# Micrium\_STM32F107\_uCOS-III源码文件结构

## 1.工程总目录



## 2.EvalBoards文件夹

这个文件夹是关于STM32F107的工程文件，其中的app\_cfg.h、cpu\_cfg.h、includes.h、lib\_cfg.h、os\_app\_hooks.c、os\_app\_hooks.h、os\_cfg.h、os\_cfg\_app.h 这八个文件是我们移植需要的



## 3.uC\_CPU文件夹：

1、cpu\_core.c文件：该文件包含了适用于所有CPU架构的C代码。包含了用来测量中断关闭事件的函数（中断关闭和打开分别由CPU\_CRITICAL\_ENTER() 和 CPU\_CRITICAL\_EXIT()两个宏实现），还包含了一个可模仿前导零计算的函数（防止CPU不提供这样的指令），以及一些其他的函数功能。

2、cpu\_core.h文件：包含cpu\_core.c中的函数原型声明，以及用来测量中断关闭时间变量的定义。

3、cpu\_def.h文件：包含uC/CPU模块使用的各种#define常量。

4、ARM\_Cortex\_M3文件夹：其中有3个文件GUN、IAR、RealView，我们用的是Keil MDK编译器，所以RealView是我们需要的。



RealView中有3个文件：

cpu.h包含了一些类型的定义，使UCOSIII和其他模块可与CPU架构和编译器字宽无关。在该文件中用户能够找到CPU\_INIT16U、CPU\_INIT32U、CPU\_FP32等数据类型的定义。该文件还指定了CPU使用的是大端模式还是小端模式，定义了UCOSIII使用的CPU\_STK数据类型，定义了CPU\_CRITICAL\_ENTER()和CPU\_CRITICAL\_EXTI()，还包括了一些与CPU架构相关的函数声明。

cpu\_a.asm文件：包含了一些汇编语言编写的函数，可用来开中断和关中断，计算前导零（如果CPU支持这条指令），以及其他一些只能用汇编语言编写的与CPU相关的函数，这个文件中的函数可以从C代码库中调用。

cpu\_c.c文件：包含了一些基于特定CPU架构的但为了可移植而用C语言编写的函数C代码，作为一个普通原则，除非汇编语言能显著提高性能，否则尽量用C语言编写函数。



## 4.uC\_LIB文件夹

其由一些可移植并且与编译器无关的函数组成，UCOSIII 中不使用uC\_LIB中的函数，但是这里假定lib\_def.h存在，uC\_LIB中包含以下文件：

lib\_ascii.h和lib\_ascii.c文件、lib\_def.h文件、lib\_math.h和lib\_math.c文件、lib\_mem.c和lib\_mem.h文件、lib\_str.c和lib\_str.h文件、lib\_mem\_a.asm文件。以上文件有兴趣的同志们自己去查有关资料。