

零死角玩转STM32



MDK的编译过程及文件类型全解

淘宝：firestm32.taobao.com

论坛：www.chuxue123.com



扫描进入淘宝店铺

主讲内容



01

编译过程

02

程序的组成、存储与运行

03

编译工具链

04

MDK工程的文件类型

05

实验：自动分配变量到外部SDRAM

06

**实验：优先使用内部SRAM并
分配堆到SDRAM**

MDK的编译过程及文件类型全解



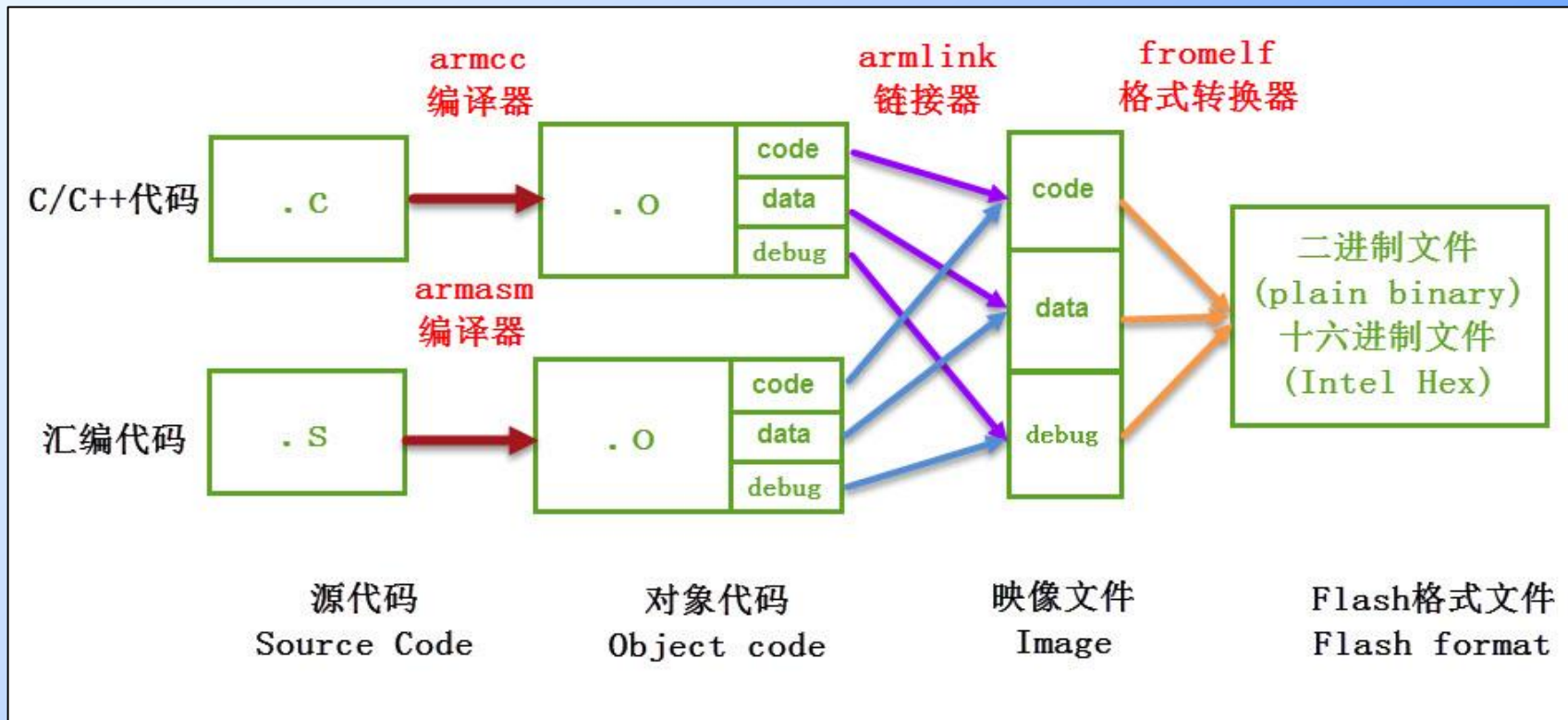
学习到这里，大家已经非常熟练地使用MDK创建应用程序了，平时使用MDK编写源代码，然后编译生成机器码，再把机器码下载到STM32芯片上运行，但是这个编译、下载的过程MDK究竟做了什么工作？它编译后生成的各种文件又有什么作用？本章节将对这些过程进行讲解，了解编译及下载过程有助于理解芯片的工作原理，这些知识对制作IAP(bootloader)以及读写控制器内部FLASH的应用时非常重要。

MDK的编译过程及文件类型全解

编译过程

编译过程简介

首先简单了解下MDK的编译过程，它与其它编译器的工作过程是类似的，该过程图例如下：



MDK的编译过程及文件类型全解



编译过程

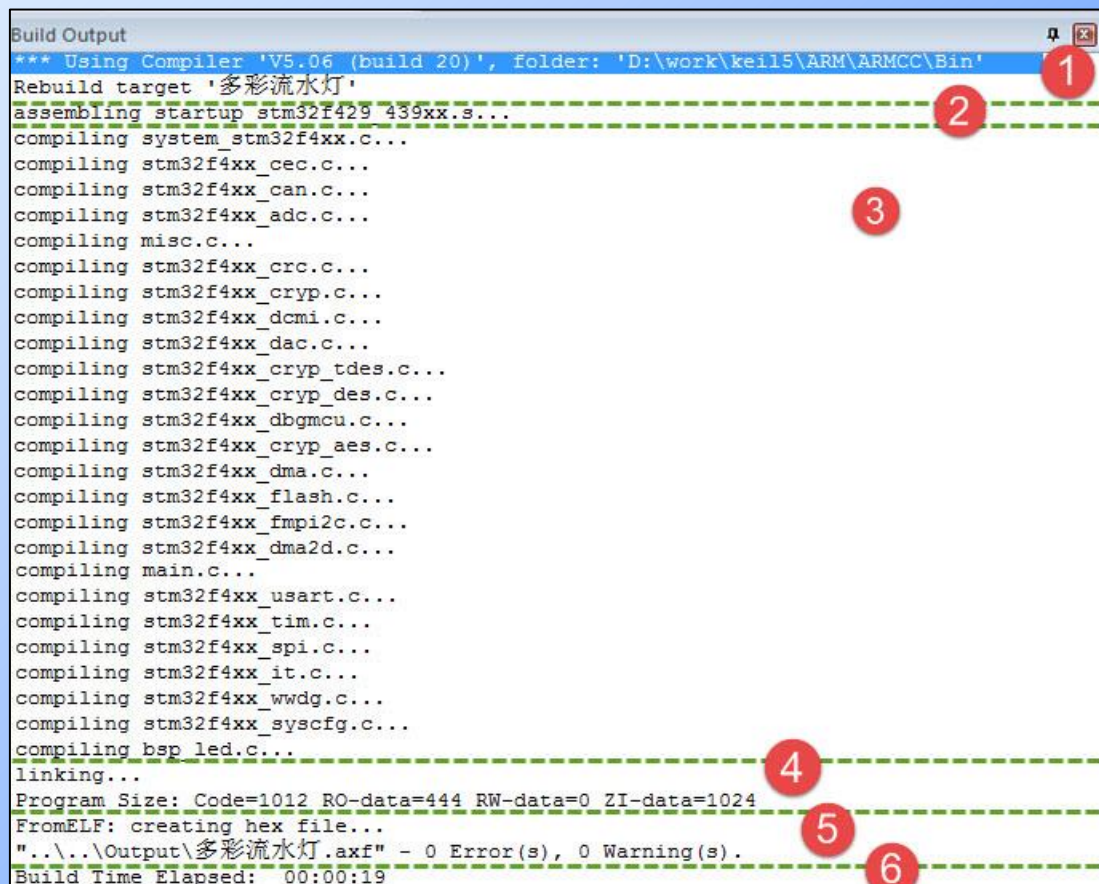
编译过程主要流程如下：

- **编译：**MDK软件使用的编译器是armcc和armasm，它们根据每个c/c++和汇编源文件编译成对应的以“.o”为后缀名的对象文件(Object Code，也称目标文件)，其内容主要是从源文件编译得到的机器码，包含了代码、数据以及调试使用的信息；
- **链接：**链接器armlink把各个.o文件及库文件链接成一个映像文件“.axf”或“.elf”；
- **格式转换：**一般来说Windows或Linux系统使用链接器直接生成可执行映像文件elf后，内核根据该文件的信息加载后，就可以运行程序了，但在单片机平台上，需要把该文件的内容加载到芯片上，所以还需要对链接器生成的elf映像文件利用格式转换器fromelf转换成“.bin”或“.hex”文件，交给下载器下载到芯片的FLASH或ROM中。

MDK的编译过程及文件类型全解

具体工程中的编译过程

下面打开“多彩流水灯”的工程，以它为例进行讲解，其它工程的编译过程也是一样的，只是文件有差异。打开工程后，点击MDK的“rebuild”按钮，它会重新构建整个工程，构建的过程会在MDK下方的“Build Output”窗口输出提示信息。



```
Build Output
*** Using Compiler 'V5.06 (build 20)', folder: 'D:\work\keil5\ARM\ARMCC\Bin'
Rebuild target '多彩流水灯'
assembling startup_stm32f429_439xx.s...
compiling system_stm32f4xx.c...
compiling stm32f4xx_cec.c...
compiling stm32f4xx_can.c...
compiling stm32f4xx_adc.c...
compiling misc.c...
compiling stm32f4xx_crc.c...
compiling stm32f4xx_cryp.c...
compiling stm32f4xx_dcmi.c...
compiling stm32f4xx_dac.c...
compiling stm32f4xx_cryp_tdes.c...
compiling stm32f4xx_cryp_des.c...
compiling stm32f4xx_dbgmcu.c...
compiling stm32f4xx_cryp_aes.c...
compiling stm32f4xx_dma.c...
compiling stm32f4xx_flash.c...
compiling stm32f4xx_fmpi2c.c...
compiling stm32f4xx_dma2d.c...
compiling main.c...
compiling stm32f4xx_usart.c...
compiling stm32f4xx_tim.c...
compiling stm32f4xx_spi.c...
compiling stm32f4xx_it.c...
compiling stm32f4xx_wwdg.c...
compiling stm32f4xx_syscfg.c...
compiling bsp_led.c...
linking...
Program Size: Code=1012 RO-data=444 RW-data=0 ZI-data=1024
FromELF: creating hex file...
"..\..\Output\多彩流水灯.axf" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:19
```

MDK的编译过程及文件类型全解



具体工程中的编译过程

构建工程的提示输出主要分6个部分，说明如下：

- 提示信息的第一部分说明构建过程调用的编译器。图中的编译器名字是“V5.06(build 20)”，后面附带了该编译器所在的文件夹。在电脑上打开该路径，可看到该编译器包含下图中的各个编译工具，如armar、armasm、armcc、armlink及fromelf，其中前面没有提到的armar是用于把.o文件打包成lib文件的。
- 使用armasm编译汇编文件。图中列出了编译startup启动文件时的提示，编译后每个汇编源文件都对应有一个独立的.o文件。
- 使用armcc编译c/c++文件。图中列出了工程中所有的c/c++文件的提示，同样地，编译后每个c/c++源文件都对应有一个独立的.o文件。

MDK的编译过程及文件类型全解

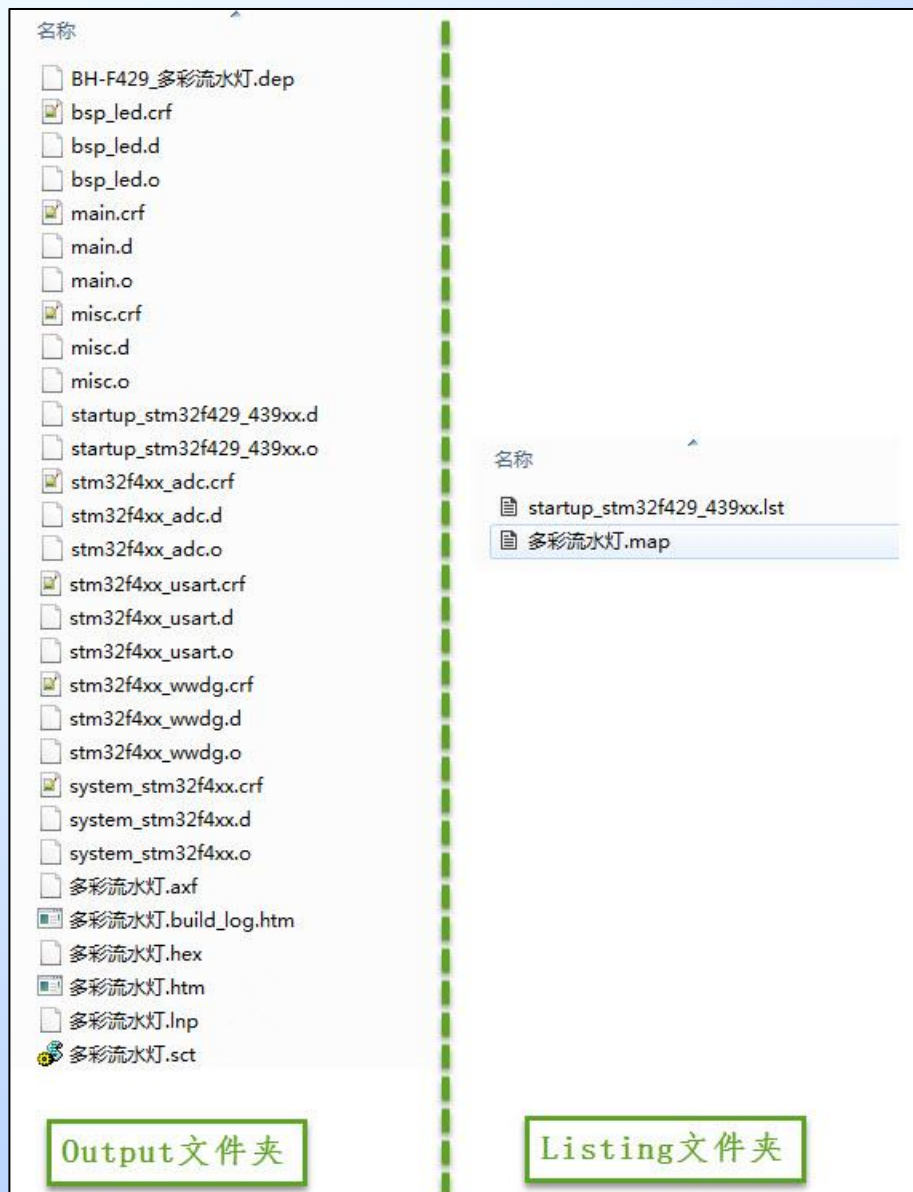


具体工程中的编译过程

- 使用armlink链接对象文件，根据程序的调用把各个.o文件的内容链接起来，最后生成程序的axf映像文件，并附带程序各个域大小的说明，包括Code、RO-data、RW-data及ZI-data的大小。
- 使用fromelf生成下载格式文件，它根据axf映像文件转化成hex文件，并列出生成过程出现的错误(Error)和警告(Warning)数量。
- 最后一段提示给出了整个构建过程消耗的时间。

构建完成后，可在工程的“Output”及“Listing”目录下找到由以上过程生成的各种文件，如下图：

MDK的编译过程及文件类型全解



可以看到，每个C源文件都对应生成了.o、.d及.crf后缀的文件，还有一些额外的.dep、.hex、.axf、.htm、.lnp、.sct、.lst及.map文件。

零死角玩转STM32



THANKS

论坛：www.chuxue123.com

淘宝：firestm32.taobao.com



扫描进入淘宝店铺