

****

软 件 学 院

《实用操作系统》实验报告

**题　　目 实验五 串口移植**

**姓　　名 陈澄**

**学　　号 32420212202930**

**班　　级 软工三班**

**实验时间 2023/11/08**

**2023 年 11 月 08 日**

# 实验目的

我们的目标是：让最小系统启动。那么对于串口，不需要考虑得很全面：

* 不需要初始化串口：u-boot已经初始化串口了
* 不需要动态配置串口：固定使用某个波特率等配置就可以(在u-boot里设置过了)

移植工作只需要实现这几点：

* 串口发送单个字符
* 注册串口接收中断函数：确定中断号、使能中断、在中断函数中读取数据

# 实验步骤

**一、修改Makefile**

1.进入/openharmony/vendor/democom/demochip/driver

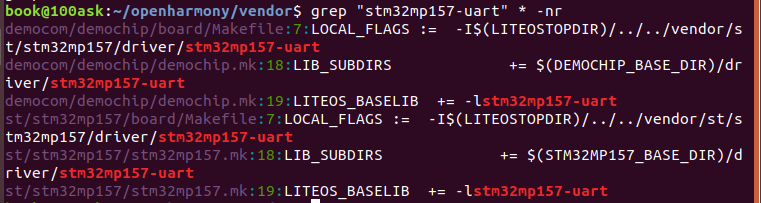
/uart(原来为stm32mp157\_uart)修改文件名如下，另一个uart\_stm32mp157.c同理



修改同目录下的makefile文件



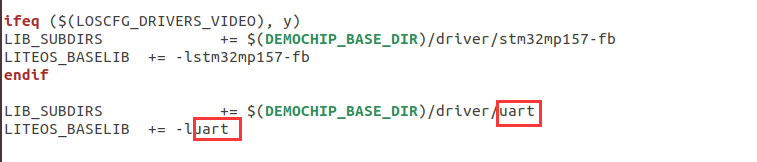
2.通过grep指令搜索“stm32mp157-uart”得到需要修改的文件



进入/openharmony/vendor/democom/demochip/board/Makefile修改



进入/openharmony/vendor/democom/demochip/demochip.mk修改

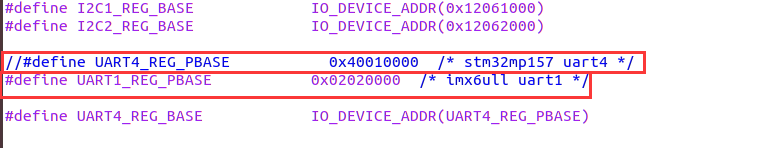


**二、修改代码**

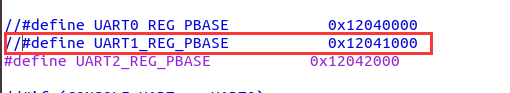
1.进入/openharmony/vendor/democom/demochip/board/include

/asm/platform.h

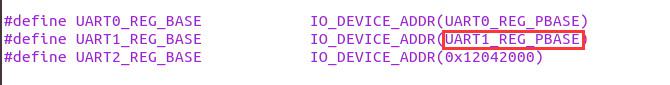
注释掉stm32mp157所使用的4号串口，手动添加一行imx6ull使用的1号串口



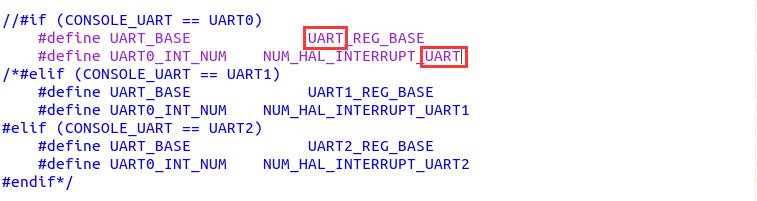
把后面这行注释掉



将物理地址映射为虚拟地址



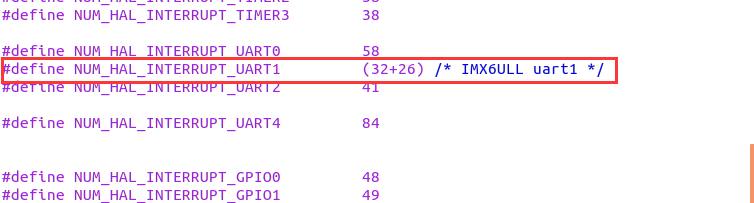
注释掉不需要的代码，将UART0改为UART



2.进入/openharmony/vendor/democom/demochip/board/include

/asm/hal\_platform\_ints.h

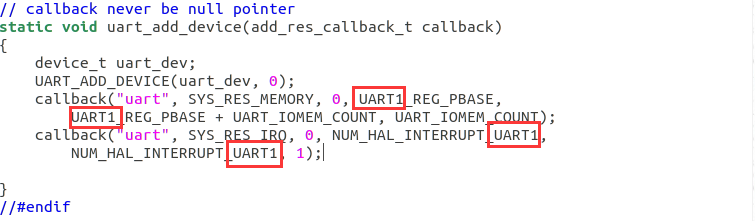
修改串口1的中断



3.进入/openharmony/vendor/democom/demochip/board

/bsd\_board.c

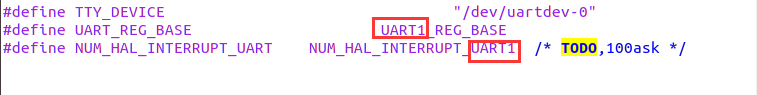
将uart\_add\_device方法内的串口参数全部改为1号串口



4.进入/openharmony/vendor/democom/demochip/board/include

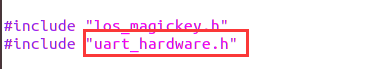
/uart.h

修改串口中断号的宏为串口1

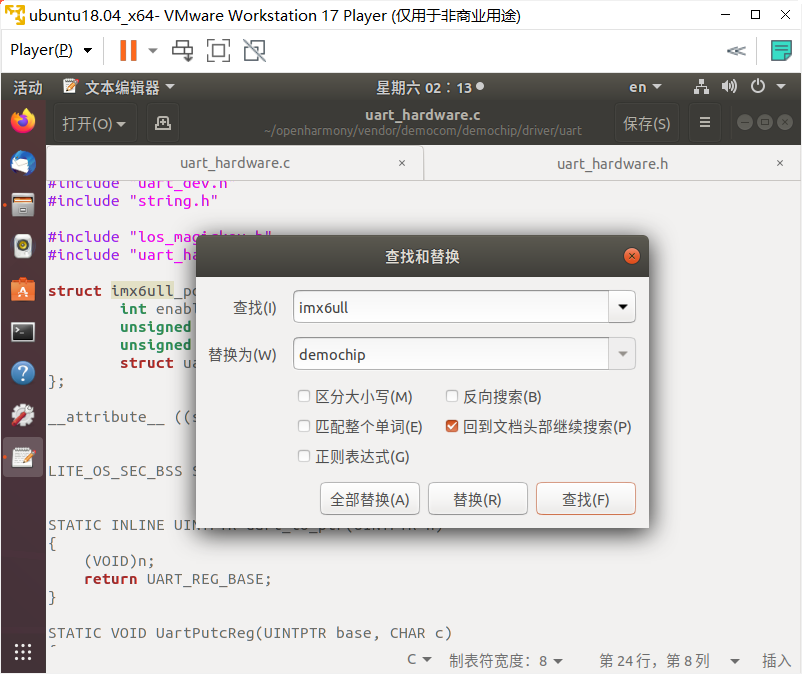


5.将/openharmony/vendor/nxp/imx6ull/driver/imx6ull-uart下的uart\_imx6ull.c和imx6ull\_uart.h的内容复制到/openharmony/vendor/democom/demochip/driver/uart下的uart\_hardware.c和uart\_hardware.h

在uart\_hardware.c中修改引用的头文件

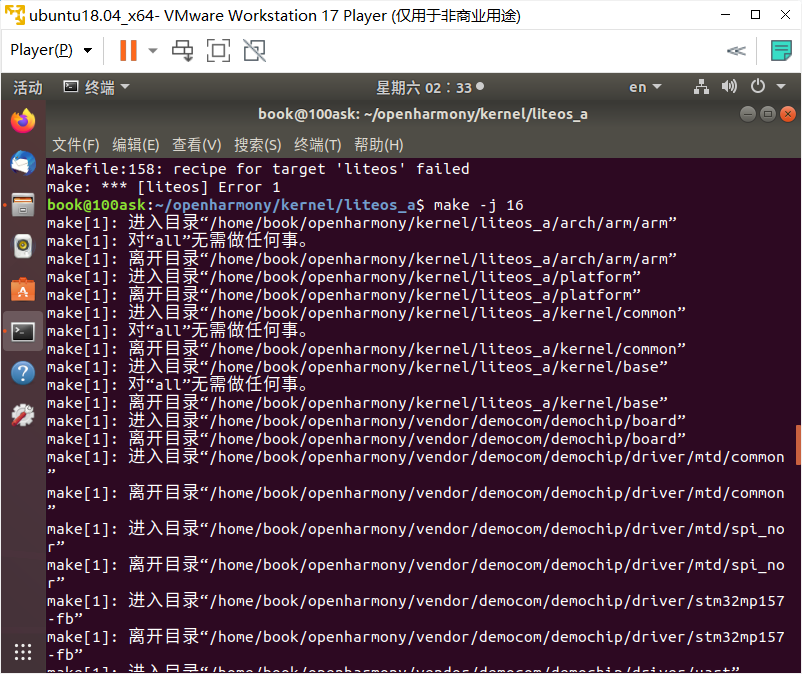


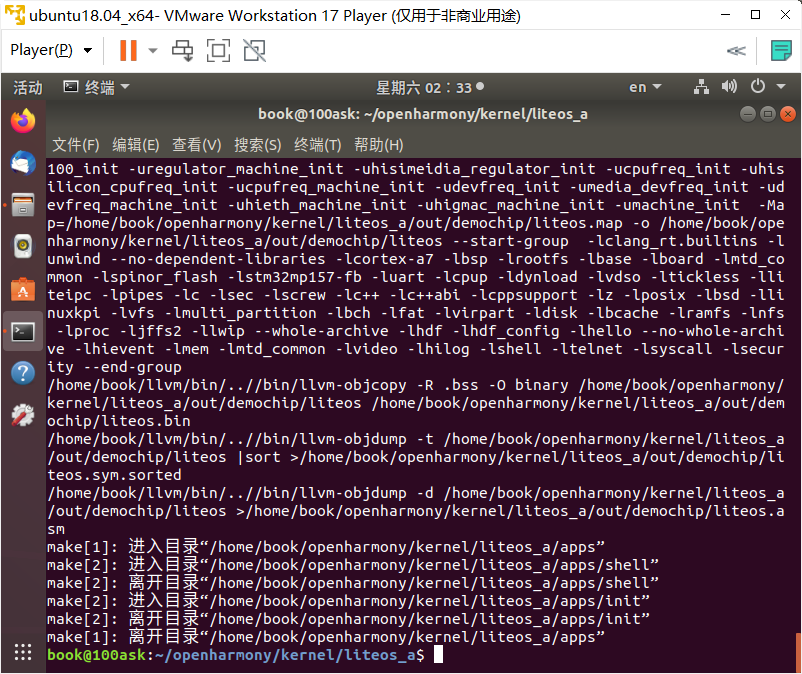
将uart\_hardware.c的imx6ull全部替换为demochip



**三、是否编译通过**

1.成功编译通过



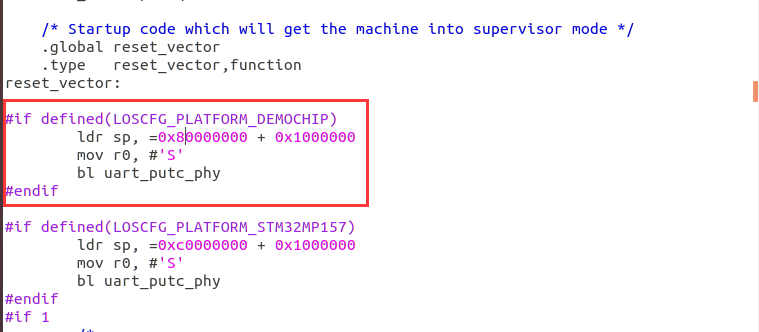


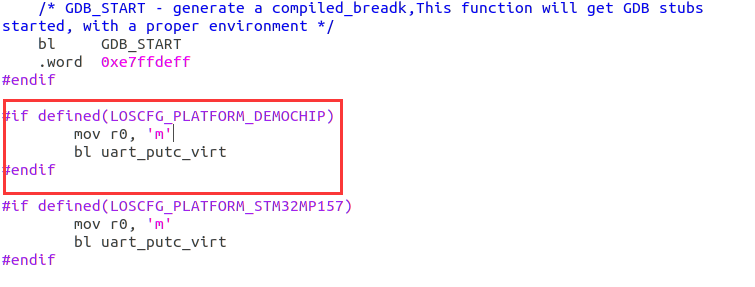
2.测试

(1)进入/openharmony/kernel/liteos\_a/arch/arm/arm/src/startup

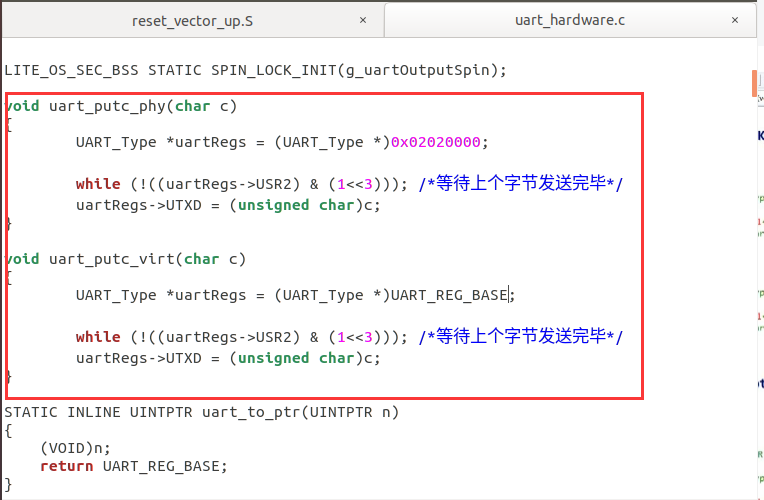
/reset\_vector\_up.S

仿照添加一段DEMOCHIP的代码，当启动的是DEMOCHIP时输出一个字符





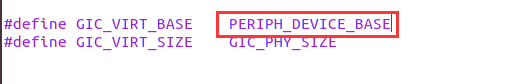
(2)回到uart\_hardware.c添加一个方法实现上面的uart\_putc\_phy



(3)进入/openharmony/kernel/liteos\_a/kernel/base/include

/los\_vm\_zone.h

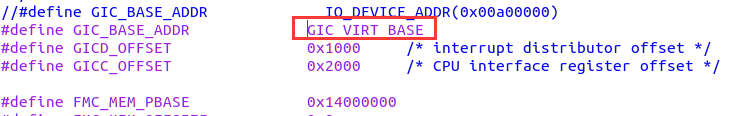
修改宏GIC\_VIRT\_BASE



(4)进入/openharmony/vendor/democom/demochip/board/include/asm

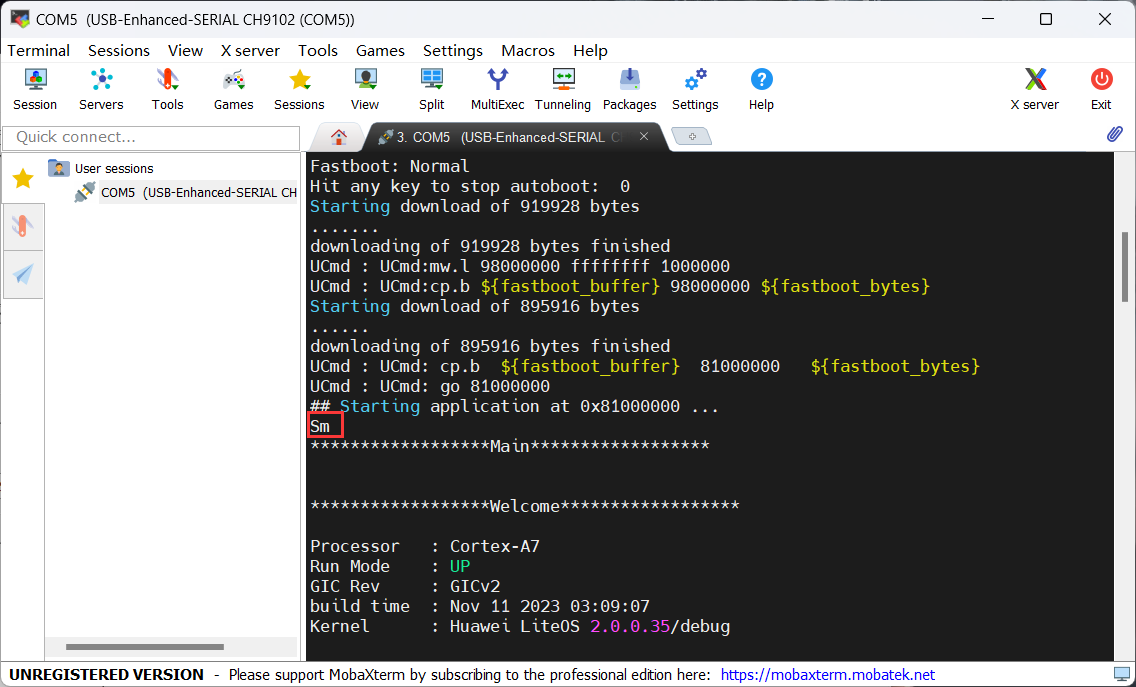
/platform.h

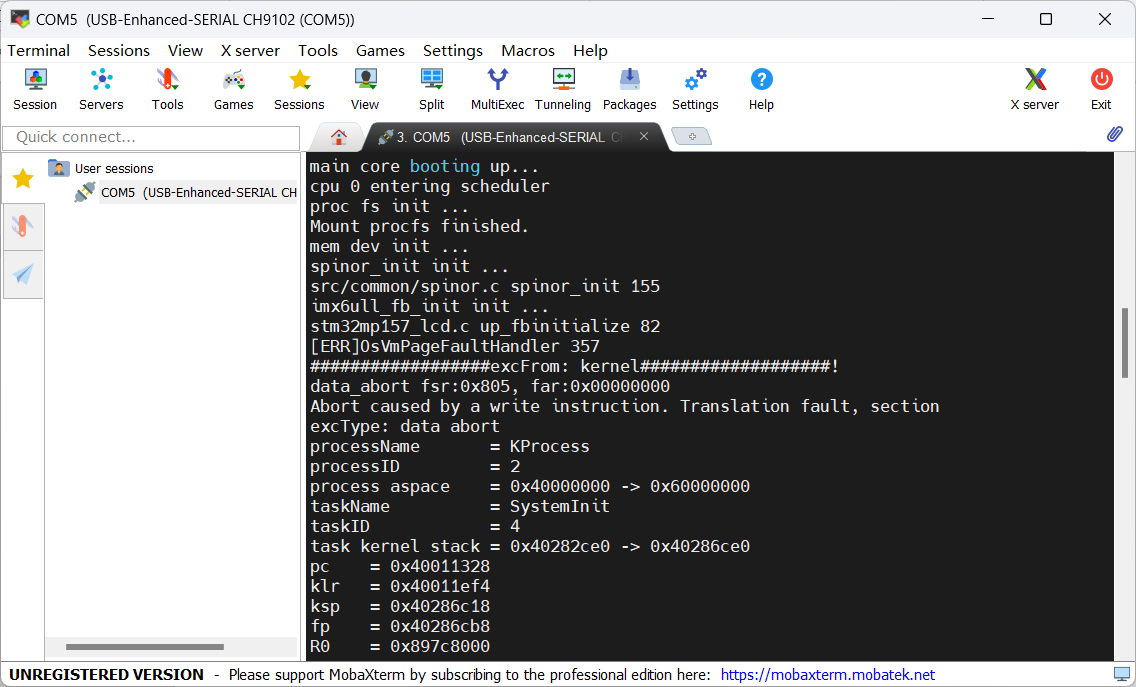
修改GIC\_BASE\_ADDR宏

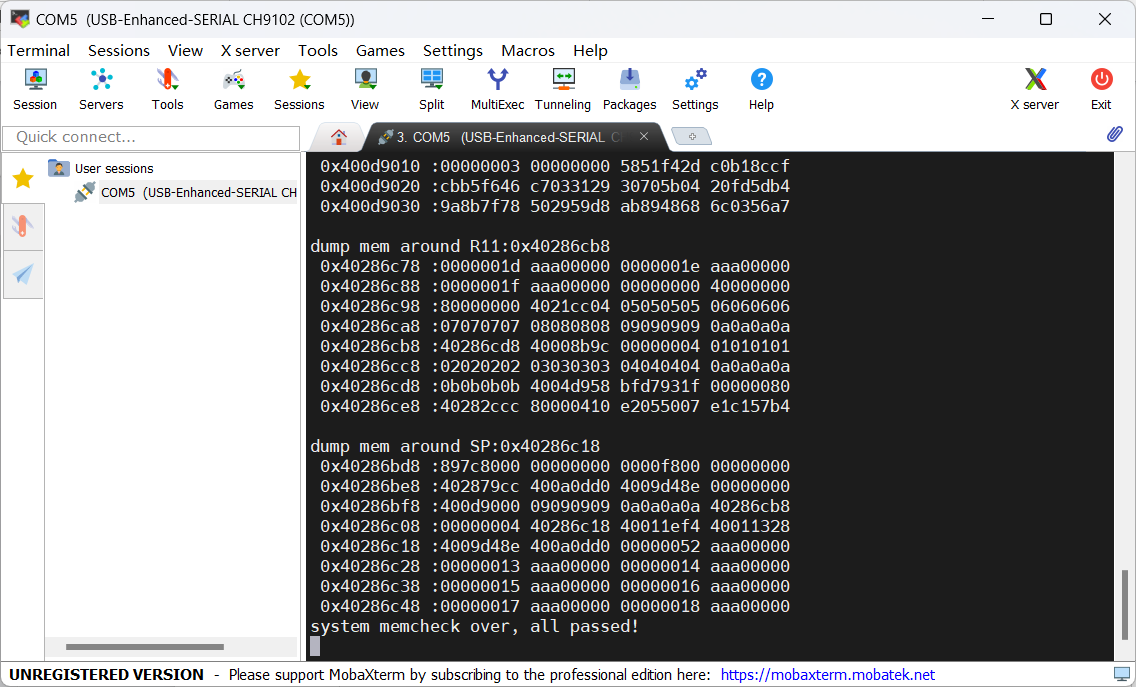


**四、运行结果**

成功显示之前调用uart\_putc\_phy显示的字符’S’以及uart\_putc\_virt显示的字符’m’，也成功进入主函数main，已达到实验要求。







# 实验遇到的问题及其解决方法

无

# 我的体会

通过这次实验，我了解了鸿蒙Liteos的内核是如何调用的控制台以及如何使用控制台输出单个字符。学会了如何编写方法修改控制台的输出字符，也学会了如何修改串口参数使最小的鸿蒙系统启动。