

试图搞懂MDK程序下载到flash(三)--MDK配置选项Linker、Target、Debug的理解分析

原创2013年05月20日 19:26:32

392810

在写这一节时，我还是想再重复一下自己写这篇文章的目的，我的目的就是为实现将MDK编写的裸机程序可不可以脱机运行，也就是不用调试的方法，因为调试的话程序默认是在SRAM中运行的，掉电丢失。而要脱机运行，就得将程序编译后的文件下载到flash中，最好能是nand flash。如果要下载到nand flash，那么就要编译生成一个bin文件，而不是用axf文件。那么剩下的问题就是，怎么生成一个完整、正确的bin文件？所以我现在就需要看能否通过编写一个分散加载文件，解决bin文件的地址问题。生成了bin文件，那么我就可以利用韦东山的方法，利用supervivi的v选项，配合DNW将程序下载到nand flash中并运行！

现在咱们就分析一下MDK配置选项，首先解释一下MDK中三种linker之间的区别：

- 1、采用Target对话框中的RAM和ROM地址。
- 采用此方式，需在Linker选项卡中勾选Use Memory Layout from Target Dialog选项（选中这一项实际上是默认在Target中对Flash和RAM的地址配置，编译链接时会产生一个默认脚本文件），并且在Target中设置好RAM、ROM地址，图2所示。MDK会根据Target选项中设定的RAM和ROM地址自动加载生成一个加载文件。最后链接器会根据此文件中的信息对目标文件进行链接，生成axf镜像文件。



狗熊王 (http://blog.csdn...)

+ 关注

(http://blog.csdn.net/mybelief321)

码云

原创	粉丝	喜欢	未开通
135	507	0	(https://github.com/mybelief321)

- 他的最新文章
- 更多文章 (http://blog.csdn.net/mybelief321)
- 生活何止是苟且
- (/mybelief321/article/details/78024427)
- 微信小程序免费HTTPS证书申请搭建教程（2）---安装SSL并使用HTTPS访问
- (/mybelief321/article/details/54561974)
- 微信小程序免费HTTPS证书申请搭建教程（1）---申请SSL
- (/mybelief321/article/details/54429314)
- C#\WPF高仿QQ音乐V12.8界面篇《2》
- (/mybelief321/article/details/53557127)



实木双层床价格



编辑推荐

最新专栏

- Keil MDK从未有过的详细使用讲解 (/y...
- MDK linker和debug的设置以及在RA...
- 试图搞懂MDK程序下载到flash(二)--分...
- 试图搞懂MDK程序下载到Flash(四)--生...
- 试图搞懂MDK程序下载到flash(五)--使...

在线课程





架构演进
http://edu.csdn.net/course/series_detail/
utm_source=blog9
http://edu.csdn.net/hu
iyiCourse/series_detail/
ce=blog9



python
可以这样学
http://edu.csdn.net/course/detail/4803?
(http://edu.csdn.net/co
utm_source=blog9
http://edu.csdn.net/4803?
utm_source=blog9)

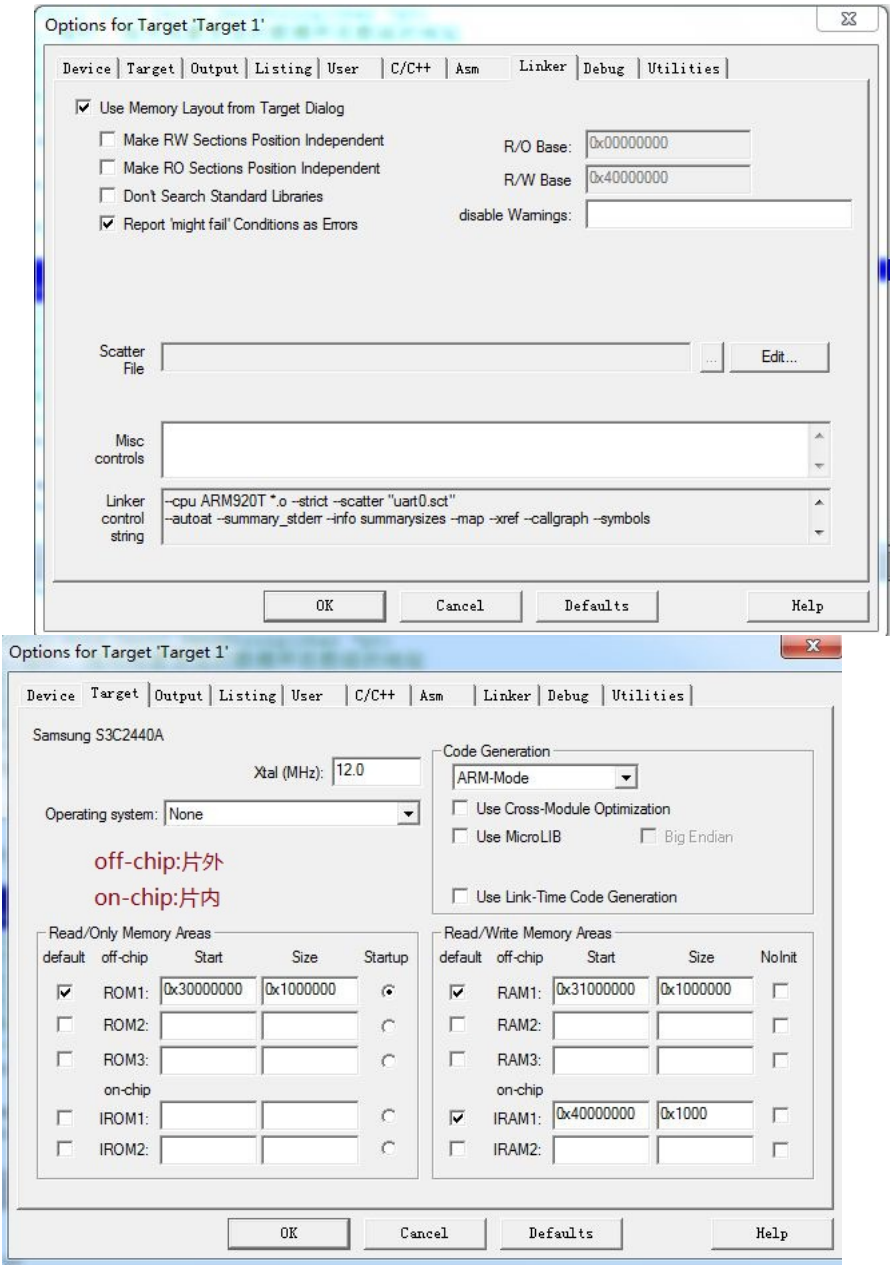


图2

至于ROM和RAM是片内还是片外、容量(Size)多大，就需要根据芯片和开发板来决定了。

- 2、直接通过Linker选项卡中的R/O Base和R/W Base来设定链接信息。
- 链接器最后可根据此处指定的地址信息进行链接，链接的文件应该是顺序存放了，最多RO和RW分开。此时需要注意的是应将Use Memory Layout from Target Dialog前的勾去掉，且保证Scatter File一栏中未包含分散加载文件，并且要在Misc controls中设定镜像文件的入口点，如：--first 2440init.o(Init) 对于这个括号内的填写依据，我暂时还不懂。

喜欢

收藏

评论

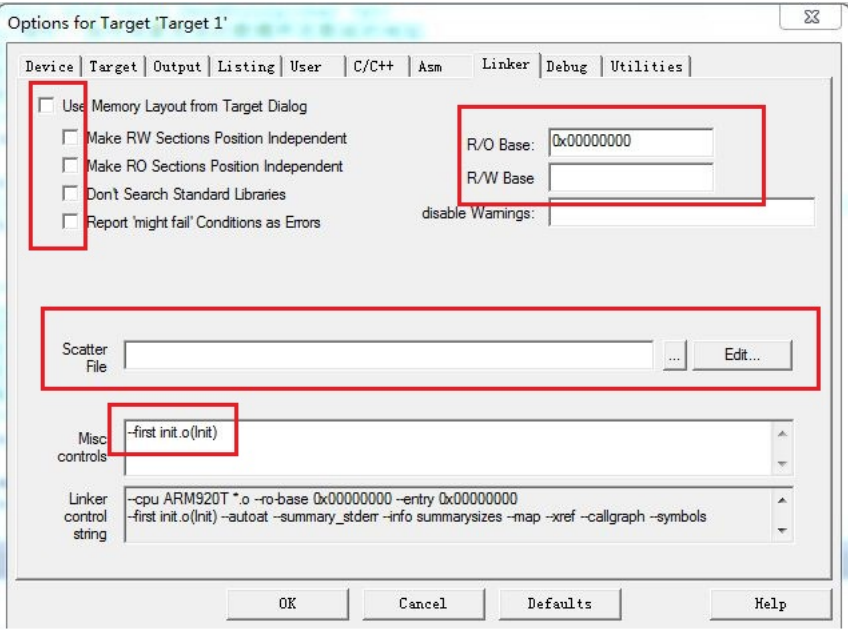


图3

图3中的R/W Base空着没填，意思就是说紧跟着R/O Base地址。

3、最后一种为直接采用分散加载文件。

在设置Linker的时候可以注意下Linker control string的信息，看看Linker的输入信息是否符合自己的要求。

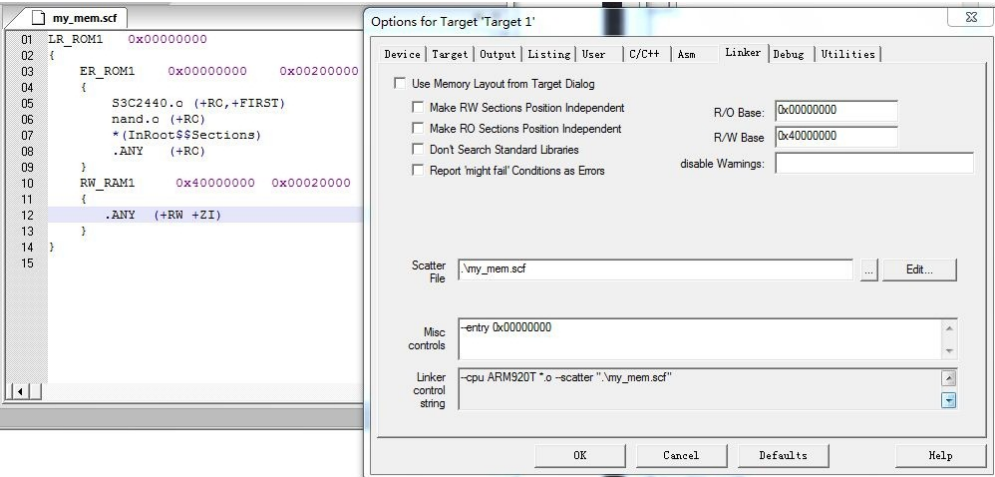


图4

此处应该是只要选择使用scatter file文件，那么其他链接方式的设置自动失效。可以从Linker control string的信息可以看出来。

现在试图着对mini2440开发板进行设置

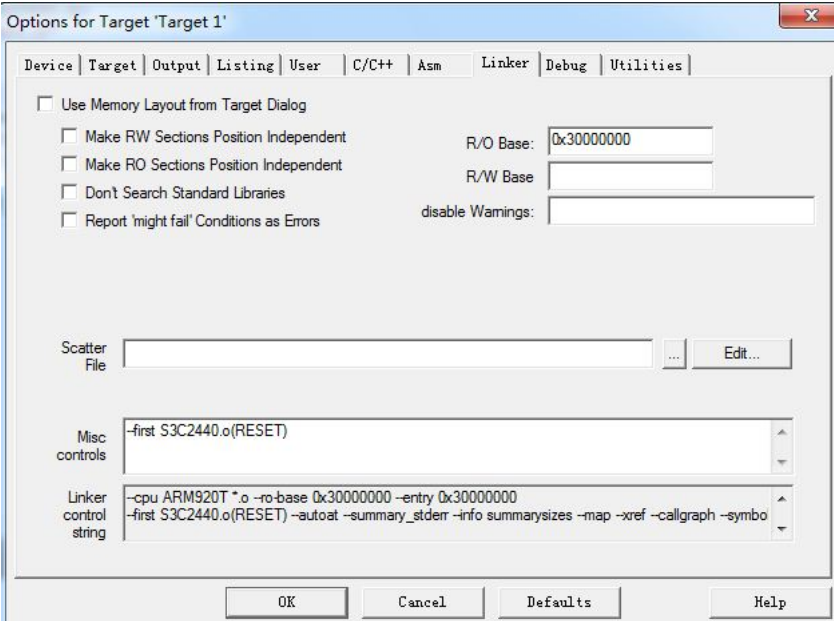
直接通过Linker选项卡中的R/O Base和R/W Base来设定链接信息

①由于mini2440开发板采用的是片外的ram和片外的flash，所以我这里直接采用前面说到的第2种方法，就是通过手动设置RO的地址，其中0x30000000就是片外ram的地址，也就是咱们的SDRAM。RW未指定，则说明RW数据顺序存放放到RO段之后。如下图。

喜欢

收藏

评论

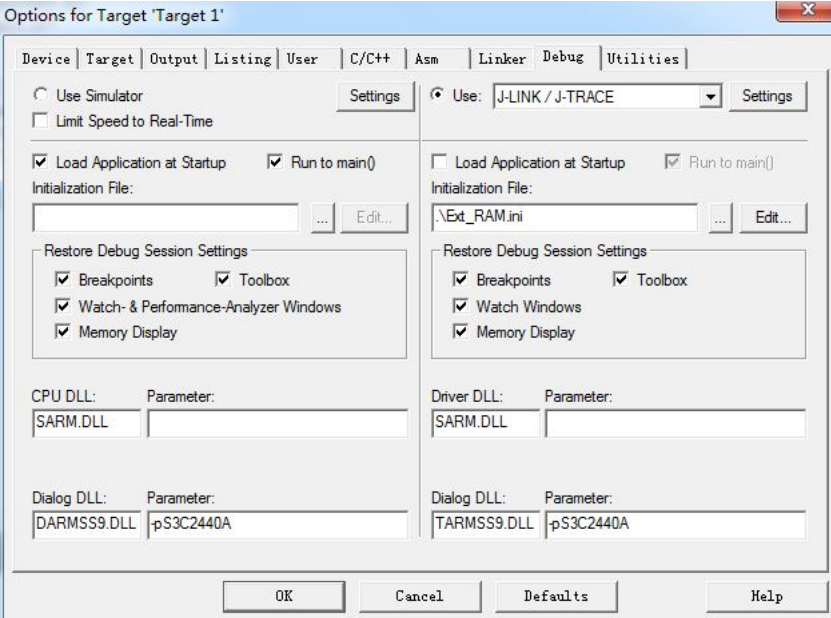


注意Misc controls的内容--first的意思是程序的入口地址，S3C2440.o是项目入口程序所在文件名的目标文件，即入口文件后缀名改为.o，（ RESET ）是程序入口段的段名，我在S3C2440.c的文件里定义的入口段的段名为RESET。如果这里设置不正确，那么编译的时候就会提醒：

Warning:Ignoring --first command. Cannot find section S3C2440.o(init)

②在option的debug选项卡下，将墨点调到右边，然后再右边的下拉框中选中J-LINK/TRACE，然后点击Setting，找到Info中的JLINK，点击查看MDK和Jlink是否能连接起来，再点击下面的target选项查看cpu。

Load Application at startup貌似去掉和不去掉的效果是一样的，猜测这个选项和前面的Target中的rom地址有关，因为ROM地址的后面有startup这个选项。我这里未采用Target方式连接，所以将此勾去掉，如下图所示。



③下面按说该接着编译了，但是编译不成功，出现了如下的错误：

Error:Undefined symbol Image

$ER_{ROM}1$

RO\$\$Length (referred from s3c2440.o)

Error:Undefined symbol Image

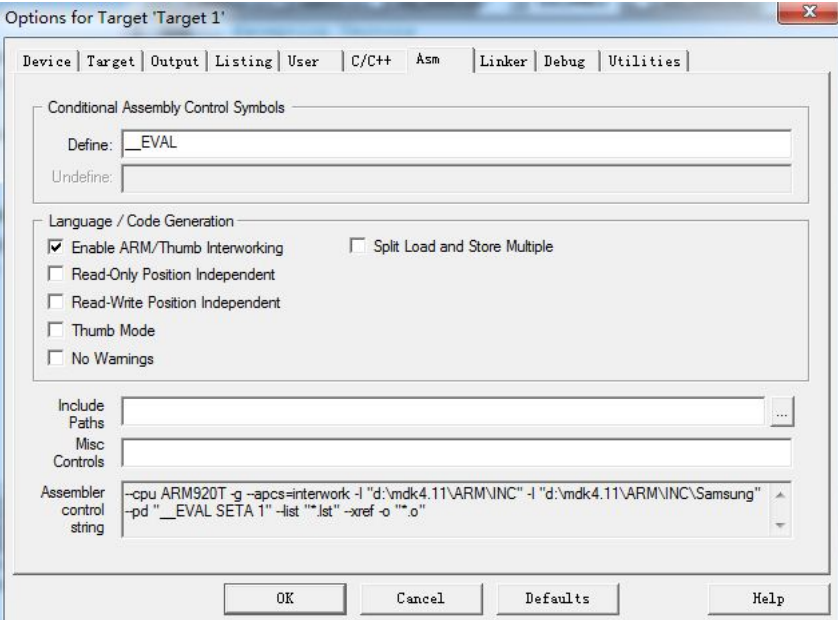
$RW_{RAM}1$

RW\$\$Length (referred from s3c2440.o)

报错是因为启动代码有一段如下：

```
0856      IF      :DEF:__EVAL
0857          DCD    0x4000
0858      ELSE
0859          DCD    ||Image$$ER_ROM1$$RO$$Length||+\
0860              ||Image$$RW_RAM1$$RW$$Length||
0861      ENDIF
```

意思是如果没有定义__EVAL，则需要引入引起错误的这两个标号，可以再Option框的Asm选项卡下的Define栏中输入__EVAL，即定义__EVAL。



修改完这个之后就可以编译通过了，然后调试就可以运行程序了。

直接采用分散加载文件

在使用这种方法时，先介绍一些知识：

- a、 Realview MDK链接程序使用了两种方式控制程序的链接，即链接控制命令选项和链接脚本文件。当使用链接控制命令选项时，链接器定义了|Image

Base|、|Image

Limit|、|Image

RO

Limit|、|Image

RO

Base|、|Image

ZI

Base|、|Image

ZI

Linit|这6个段地址描述符，这6个描述符可以直接在程序中引用。而在使用链接脚本文件后，这6个描述符就没有了，取而代之的是链接脚本文件中的段描述符，格式为：Image

段名

Base和Image



段名

Limit。具体的可以参看MDK版主文档中关于链接器定义的符号这一章！
b、RealView MDK中如何获得RO、RW、ZI的地址和长度。
在不使用Scatter文件时，默认的为Image

Base等6个地址，它的长度这样计算：Length=(Image

Limit-Image

Base)
在使用Scatter文件后，上述的6个默认地址没有了，取而代之的是Image

Base、Image

Limit表示的笛子，长度计算的方法和上述一样，即Length=(Image

Limit-Image

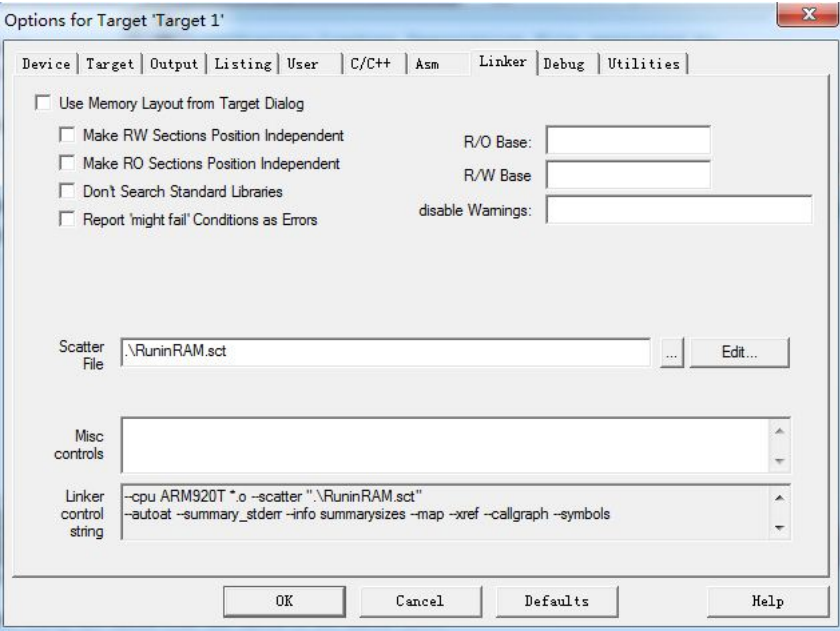
Base)
下面开始使用分散加载文件调试程序
①编写自己的分散加载文件，并命名为RuninRAM.sct，以下为RuninRAM.sct的内容：

```
*****  
;***Scatter-Loading Description File generated by  
*****  
;Run in RAM  
LR_ROM1 0x30000000 ;加载域为0x3000 0000  
{  
    ER_ROM1 0x30000000 0x0800000 ;运行域为0x3000 0000大小为0x80 0000  
    {  
        *.o (RESET,+First) ;将入口程序的RESET段放在首位  
        *(InRoot$$Sections)  
        .ANY (+RO)  
    }  
    RW_RAM1 0x30800000 0x0800000 ;RW data  
    {  
        .ANY (+RW +ZI)  
    }  
    RW_IRAM1 0x40000000 0x00001000  
  
    {  
  
        .ANY (+RW +ZI)
```


}

}

②配置选项Options/Linker中加入咱们这个分散加载文件，如下图所示



③Options/Target选项中关于内存地址的设置已经无所谓了

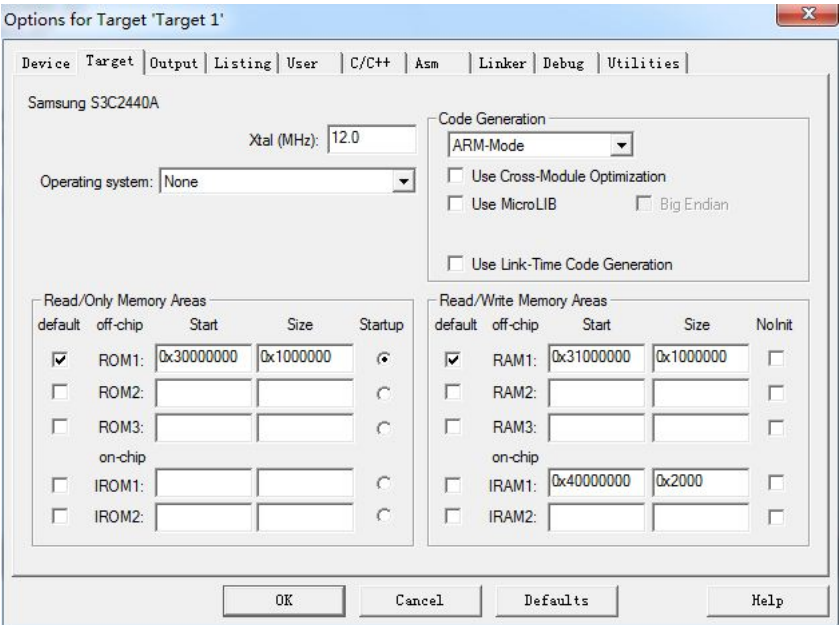
④编译调试就可以了。

到这里，咱们已经会了分散加载文件的编写以及使用。

部分问题进一步分析

关于方式1中采用Target选项中的RAM地址和ROM地址，它默认的是生成一个分散加载文件，咱们现在通过改变RAM地址的设置，进一步认识一下分散加载文件

第一种：



相对应的分散加载文件



```
1 ; *****
2 ; *** Scatter-Loading Description File generated by uVision ***
3 ; *****
4
5 LR_ROM1 0x30000000 0x02000000 { ; load region size_region
6   ER_ROM1 0x30000000 0x02000000 { ; load address = execution address
7     *.o (RESET, +First)
8     *(InRoot$$Sections)
9     .ANY (+RO)
10  }
11 RW_RAM1 0x32000000 0x01000000 { ; RW data
12   .ANY (+RW +ZI)
13 }
14 RW_IRAM1 0x40000000 0x00001000 {
15   .ANY (+RW +ZI)
16 }
17 }
```

第二种，只是在第一种的基础上，将IRAM1勾选

Options for Target 'Target 1'

Device Samsung S3C2440A

Xtal (MHz): 112.0

Operating system: None

Code Generation

ARM-Mode

Use Cross-Module Optimization

Use MicroLIB

Big Endian

Use Link-Time Code Generation

Read/Only Memory Areas

default off-chip Start Size Startup

ROM1: 0x30000000 0x1000000

ROM2:

ROM3:

on-chip

IROM1:

IROM2:

Read/Write Memory Areas

default off-chip Start Size NoInit

RAM1: 0x31000000 0x1000000

RAM2:

RAM3:

on-chip

IRAM1: 0x40000000 0x2000

IRAM2:

OK Cancel Defaults Help

相对应的分散加载文件

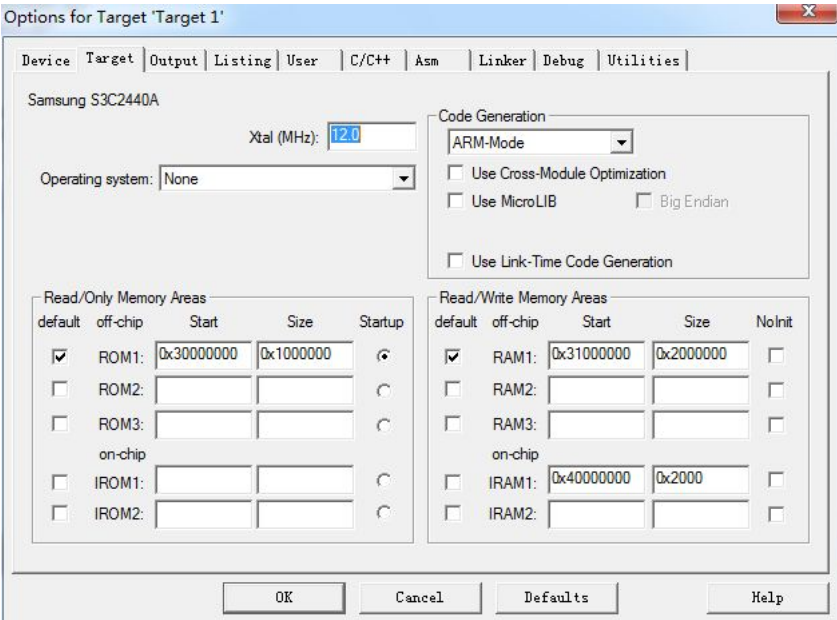
```
1 ; *****
2 ; *** Scatter-Loading Description File generated by uVision ***
3 ; *****
4
5 LR_ROM1 0x30000000 0x01000000 { ; load region size_region
6   ER_ROM1 0x30000000 0x01000000 { ; load address = execution address
7     *.o (RESET, +First)
8     *(InRoot$$Sections)
9     .ANY (+RO)
10  }
11 RW_RAM1 0x31000000 0x01000000 { ; RW data
12   .ANY (+RW +ZI)
13 }
14 }
```

第三种，在第一种的基础上，将RAM1的范围改变一下：

喜欢

收藏

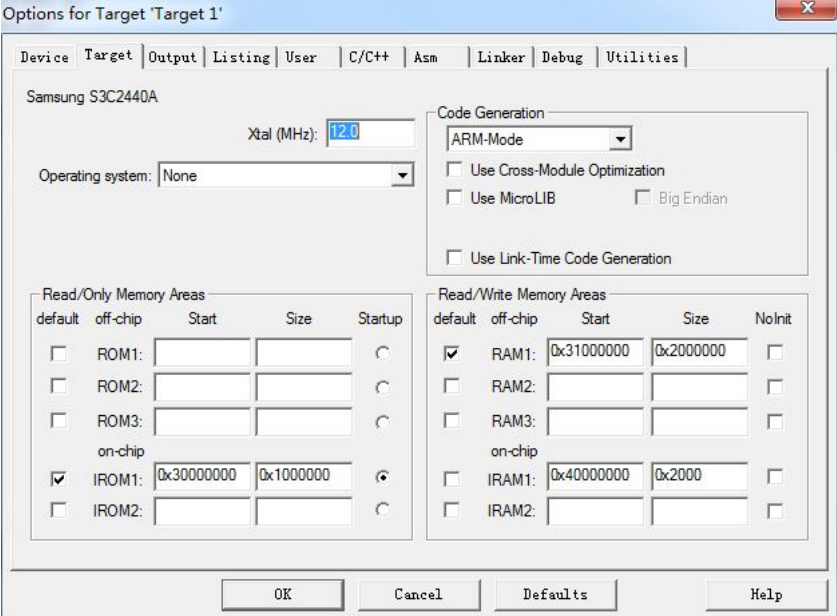
评论



相对应的分散加载文件

```
1 ; *****
2 ; *** Scatter-Loading Description File generated by uVision ***
3 ; *****
4
5 LR_ROM1 0x30000000 0x01000000 { ; load region size_region
6   ER_ROM1 0x30000000 0x01000000 { ; load address = execution address
7     *.o (RESET, +First)
8     *(InRoot$$Sections)
9     .ANY (+RO)
10  }
11  RW_RAM1 0x31000000 0x01000000 { ; RW data
12    .ANY (+RW +ZI)
13  }
14  RW_IRAM1 0x40000000 0x00002000 {
15    .ANY (+RW +ZI)
16  }
17 }
```

第四种，在第一种的基础上，将ROM1换为IROM1



这时候编译不通过了，产生如下的错误

Build Output

Build target 'Target 1'
linking...
uart0.axf: Error: L6218E: Undefined symbol Image\$\$ER_ROM1\$\$RO\$\$Length (referred from s3c2440.o).
uart0.axf: Finished: 0 information, 0 warning and 1 error messages.
Target not created

由此错误可以看出它产生的这个标号与ROM有关，而不是IROM

版权声明：本文为博主原创文章，未经博主允许不得转载。


举报

本文已收录于以下专栏：min2440专栏 (<http://blog.csdn.net/column/details/mybelief321.html>)

喜欢

收藏

评论



afuhanxiaozhu (/afuhanxiaozhu)

2013-10-30 22:28

1楼


(/afuhanxiaozhu)那个自动生成分散加载文件在哪里可以看到，或者说哪里可以设置看到，谢谢了。

回复

相关文章推荐

Keil MDK从未有过的详细使用讲解 (/yx_l128125/article/details/9144887)

转自博客：<http://blog.csdn.net/zhzht19861011/article/details/5846510> 这博主关于MDK 的使用的文章，写的得TM的好 TM的实用！真心收...

 yx_l128125 (http://blog.csdn.net/yx_l128125) 2013-06-21 14:08 41503

MDK linker和debug的设置以及在RAM中调试 (/rockrockwu/article/details/7093435)

有误或者表述不清楚请指出，谢谢 硬件：TQ2440开发板、jlink V8 固件 软件：J-LINK ARM 4.08i、MDK4.20 先解释下MDK中三种link...

 rockrockwu (<http://blog.csdn.net/rockrockwu>) 2011-12-21 20:33 9386




我是如何成为一名python大咖的？ (http://edu.csdn.net/topic/python1?utm_source=blog10)

人生苦短，都说必须python，那么我分享下我是如何从小白成为Python资深开发者的吧。2014年我大学毕业...

试图搞懂MDK程序下载到flash(二)--分散加载文件scatter (/mybelief321/article/details/8941238)

分散加载文件概念 对于分散加载文件的概念，在《ARM体系结构与编程》书第11章有明确介绍。 分散加载文件(即 scatter file,后缀为 .scf)是一个文本文件，通过编...

 mybelief321 (<http://blog.csdn.net/mybelief321>) 2013-05-19 18:09 4976

试图搞懂MDK程序下载到Flash(四)--生成bin文件下载到Nor Flash (/mybelief321/article/details/8952306)

喜讯啊！！下载到flash中的一种实现了啊！！鸡冻了！终于可以脱机运行了，尽管是下载到了Nor Flash中运行，还没有下载到Nand Flash中运行，但是这也是一个进步了吧，下面详细介绍一下MD...

