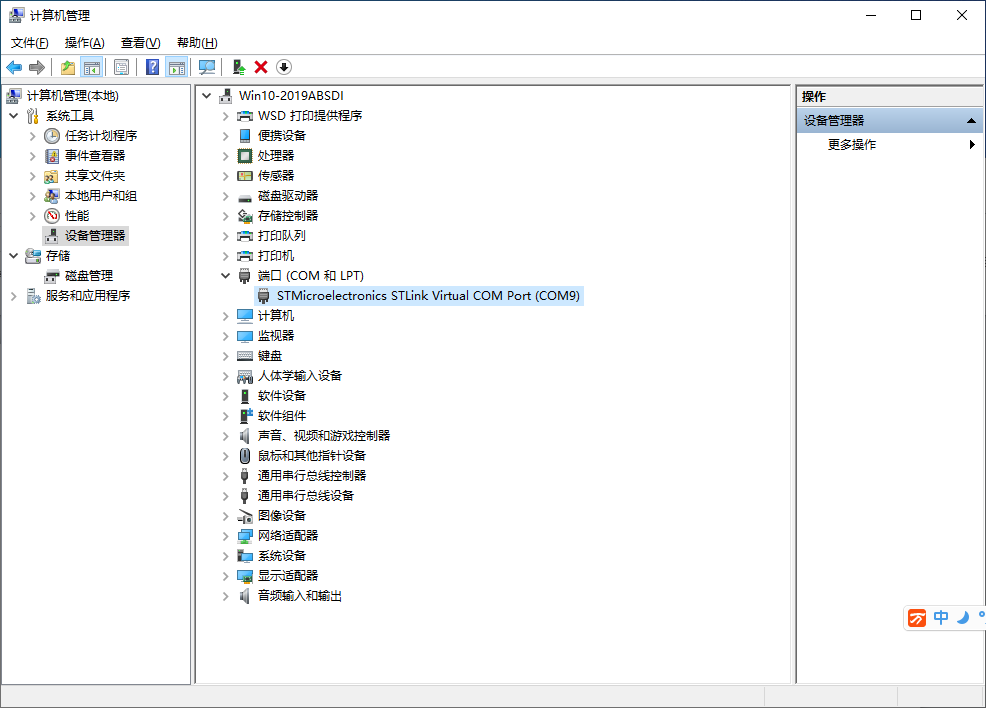
按《hand-on RTOS with Microcontrollers Building real-time embedded system using freeRTOS, STM32 mcus, and SEGGER debug tools 》设计实验

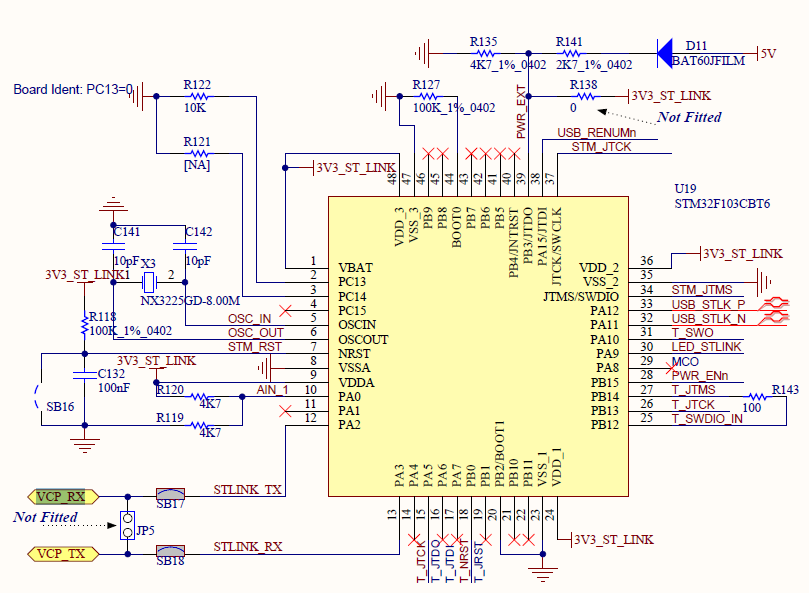
实验一：UART pull mode

Step1: 确定与PC间的串口号 （COM9）



Step 2: 通过原理图确定PC上的串口号对应芯片上的哪个串口

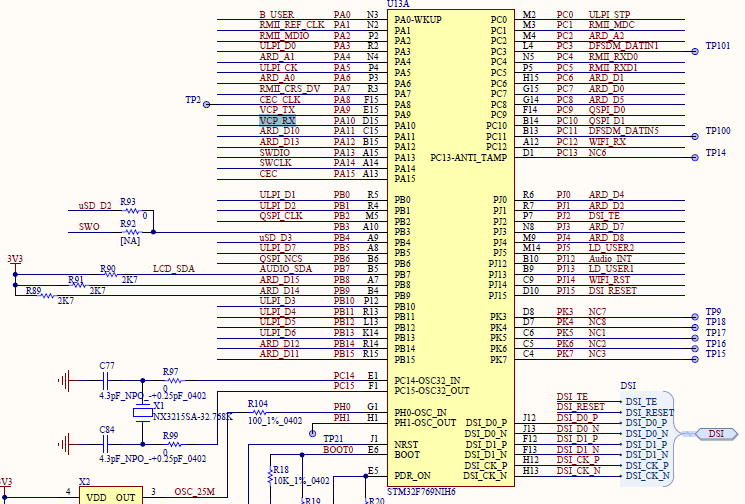
Step 2.1 通过原理图确定USB-uart的桥接芯片



通过USB与PC进行串口通讯

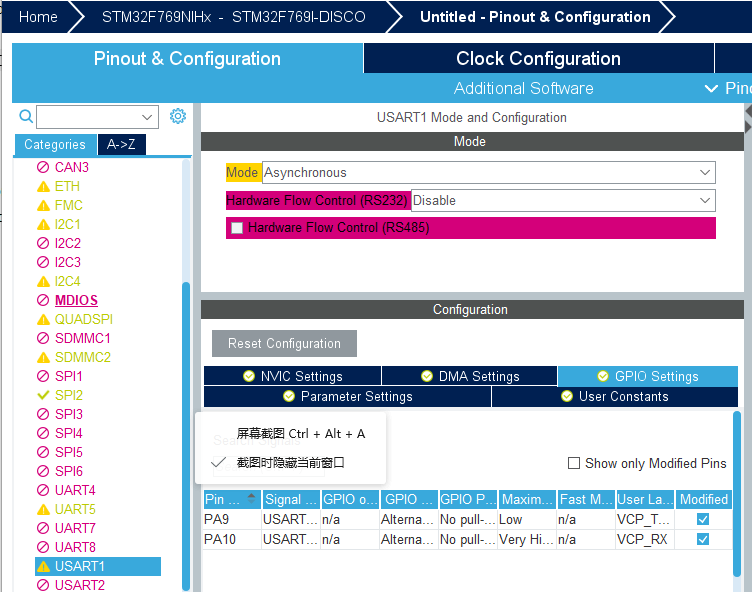
与STM32主芯片进行串口通讯

Step 2.2 确定主芯片的引脚

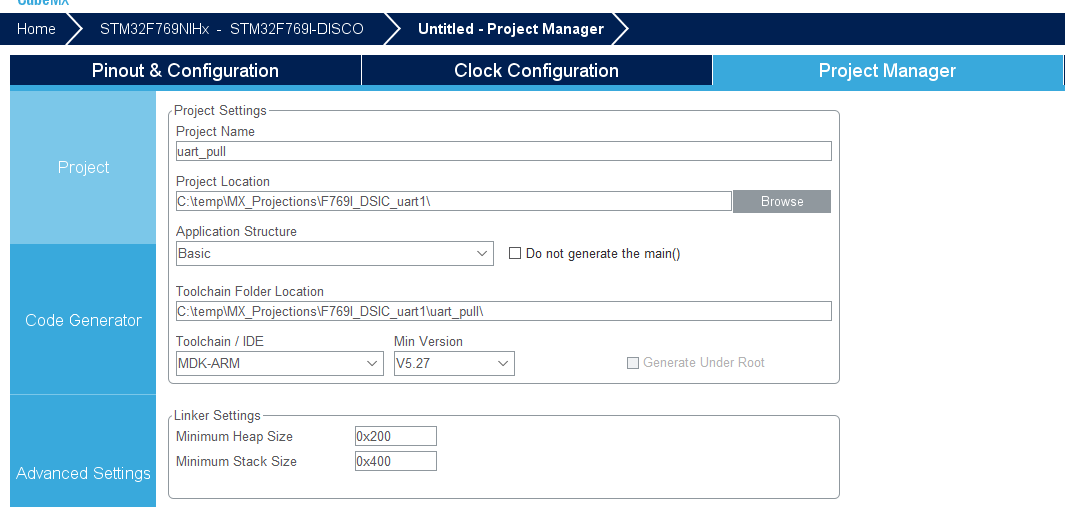


主芯片进行串口通讯引脚PA9,10

Step 3: stm32cubeMX上确定主芯片串口号

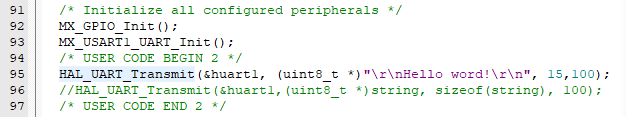


Step 4: 配置cubeMX的project manager

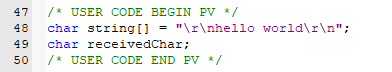


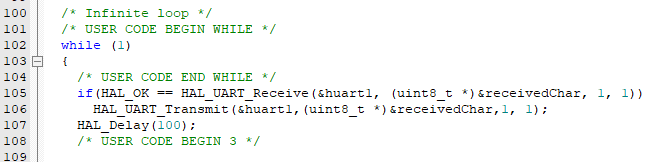
然后生成代码，转至mdk

Step 5: 使用HAL库通过串口输出。



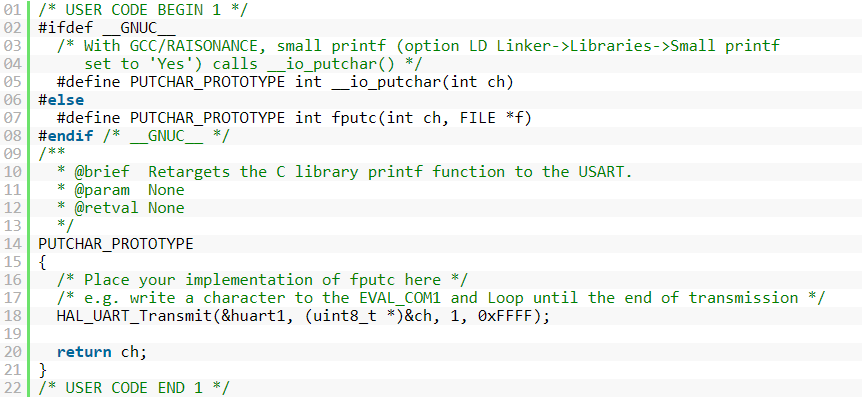
Step 6: 串口接收



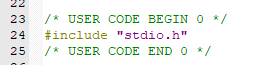


实验二：使用printf控制输出格式

在usart.c文件后面添加如下代码，代码中添加了#ifdef宏定义进行条件编译，如果使用GUNC编译，则PUTCHAR\_PROTOTYPE 定义为int \_\_io\_putchar(int ch)函数，否则定义为int fputc(int ch, FILE \*f)函数。



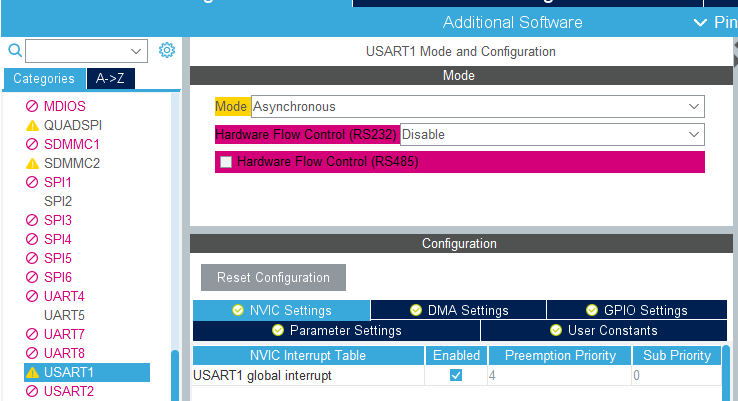
注意：mdk5以上版本，还需要增加stdio.h头文件



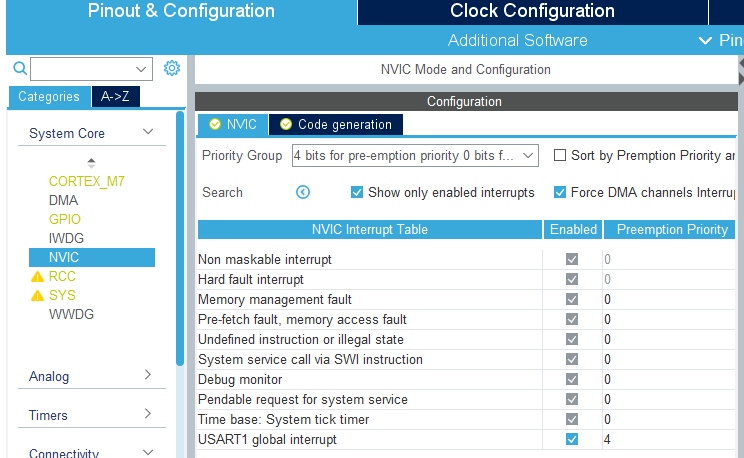
在main.c中可以使用printf格式化输出至串口，同样mdk5以上版本，main.c中也需要增加stdio.h头文件

**实验三：UART interrupt receive**

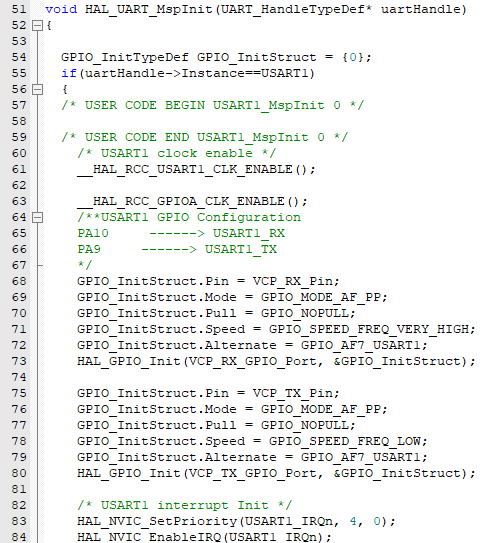
Step1: 在MX中的usart1使能uart的中断



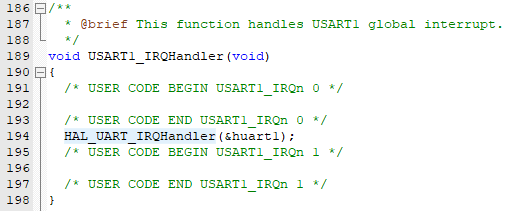
Step 2: 在MX中的NVIC中配置usart1中断优先级



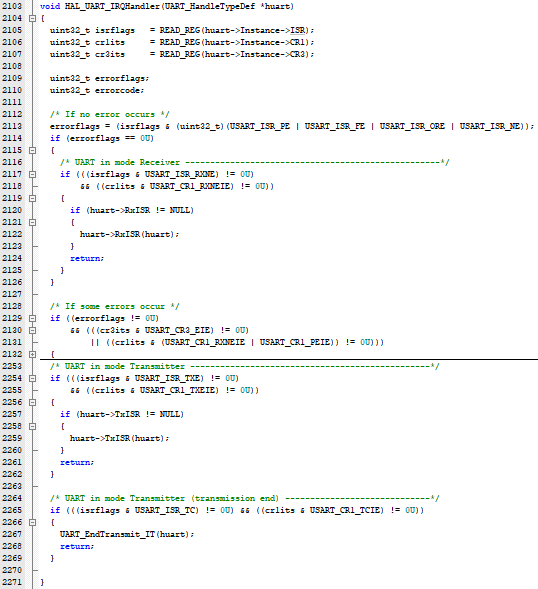
Step 3: MX生成代码后转至mdk5，在usart.c中可查看MX生成的代码，注意这里以为我们生成好了usart1的interrupt初始化代码



Step 4: 在stm32f7xx\_it.c中查看中断响应handles



Step 5: 跟踪HAL\_UART\_IRQHandler(&huart1);



共四段内容

1. 2113-2126： 处理接收相关
2. 2128-2132： 处理错误相关
3. 2153-2262： 处理发送相关
4. 2264-2269： 处理发送完成

接收相关分析如下：

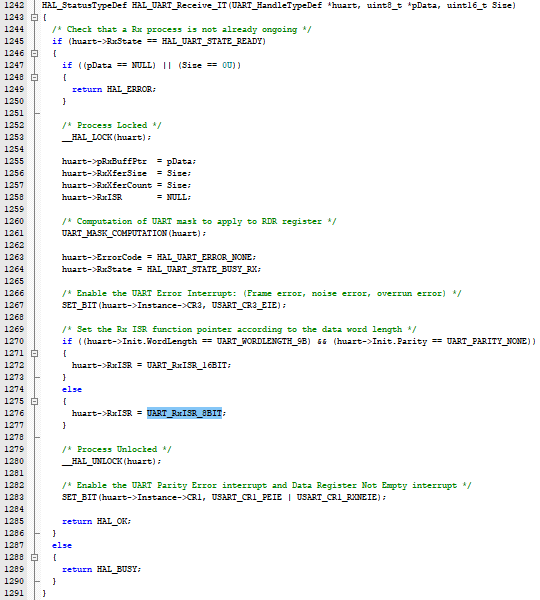
首先，无错误发生，确认USART\_ISR\_RXNE（接收非空，Rx No Empty）和USART\_CR1\_RXNEIE(接收非空中断为使能状态，Rx No Empty Interrupt Enabled)存在。

然后，确定是否已配置RxISR处理函数，如存在，将huart句柄传递给该处理函数。

那么，问题是RxISR处理函数必须先于此HAL\_UART\_IRQHandler前被赋值。这是由main.c中的两条函数调用实现：

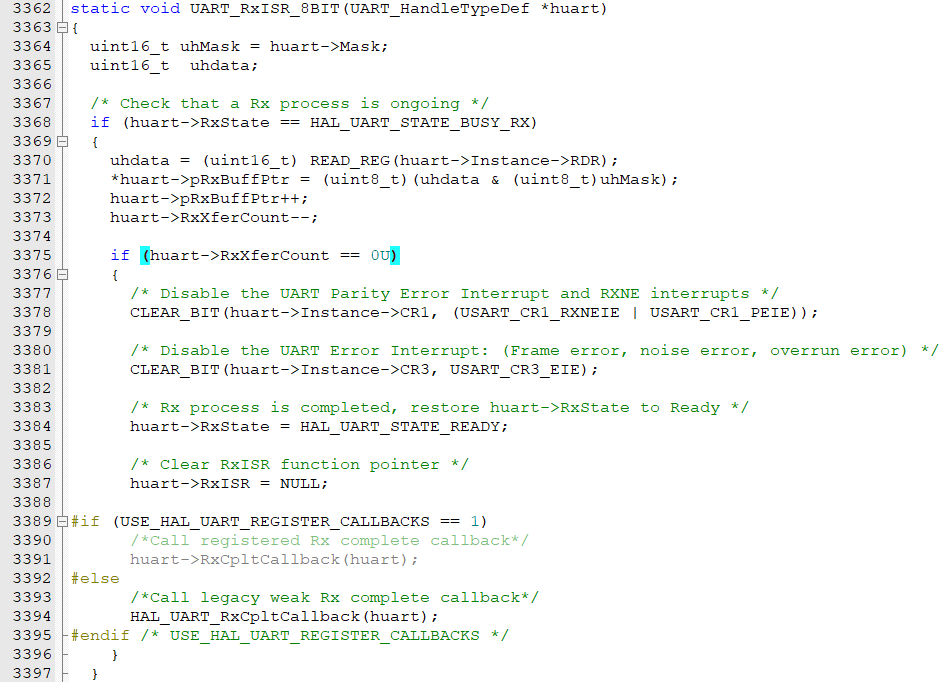
HAL\_UART\_Transmit\_IT和HAL\_UART\_Receive\_IT

Step 6： HAL\_UART\_Receive\_IT初始化huart->RxISR处理函数



如上图1272和1276行，根据uart的初始设置，huart->RxISR实际由UART\_RxISR\_8BIT或UART\_RxISR\_16BIT执行

Step 7：UART\_RxISR\_8BIT处理中断接收



其中3375行确定是否指定的接收长度达到，即如指定需要10个char, 如达到3387解除本次中断接收执行关系，最后调用HAL\_UART\_RxCpltCallback(huart)用于完成后需要的动作。

Step 8：发送中断

发送中断分两种情况，一是启动发送，二是发送完成，因此，需要两个处理函数，UART\_TxISR\_8BIT和独立的UART\_EndTransmit\_IT。前者，仅处理开始发送中断使能以及具体发送数据入TDR;后者，最终调用HAL\_UART\_TxCpltCallback(huart);

实验时，注意使用uart1的GPIO是否为PA9,10，如不是，先disable usart1, 在引脚规划图中选中PA9,设置为usart1 的tx， 选中PA10,设置为usart1 的rx；MX缺省的uart1的抢断中断优先级为0，可以通过NVIC进行调整，如调整为3

**实验四：UART DMA receive**