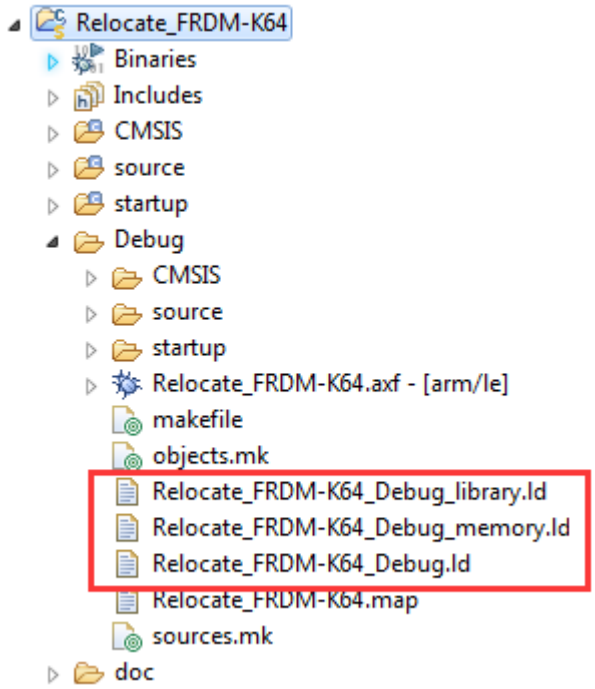
Relocating Code and Data Using the MCUXpresso IDE for Kinetis

1. Introduce the Liker File of MCUXpresso IDE

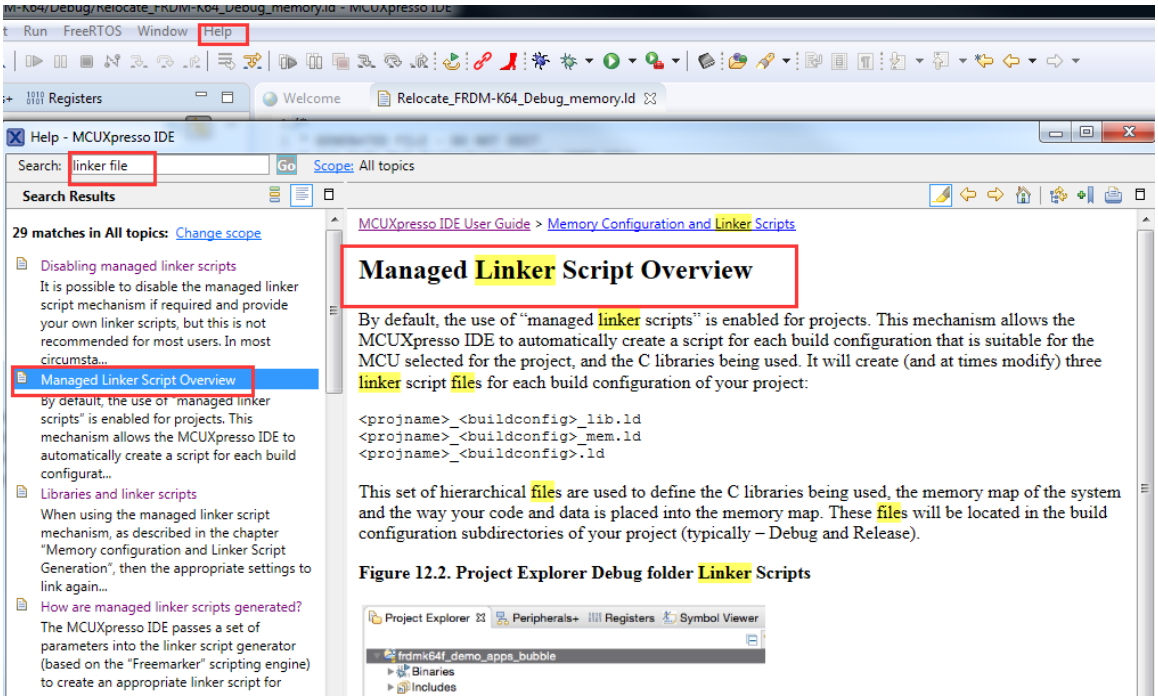
1.1 Different from KDS, the MCUXpresso IDE separates the linker file into 3 files:

<projname>_<buildconfig>_lib.ld
<projname>_<buildconfig>_mem.ld
< projname>_<buildconfig>.ld

For example in the project of “Relocate_FRDM-K64”, the 3 files are as below:

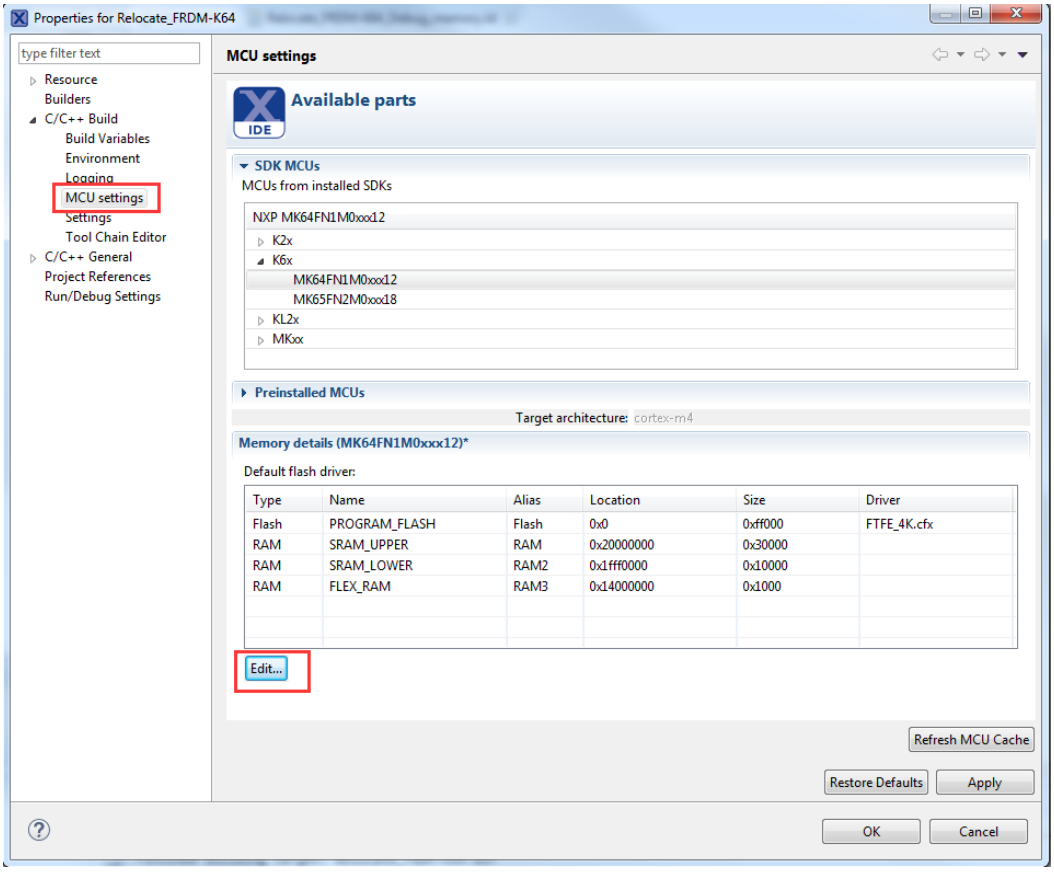


About the detail of "Linker file", please search “Linker file” in MCUxpresso IDE User Guide: click “Help”-> “MCUxpresso IDE User Guide”.



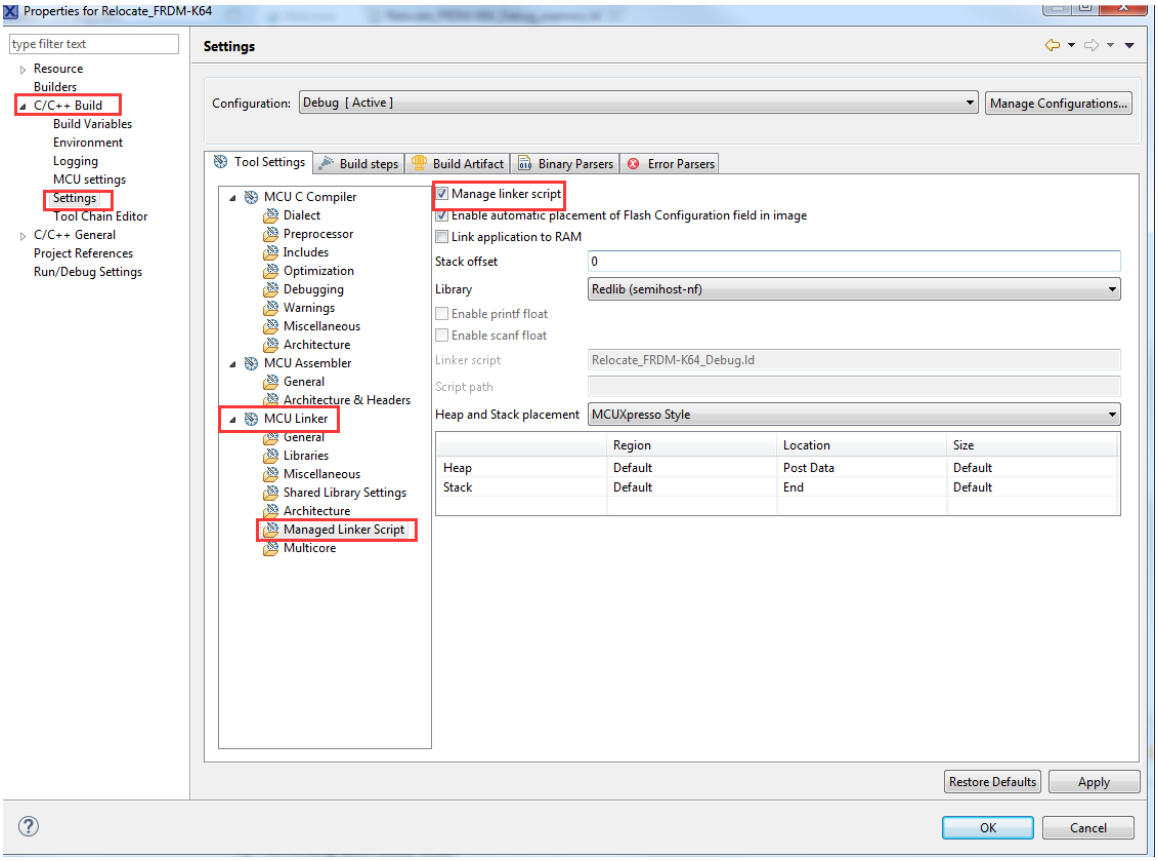
1.2 There are two methods to change the linker files.

- One method is editing the “Memory detail” in “MCU setting” as below:



Click the “Edit…” button to add delete or change memory size. By default, the use of “Manage linker script” is enabled for projects. This mechanism allows the MCUXpresso IDE to automatically create a script for each build configuration. So after finish configuring, build, it will create three linker script files for the memory details configuration .

The “Manage linker script” option is under Properties/C/C++ Build/Settings/MCU Linker/Managed Linker Script :



- Another method is disable “Manage linker script”, we direct configure the three linker script files by hand, the same with KDS.

2. Relocating code steps

The mainly steps and syntax are the same with KDS, please refer to the DOC “

Relocating Code and Data Using the KDS GCC Linker File for Kinetis"

https://community.nxp.com/docs/DOC-104433

The only difference is configure the linker files, it's better configure the memory segment in "Memory configuration editor" view(This is more simple than setting in linker file by hand.).

In order to know the steps clearly, I take the "**Relocating Code in ROM**" as example to show the detail steps.

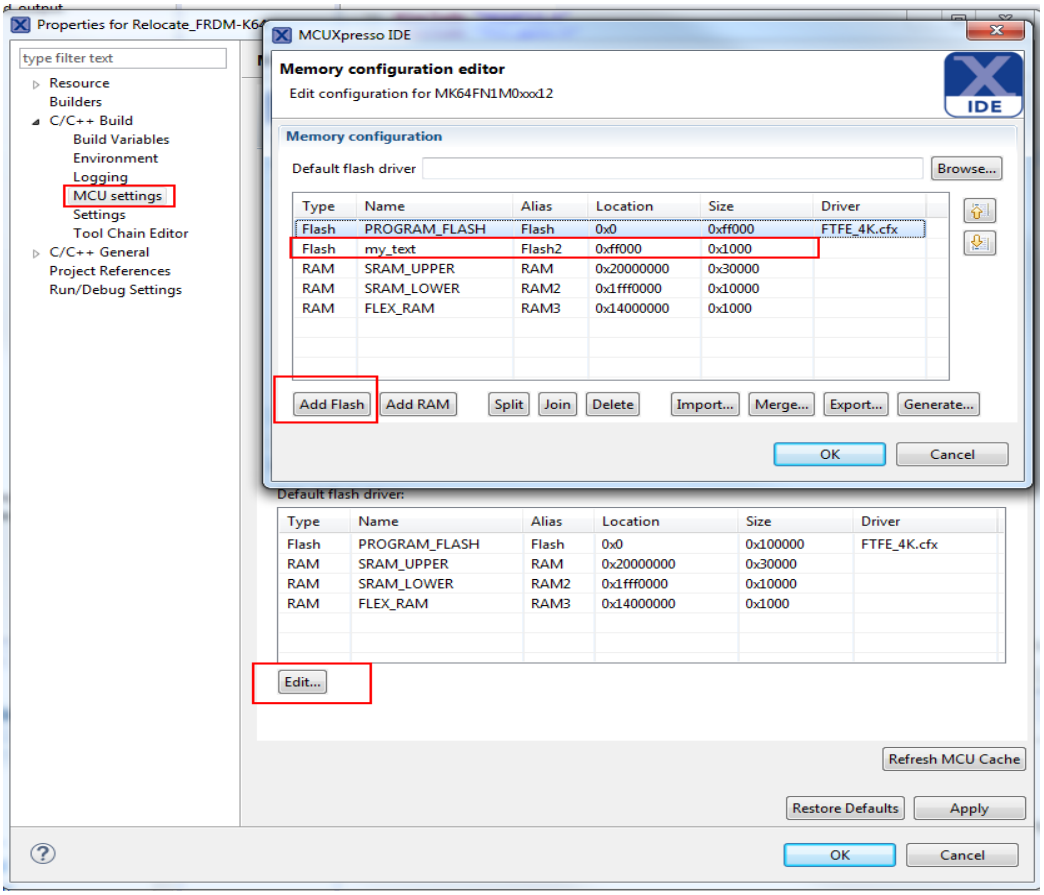
This example is relocating function "delay()" into 0xff000.

Steps 1: Create a project on MCUXpresso IDE, write your code, if you first use the MCUXpresso IDE, please refer to the user guide of MCUXpresso IDE.

Steps 2: Use the keyword "__attribute__" create a section named ".myROM", and use it to relocate the function of "delay()":

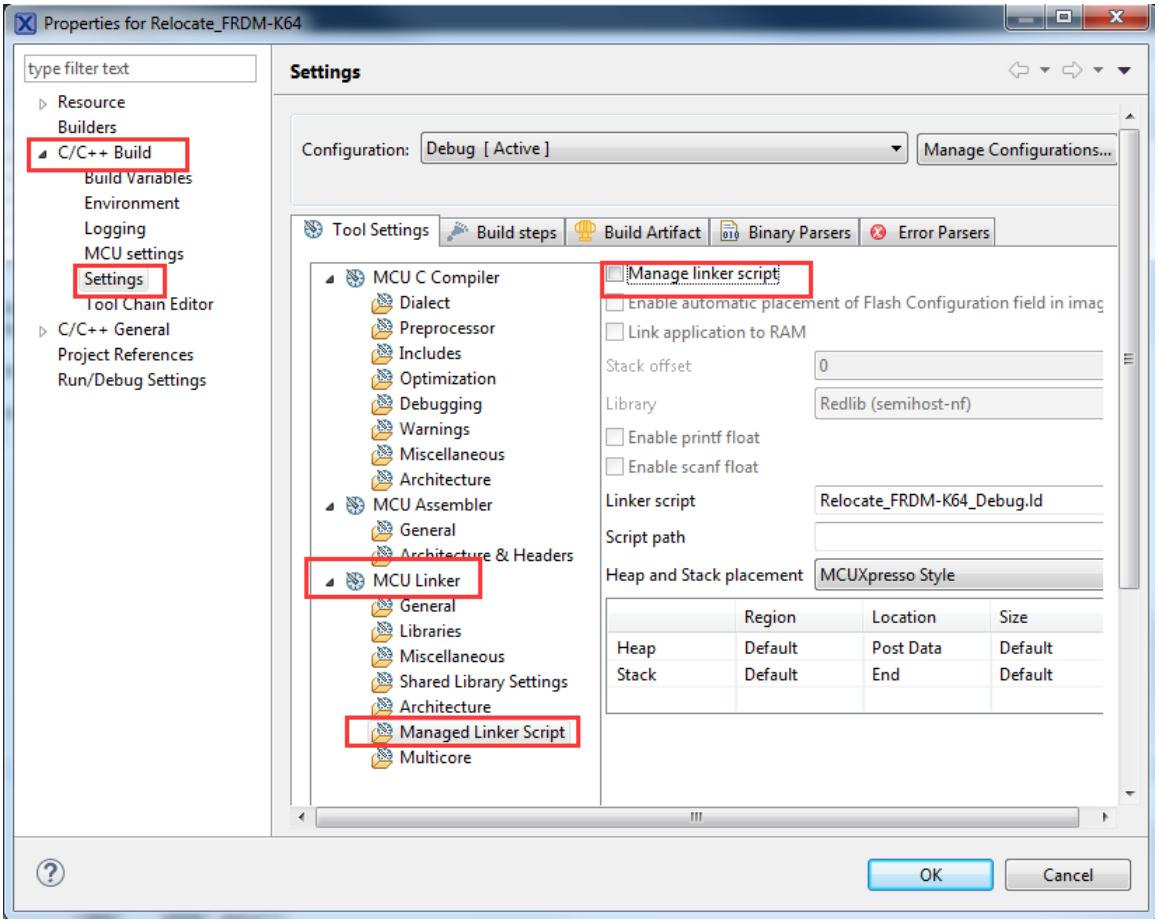
```
48  __attribute__ ((section(".myROM"))) void delay(void);
49
50  void delay(void)
51  {
52      volatile uint32_t i = 0;
53      for (i = 0; i < 800000; ++i)
54      {
55          __asm("NOP"); /* delay */
56      }
57  }
```

Steps 3: Create a new segment named "my_text", located at 0xff000, the size is 0x1000.



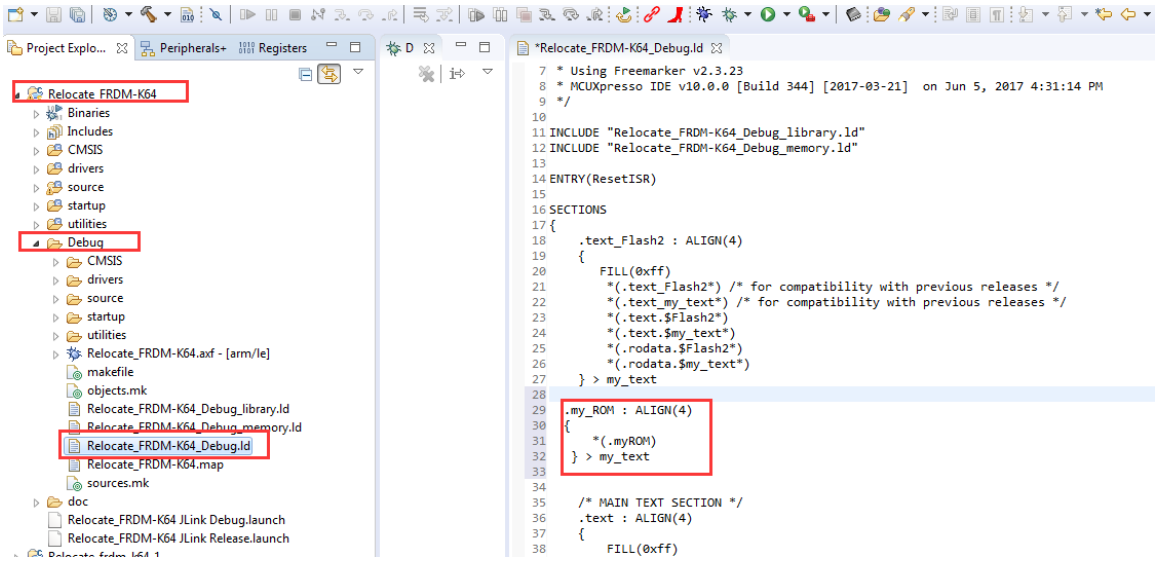
Click “OK” button, then build project.

Steps 3: Disable “Manage linker script” in MCU Linker Settings:

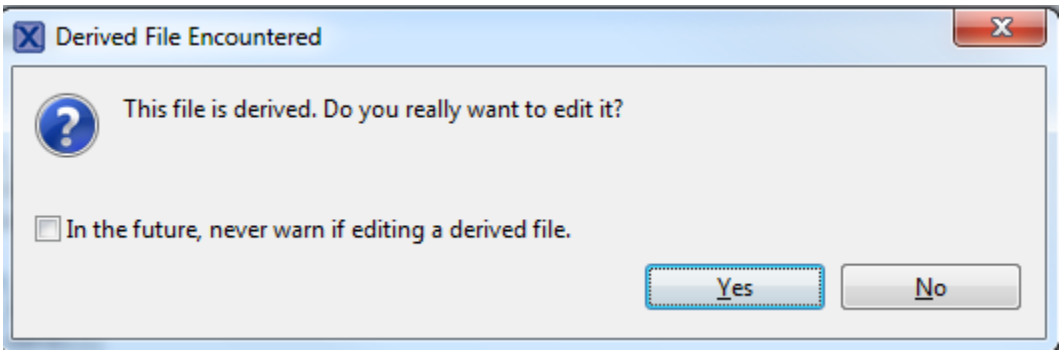


Click “OK”.

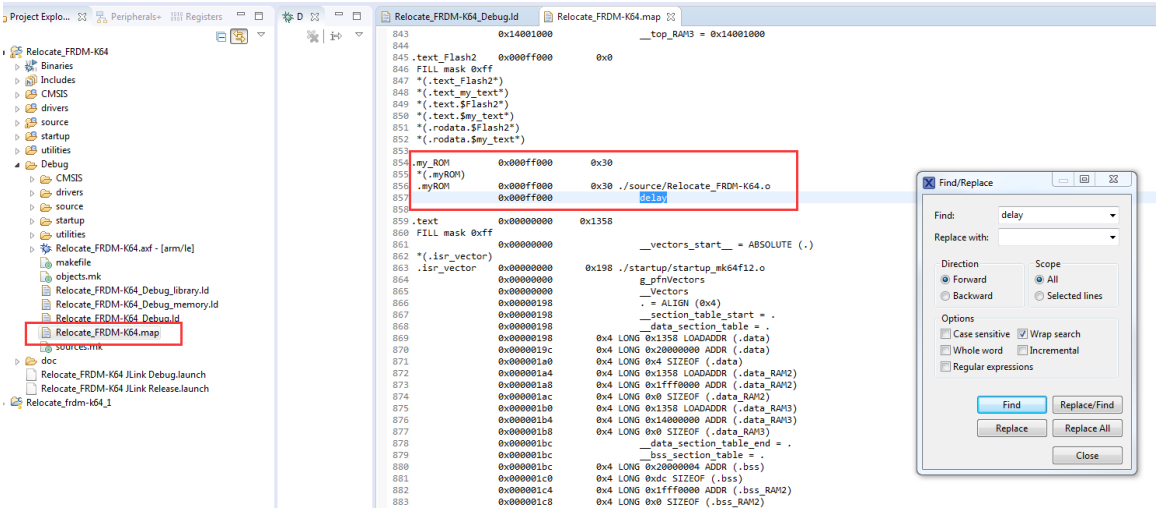
Steps 4: Create a new section(.my_ROM) in the file of “Relocate_FRDM-K64_Debug.ld” (“Relocate_FRDM-K64” is the project name made by myself) to place “.myROM”.



When you edit the *.ld file, it will shows the below section, please choose “Yes”:



Steps 5: Build the project, open “Relocate_FRDM-K64.map” file , search “delay”, we can see it is relocated in the memory of 0x000ff000:



Reference:

“Relocating Code and Data Using the KDS GCC Linker File for kinetis”

(https://community.nxp.com/docs/DOC-104433)