

Keil 工程文件的建立、设置与目标文件的获得

单片机开发中除必要的硬件外，同样离不开软件，我们写的汇编语言源程序要变为 CPU 可以执行的机器码有两种方法，一种是手工汇编，另一种是机器汇编，目前已极少使用手工汇编的方法了。机器汇编是通过汇编软件将源程序变为机器码，用于 MCS-51 单片机的汇编软件有早期的 A51，随着单片机开发技术的不断发展，从普遍使用汇编语言到逐渐使用高级语言开发，单片机的开发软件也在不断发展，Keil 软件是目前最流行开发 MCS-51 系列单片机的软件，这从近年来各仿真机厂商纷纷宣布全面支持 Keil 即可看出。Keil 提供了包括 C 编译器、宏汇编、连接器、库管理和一个功能强大的仿真调试器等在内的完整开发方案，通过一个集成开发环境（uVision）将这些部份组合在一起。运行 Keil 软件需要 Pentium 或以上的 CPU，16MB 或更多 RAM、20M 以上空闲的硬盘空间、WIN98、NT、WIN2000、WINXP 等操作系统。掌握这一软件的使用对于使用 51 系列单片机的爱好者来说是十分必要的，如果你使用 C 语言编程，那么 Keil 几乎就是你的不二之选（目前在国内你只能买到该软件、而你买的仿真机也很可能只支持该软件），即使不使用 C 语言而仅用汇编语言编程，其方便易用的集成环境、强大的软件仿真调试工具也会令你事半功倍。

我们将通过一些实例来学习 Keil 软件的使用，在这一部份我们将学习如何输入源程序，建立工程、对工程进行详细的设置，以及如何将源程序变为目标代码。图 1 所示电路图使用 89C51 单片机作为主芯片，这种单片机性属于 MCS-51 系列，其内部有 4K 的 FLASH ROM，可以反复擦写，非常适于做实验。89C51 的 P1 引脚上接 8 个发光二极管，P3.2~P3.4 引脚上接 4 个按钮开关，我们的第一个任务是让接在 P1 引脚上的发光二极管依次循环点亮。

一、Keil 工程的建立

首先启动 Keil 软件的集成开发环境，这里假设读者已正确安装了该软件，可以从桌面上直接双击 uVision 的图标以启动该软件。

UVision 启动后，程序窗口的左边有一个工程管理窗口，该窗口有 3 个标签，分别是 Files、Regs、和 Books，这三个标签页分别显示当前项目的文件结构、CPU 的寄存器及部份特殊功能寄存器的值（调试时才出现）和所选 CPU 的附加说明文件，如果是第一次启动 Keil，那么这三个标签页全是空的。

1、源文件的建立

使用菜单“File->New”或者点击工具栏的新建文件按钮，即可在项目窗口的右侧打开一个新的文本编辑窗口，在该窗口中输入以下汇编语言源程序，例 1：

```

MOV    A, #0FEH
MAIN:  MOV    P1, A
        RL     A
        LCALL  DELAY
        AJMP   MAIN
DELAY:  MOV    R7, #255
D1:     MOV    R6, #255
        DJNZ   R6, $

```

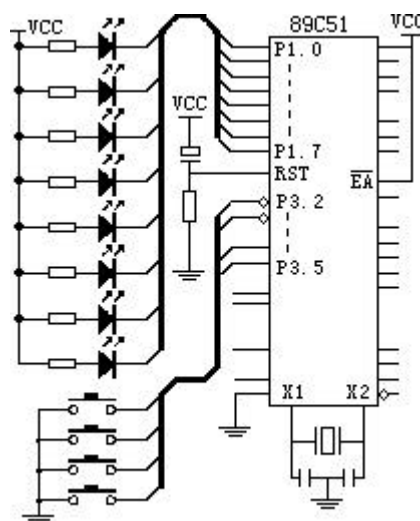


图 1 简单的键盘、显示板

```
DJNZ    R7,D1
RET
END
```

END

保存该文件，注意必须加上扩展名（汇编语言源程序一般用 asm 或 a51 为扩展名），这里假定将文件保存为 exam1.asm。

需要说明的是，源文件就是一般的文本文件，不一定使用 Keil 软件编写，可以使用任意文本编辑器编写，而且，Keil 的编辑器对汉字的支持不好，建议使用 UltraEdit 之类的编辑软件进行源程序的输入。

2、建立工程文件

在项目开发中，并不是仅有一个源程序就行了，还要为这个项目选择 CPU（Keil 支持数百种 CPU，而这些 CPU 的特性并不完全相同），确定编译、汇编、连接的参数，指定调试的方式，有一些项目还会有多个文件组成等，为管理和使用方便，Keil 使用工程（Project）这一概念，将这些参数设置和所需的所有文件都加在一个工程中，只能对工程而不能对单一的源程序进行编译（汇编）和连接等操作，下面我们就一步一步地来建立工程。

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 exam1），不需要扩展名。点击“保存”按钮，出现第二个对话框，如图 2 所示，这个对话框要求选择目标 CPU（即你所用芯片的型号），Keil 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，出现一个下拉菜单，如图 3 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，而我们的文件是以 asm 为扩展名的，所以在列表框中找不到 exam1.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 exam1.asm 文件了。

双击 exam1.asm 文件，将文件加入项目，注

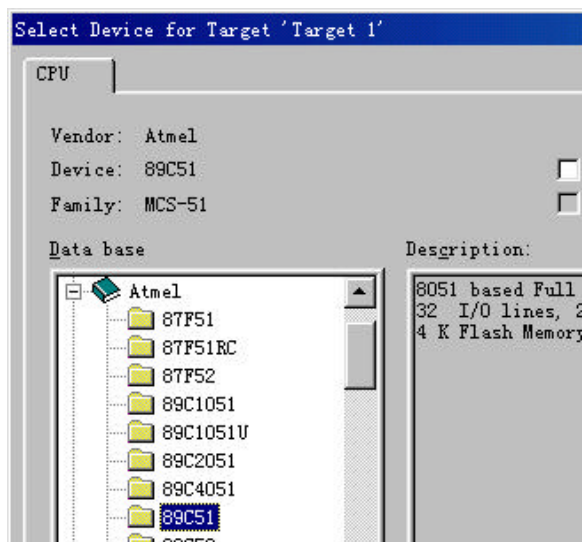


图 2 选择目标 CPU

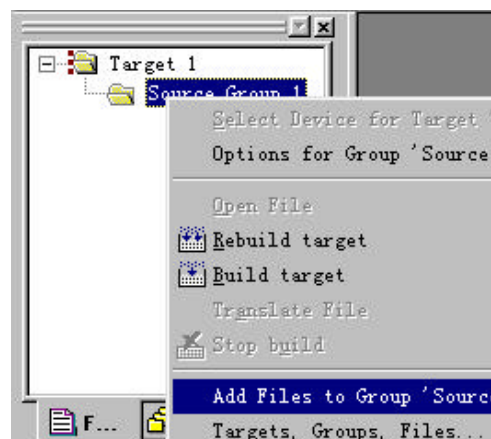


图 3 加入文件

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 4 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。

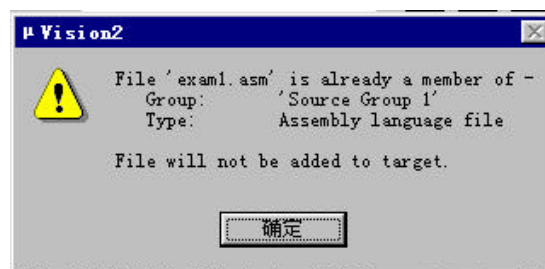


图 4 重复加入文件的错误

二、工程的详细设置

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 5 所示，Xtal 后面的数值是晶振频率值，默认值是所选目标 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

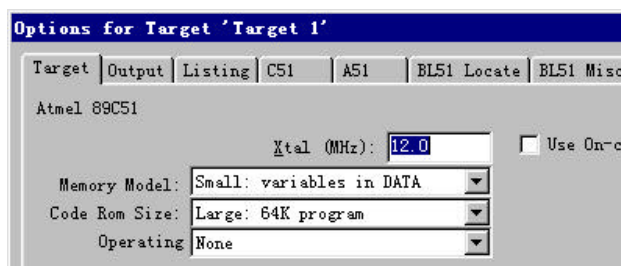


图 5 对目标进行设置

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。

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 Target 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；Target 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 6 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 7 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。

设置完成后按确认返回主界面，工程文件建立、设置完毕。

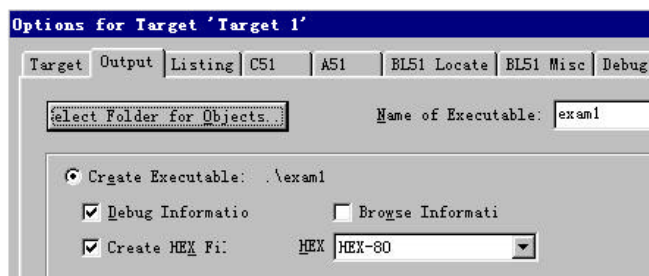


图 6 对输出进行控制

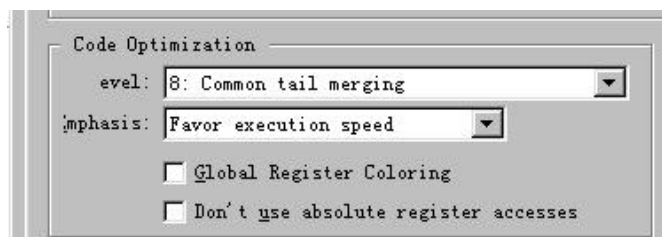


图 7 代码生成控制

三、编译、连接

在设置好工程后，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 8 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。

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

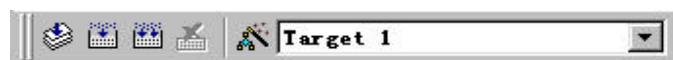


图 8 有关编译、连接、项目设置的工具条

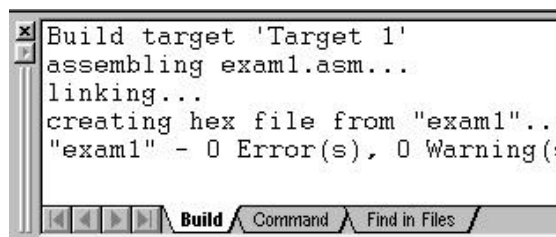


图 9 正确编译、连接之后的结果

Keil 实例教程（一）

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 9 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。