

STM32F4XX 固件库之 DSP 库的说明

为了充分发挥 STM32F4XX（Cortex-M4F）处理器浮点运算性能，建议使用固件库自带的 arm_math.h 头文件而非编译器自带的 math.h。此文件位于\Libraries\CMSIS\Lib 文件夹。

硬件准备：

需要开启 STM32F4XX 的 FPU，所以要确定 system_stm32f4xx.c 文件中的 SystemInit()函数有如下代码，如果没有则加上：

```
#if (__FPU_PRESENT == 1) && (__FPU_USED == 1)
    SCB->CPACR |= ((3UL << 10*2)|(3UL << 11*2)); /* set CP10 and CP11 Full Access */
#endif
```

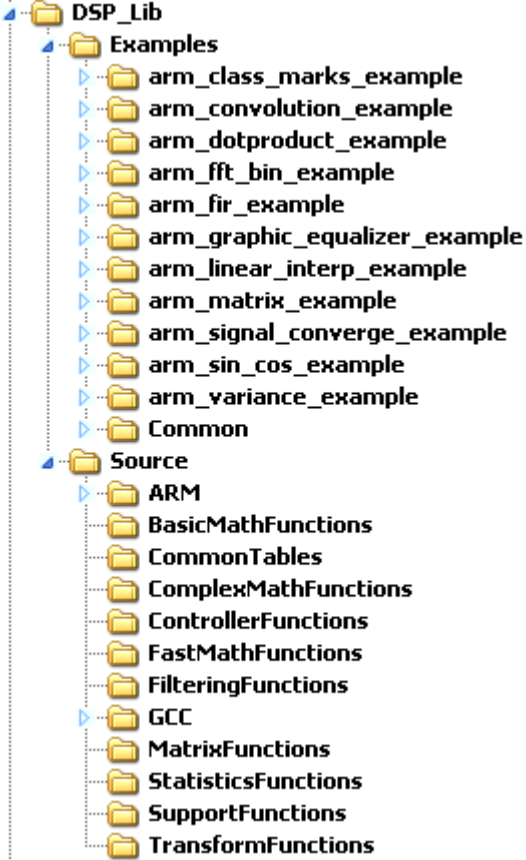
编译控制：

根据 arm_math.h 中的要求（备注 1），在工程编译选项（Option）之 C/C++中加入如下编译控制定义：

```
__FPU_PRESENT=1
__FPU_USED =1
ARM_MATH_CM4
__CC_ARM
```

对于 Cortex-M0 或者 M3 处理器类型，由于没有 FPU 因此无法直接进行浮点运算，只能将浮点数进行 Q 规格化(q7、q15 或 Q31)处理，如开平方运算：M0/M3 只能通过迭代法（标准数学函数库）计算，而 M4F 直接调用 VSQRT 指令完成。

文件结构：

	<p>例程组：各种实际应用的例程，如 FFT 变换，FIR 滤波，矩阵处理等等</p>
	<p>公共文件 math_helper，提供例程需要的辅助函数</p> <p>源文件组：各种功能函数的定义或原型。 ARM 和 GCC 文件夹的工程将所有源文件编译为库 (lib)，在应用时供调用</p>

事实上 arm_math.h 文件中的定义就为源文件组使用的。因此在将源文件组编译为库时，在应用工程中添加这个库和 arm_math.h 即可访问所有 DSP 库功能。也可以根据需要只添加 arm_math.h 和需要的源文件，以缩短程序长度。

BasicMathFunctions

提供浮点数的各种基本运算函数，如加减乘除等运算。对于 M0/M3 只能用 Q 运算，即文件夹下以_q7、_q15 和_q31 结尾的文件；而 M4F 能直接硬件浮点计算，属于文件夹下以_f32 结尾的文件。

CommonTables

arm_common_tables.c 文件提供位翻转或相关参数表。

ComplexMathFunctions

复述数学功能，如向量处理，求模运算的。

ControllerFunctions

控制功能，主要为 PID 控制函数。arm_sin_cos_f32/-q31.c 函数提供 360 点正余弦函数表和任意角度的正余弦函数值计算功能。

FastMathFunctions

快速数学功能函数，提供 256 点正余弦函数表和任意任意角度的正余弦函数值计算功能，和 Q 值开平方运算：

Arm_cos_f32/_q15/_q31.c: 提供 256 点余弦函数表和任意角度余弦值计算功能。

Arm_sin_f32/_q15/_q31.c: 提供 256 点正弦函数表和任意角度正弦值计算功能。

Arm_sqrt_q15/q31.c: 提供迭代法计算平方根的函数。

对于 M4F 的平方根运算，通过执行 VSQRT 指令完成。

FilteringFunctions

滤波函数功能，主要为 FIR 和 LMS（最小均方根）滤波函数。

MatrixFunctions

矩阵处理函数。

StatisticsFunctions

统计功能函数，如求平均值、计算 RMS、计算方差/标准差等。

SupportFunctions

支持功能函数，如数据拷贝，Q 格式和浮点格式相互转换，Q 任意格式相互转换。

TransformFunctions

变换功能。包括复数 FFT (CFFT) /复数 FFT 逆运算 (CIFFT)、实数 FFT (RFFT) /实数 FFT 逆运算 (RIFFT)、和 DCT（离散余弦变换）和配套的初始化函数。