

上海沪生电子产品使用说明书

<http://www.husoon.com>

电话：021-61021969

网站：www.husoon.com

仿真机使用说明

学习单片机离不开实验，以往单片机的实验往往依赖于仿真机和单片机学习系统，价格昂贵，至使学习单片机的门槛很高，很多人不能跨入单片机学习的大门。美国 SST 公司推出的 SST 系列单片机更是集成了仿真功能，配合 Keil 软件，可以做成一个功能强大，使用简单的仿真机。

1. 简介

本仿真机的 CPU 采用 SST 公司的 P89E554RC 芯片制作，该芯片具有 IAP 功能，可在应用中编程。图 1 是仿真机的外形图。



图 1 Min51 迷你型 51 仿真机外形图

特点

- 直接支持 KEIL C51 的 IDE 开发仿真环境；
- 可单步、断点、全速、可参考变量、RAM 变量；
- 支持汇编、C 语言，混合调试；
- 监控程序占用用户的资源少,全速运行不占用资源；
- 完全保留单片机特性，避免仿真正常而实际烧录芯片却不正常的问题；
- 可以仿真 89C51、89C52、89S51、89S52、89C58 等 51 内核的单片机；
- 有配套的单片机实验板可廉价提供，构成单片机学习、开发的完整方案。

2. 硬件结构

本机的硬件结构非常简单，整个仿真机就是一个芯片大小，内置了由 MAX202 等构成的一个串行口电路，一个复位电路。

2. 1 复位电路设计

之所以在这么一个小小的仿真机上采用复位电路，是因为考虑到很多用户板采用简单的阻容复位，正常工作时，不需要反复断电，并且通常会在一次断电后稍候 2~3 秒再通电（很多仪器要求断电后至少 30 秒再通电），而在使用仿真机调程序时，很少有人有耐心等上 2~3 秒钟，因此，上电有时并不能保证可靠复位。此外在仿真机加上复位按钮，省却了装载程序时反复断电的麻烦，设计虽小却很贴心。R2 使用了 100Ω 的电阻，通常足以保证正常复位。

如果用户板使用了专用复位芯片，则可以保证上电时可靠复位，不需要使用复位按钮。如果不小心误按下该按钮，100Ω 的上拉电阻也不会使复位芯片损坏，我们做过测试，在常用的几种复位芯片的复位端长时间接远小于 100Ω 的上拉电阻，没有造成芯片的任何问题。当然，如果系统中使用了复位芯片，按下按钮，不足以产生使 CPU 复位的高电平，这个复位按钮就不再起作用了，如需复位，只能给系统断电后重新上电。

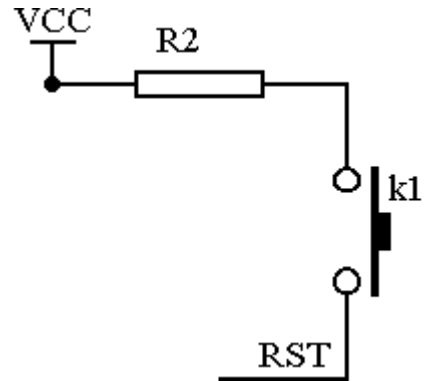


图 2 复位电路

2. 2 仿真头设计

仿真插头采用了金针，这种针接触良好，针体纤细，不会损坏用户板上的集成电路插座，比之常见的扁针要好。但这种针的坏处是易折断，一旦折断很难修复，很多同类仿真机要么使用了扁针，要么回避这个问题。实际工作中我们自己曾多次出现断针的问题，深为此苦恼，因此，在该仿真机设计时，特采用了针、机分离的方式，如图 3 所示。在仿真机的背面，加

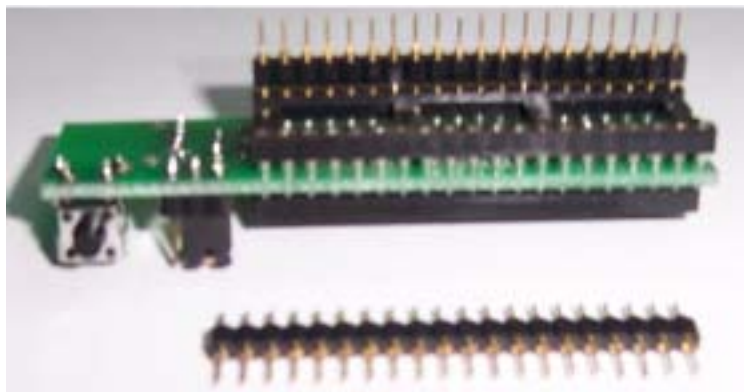


图 3 仿真机背面

装有一个优质圆孔插座，然后再插入金针，这样即做使不小心折断了针，只要自行购买一条金针，拨去折断的针，插上新的针即可修复，且成本不高。

2. 3 接线端子处理

本机由一条串口线将本机与电脑相连，串口线通过接线端子与仿真机相连，这个接线端子很难取，则开始我们选择了 USB 接线端子，但使用效果不佳，现改用常见的三端子插针、座。使用这种接线端子的缺点是线易折断，如不加处理，时间稍长，断线几乎是不可避免的。为此在制作中采取了特别的措施，大大减少了线折断的机会。图 4 所示是该仿真机的正面清晰图。



图 4 仿真机正面清晰图

2. 4 可切换晶振设计

本机自带了一个晶振，通常是 24M 的，也可以通过板上的插针切换为用户板上的晶振。由于本仿真机与用户板紧密集成，因此，用户板上晶振的使用几乎不受限制，之所以加上这样的设计，主要是因为我们在长期教学工作中发现有很多初学者难以判断自己设计的电路板晶振工作时是否正常，因此，通常情况下，可以选择用户板晶振，如果对用户板晶振工作时是否正常有怀疑时，可选用本机自带晶振进行测试。

3. 可仿真 CPU 清单

本仿真机采用的是 SST 公司的 SST89E554RC，该芯片具有片内 32K+8K Flash ROM，1K 字节 RAM，4 级 8 个中断源，双 DPTR，内置 PCA 功能，内置 WDT 等功能，原则上，只要被仿真芯片内部资源少于该 CPU 所具有的资源，就可以完全仿真，如果被仿真 CPU 有部分不同于该 CPU 的特殊功能，那么该仿真机不能仿真该特殊功能，但其他通用功能依然可以仿真。

以下列出部分可以仿真的 CPU 清单：

Atmel

AT89C51 AT89C52 AT89S51 AT89S52 AT89LV52 AT89S53 AT89LS53 AT89C55
AT89LV55

Philips

P80C54 P80C58 P87C54 P87C58 P87C524 P87C528

Winbond

W78C54 W78C58 W78E54 W78E54

SST

SST89C54 SST89C58 SST89E554RC SST89V554RC

以上所列型号是根据相关资料整理而成，并未全部经过测试。

4. 使用

4. 1 硬件连接

将本机插入到用户板上，注意方面不要插错。然后将通讯电缆一端插在仿真机上，另一端插于电脑的任一个串行口上（下面以插入 COM1 口为例）。

由于本机不带有电源供电端，必须依靠用户板供电，因此，要给用户板接上正确的电源，要求电源电压为 5V，不能超过±%5，即不低于 4.5V 不高于 5.5V。

检查晶振插针，通常应处于用户板位置，如确有必要，可以切换到本机晶振位置，当仿真机如图 4 放置时，短路子插于左侧为选择用户板晶振，插于右侧为选择本机晶振。

4. 2 软件端的使用

以下以本站提供的 DSB-1A 型单片机实验板为例，介绍如何使用仿真机。

DSB-1A 型实验板的 P1 口接有 8 个 LED，这 8 个 LED 的阴极分别连到 P1.0~P1.7，阳极则通过限流电阻接正电源，因此，只要 P1 某引脚为低电平，该引脚上接的 LED 便点亮。

打开任意字处理软件，输入以下源程序。

```
*****
;
;   平凡单片机工作室
;   http://www.mcustudio.com
;   Copyright 2003 pingfan's McuStudio
;   All rights Reserved
;作者：周坚
;lsd.asm
;流水灯程序
*****
      ORG    0000H
      LJMP  START
      ORG    30H
START: MOV    A,#0FEH
LOOP:  MOV    P1,A
      RL  A
      LCALL DELAY
      LJMP  LOOP
;以下子程序
DELAY: MOV    R7,#250
```

```

D1:    MOV    R6,#250
D2:    DJNZ   R6,D2
      DJNZ   R7,D1
      RET
      END

```

以 lsd.asm 为文件名存盘，注意，必须加上扩展名。

点击“Project->New Project...”菜单，出现一个对话框，要求给将要建立的工程起一个名字，你可以在编辑框中输入一个名字（设为 lsd），不需要扩展名。点击“保存”按钮，出现第二个对话框，如图 5 所示，这个对话框要求选择目标 CPU（即你所用芯片的型号），Keil

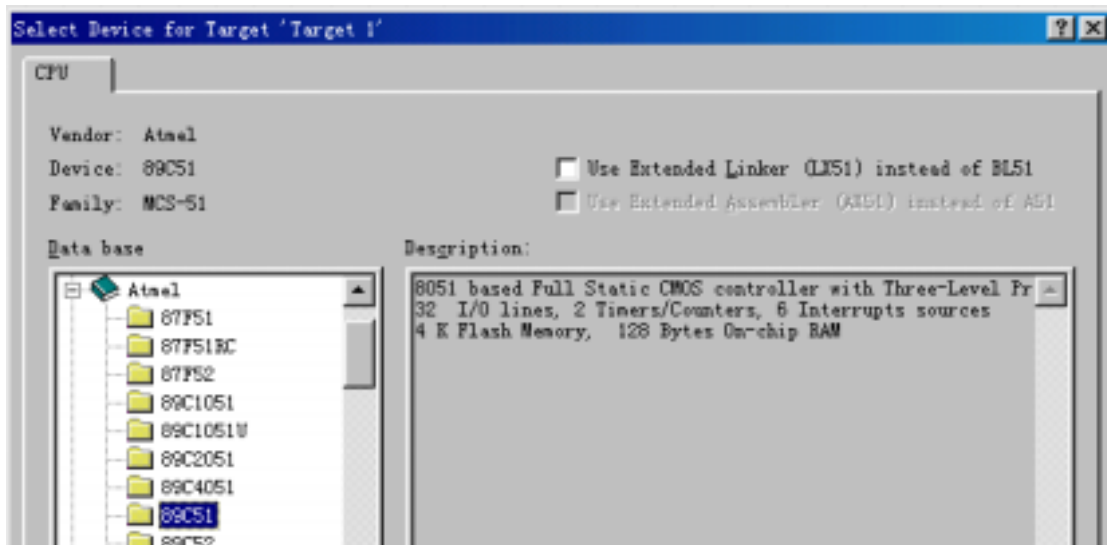


图 5 选择目标 CPU

支持的 CPU 很多，我们选择 Atmel 公司的 89C51 芯片。点击 ATMEL 前面的“+”号，展开该层，点击其中的 89C51，然后再点击“确定”按钮，回到主界面，此时，在工程窗口的文件页中，出现了“Target 1”，前面有“+”号，点击“+”号展开，可以看到下一层的“Source Group1”，这时的工程还是一个空的工程，里面什么文件也没有，需要手动把刚才编写好的源程序加入，点击“Source Group1”使其反白显示，然后，点击鼠标右键，出现一个下拉

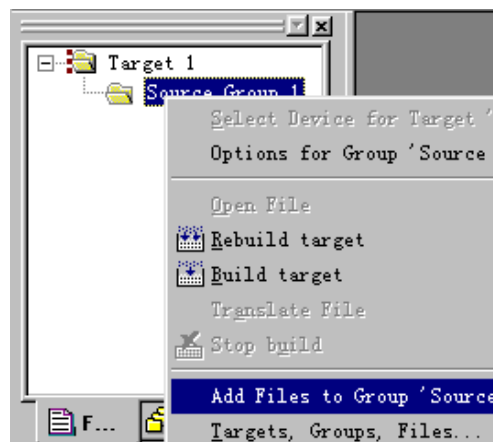


图 6 加入文件

菜单，如图 6 所示。选中其中的“Add file to Group”Source Group1”，出现一个对话框，要求寻找源文件，注意，该对话框下面的“文件类型”默认为 C source file(*.c)，也就是以 C

为扩展名的文件，而该文件是以 asm 为扩展名，所以在列表框中找不到 lsd.asm，要将文件类型改掉，点击对话框中“文件类型”后的下拉列表，找到并选中“Asm Source File(*.a51,*.asm)”，这样，在列表框中就可以找到 lsd.asm 文件了。

双击 lsd.asm 文件，将文件加入项目，**注意：**在文件加入项目后，该对话框并不消失，等待继续加入其它文件，但初学时常会误认为操作没有成功而再次双击同一文件，这时会出现如图 7 所示的对话框，提示你所选文件已在列表中，此时应点击“确定”，返回前一对话框，然后点击“Close”即可返回主界面，返回后，点击“Source Group 1”前的加号，会发现 exam1.asm 文件已在其中。双击文件名，即打开该源程序。



图 7 重复加入源文得到的提示

工程建立好以后，还要对工程进行进一步的设置，以满足要求。

首先点击左边 Project 窗口的 Target 1，然后使用菜单“Project->Option for target ‘target1’”即出现对工程设置的对话框，这个对话框可谓非常复杂，共有 8 个页面，要全部搞清可不容易，好在绝大部份设置项取默认值就行了。

设置对话框中的 Target 页面，如图 8 所示，Xtal 后面的数值是晶振频率值，默认值是所选择目标 CPU 的最高可用频率值，对于我们所选的 AT89C51 而言是 24M，该数值与最终产生的目标代码无关，仅用于软件模拟调试时显示程序执行时间。正确设置该数值可使显示时间与实际所用时间一致，一般将其设置成与你的硬件所用晶振频率相同，如果没必要了解程序执行的时间，也可以不设，这里设置为 12。

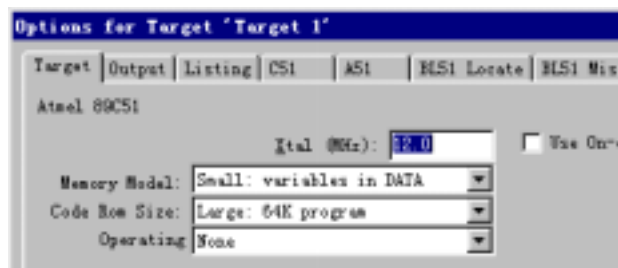


图 8 对工程进行设置

Memory Model 用于设置 RAM 使用情况，有三个选择项，Small 是所有变量都在单片机的内部 RAM 中；Compact 是可以使用一页外部扩展 RAM，而 Large 则是可以使用全部外部的扩展 RAM。Code Model 用于设置 ROM 空间的使用，同样也有三个选择项，即 Small 模式，只用低于 2K 的程序空间；Compact 模式，单个函数的代码量不能超过 2K，整个程序可以使用 64K 程序空间；Large 模式，可用全部 64K 空间。Use on-chip ROM 选择项，确认是否仅使用片内 ROM（注意：选中该项并不会影响最终生成的目标代码量）；Operating 项是操作系统选择，Keil 提供了两种操作系统：Rtx tiny 和 Rtx full，关于操作系统是另外一个很大的话题了，通常我们不使用任何操作系统，即使用该项的默认值：None（不使用任何操作系统）；Off Chip Code memory 用以确定系统扩展 ROM 的地址范围，Off Chip xData memory 组用于确定系统扩展 RAM 的地址范围，这些选择项必须根据所用硬件来决定，由

于该例是单片应用，未进行任何扩展，所以均不重新选择，按默认值设置。

设置对话框中的 OutPut 页面，如图 9 所示，这里面也有多个选择项，其中 Creat Hex file 用于生成可执行代码文件（可以用编程器写入单片机芯片的 HEX 格式文件，文件的扩展名为.HEX），默认情况下该项未被选中，如果要写片做硬件实验，就必须选中该项，这一点是初学者易疏忽的，在此特别提醒注意。选中 Debug information 将会产生调试信息，这些信息用于调试，如果需要对程序进行调试，应当选中该项。Browse information 是产生浏览信息，该信息可以用菜单 view→Browse 来查看，这里取默认值。按钮“Select Folder for objects”是用来选择最终的目标文件所在的文件夹，默认是与工程文件在同一个文件夹中。Name of Executable 用于指定最终生成的目标文件的名字，默认与工程的名字相同，这两项一般不需要更改。

工程设置对话框中的其它各页面与 C51 编译选项、A51 的汇编选项、BL51 连接器的连接选项等用法有关，这里均取默认值，不作任何修改。以下仅对一些有关页面中常用的选项作一个简单介绍。

Listing 标签页用于调整生成的列表文件选项。在汇编或编译完成后将产生 (*.lst) 的列

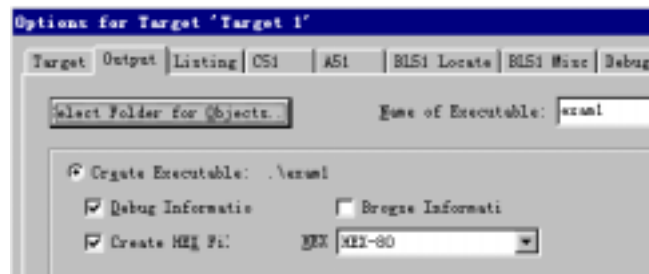


图 9 对输出进行控制

表文件，在连接完成后也将产生 (*.m51) 的列表文件，该页用于对列表文件的内容和形式进行细致的调节，其中比较常用的选项是“C Compile Listing”下的“Assamble Code”项，选中该项可以在列表文件中生成 C 语言源程序所对应的汇编代码。

C51 标签页用于对 Keil 的 C51 编译器的编译过程进行控制，其中比较常用的是“Code Optimization”组，如图 10 所示，该组中 Level 是优化等级，C51 在对源程序进行编译时，可以对代码多至 9 级优化，默认使用第 8 级，一般不必修改，如果在编译中出现一些问题，可以降低优化级别试一试。Emphasis 是选择编译优先方式，第一项是代码量优化（最终生成的代码量小）；第二项是速度优先（最终生成的代码速度快）；第三项是缺省。默认的是速度优先，可根据需要更改。

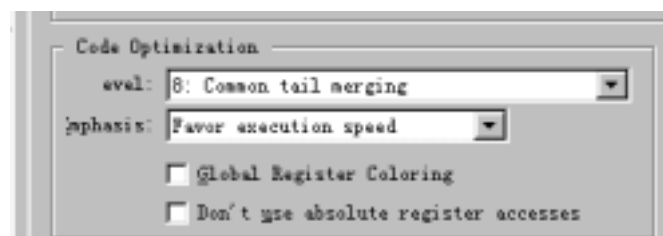


图 10 代码生成控制

Debug 页用来设置调试器。左侧的“Use Simulator”是选择 Keil 内置的模拟调试，可以使用 Keil 的模拟调试器进行软件调试工，这里对此不作详细说明。当需要使用本机的仿真功能时，需要点击右侧的“Use Keil Monitor-51 Drive”，如图 11 所示。通常正常安装完成后，Use 后的下拉列表就是显示“Keil Monitor-51 Drive”，如果你发现是其他参数，可以点击下拉列表，选择“Keil Monitor-51 Drive”，如图 12 所示。选择完成后，点击“Setting”按

钮，选择你所用的 PC 上的串口，波特率（通常可以使用 38400），其他设置一般不需要更改，如图 13 所示。点击“OK”回到 Debug 页面后，选中“Load Application at Start”和“Go Till main”，使其如图 11 所示。

设置好工程后，即可进行编译、连接。选择菜单 Project→Build target，对当前工程进行连接，如果当前文件已修改，软件会先对该文件进行编译，然后再连接以产生目标代码；如果选择 Rebuild All target files 将会对当前工程中的所有文件重新进行编译然后再连接，确保最终生产的目标代码是最新的，而 Translate ...项则仅对该文件进行编译，不进行连接。

以上操作也可以通过工具栏按钮直接进行。图 14 是有关编译、设置的工具栏按钮，从左到右分别是：编译、编译连接、全部重建、停止编译和对工程进行设置。

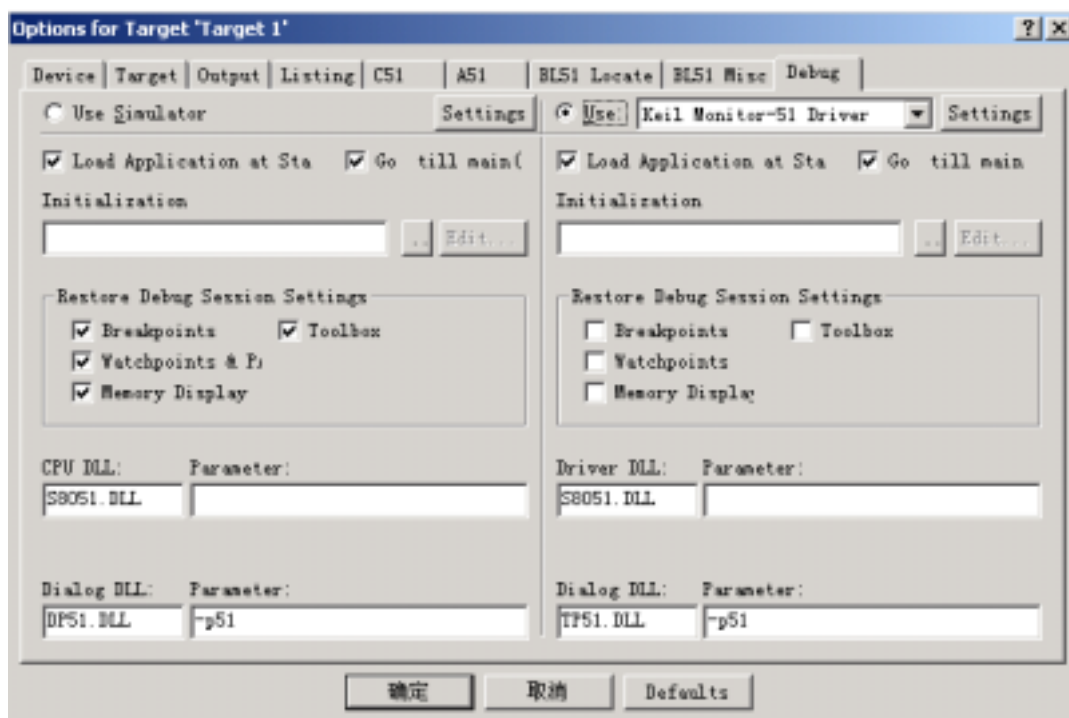


图 11 设置 Debug 页

编译过程中的信息将出现在输出窗口中的 Build 页中，如果源程序中有语法错误，会有错误报告出现，双击该行，可以定位到出错的位置，对源程序反复修改之后，最终会得到如图 15 所示的结果，提示获得了名为 exam1.hex 的文件，该文件即可被编程器读入并写到芯片中，同时还产生了一些其它相关的文件，可被用于 Keil 的仿真与调试，这时可以进入下一步调试的工作。点击 Debug→Start/Stop Debug Session 即可进入调试界面，如图 16 所示。

有关 Keil 使用的更多知识，请参考《Keil 实例教程》。

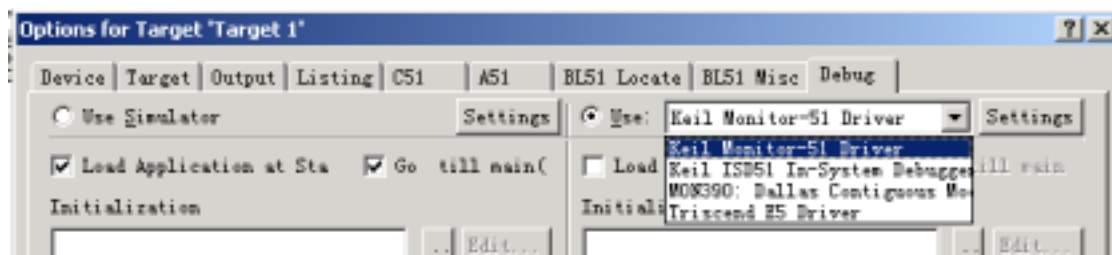


图 12 选中“Keil Moniotr-51 Driver”

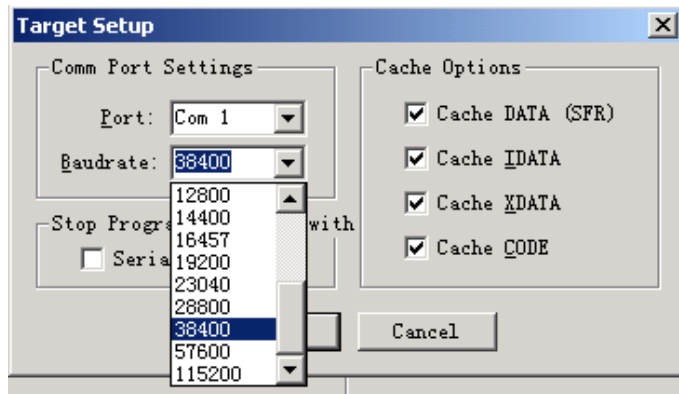


图 13 选择串口、波特率及其他选项



图 14 有关编译、连接、项目设置的工具条



图 15 成功地编译、连接后的信息

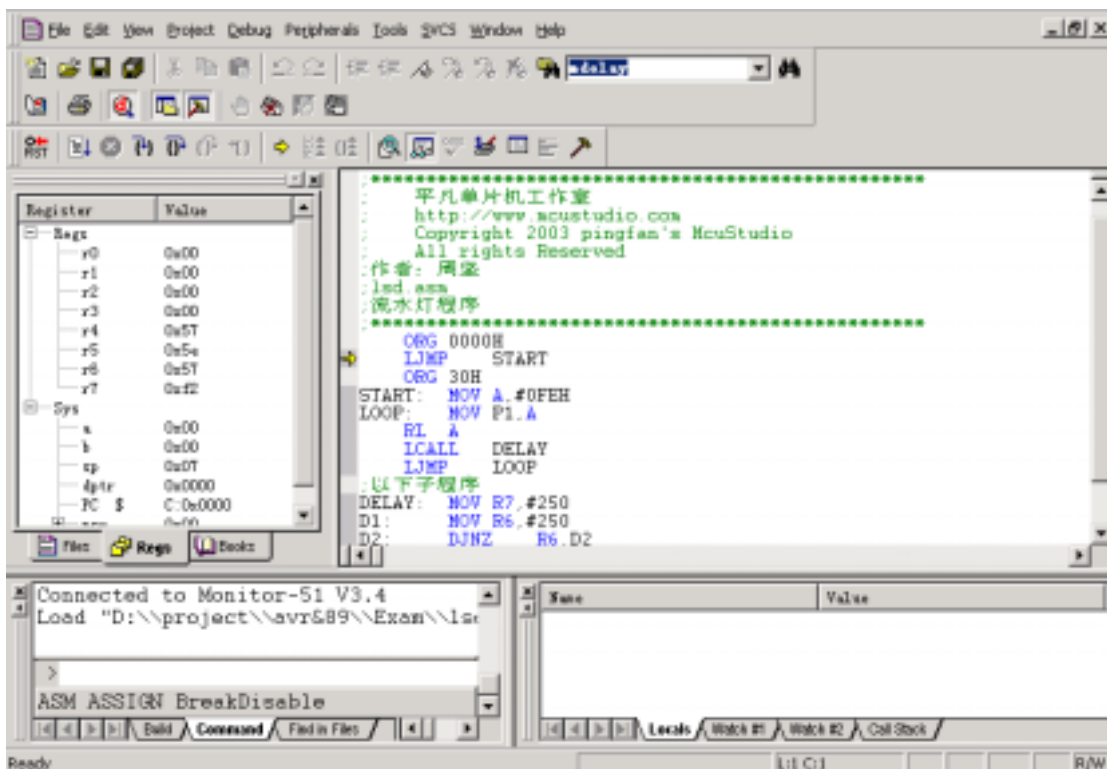


图 16 正确进入调试后的界面