* + - 1. 将demo目录\sdk\plactform\driver下的uart2文件复制到客户工程对应的目录下。
      2. 将demo目录\sdk\plactform\reg下的reg\_uart2.h文件复制到客户工程对应的文件，如果客户工程对应的目录已经有reg\_uart2.h文件，则直接替换。
      3. 在BK3435\_reg.h添加 宏定义

#define REG\_APB3\_UART2\_CFG (\*((volatile unsigned long \*) 0x00806A00))

#define REG\_APB3\_UART2\_FIFO\_CFG (\*((volatile unsigned long \*) 0x00806A04))

#define REG\_APB3\_UART2\_FIFO\_STAT (\*((volatile unsigned long \*) 0x00806A08))

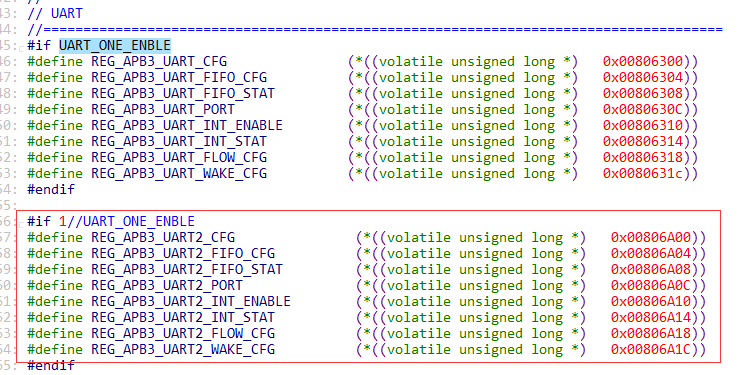
#define REG\_APB3\_UART2\_PORT (\*((volatile unsigned long \*) 0x00806A0C))

#define REG\_APB3\_UART2\_INT\_ENABLE (\*((volatile unsigned long \*) 0x00806A10))

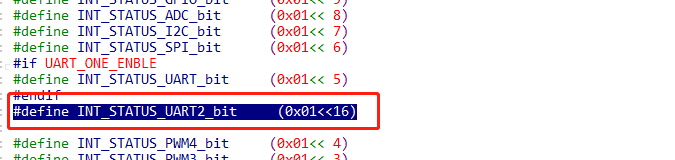
#define REG\_APB3\_UART2\_INT\_STAT (\*((volatile unsigned long \*) 0x00806A14))

#define REG\_APB3\_UART2\_FLOW\_CFG (\*((volatile unsigned long \*) 0x00806A18))

#define REG\_APB3\_UART2\_WAKE\_CFG (\*((volatile unsigned long \*) 0x00806A1C))



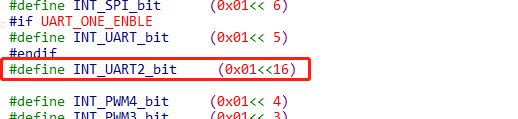
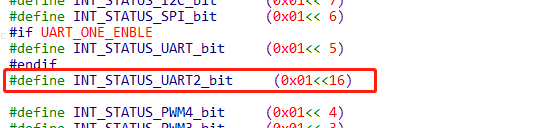
#define INT\_STATUS\_UART2\_bit (0x01<<16)



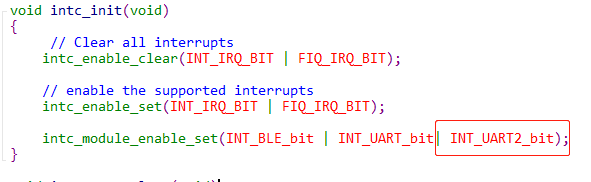
* + - 1. 在int.h文件添加宏定义

#define INT\_STATUS\_UART2\_bit (0x01<<16)

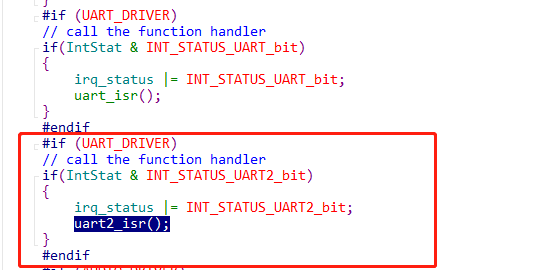
#define INT\_UART\_bit (0x01<<16)



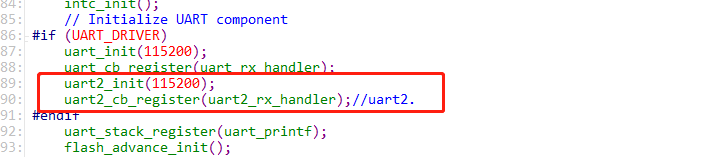
* + - 1. 在int.c文件修改void intc\_init(void)函数



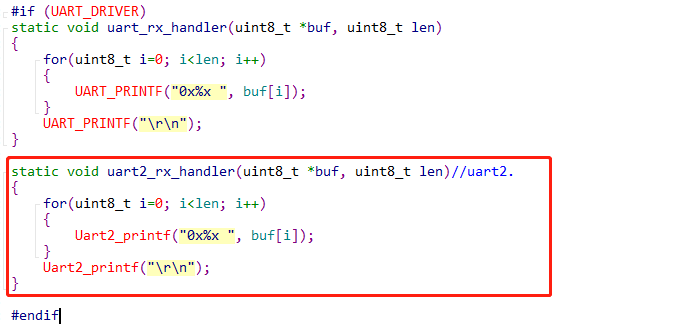
在oid IRQ\_Exception(void)函数调用中断函数uart2\_isr();

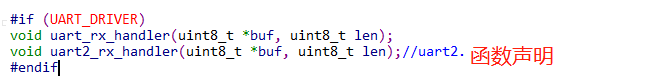


* + - 1. 在arch\_main.c文件rw\_main()函数里调用uart2\_init(115200);初始化函数和uart2\_cb\_register(uart2\_rx\_handler);函数

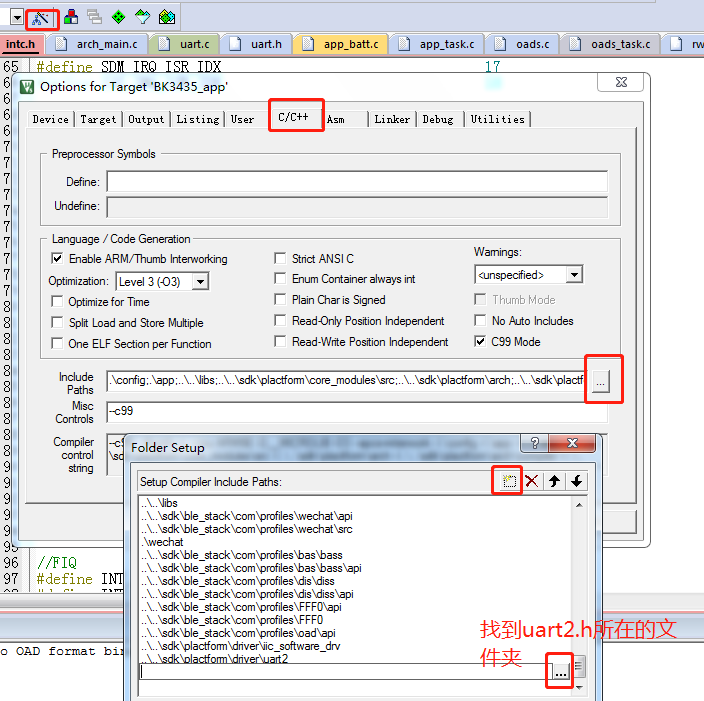


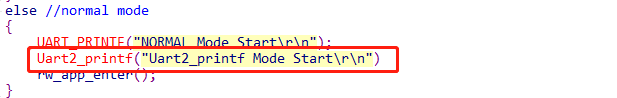
* + - 1. 增加static void uart2\_rx\_handler(uint8\_t \*buf, uint8\_t len)并且声明





* + - 1. 在相应的文件下调用#include "uart2.h"头文件声明。注意P17、P16不要做任何配置。
      2. 在keil工程里添加uart2.c和uart2.h文件。



* + - 1. 在arch\_main.c文件rw\_main()函数里调用Uart2\_printf("Uart2\_printf Mode Start\r\n");UART2打印函数。
      2. 
      3. 打开串口助手，串口助手发送数据后，芯片会通过UART2将数据原样返回。

