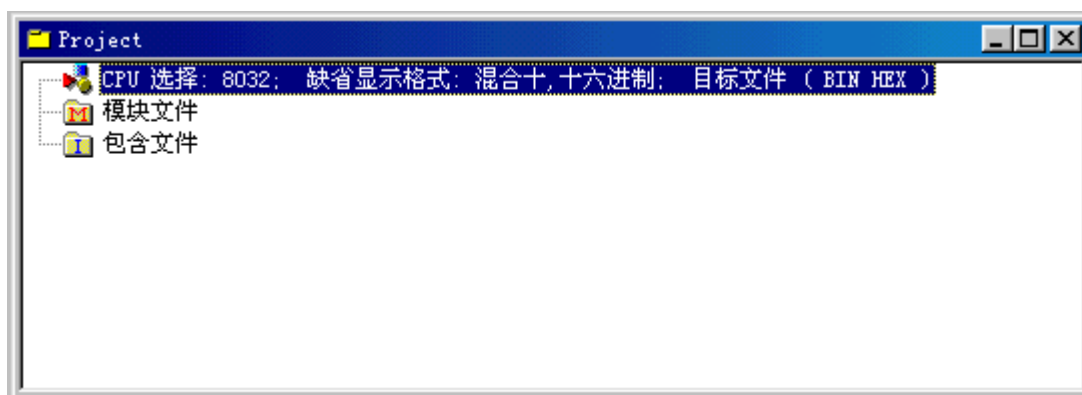
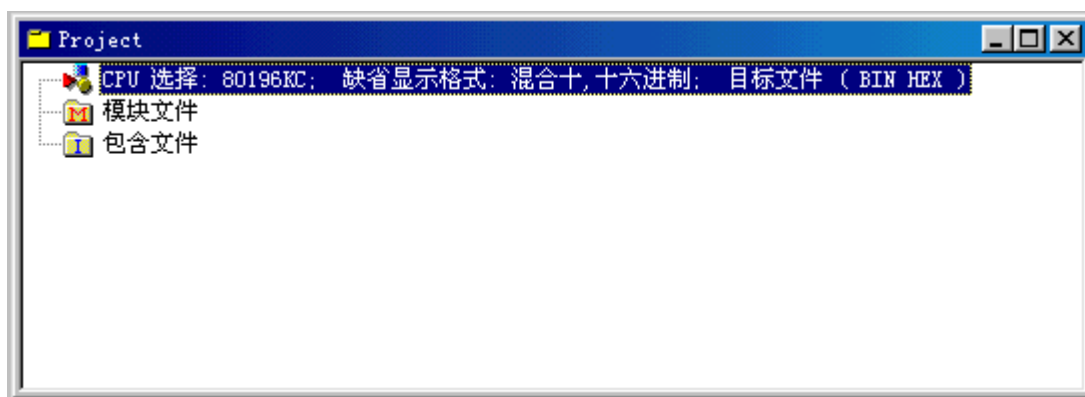


伟福环境下反汇编范例

1、关闭当前项目或者新建一个空项目，设置项目的 CPU 类型（如下图所示）：

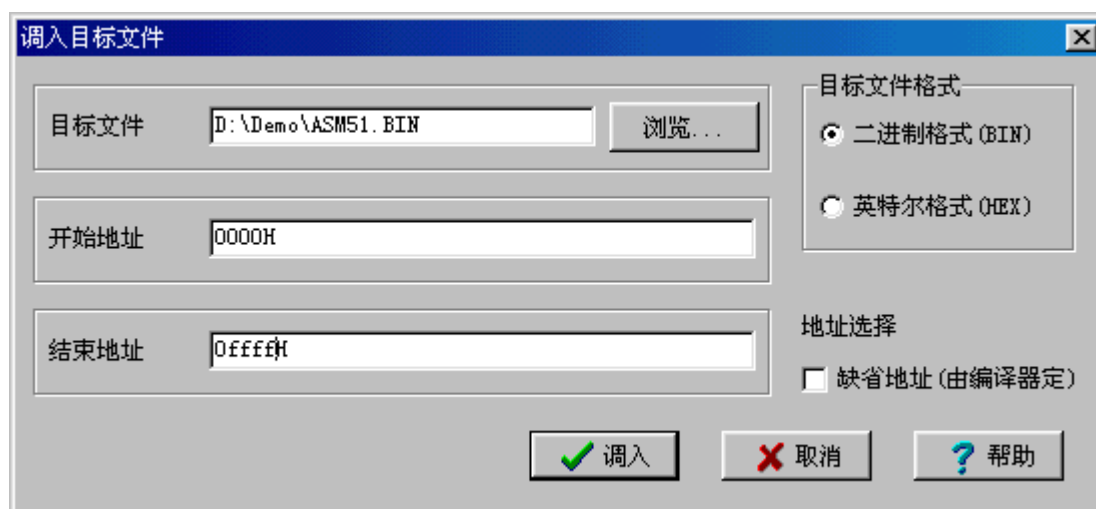


进行 MCS-51 反汇编时的项目设置



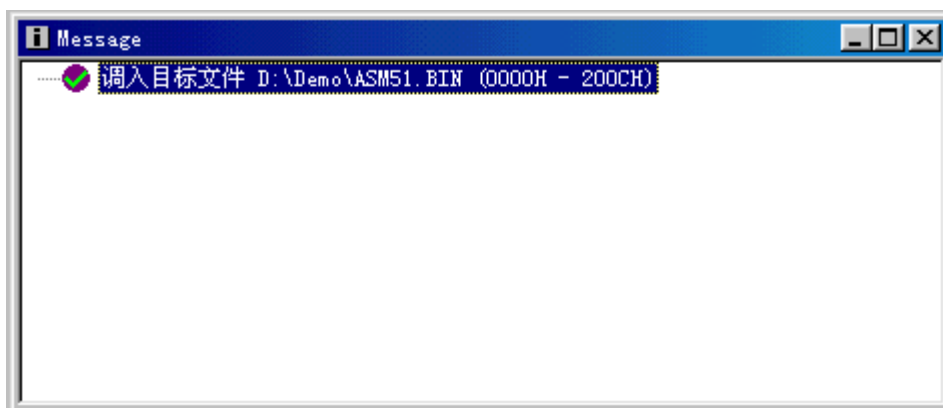
进行 MCS-96 反汇编时的项目设置

2、选择“文件\调入目标文件”，出现如下图示的对话框：

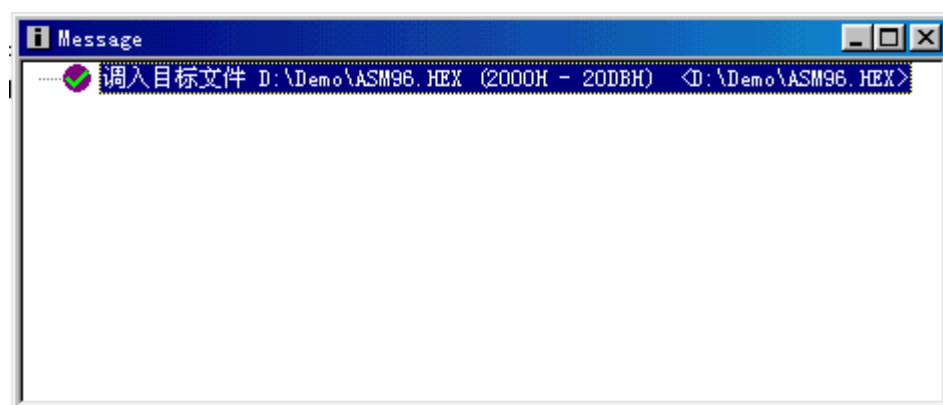


调入目标文件对话框

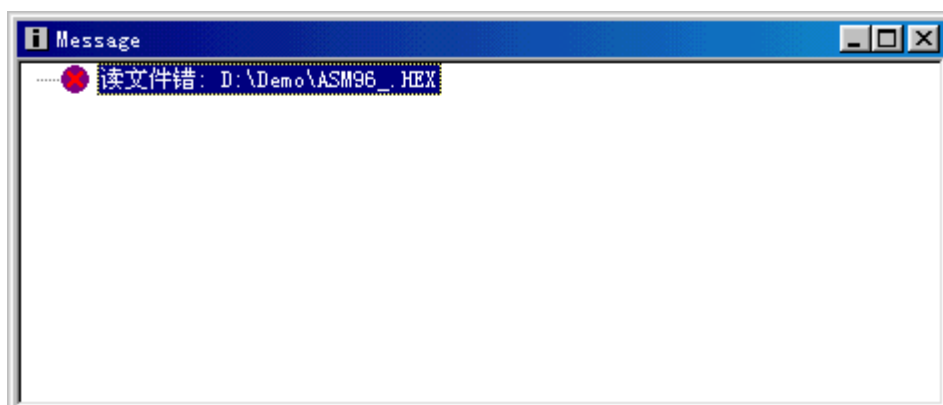
在上述的对话框中，选择好目标文件和目标文件的格式（BIN 或 HEX），地址选择一般使用缺省地址，即由编译器确定（如果完全确定目标文件的地址或者只需要调入某一段目标码，可以取消缺省地址前面的复选框选择，在“开始地址”和“结束地址”栏填入地址），点击调入按钮。调入成功后，会在信息窗口中显示“调入目标文件 目标文件路径和文件名（xxxxH - xxxxH）”（如下图所示）。如果目标文件有错，会出现相应的错误信息（如下图所示）。注意：目标文件格式选择不对，可以被调入，但是目标码会不对；



读入 BIN 文件

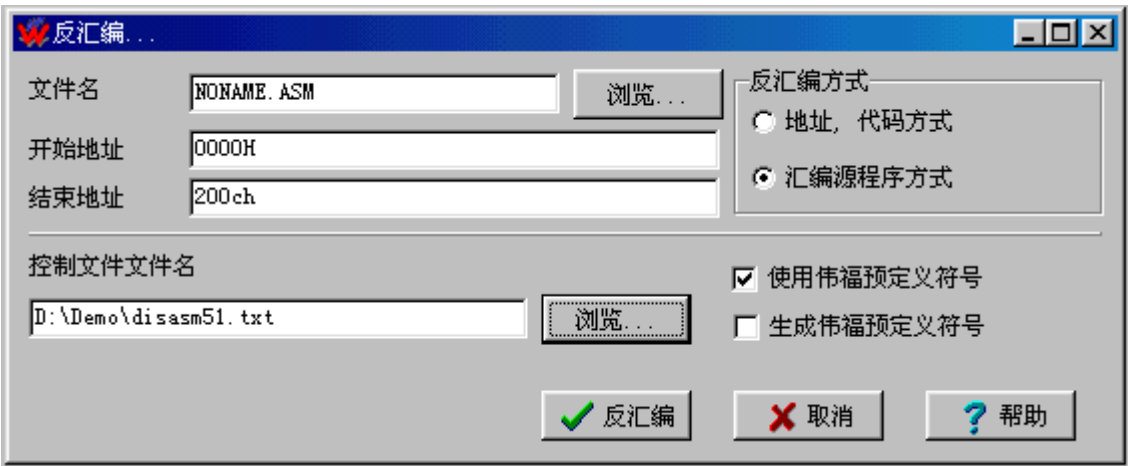


读入 HEX 文件



读入目标文件错误

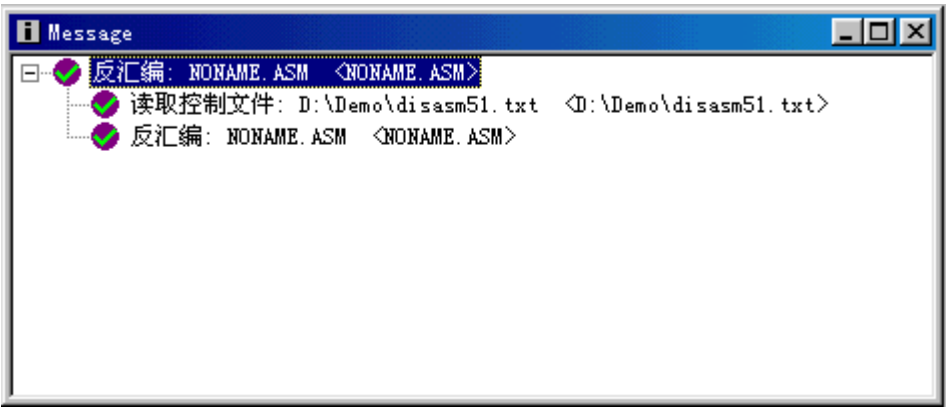
3、目标文件调入成功后，选择“文件\反汇编”，弹出如图对话框：



反汇编对话框

图中文件名以及所在路由用户自己设置，开始地址和结束地址在打开反汇编的时候有软件自动调入，如果用户只需反汇编目标码中的某一段代码，可以手工修改起始地址或者结束地址。控制文件用于优化反汇编后源程序可读性而增加的用户控制文件，需要用户自定义，但是必须符合本文件末的附录的伟福反汇编控制文件语法规定。反汇编方式选为汇编源程序方式。为了增加程序的通用性，两个复选框可以不选。

4、完成了上述的设置，可以点击“反汇编”按钮，即可完成反汇编工作，成功后信息窗口如下图所示，双击相应的文件可以分别打开控制文件和反汇编文件以便修改。



反汇编后的信息窗口

5、用户可以通过分析上一次反汇编后的源程序，建立、修改反汇编控制文件，使目标码在用户控制文件的控制下进行智能的反汇编，是反汇编文件达到用户满意的可读性。

附录——伟福反汇编控制文件语法规则

伟福反汇编控制文件只能是仅使用伟福反汇编命令字的文本文件，反汇编过程对控制文件的后缀名不敏感，如不符合语法规则，将无法进行反汇编。

伟福反汇编控制文件命令字：

1. REM 注释

格式: REM 注释语句

2. CODE 标号定义

格式: CODE ('标号名称', 地址, '说明语句')

3. DATA 数据区标号定义

格式: DATA ('数据标号名称', 地址, '说明语句')

4. XDATA 外部数据标号定义(仅适用于 51 系列)

格式: XDATA ('外部数据标号名称', 地址, '说明语句')

5. BIT 位标号定义(仅适用于 51 系列)

格式: BIT ('位标号名称', 地址, '说明语句')

6. DB 程序区数据字节定义,将数据定义成字节形式

格式: DB ('数据标号', 开始地址 - 结束地址, '说明语句')

7. DW 程序区数据双字节定义, 将数据定义成双字节形式

格式: DW ('数据标号', 开始地址 - 结束地址, '说明语句')

8. DS 程序区数据字符串定义, 将数据定义成字符串形式

格式: DS ('数据标号', 开始地址 - 结束地址, '说明语句')

9. NULL 无效数据区, 反汇编时跳过这段数据

格式: NULL (开始地址 - 结束地址)

以下是一个反汇编控制字的例子:

文件名: DISASM51.TXT

```
code  ('START', 1ADh, '==== This is MAIN')
code  ('Read_RX', 30h, '==== test if sio in')
code  ('main_loop', 1eah, '==== here waiting command')
code  ('Wait_RX', 4eh, ")
code  ('Process_11H', 1fbh, '====')
code  ('Process_12H', 228h, '====')
code  ('Main_Back', 365h, ")
code  ('Dly35ms', 111H, '==== delay 35 ms')
code  ('Dly140ms', 11aH, '==== delay 140 ms')
rem code  ('addr132e', 132eh, ")
rem code  ('JUMPTBL0', 1462h, '==== function jump')
bit   ('Has_RX', 73h, ")
db    ('NoUse', 3h-0fh)
db    ('xx1Table', 0b8h-0c7h)
rem db    ('xx3Table', 09c1fh-9c3ch)
rem db    ('xx4Table', 09c3dh-9c5ah)
null  (136ch-1fffh)
```