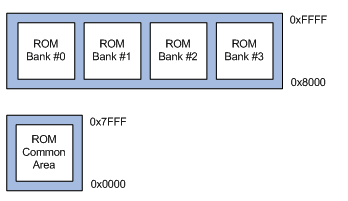


1、A standard 8051 device has an address range of 64 KBytes for code space. To expand program code beyond this 64 KB limit, the Keil 8051 tools support code banking. **This technique lets you manage one common area and 32 banks of up to 64 KBytes each for a total of 2 MBytes of bank-switched memory.**

译：一个标准的8051设备有一个地址范围64 KBytes的代码空间。为了扩展程序代码超过64kb的限制，Keil 8051工具组支持代码库。**这种技术允许您管理一个公共区域和32个库，每个多达64 KB，总共2MBytes的可切换的存储库。**(注：code banking :我翻译成“代码库”)



2、For example, your hardware design may include one 32 K ROM mapped from address 0x0000 to 0x7FFF (known as the common area or common ROM) and four 32K ROMs, mapped from code address 0x8000 to 0xFFFF (known as the code bank ROMs). Typically, the code bank ROMs are selected via port bits. The figure above shows this memory structure.

2、例如，您的硬件设计可能包括一个地址从0x0000映射到0x7FFF(称为公共区域或公共ROM)的32 K ROM，和四个32K ROMs，他们的代码地址从0x8000映射到0xFFFF(被称为code bank ROMs)。通常，代码库rom是通过端口位选择的。上图显示了这个内存结构。

3、Code Banking is enabled and configured in the **Project — Options for Target — Target** dialog. Enter the number of code banks and the start and end address of the code banking area as shown below:

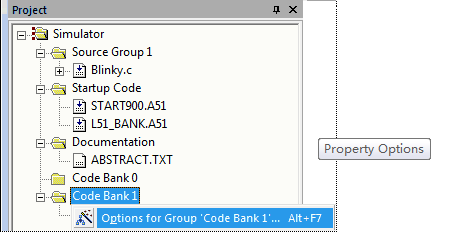
3、在**Project — Options for Target — Target对话框**中启用和配置代码库。输入代码库的数量和代码库区域的开始和结束地址如下:

4、To configure the banking hardware, add the file **C51\LIB\L51\_BANK.A51** to your project. Copy this file into the project source file folder and add the file to a project group. The **L51\_BANK.A51** file must be configured to match your target hardware.

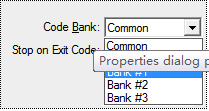
4、要配置库硬件，请添加文件**C51\LIB\L51\_BANK.A51**到项目中。将此文件复制到项目源文件文件夹中，并将文件添加到项目组中。**L51\_BANK.A51文件**必须被配置以匹配您的目标硬件。

5、For each source file or file group you must specify the code bank. Use the dialog **Options for — Properties** for this purpose. Right click with the mouse on the file or file group and select **Options.**

5、对于每个源文件或文件组，你必须指定代码库。为了这个目的，使用对话框**Options for — Properties**。右键单击文件或文件组中的鼠标并选择**Options**。



This opens the Properties dialog page where you specify the code bank for the source file



6、The common code area can be accessed in all code banks. Usually, the common area includes **routines** and constants that must be accessible at all times. Such examples are: interrupt routines, interrupt and reset vectors, string constants, and bank switch routines.

6、公共代码区域可以在所有代码库中被访问。通常，公共区域包括常规和常量，这些例程和常量必须在任何时候都可以访问。这样的例子有:中断例程、中断和重置向量、字符串常量和库开关例程。

7、The linker locates only the program segments of a module into the bank area. However, if you can ensure that your program does access information in constant segments within a specific code bank only, you can locate these segments into this bank using the **BANKx** linker directives in the dialog **Options for Target — BL51 Misc**.

7、链接器只将模块的程序段定位到库区域。然而，如果你能确保你的程序只在某个特定的代码库内的常量段中访问信息，你就可以在**Options for Target — BL51 Misc**的对话框中使用BANKx链接器指令将这些段定位到这家银库。



8、The above steps complete the configuration of a code banking application. The µVision debugger fully supports code banking and allows you to test your program.

8、上述步骤完成了代码库应用程序的配置。µVision调试器完全支持代码库且允许您测试您的程序。