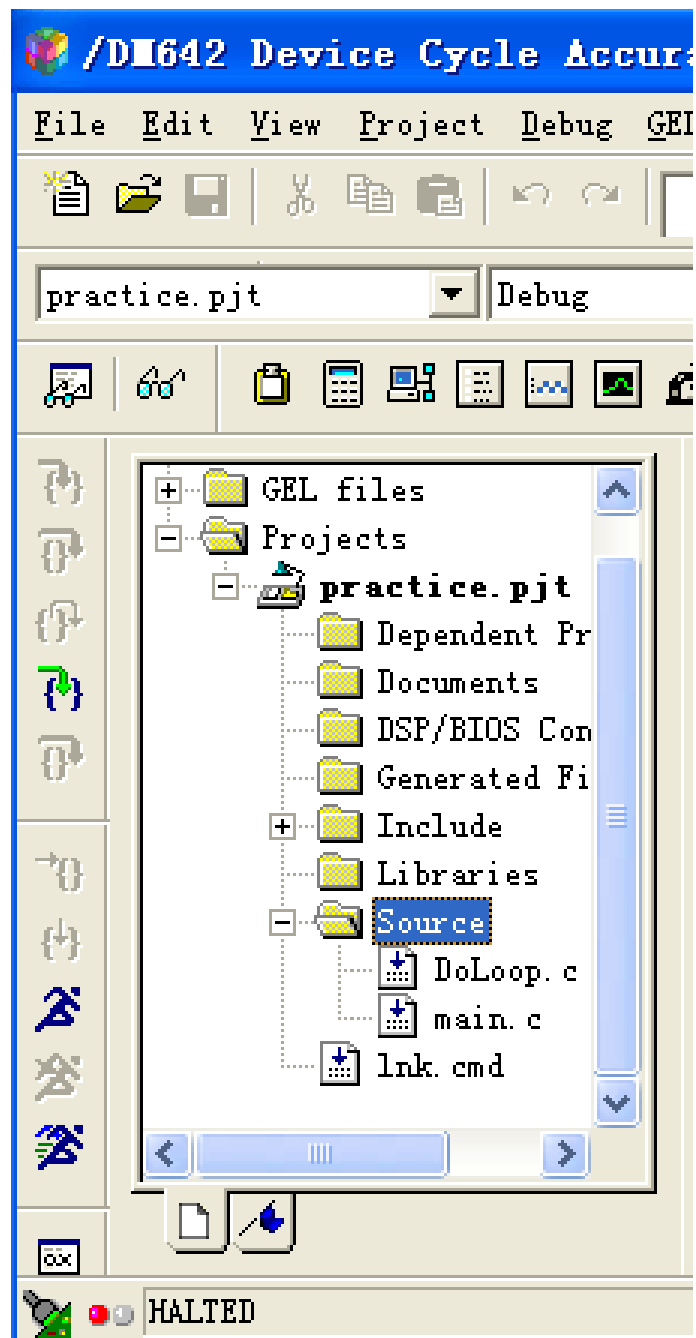




5. 向工程添加文件

添加 main.c,doloop.c,lnk.cmd 到工程。





查看源程序:DoLoop.c,main.c。

打开: lnk.cmd 文件, 编辑如下内容保存:

-stack 0x400

-heap 0x400

MEMORY

{

 ISRAM : origin=0x400, len = 0x1000000

}

SECTIONS

{

 .vectors > ISRAM

 .text > ISRAM

 .data > ISRAM

 .bss > ISRAM

 .cinit > ISRAM

 .const > ISRAM

 .far > ISRAM

 .stack > ISRAM

 .cio > ISRAM

 .systemem > ISRAM

}

或者:

```

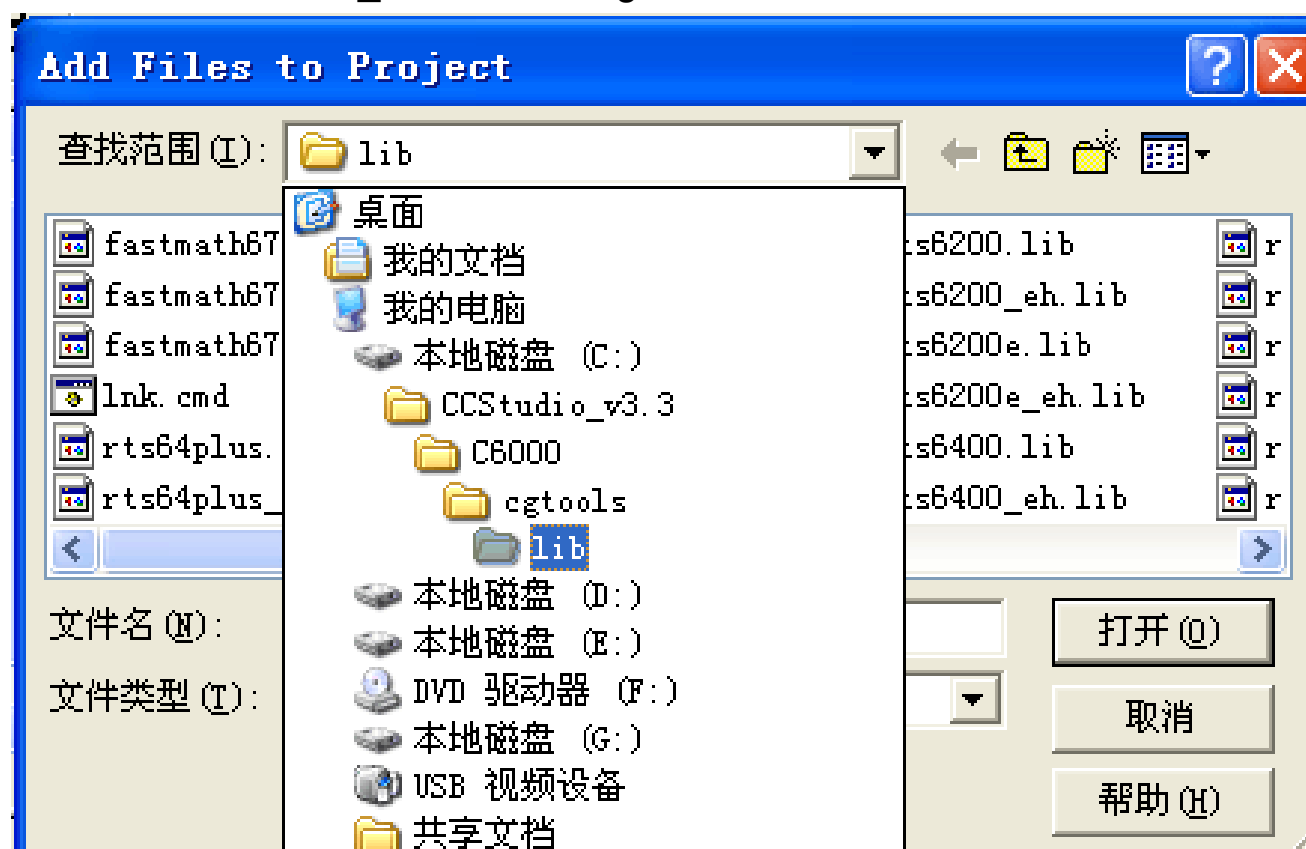
/*****
/*  c64xx_cov_lnk.cmd
/*  Copyright (c) 1996-2002    Texas Instruments Incorporated
/*****

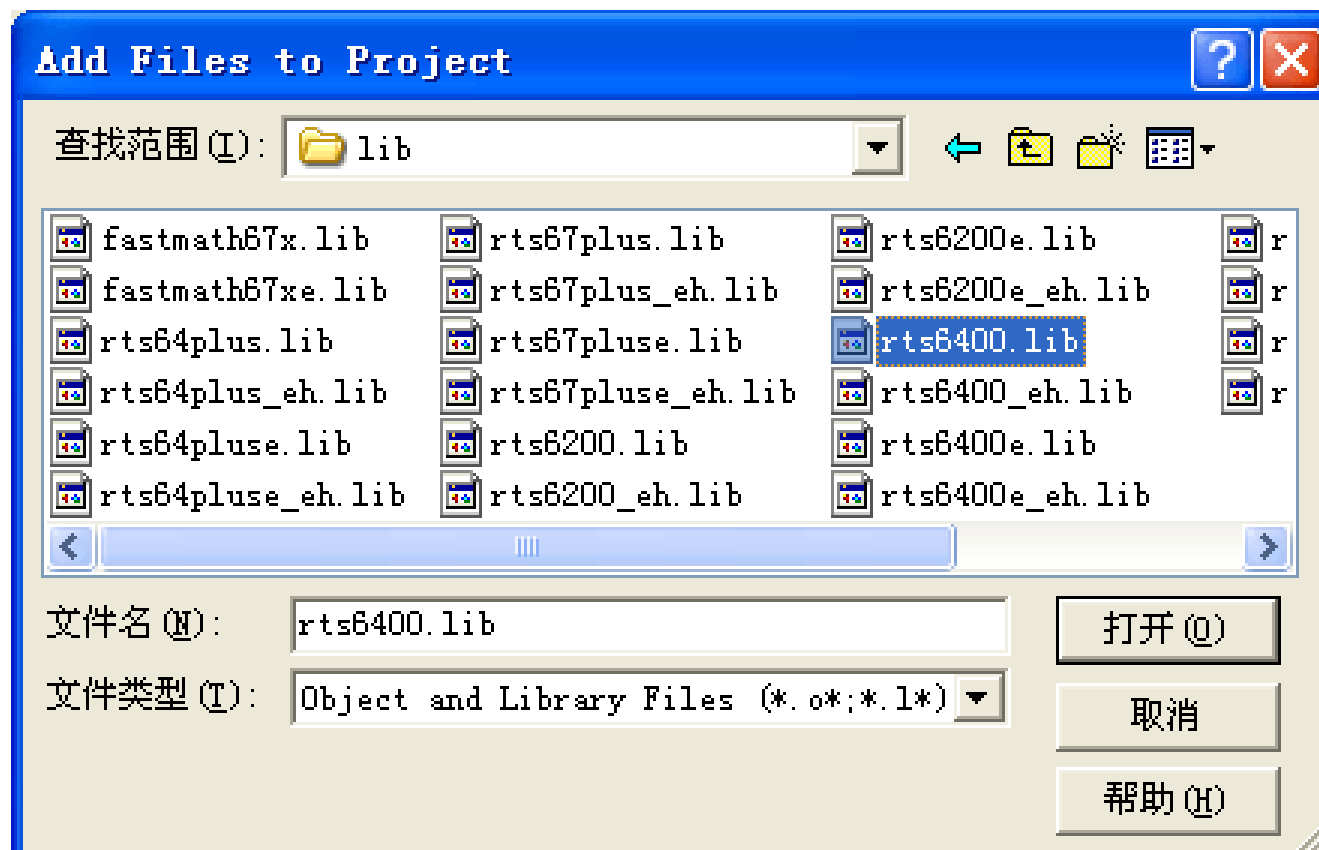
-c
-heap 5000
-stack 5000
/* Memory Map0 */
MEMORY
{
    ON_CHIP      :   origin = 00000000h           length = 00100000h
    EMIFA_CE0:   origin = 80000000h           length = 10000000h
    EMIFA_CE1:   origin = 90000000h           length = 10000000h
}
SECTIONS
{
    .text        >   ON_CHIP
    .stack       >   ON_CHIP
    .bss         >   ON_CHIP
    .cinit       >   EMIFA_CE1
    .cio         >   EMIFA_CE0
    .const       >   ON_CHIP
    .data        >   ON_CHIP
    .switch      >   ON_CHIP
    .systemem    >   ON_CHIP
    .far         >   EMIFA_CE0
}

```

5. 工程中添加库文件

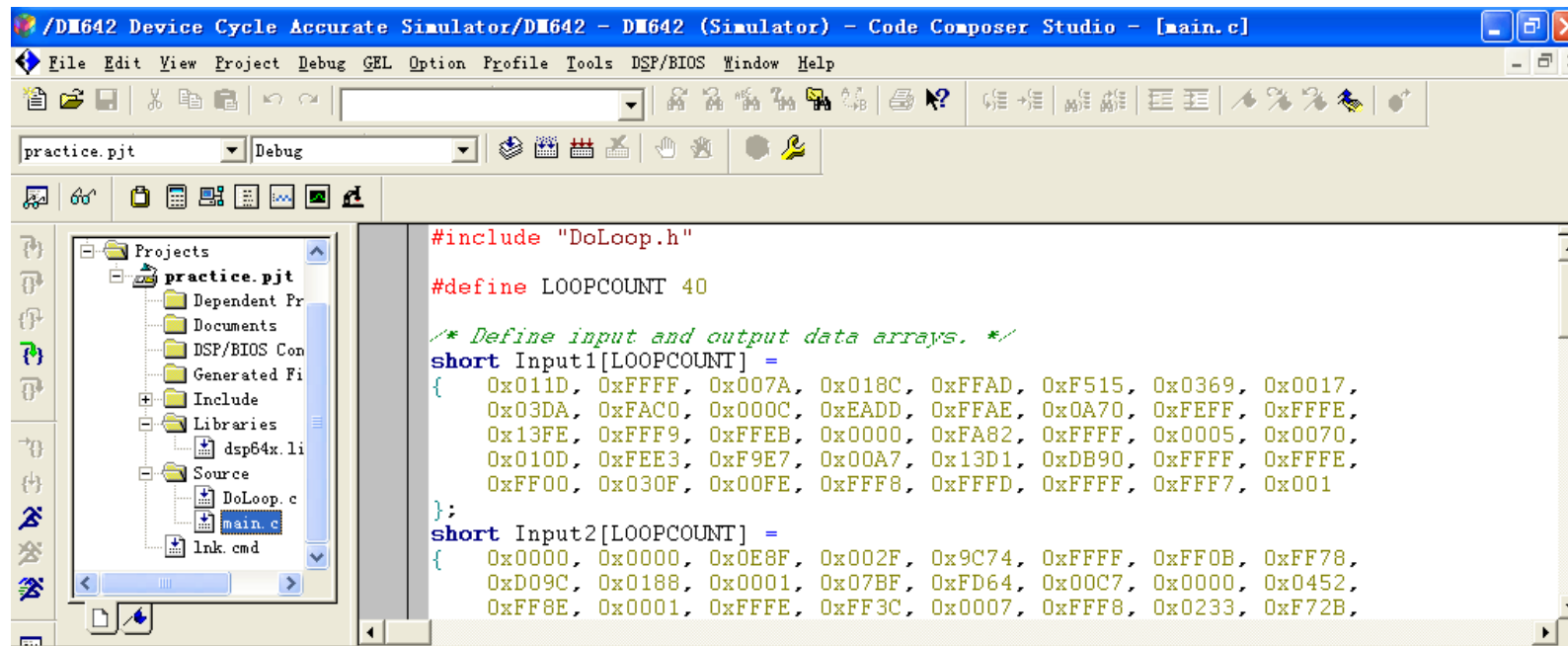
添加 C:\CCStudio_v3.3\C6000\cgtools\lib 目录下的 rts6400.lib





6. 查看源程序文件

点击工程列表下文件，可以打开源程序文件main.c 查看代码

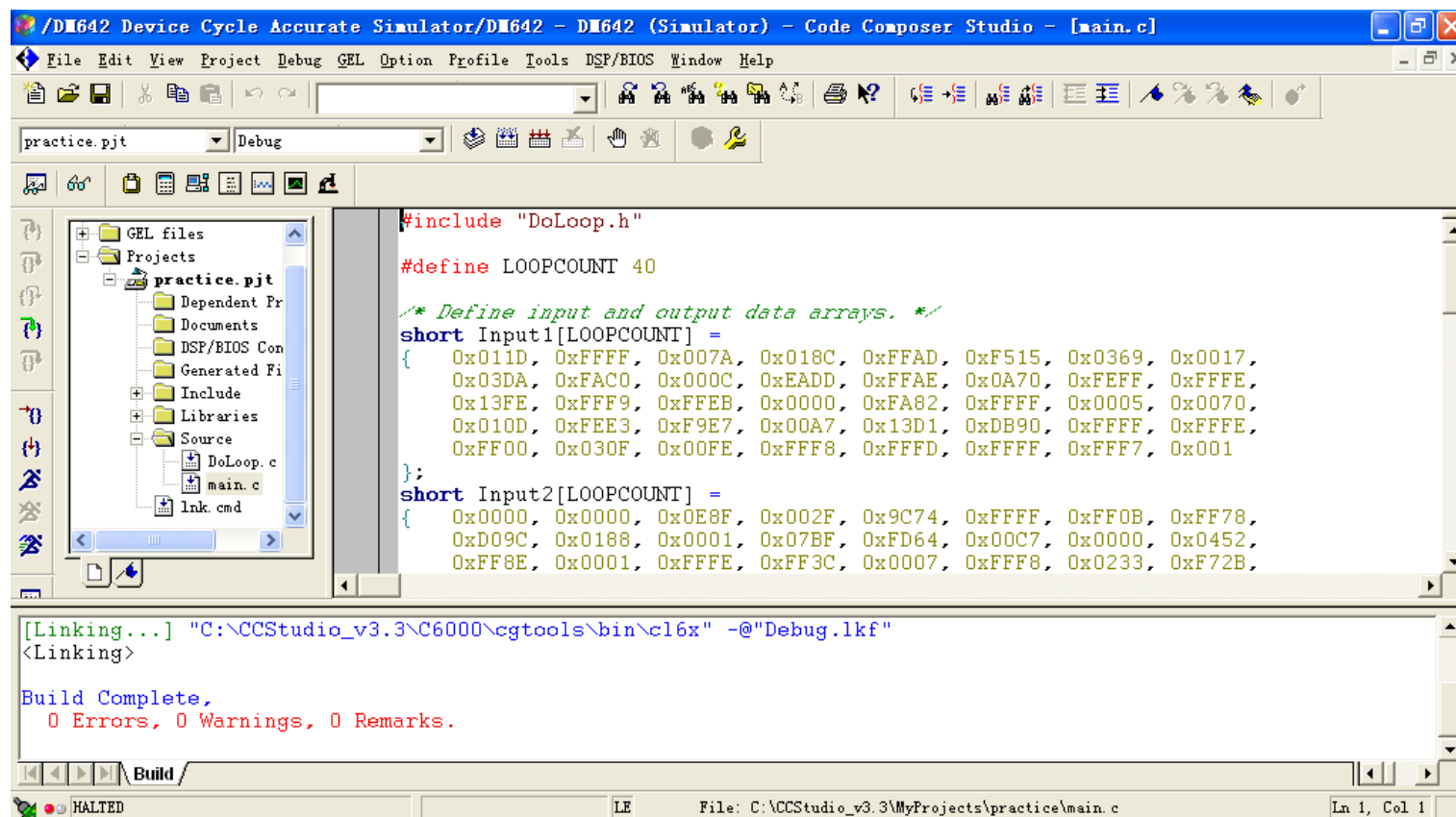


7. 编译、创建目标文件

点击菜单project->compile file,可以对文件进行编译

再点击菜单Project->Build, 建立目标文件

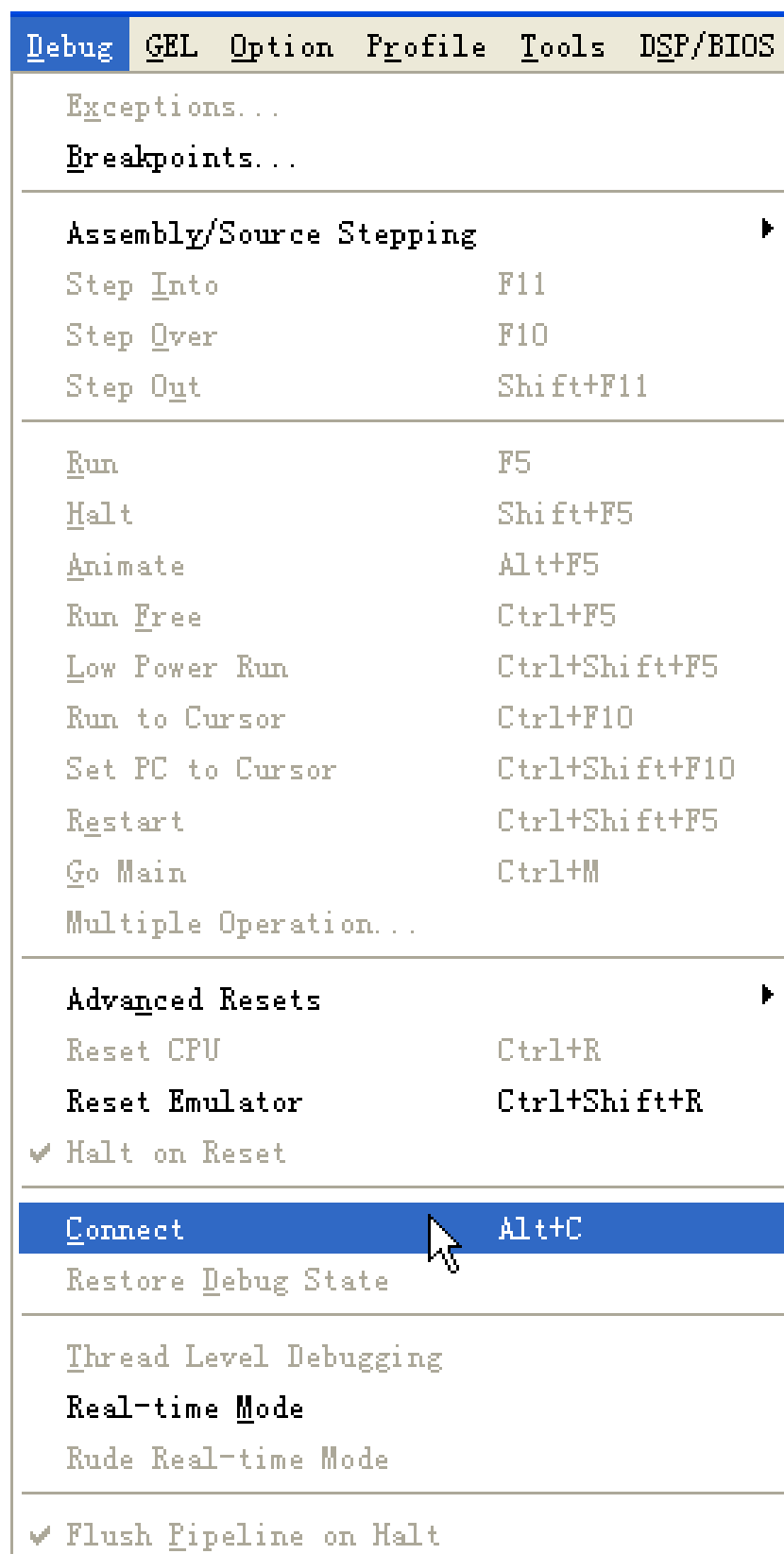
成功建立的文件应该是错误和警告都为0



8. 调用目标文件

1) 连接仿真器与目标板

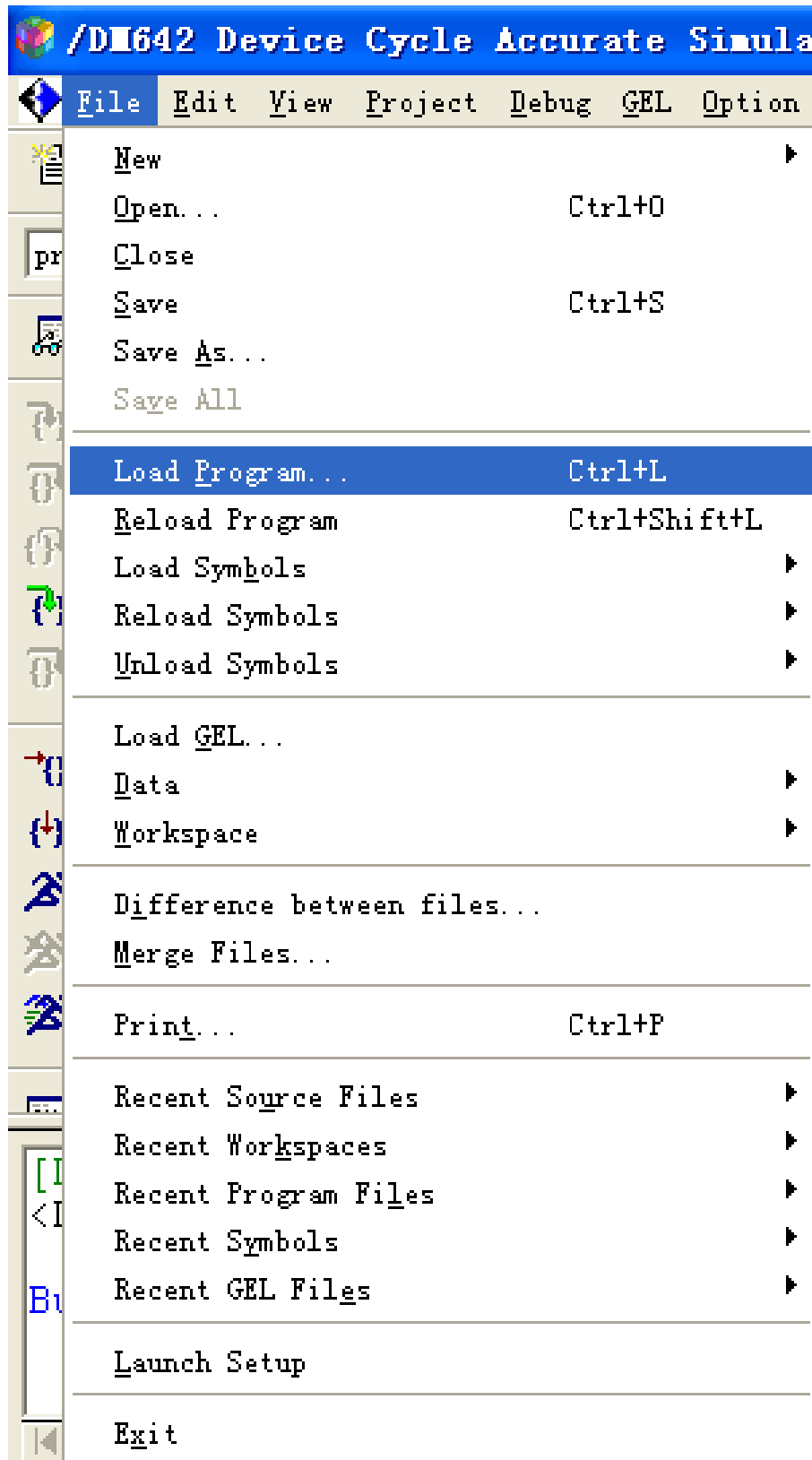
点击: **Debug->connect**,

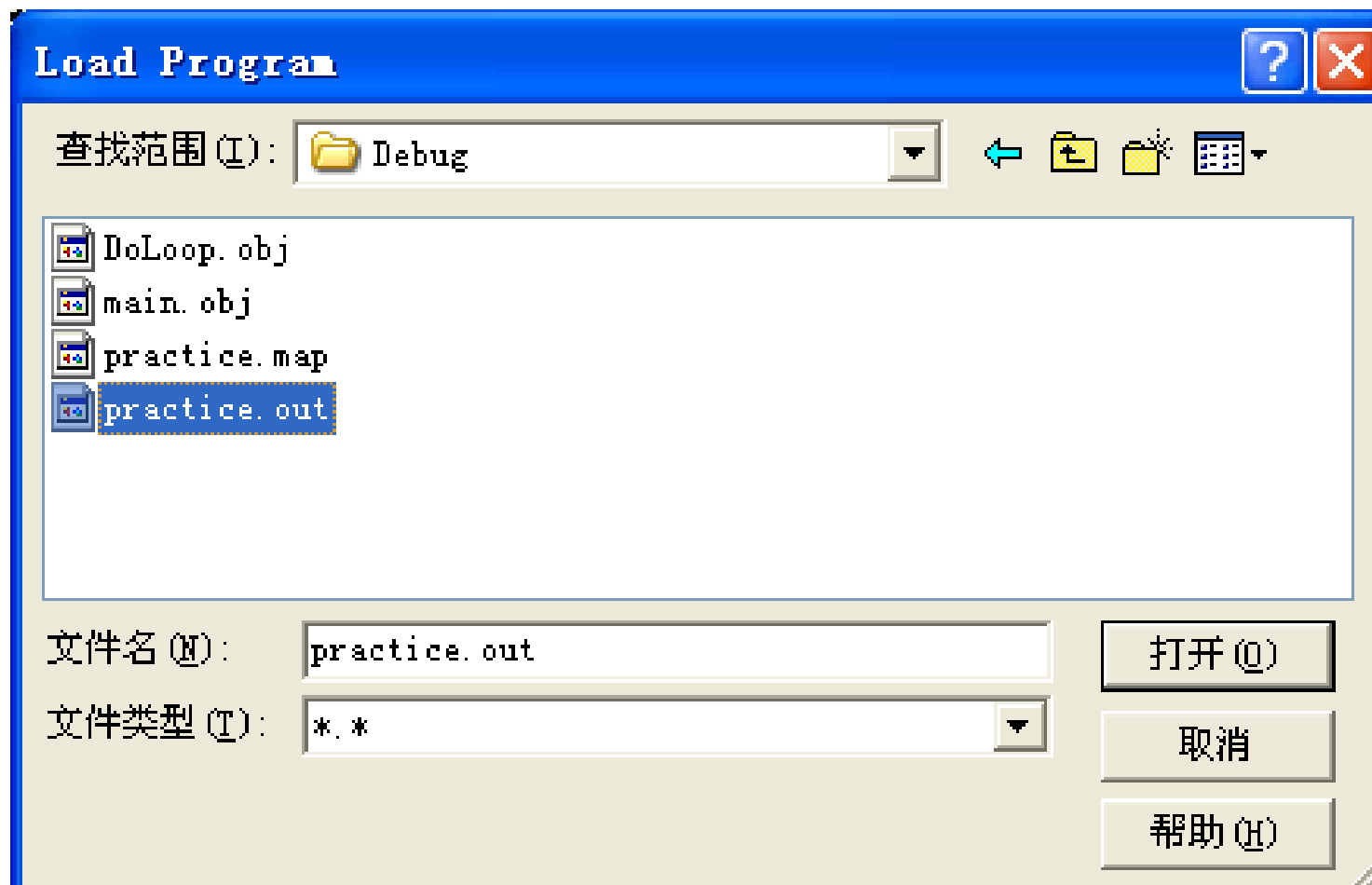




连接成功后，CCS 左下角有提示。

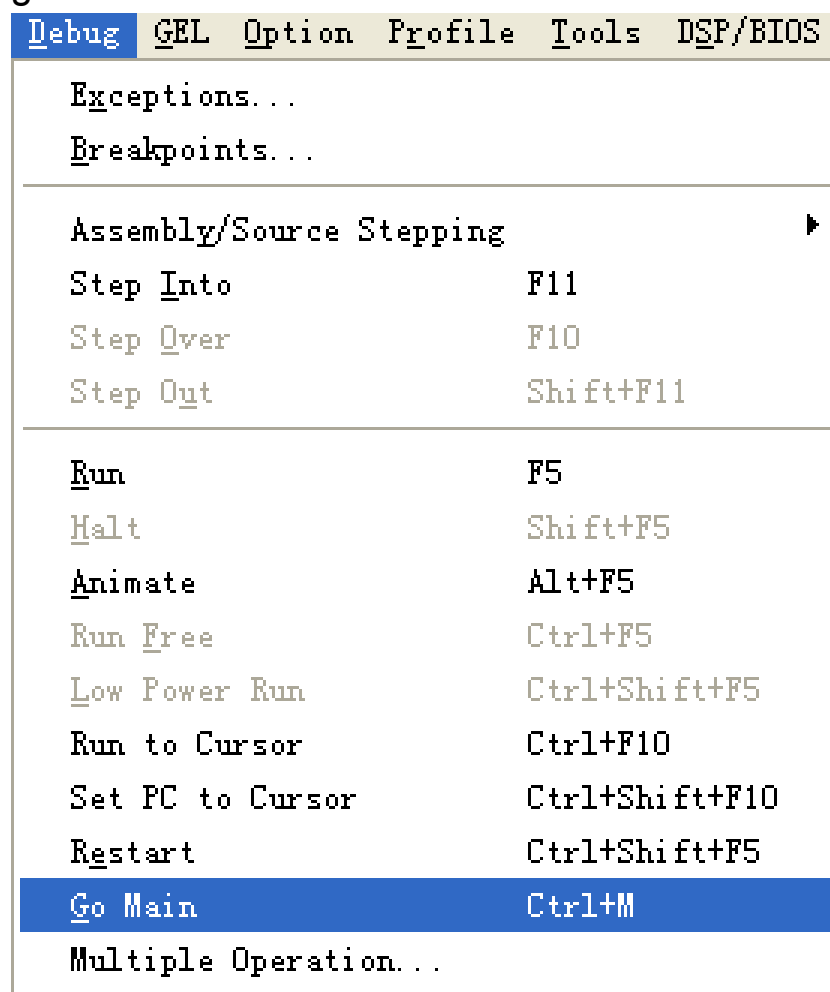
- 1) 点击 File->Load Program,选择\practice\Debug\practice.out

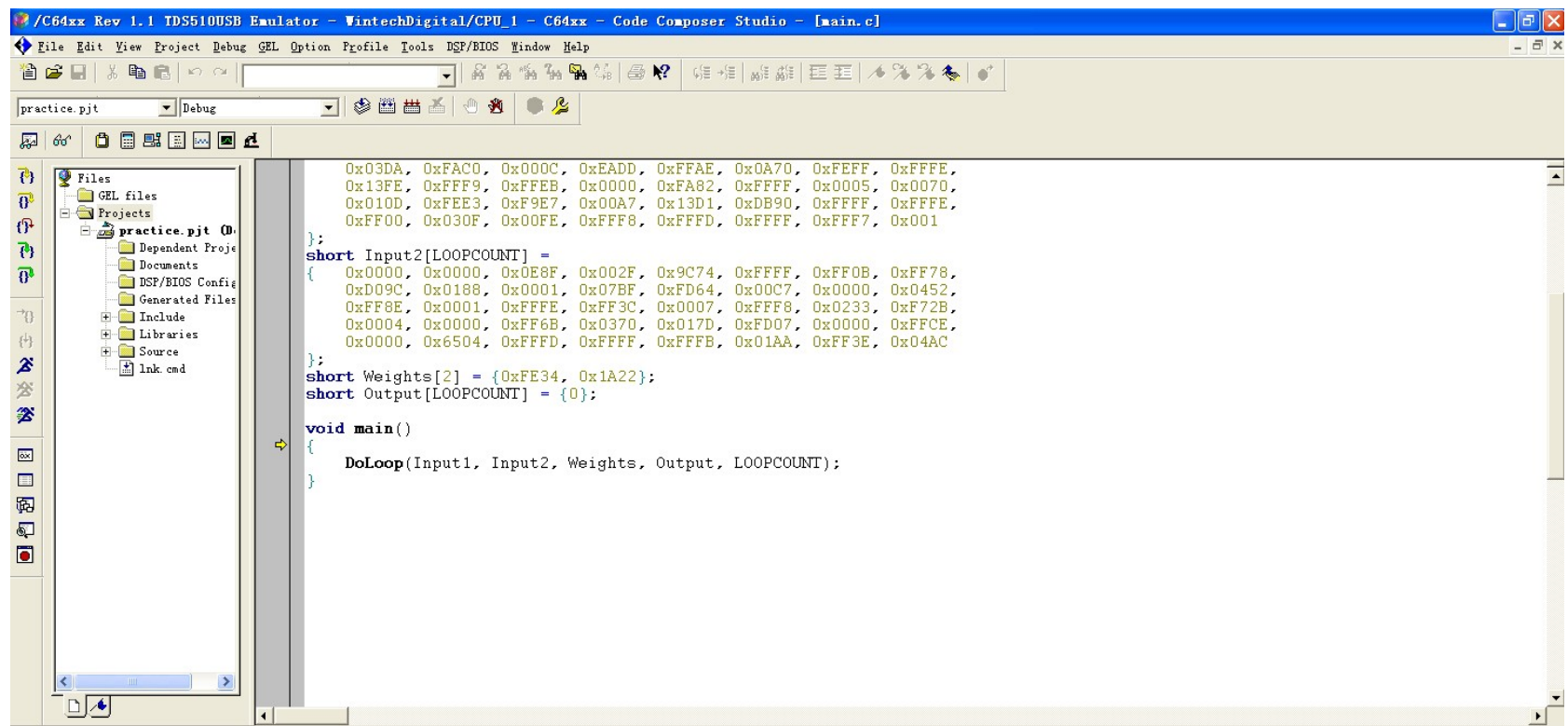




9. 调试1)

goto main



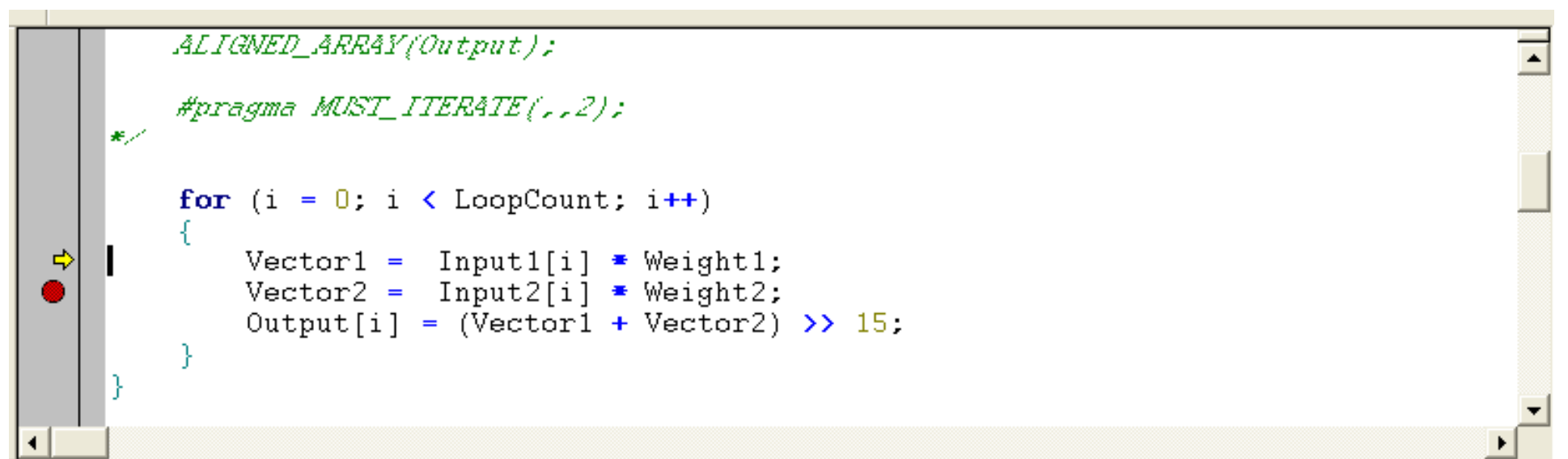


2) 设置断点

在需要设置断点的行前双击鼠标，则在行前出现红色的原点，表示设置了断点；若在已经设置了断点的行前双击鼠标，则可以取消该断点。

3) 单步运行

用 F11



4) 观察窗口

点击 view->watch window,则打开观察窗口

Name	Value	Type	R...
+ Input1	0x02000000	s...	hex
+ Input2	0x02000050	s...	hex
LoopCount	40	int	dec
+ Output	0x020000A0	s...	hex
Vector1	0	int	dec
Vector2	0	int	dec
Weight1	0	s...	dec

Watch Locals Watch 1

(五) 实验报告要求

1. 写出实验报告，包括：目的和要求、仪器设备
2. 写出实验步骤及内容

实验 4.2 汇编语言 DSP 程序设计实验

(一) 实验目的和要求:

1. 学习用汇编语言编制程序;
2. 了解在CCS 下开发TMS320C64x 汇编语言程序的步骤;
3. 掌握汇编语言数据搬移、存储的方法;
4. 学习在CCS 环境中调试汇编代码

(二) 主要仪器设备:

- 1.计算机
- 2.CCS软件
- 3.TMS320C64xx实验箱一台

(三) 实验原理与方法

1. 汇编语言程序在执行时直接从用户指定入口开始。
2. 由于 CCS 的代码链接器默认支持 C 语言,在编制汇编语言程序时,需要设置链接参数,选择非自动初始化,注明汇编程序的入口地址。

(四) 实验步骤与内容

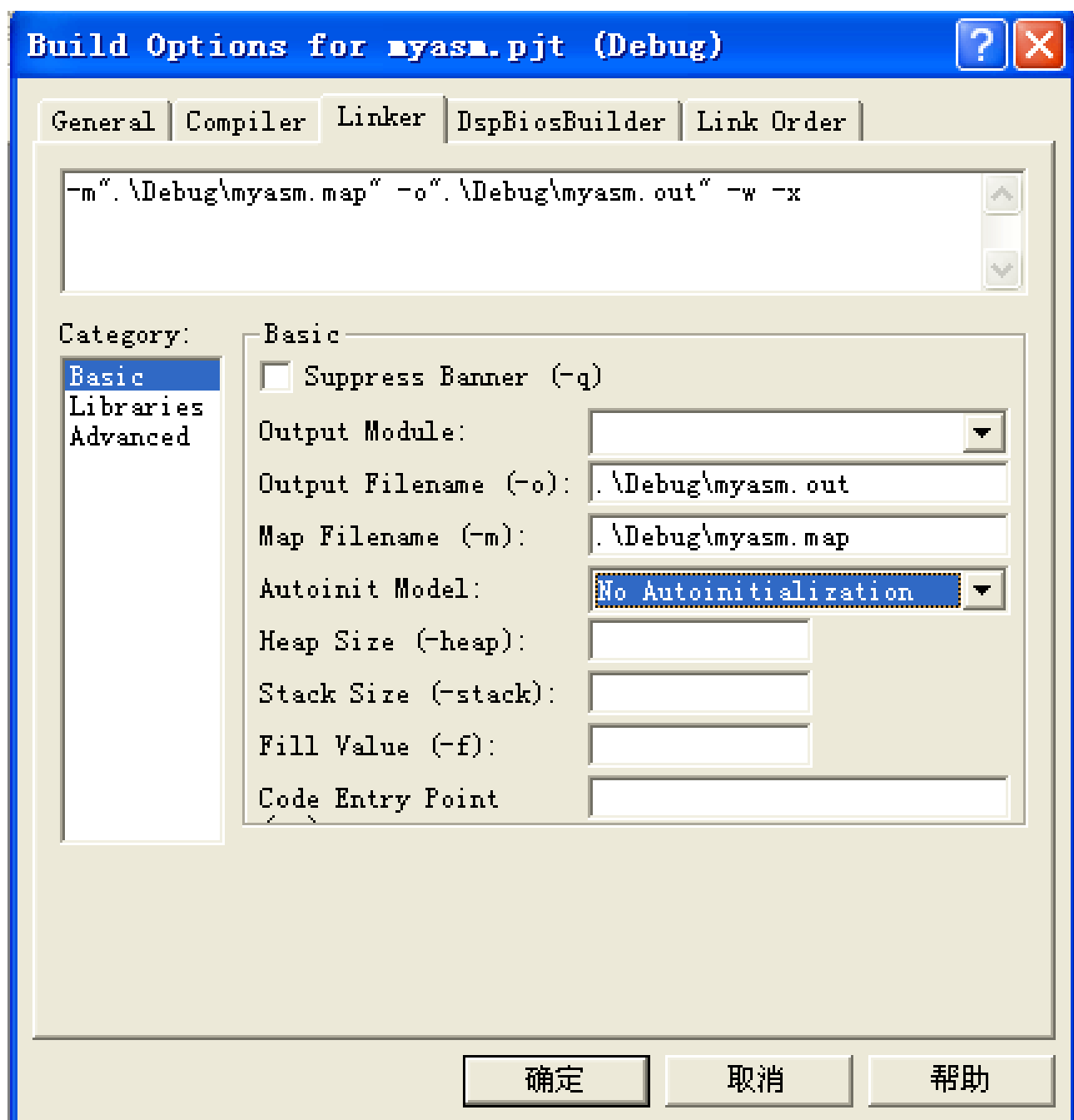
1. 运行 CCS Setup, 设置 CCS 在模拟仿真 (Simulator) 方式下运行。选择模拟仿真调DM642 Device Cycle Accurate Simulator(Endianness选little)。
2. 启动 CCS
3. 创建工程



4. 设置工程编译选项

选择 CCS 菜单 project->Build Options,点击 Linker 页, 设置Autoinit Model为:
No Autoinitialization(这点是与 C 语言程序设计不同)

注意: 若省略这一步, 也不影响程序建立, 只不过会出现警告: warning: entry point symbol _c_int00 undefined



1. 输入源程序

1) 输入汇编源程序

选择 File→New→Source File 或使用工具条的快捷按钮 , 进入文件编辑窗口。

录入汇编源程序:

```

;*****
; 输入数据存放在: x,y
; 本例程完成运算: x+y
; 结果存放: z
; 其中: x,y,z 都占 32 位
;*****
.bss x, 4      ;排版要空一格, 下同
.bss y, 4
.bss z, 4
;.def start
;.def x, y, z

.text
start:        ;排版要顶格
    mvkl x, A0;立即寻址, 将变量x 的值放到寄存器A0
    mvkh x, A0;立即寻址, 将变量x 的值放到寄存器A0
    mvkl y, B0;立即寻址, 将变量y 的值放到寄存器B0
    mvkh y, B0;立即寻址, 将变量y 的值放到寄存器B0

```

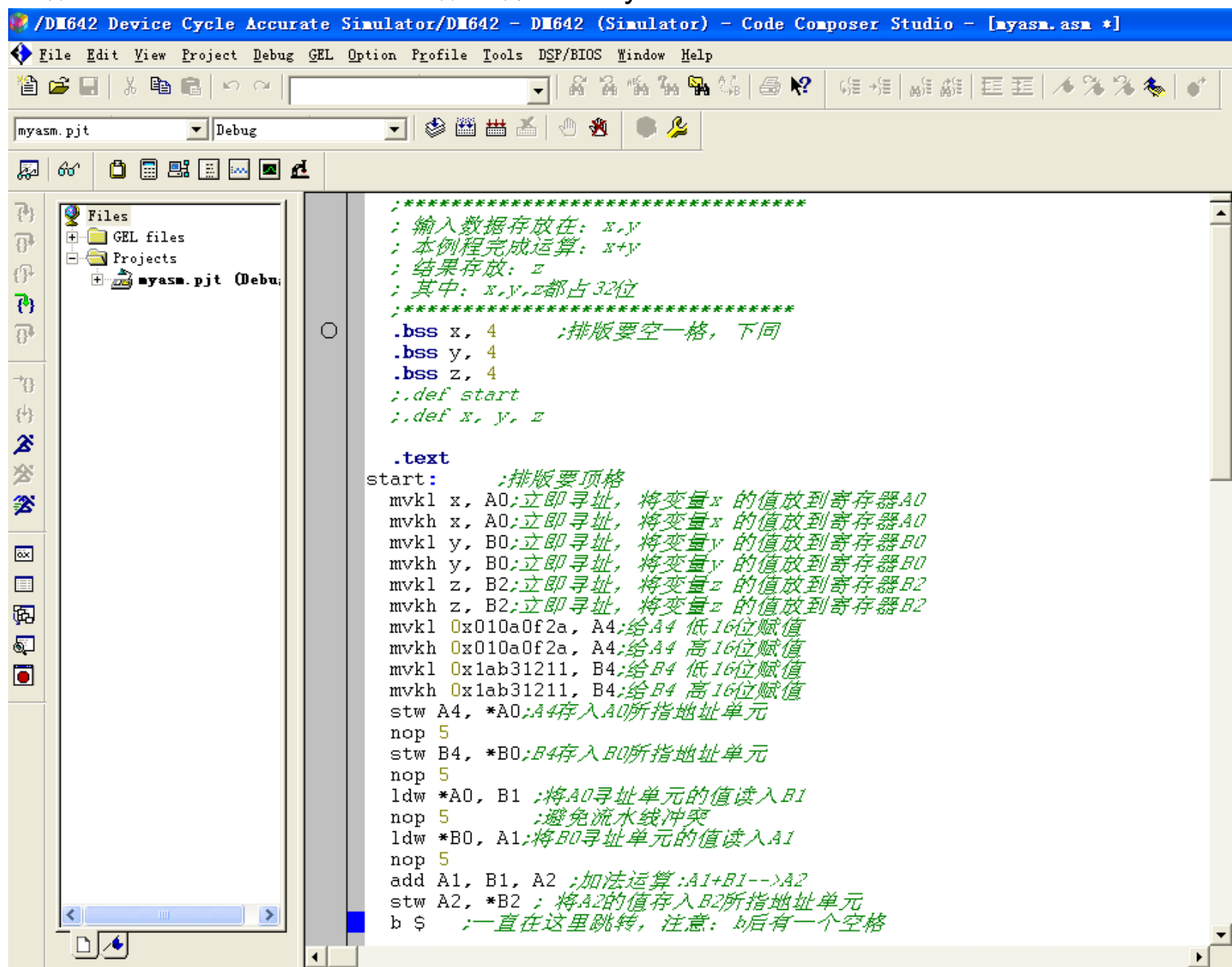
```

mvkl z, B2;立即寻址, 将变量z 的值放到寄存器B2
mvkh z, B2;立即寻址, 将变量z 的值放到寄存器B2
mvkl 0x010a0f2a, A4;给 A4 低 16 位赋值
mvkh 0x010a0f2a, A4;给 A4 高 16 位赋值
mvkl 0x1ab31211, B4;给 B4 低 16 位赋值
mvkh 0x1ab31211, B4;给 B4 高 16 位赋值
stw A4, *A0;A4 存入 A0 所指地址单元
nop 5
stw B4, *B0;B4 存入 B0 所指地址单元
nop 5
ldw *A0, B1 ;将 A0 寻址单元的值读入 B1
nop 5 ;避免流水线冲突
ldw *B0, A1;将 B0 寻址单元的值读入 A1
nop 5
add A1, B1, A2 ;加法运算:A1+B1-->A2
stw A2, *B2 ; 将 A2 的值存入 B2 所指地址单元
b $ ;一直在这里跳转, 注意: b 后有一个空格

```

注意: 汇编语句前需要留出空格, 标号顶格输入, 否则编译有错误提示。

保存源程序到工程目录下, 保存文件为: myasm.asm



2) 输入链接命令文件

点击“File/New/Source File”或使用工具条的快捷按钮, 进入文件编辑窗口。输入链接命令文件内容:

- stack 400

```

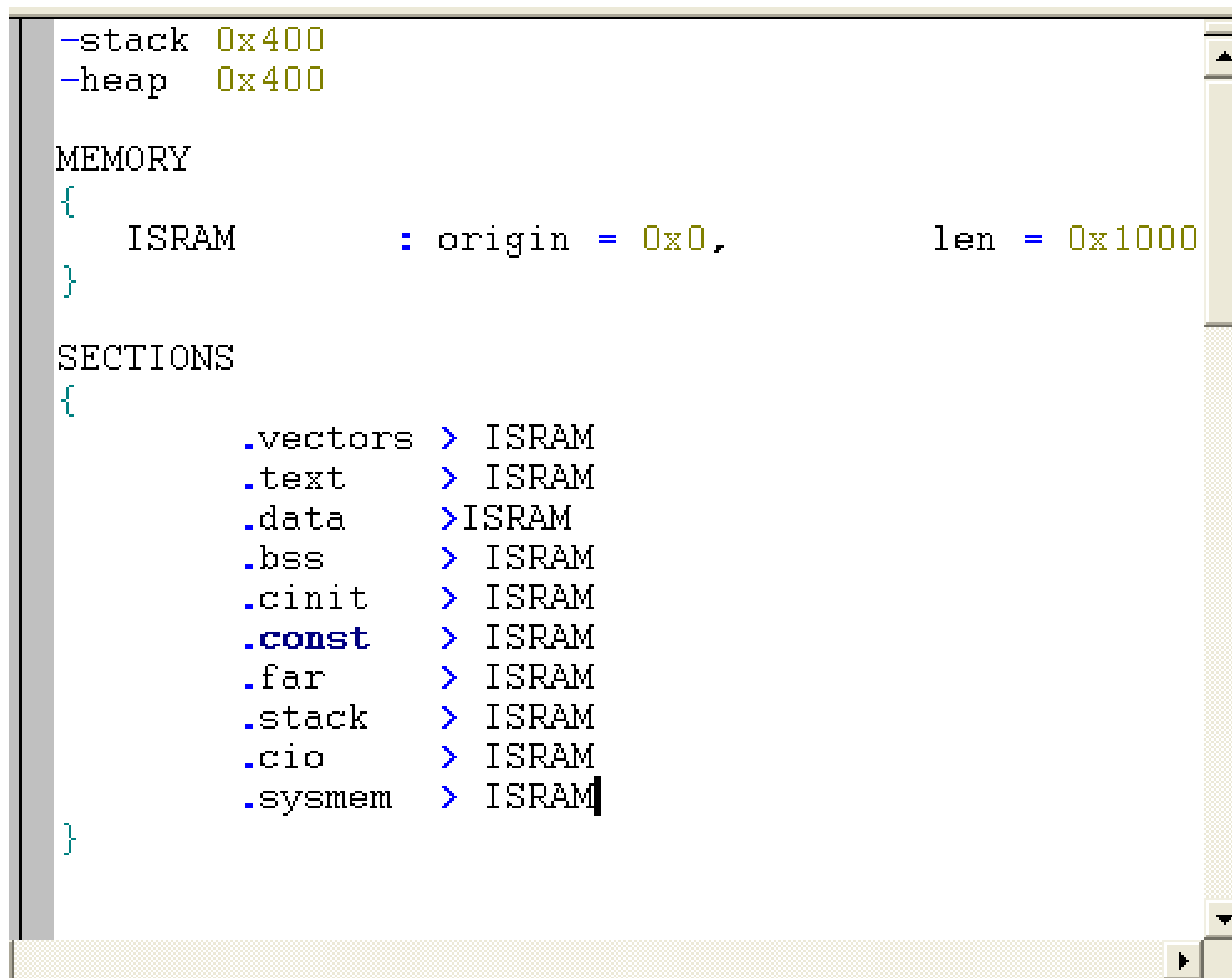
heap 400
MEMORY
{
    ISRAM      : origin = 0x0,          len= 0x1000000
}

SECTIONS
{
    .vectors > ISRAM
    .text    > ISRAM

    .bss     > ISRAM
    .cinit   > ISRAM
    .const   > ISRAM
    .far     > ISRAM
    .stack   > ISRAM
    .cio     > ISRAM
    .system  > ISRAM
}

```

保存源程序到工程目录下，命名为myasm.cmd。



```

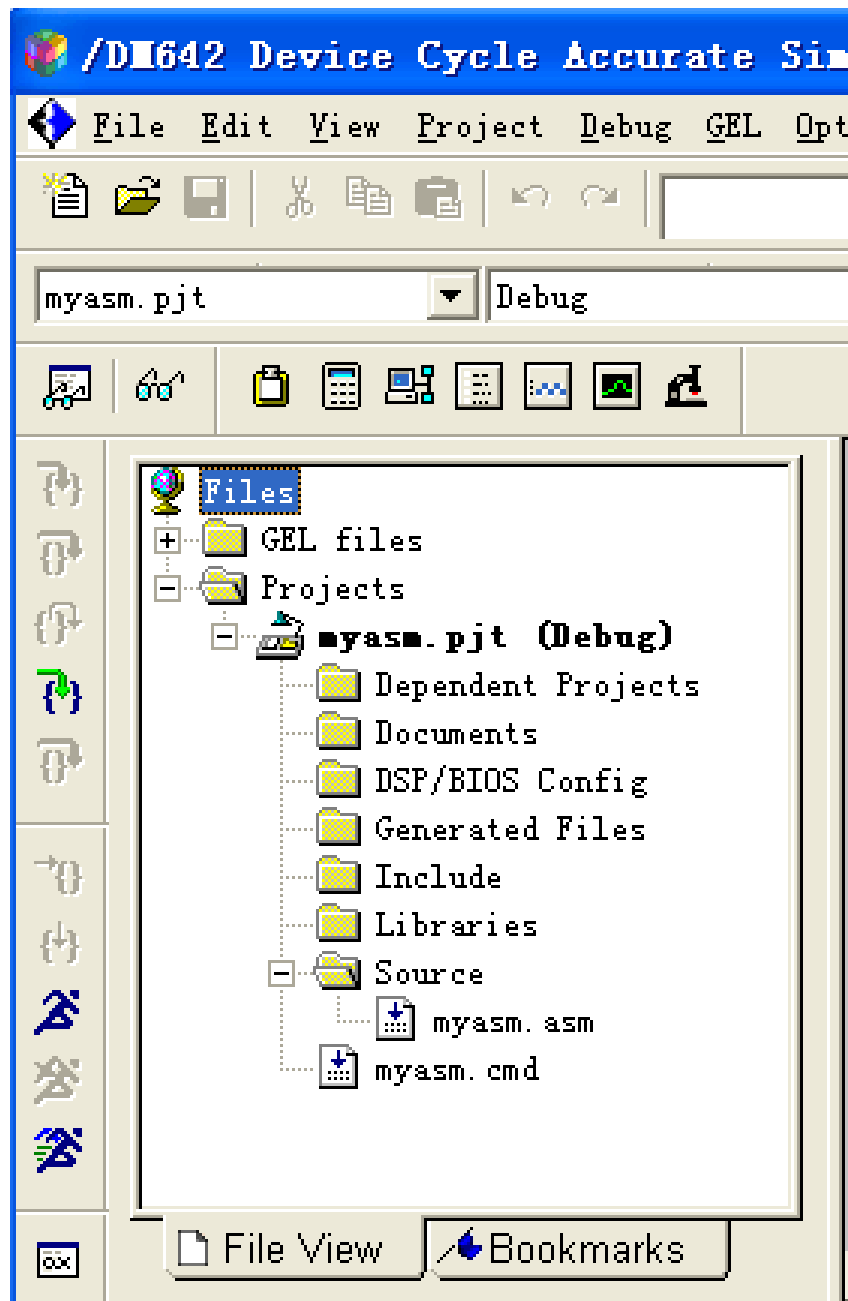
-heap 0x400
-stack 0x400

MEMORY
{
    ISRAM      : origin = 0x0,          len = 0x1000
}

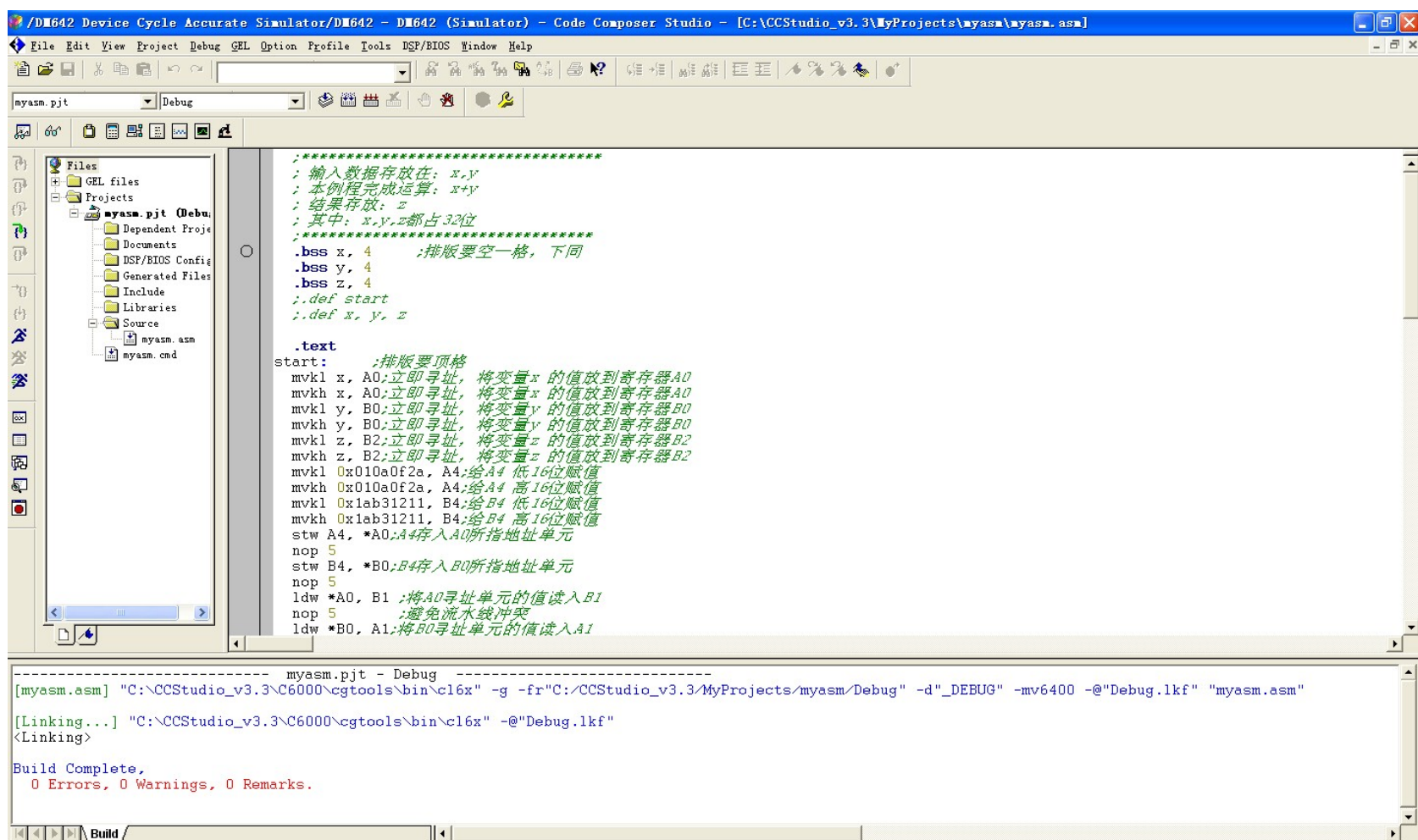
SECTIONS
{
    .vectors > ISRAM
    .text    > ISRAM
    .data    > ISRAM
    .bss     > ISRAM
    .cinit   > ISRAM
    .const   > ISRAM
    .far     > ISRAM
    .stack   > ISRAM
    .cio     > ISRAM
    .system  > ISRAM
}

```

1. 将上述编辑的源程序myasm.asm和链接命令文件myasm.cmd加入工程 myasm.pjt。

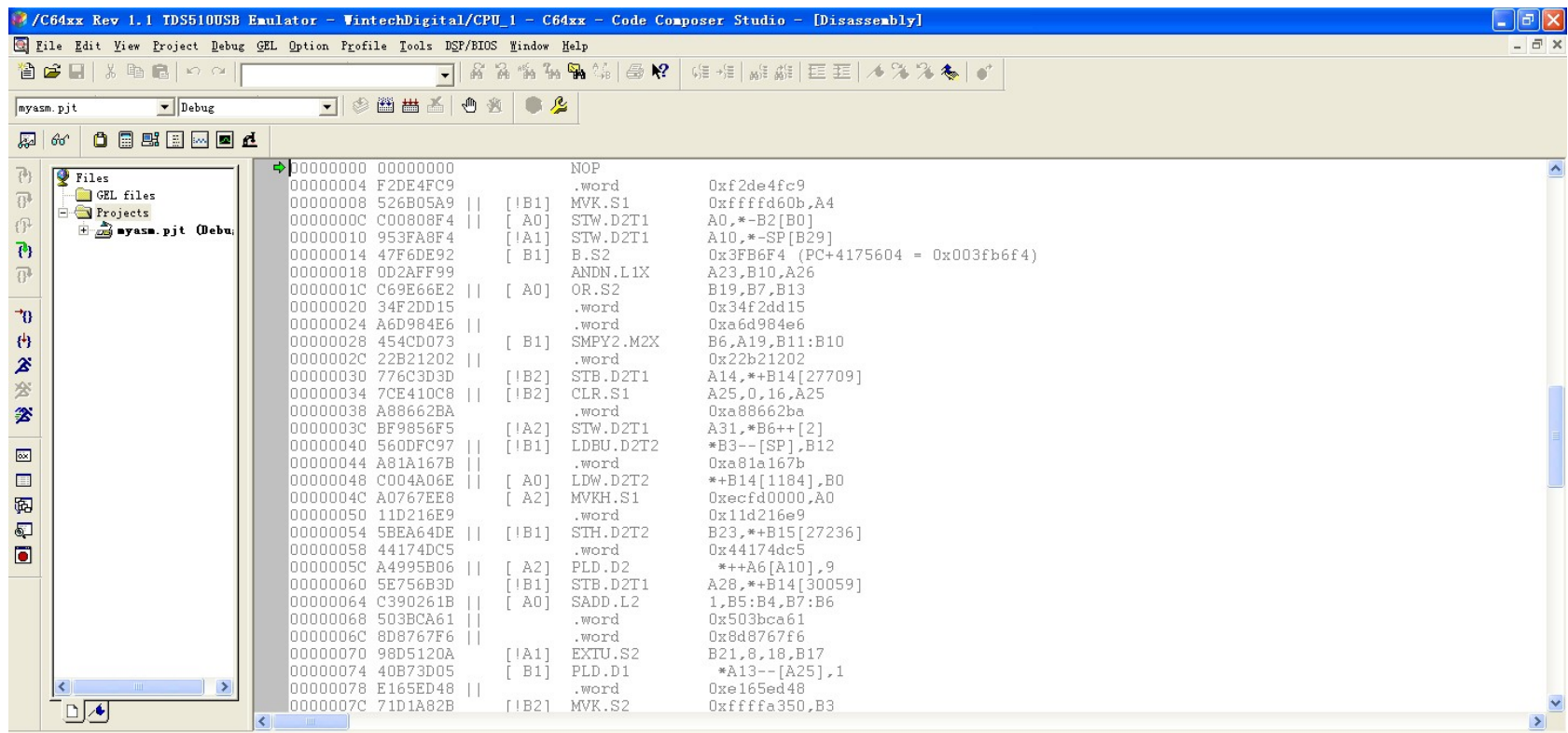


1. 编译源文件、建立目标文件 (.out)



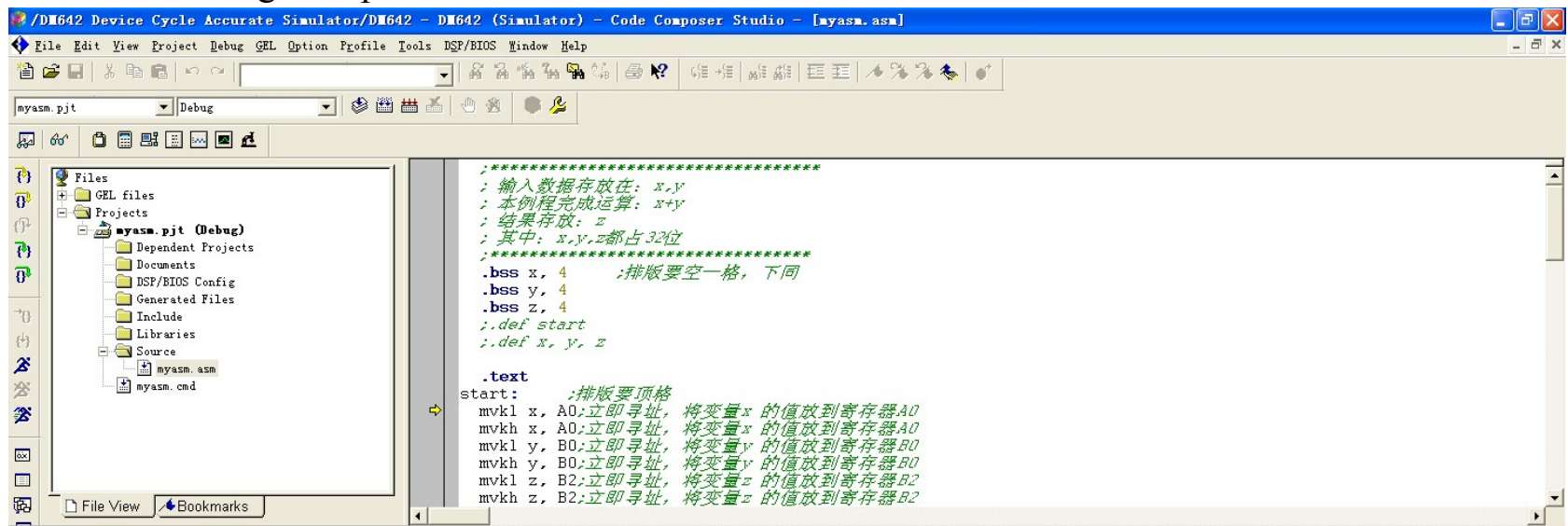
2. 调用编程可执行程序

选择File->Load Program,选择\myasm\debug\myasm.out, 点击Debug->Restart后PC此时指向00000000 00000000地址。



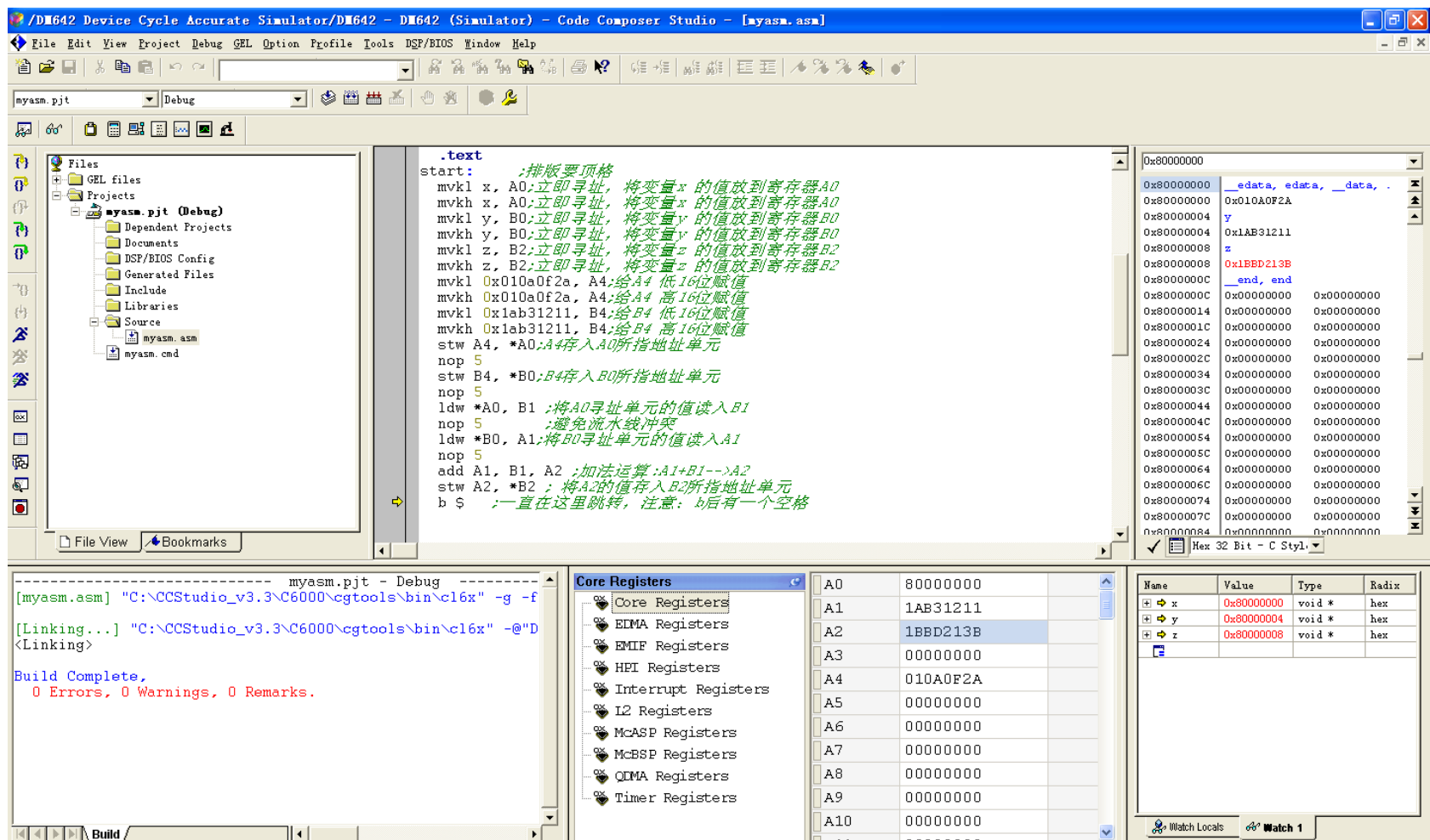
1. 调试

用菜单 debug->Step Into 或 F11 可以单步运行。



2. 观察

开启 CPU 寄存器观察窗口：单击菜单 View Registers core 使用单步运行，查看相应寄存器的值的变化。还可开启 Watch window, Memory 窗口，进行变量、存储空间的观察



3. 退出 CCS

(五) 实验报告要求

1. 写出实验报告，包括：目的和要求、仪器设备
2. 写出实验步骤及内容

实验 4.3 C 语言 DSP 程序设计实验

(一) 实验目的和要求：

1. 掌握DSP 芯片的C 语言程序设计方法；
2. 掌握在CCS 环境下用C 语言开发的步骤；
3. 学会调试CCS 下的C 语言代码

(二) 主要仪器设备：

1. 计算机
2. CCS软件
3. TMS320C64xx实验箱一台

(三) 实验原理与方法

1. 标准C 语言程序

CCS 支持使用标准C 语言开发DSP 应用程序。C 源程序文件名的后缀应为.c。

CCS 在编译标准 C 语言程序时，首先将其编译成相应汇编语言程序，再进一步编译成目标 DSP 的可执行代码。最后生成的是 COFF 格式的可下载到 DSP 中运行的文件，其文件名后缀为.out。

由于使用 C 语言编制程序，其中调用的标准 C 的库函数由专门的库提供，在编译链接时编译系统还负责构建C 运行环境。所以用户工程中需要注明使用C 的支持库。

2. 命令文件的作用

命令文件（文件名后缀为 cmd）为链接程序提供程序和数据在具体DSP 硬件中的位置分配信息。通过编制命令文件，我们可以将某些特定的数据或程序按照我们的意图放置在 DSP 所管理的内存中。命令文件也为链接程序提供了 DSP 外扩存储器的描述。在程序中使用 CMD 文件描述硬件存储区，可以只说明使用部分，但只要是说明的，必须和硬件匹配，也就是只要说明的存储区必须是存在的和可用的。

3. 内存映射（map）文件的作用

为了更精确地使用 ROM 空间，我们就需要知道程序的大小和位置，通过建立目标程序的 map 文件可以了解 DSP 代码的确切信息。当需要更改程序和数据的大小和位置时，就要适当修改cmd 文件和源程序，再重新生成 map 文件来观察结果。另外，通过观察 map 文件，可以掌握DSP 存储器的使用和利用情况，以便进行存储器方面的优化工作。

(四) 实验步骤与内容

1. 准备

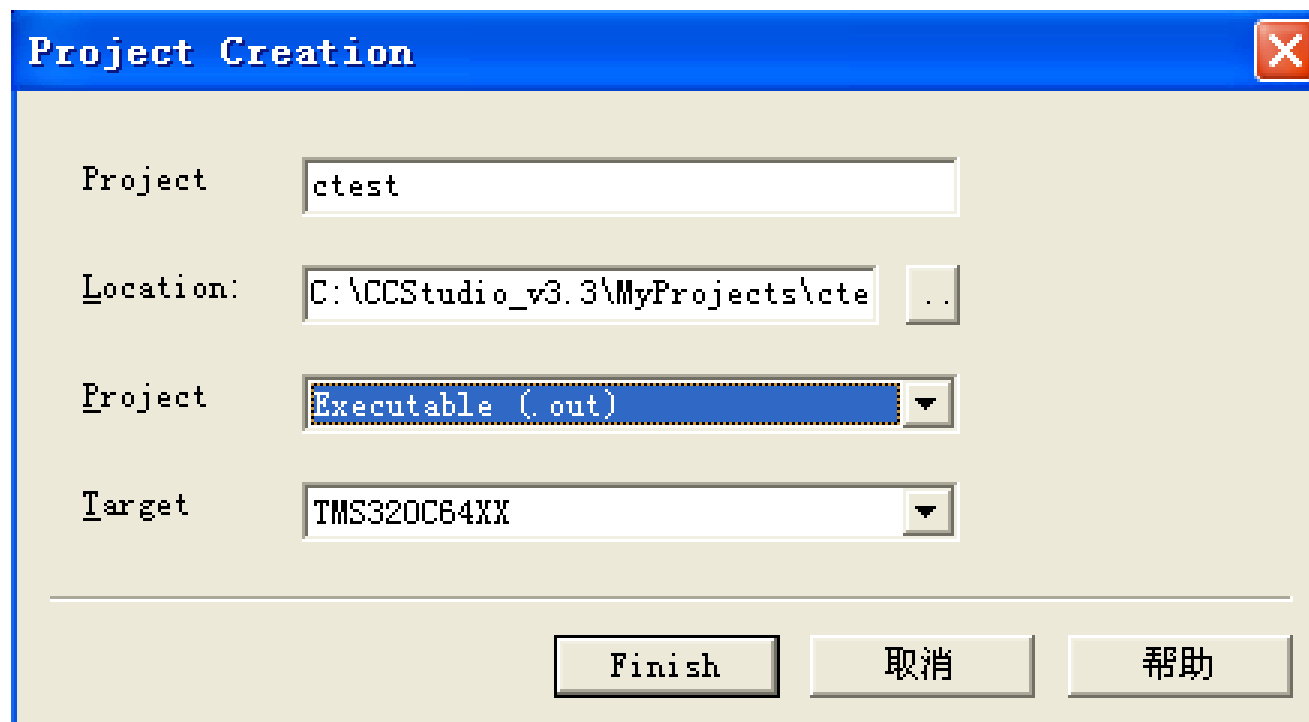
将DSP实验设备与硬件仿真器相连接,连接电源线,打开电源开关。

2. 设置CCS 在硬件仿真（Emulator）方式下运行，选择实际的仿真器。也可设置在模拟仿真（Simulator）方式下工作。

3. 启动CCS 环境

4. 创建工程。

选择菜单Project->New, 创建 ctest.pjt 工程

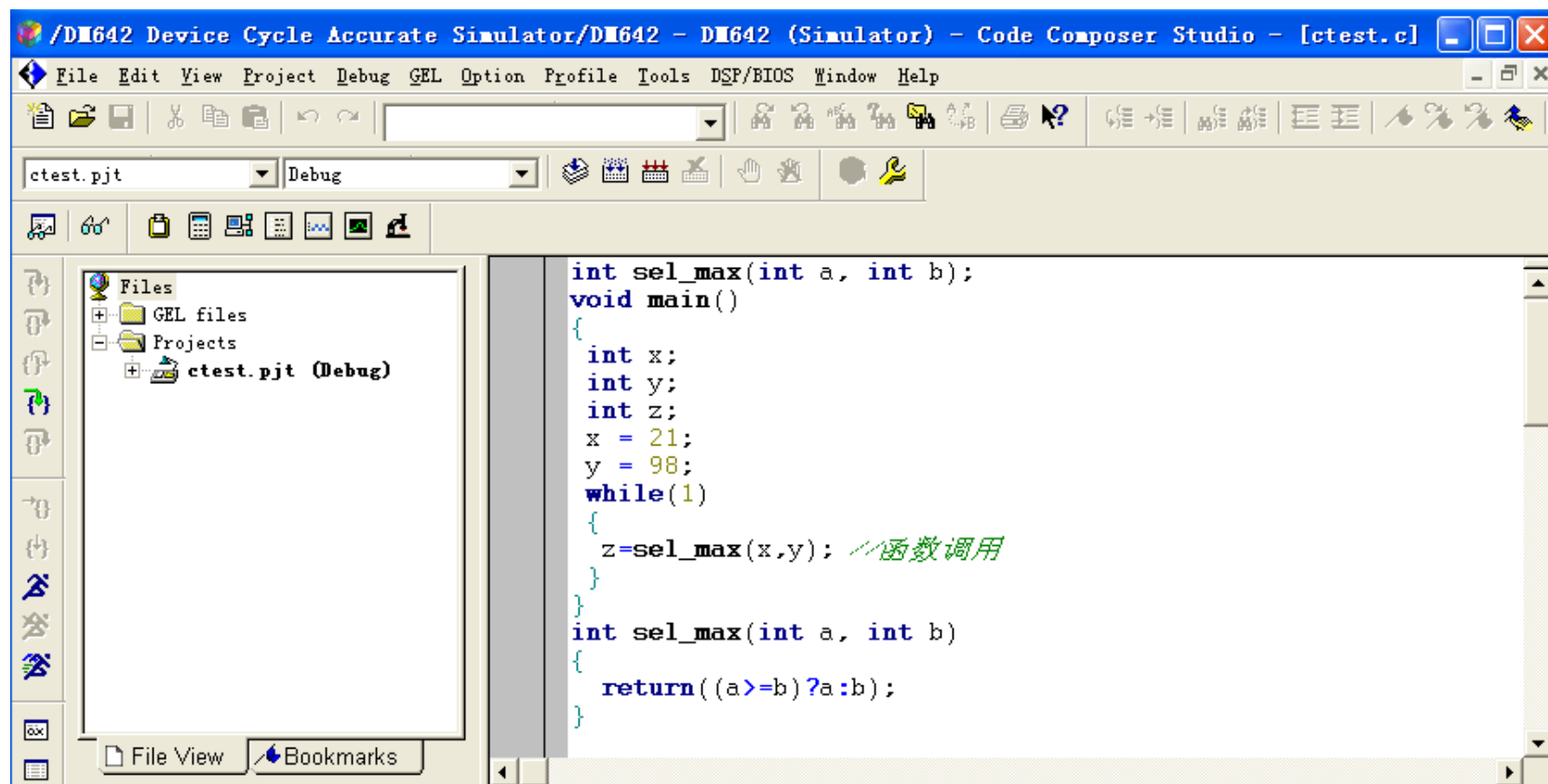


5. 建立源文件

1) 建立C 语言源程序文件。

选择 FILE->New->Source, 输入下列 C 代码, 保存到文件ctest.c。

```
intsel_max(inta,intb);  
void main()  
{  
    int x;  
    int y;  
    intz;  
    x=21;  
    y=98;  
    while(1)  
    {  
        z=sel_max(x,y); //函数调用  
    }  
}  
int sel_max(int a, intb)  
{  
    return((a>=b)?a:b);  
}
```



2) 建立链接命令文件

点击“File/New/Source File”或使用工具条的快捷按钮，打开编辑窗，建立ctest.cmd文件，内容如下：

```
stack 400
heap 400
MEMORY
{
    ISRAM      : origin = 0x0,          len= 0x1000000
}

SECTIONS
{
    .vectors > ISRAM
    .text    > ISRAM

    .bss     > ISRAM
    .cinit   > ISRAM
    .const   > ISRAM
    .far     > ISRAM
    .stack   > ISRAM
    .cio     > ISRAM
    .system  > ISRAM
}
```

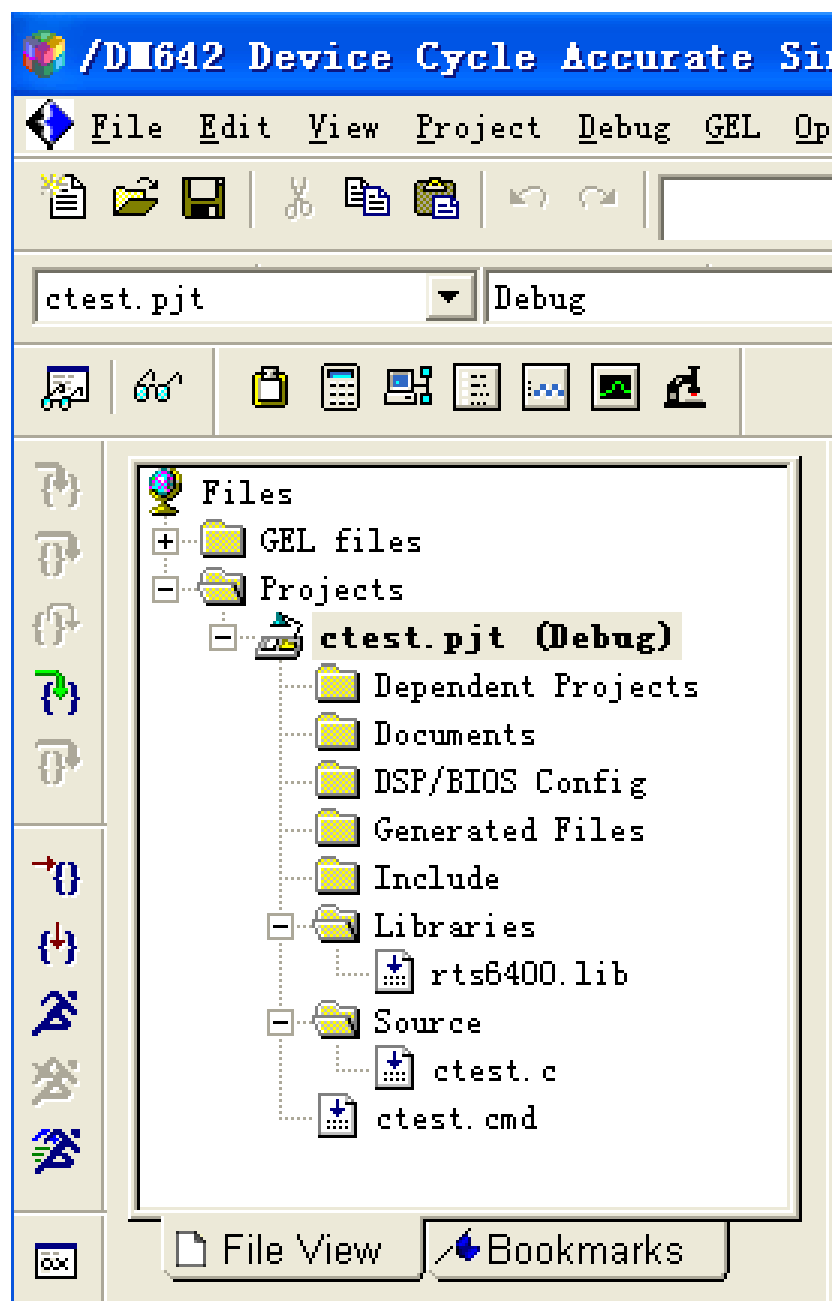
```
-stack 0x400
-heap 0x400

MEMORY
{
    ISRAM      : origin = 0x0,          len = 0x1000
}

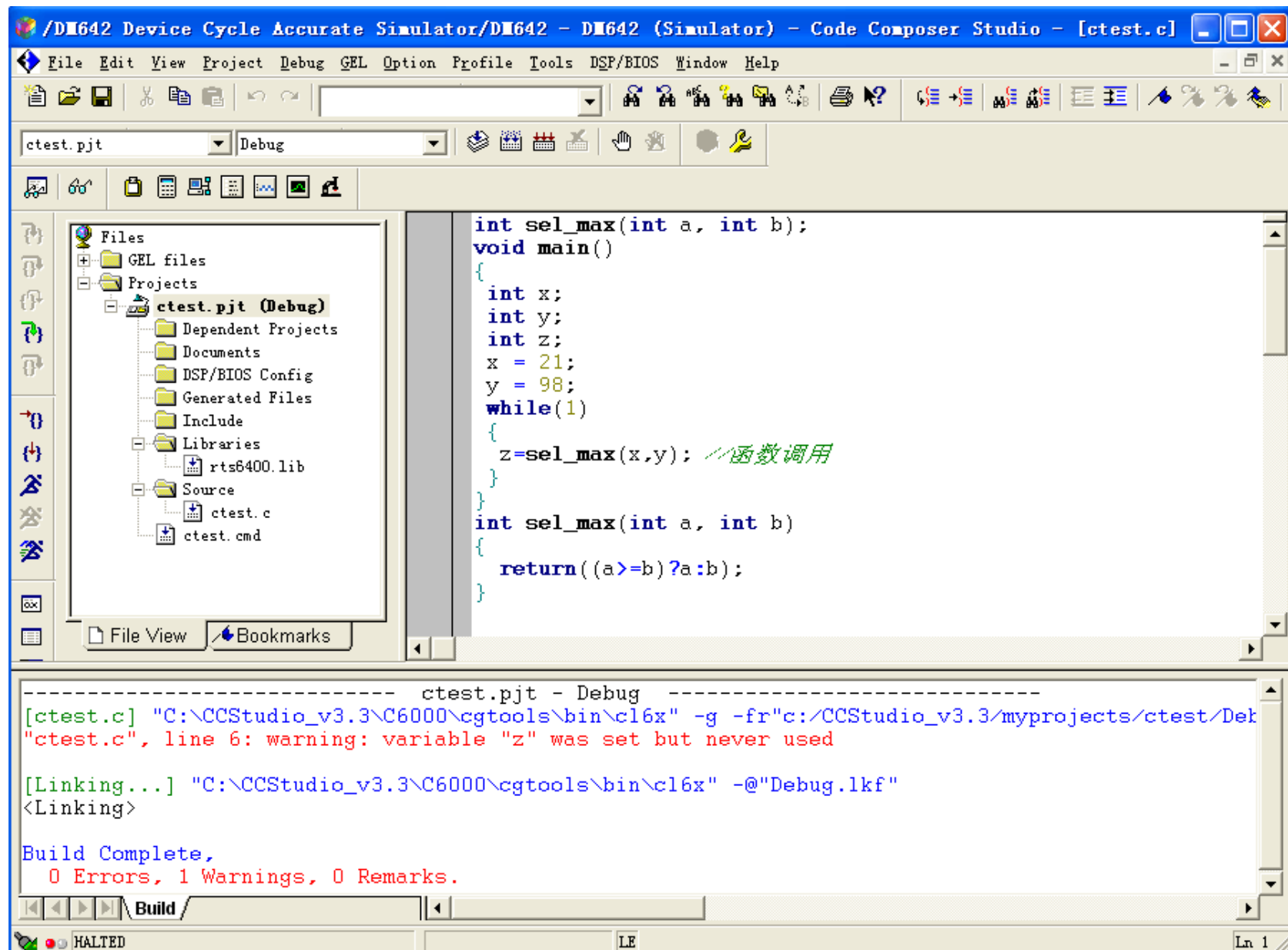
SECTIONS
{
    .vectors > ISRAM
    .text    > ISRAM
    .data    > ISRAM
    .bss     > ISRAM
    .cinit   > ISRAM
    .const   > ISRAM
    .far     > ISRAM
    .stack   > ISRAM
    .cio     > ISRAM
    .system  > ISRAM
}
```

6. 工程添加文件

点击菜单Project->AddFilestoproject,分别加入ctest.c,ctest.cmd文件,再向工程添加C:\CCStudio_v3.3\C6000\cgtools\lib下的文件rts6400.lib(这点与汇编程序设计不同)(注意:若不添加,则链接时会出现警告:warning:entrypointsymbol_c_int00undefined)



7. 编译、建立目标文件 产生ctest.out



8. 调用编程可执行程序

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：
<https://d.book118.com/825332333243011234>