

## 一．基本情况

1. 外设库：LL 库，HAL 库
2. 操作系统：FreeRTOS
3. 配置软件：STM32Cube MX

## 二．工程结构

底层代码整体上分为 bsp, driver, task 三层。

bsp 层的内容是外设配置，包含了一些对 cube 没有生成的外设配置代码的补充。driver 层是驱动层，内容包括电机的速度环、位置环，传感器读取等。**task 层是各车主要需要写代码的一层，内容应是控制逻辑。**

下面是对整个工程结构的具体描述：

Groups	Files	内容说明
Drivers/STM32F4xx_HAL_Driver	略	LL 库，HAL 库源码
Middlewares/FreeRTOS	略	FreeRTOS 源码
Drivers/CMSIS	stm32f405xx.h	F4 配置文件
	system_stm32f4xx.c	
Application/MDK-ARM	startup_stm32f405xx.s	启动文件。
MPL/User	略	MPU9250 的 MPL 库文件
Application/User	main.c	main 函数所在文件
	freertos.c	FreeRTOS 各任务的创建及任务函数的入口
	stm32f4xx_it.c	中断函数所在文件
	delay.c	delay 函数相关代码
	其他	cube 中配置外设所生成的代码
<b>BSP 层：</b> Application/UserBsp	BSPconfig.h	所有底层配置的接口
	其他	对 cube 没有生成的外设配置代码的补充
MathFunction/User	pid.c	PID 算法代码
	ramp.c	斜坡函数代码
<b>TASK 层：</b> Task/User	task_chassis.c	底盘控制逻辑
	task_check_remainmemory.c	任务剩余内存检测任务
	task_feedmotor.c	拨弹控制逻辑
	task_gimbal.c	云台控制逻辑
	task_lostcounter.c	丢失数据计数任务
	task_remote.c	遥控器逻辑
<b>DRIVER 层：</b> Driver/User	driver_chassis.c	底盘速度环，运动封装
	driver_feedmotor.c	拨弹速度环，位置环
	driver_friction.c	摩擦轮开关
	driver_gimbal.c	云台速度环，位置环
	driver_laser.c	激光开关
	driver_magazine.c	弹仓开关

	driver_remote.c	遥控器数据分割
	driver_mpu9250.c	MPU9250 数据读取

### 三．注意事项

1. 使用 cube 重新生成代码后，如果报错如下：

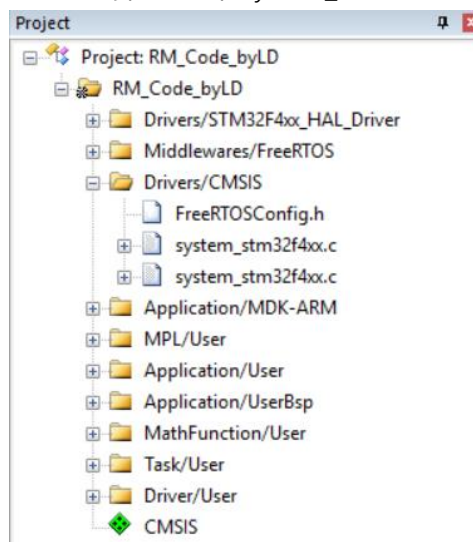
```

Build Output
Linking...
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol AHBPrescTable multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol APBPrescTable multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol SystemCoreClock multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol SysTick_Handler multiply defined (by stm32f4xx_it.o and port.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol __asm__18_system_stm32f4xx_c_5d646a67__REV16 multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol __asm__18_system_stm32f4xx_c_5d646a67__REVSH multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol __asm__18_system_stm32f4xx_c_5d646a67__RRX multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol SystemCoreClockUpdate multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
RM_Code_byLD\RM_Code_byLD.axf: Error: L6200E: Symbol SystemInit multiply defined (by system_stm32f4xx_1.o and system_stm32f4xx.o).
Not enough information to list image symbols.
Not enough information to list load addresses in the image map.
Finished: 2 information, 0 warning and 9 error messages.
"RM_Code_byLD\RM_Code_byLD.axf" - 9 Error(s), 21 Warning(s).
Target not created.
Build Time Elapsed: 00:01:18

```

解决方案有：

- a. 删去 Drivers/CMSIS 文件夹下的任意一个 system\_stm32f4xx.c 文件。



- b. 注释 FreeRTOSConfig.h 文件中的#define xPortSysTickHandler SysTick\_Handler

```

165 /* IMPORTANT: This define is commented when used with STM32Cube firmware, when timebase is systic,
166      to prevent overwriting SysTick_Handler defined within STM32Cube HAL */
167 // #define xPortSysTickHandler SysTick_Handler
168
169 /* USER CODE BEGIN Defines */
170 /* Section where parameter definitions can be added (for instance, to override default ones in FreeRTOS.h) */
171 /* USER CODE END Defines */
172
173 #endif /* FREERTOS_CONFIG_H */
174

```

2. USB CDC 使用：

若插上电脑后无法识别，将启动文件 startup\_stm32f405xx.s 中 Heap\_Size 由 0x200 改为 0x400。

```

52
53 ; <h> Heap Configuration
54 ;   <o> Heap Size (in Bytes) <0x0-0xFFFFFFFF:8>
55 ; </h>
56
57 Heap_Size      EQU      0x400
58

```

2019 年 01 月 04 日