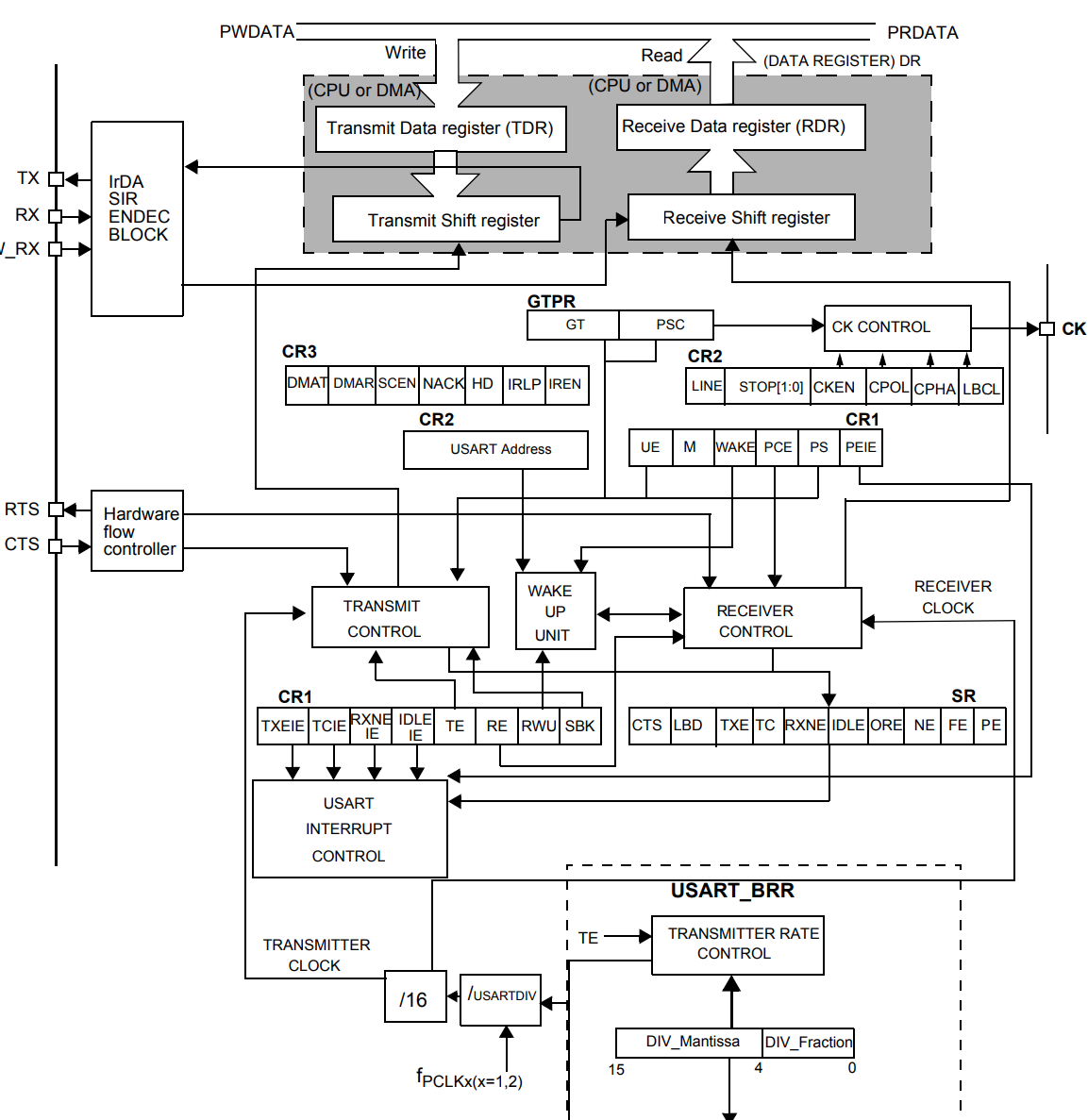
异步串行通信协议

* 起始位
* 数据位
* 奇偶校验位
* 停止位
* 波特率设置（速度）

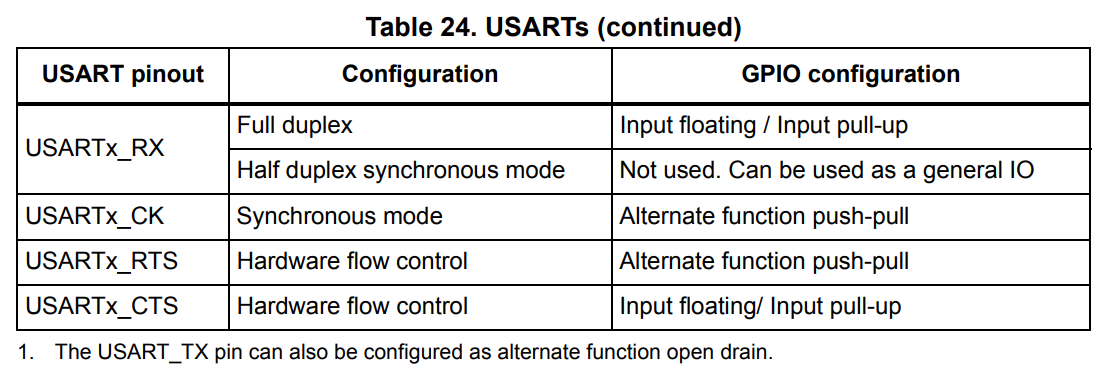
通信的甲乙双方只有在以上五种设置为相同的数据时，才可以实现数据通信

硬件实现原理

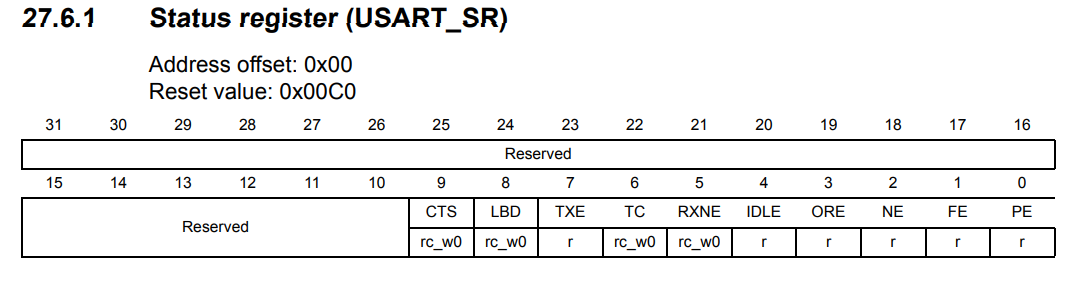


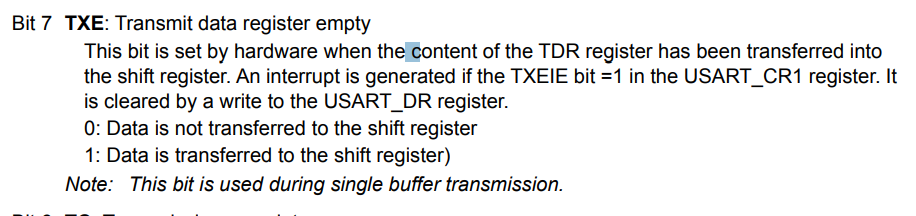
在Receive Shift register中，数据会先将它填满或者接收结束，才会将数据转移到Receive Data Register中，并接入总线，提供read的数据读取服务

时钟信号通过除以16之后传送到Receive Control & Transmit Control



寄存器和功能设置

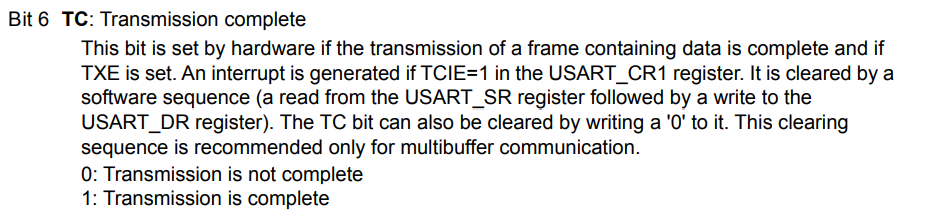




发送数据为空寄存器

1：有数据发送

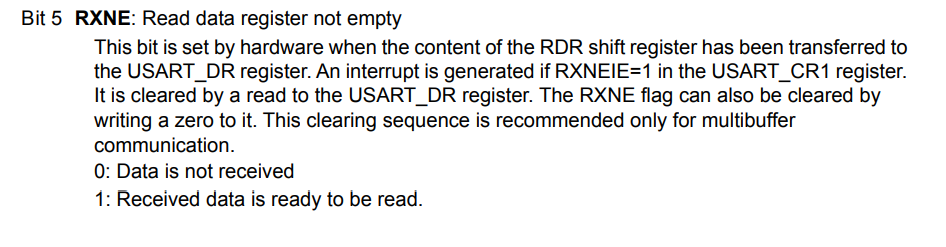
【在执行发送的时候由机器自动清除】



数据发送完成寄存器

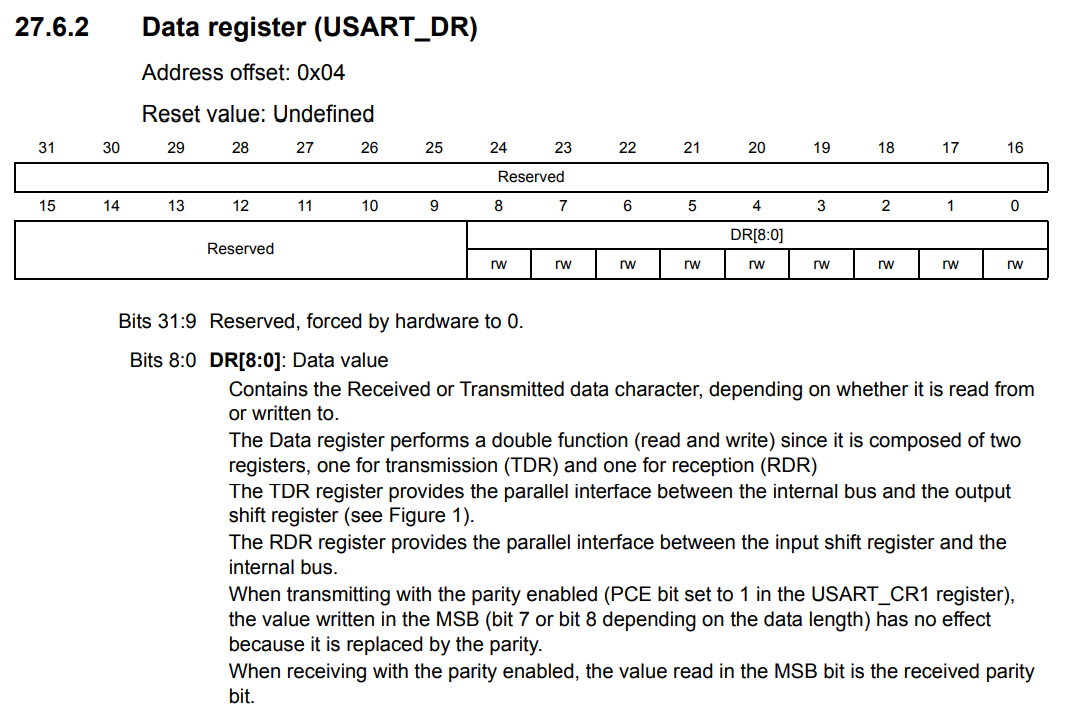
1：数据未全部发送完成

0：数据全部发送完成

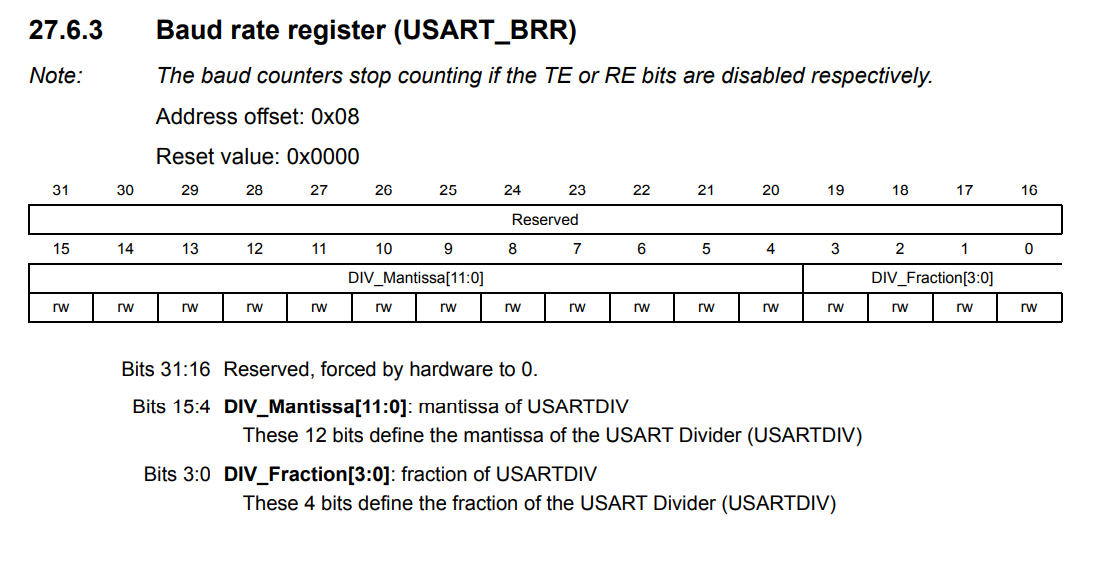


接收数据非空寄存器

1：数据已经全部接收，并转移到Receive Data Register中

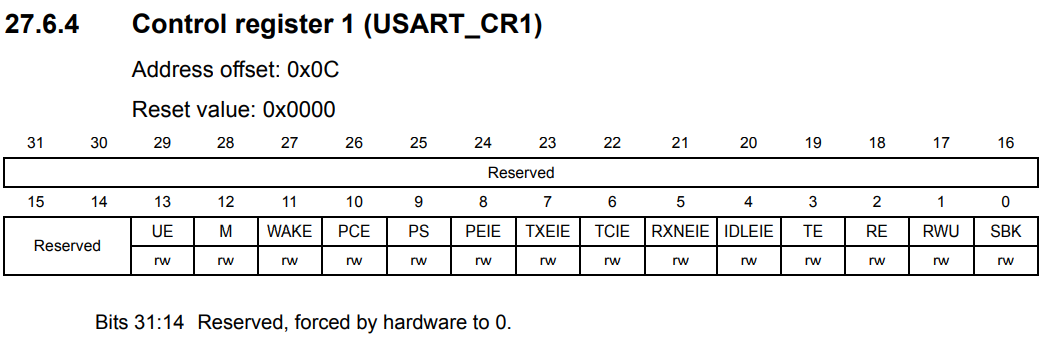


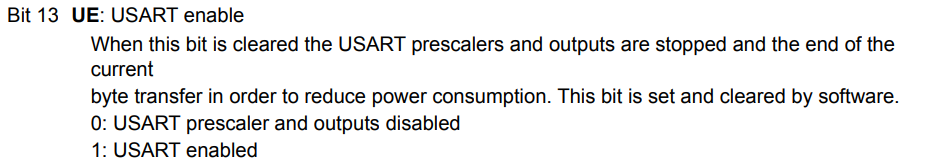
存放收发的数据，该寄存器可收可发，取决于发送源或者接收方。支持读写两种函数



波特率控制

***控制寄存器***



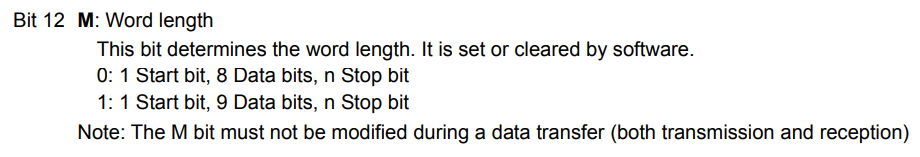


USART使能寄存器

0：停止功能

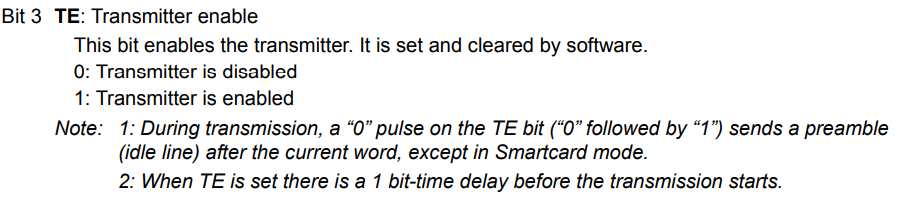
1：允许USART

【设置为1启动】



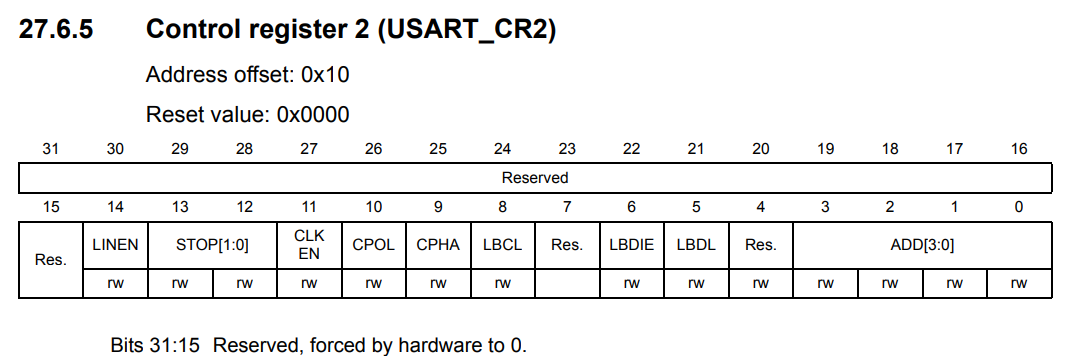
数据模式选择

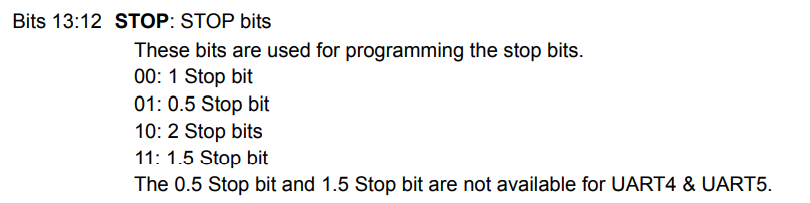
【1：选择1使用标准模式】



发送使能：

【设置位1允许发送】





选择停止位样式

发送过程阐释：

TXE寄存器标志有无数据被写入寄存器等待发送

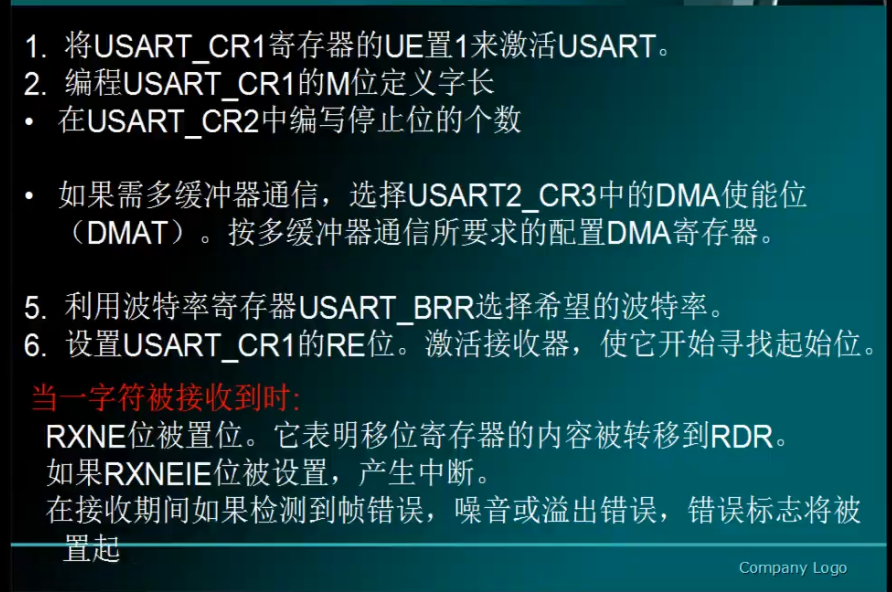
当数据（1byte）被写入寄存器准备发送的时候，TXE寄存器自动设置位0，表示端口发

发送占用。当发送完成，寄存器清零，TXE寄存器置1，程序将探测这一位，并执行后

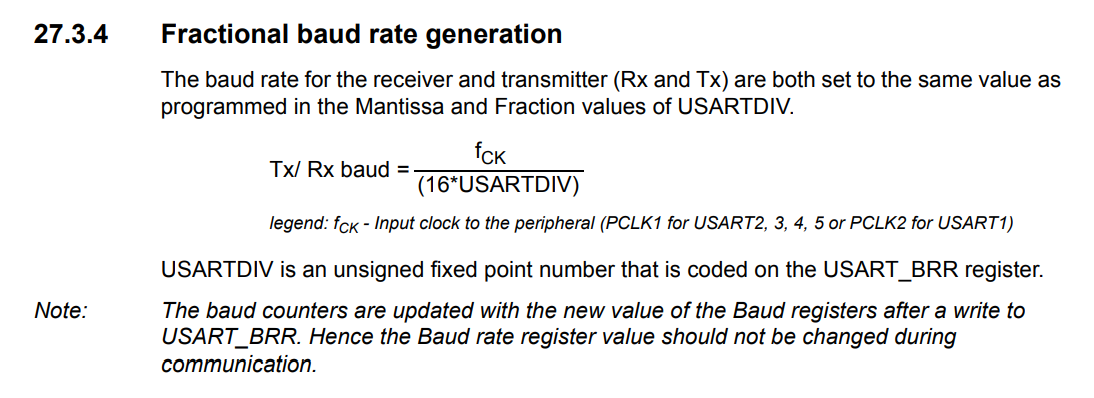
续发送



接收过程阐述：



波特率计算公式：



UART模块选择

选择USART1

设置初始化

USART1 -> CR1 |= (1<<13); // 13位置一，开启USART使能

USART1 -> CR1 &= ~(1<<12); // 12位置零，使用普通模式

USART1 -> CR2 &= ~(3<<12); //设置停止标志长度(1<<12)(1<<13)

USART1 -> CR1 |= (1<<3); // 允许发送

// 设置波特率9600

// 设置时钟

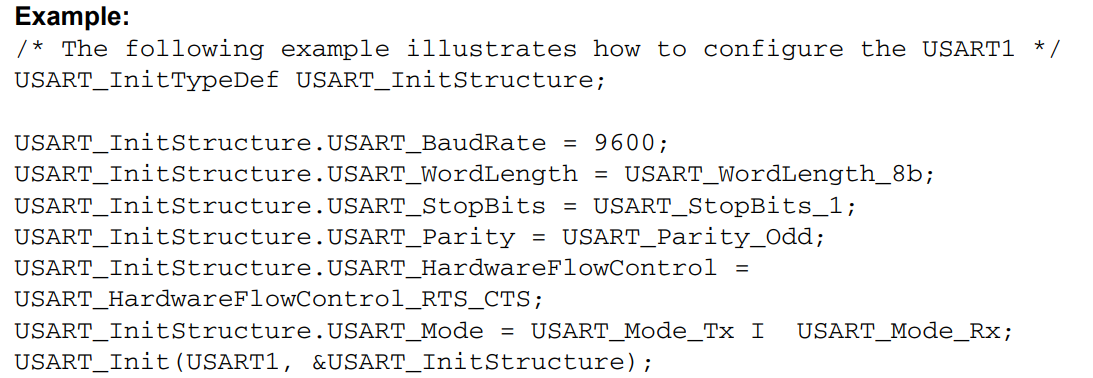
/\*发送数据\*/

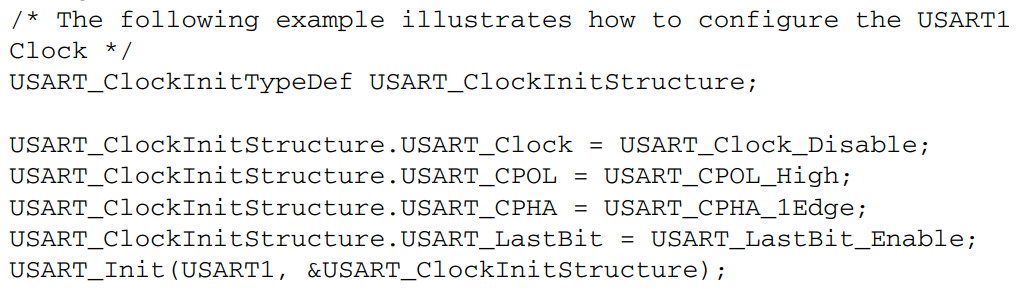
USART -> DR = ‘A’; //将数据发送到DR寄存器中

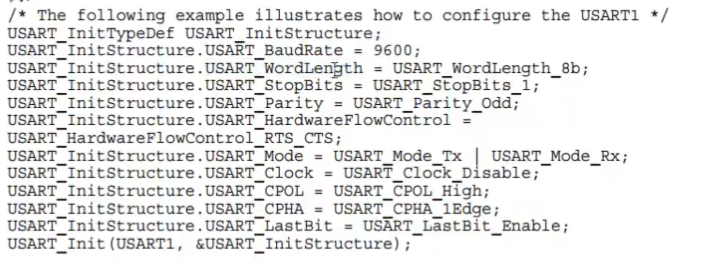


GPIO设置

库函数编程







（上面是旧版本的实现方法）

Printf的重定向输出

目标：将printf输出的对象从屏幕改为串口

函数定义位置：stdio.lib FILE文件流

通过修改该函数进行重定向c库printf

\_sys\_exit \_sys\_open int fputc(int ch, FILE \*f);

Int fputc(int ch, FILE \*f) {

// ch发送给USART

USAER\_Send(USART1, ch); // 固件库函数

// 等待发送完毕

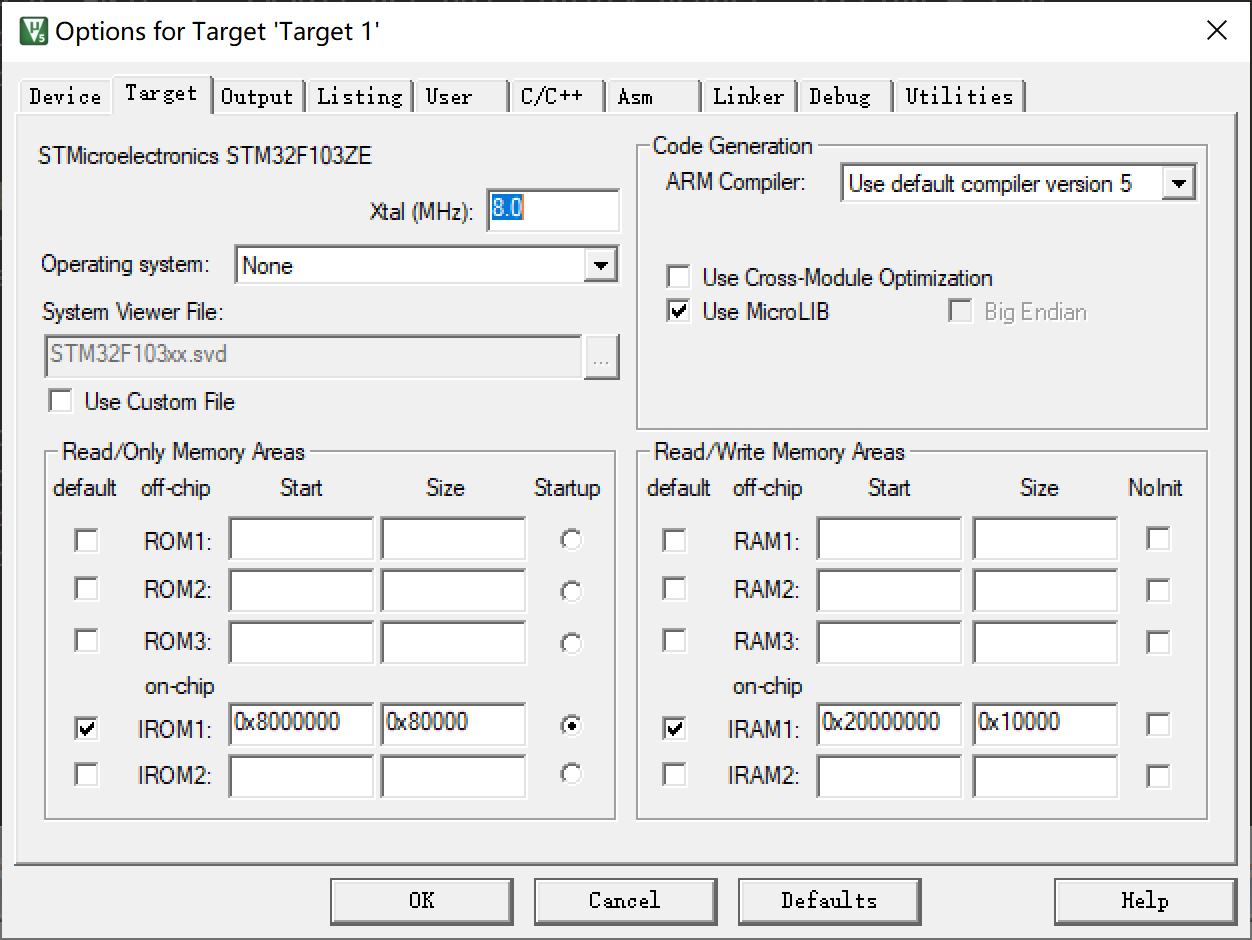
While (USAET\_GetFlagStatus(USART1, USART\_FLAG\_TC) == RESET);

// 返回ch

return(ch);

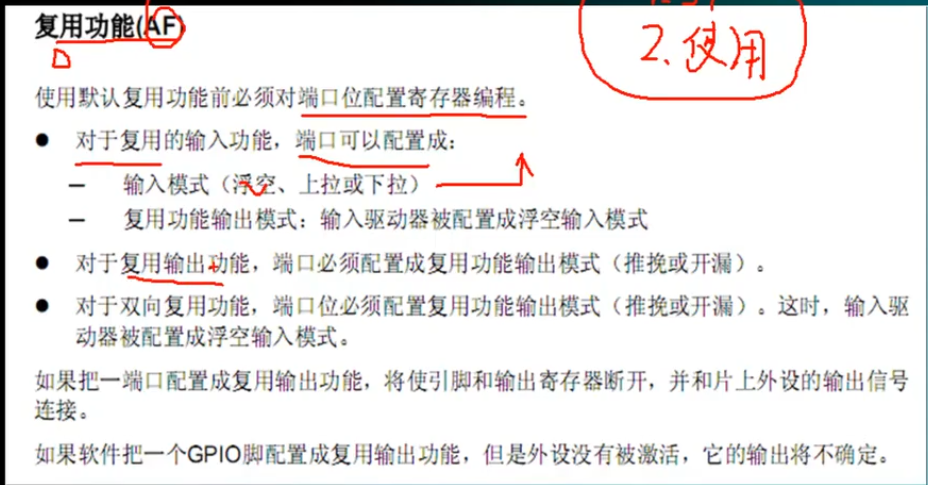
}

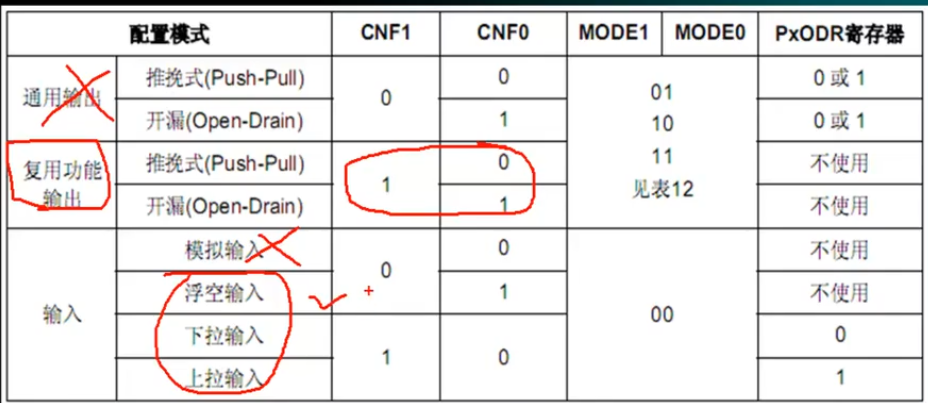
工程配置修改：



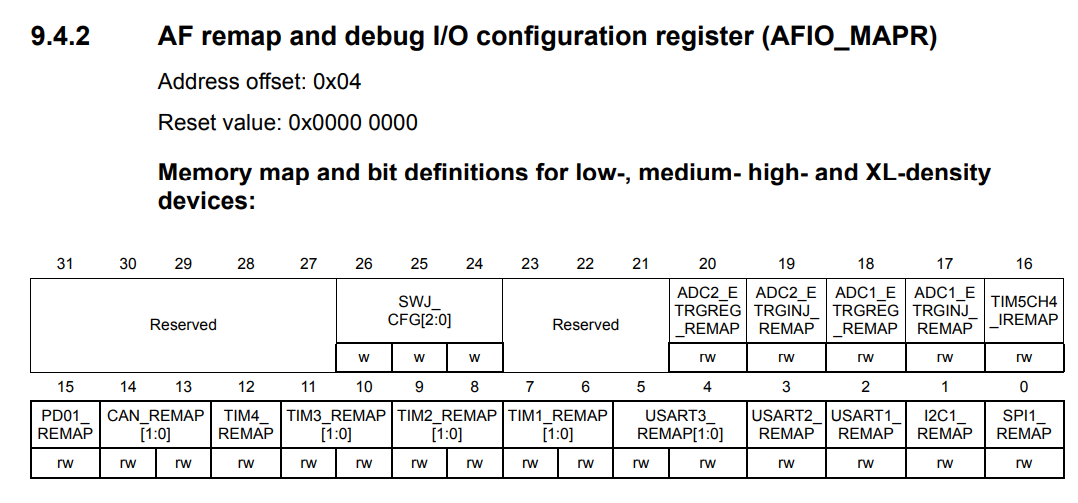


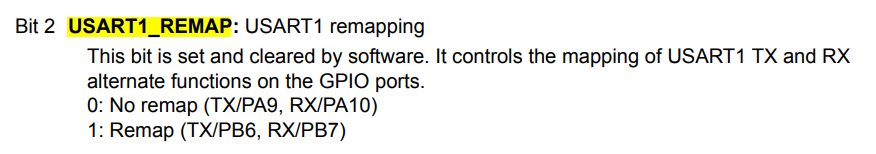
AF: 端口复用

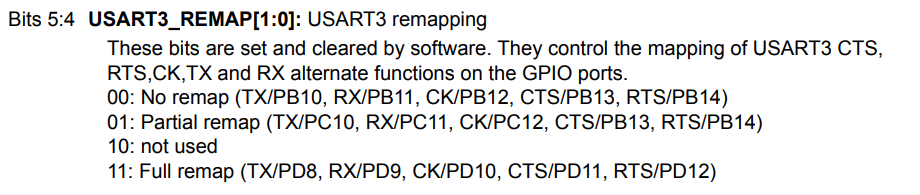




关于USART的函数复用





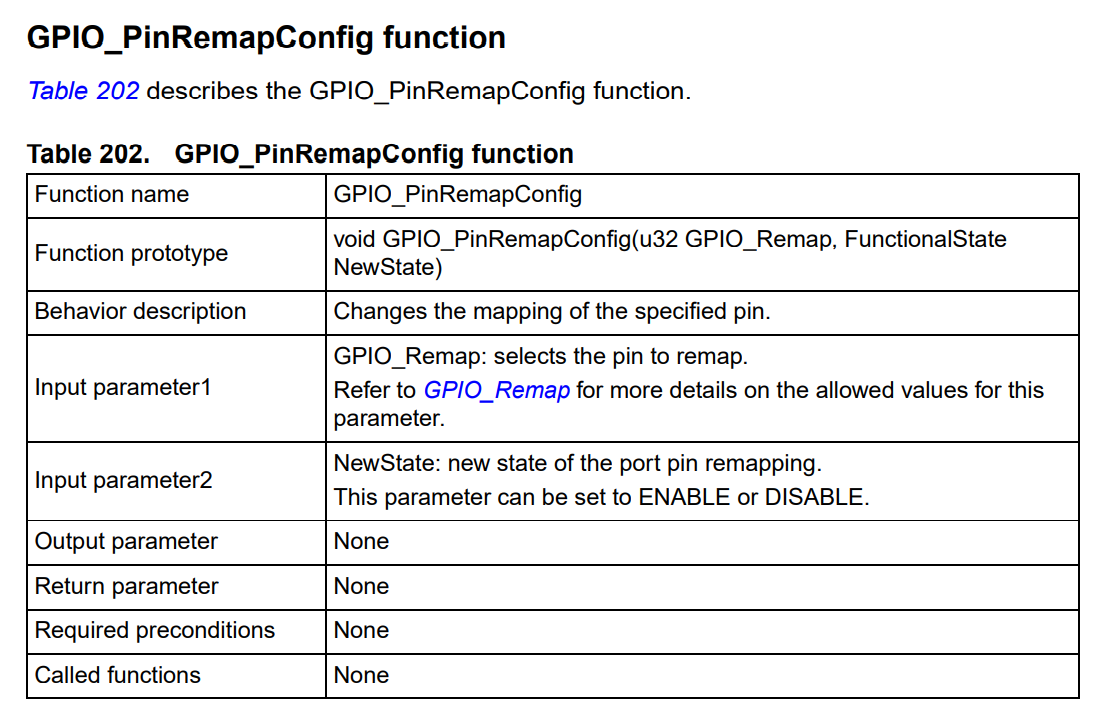


如何查找影响的部分映像和完全映像

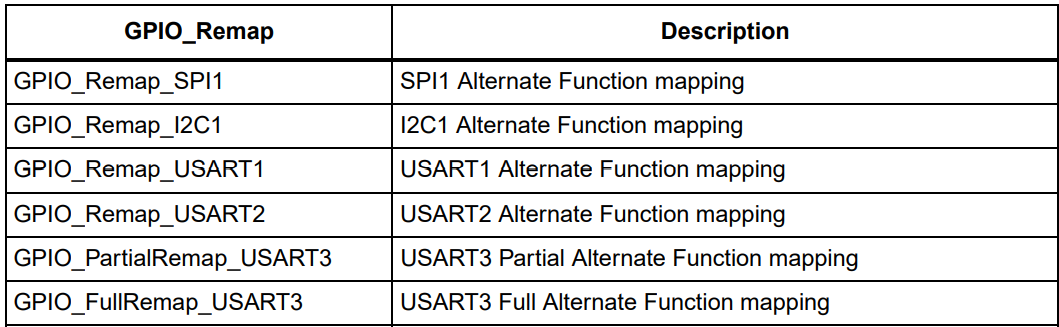
查找开发板原理图，找到指定USART对应的GPIO口，再返回手册做比较

例如：开饭板USART3使用PC10和PC11作为数据输入口，则对照上表可知，使用的方式时部分复用

端口重映像固件函数



查找参数input parameter 1

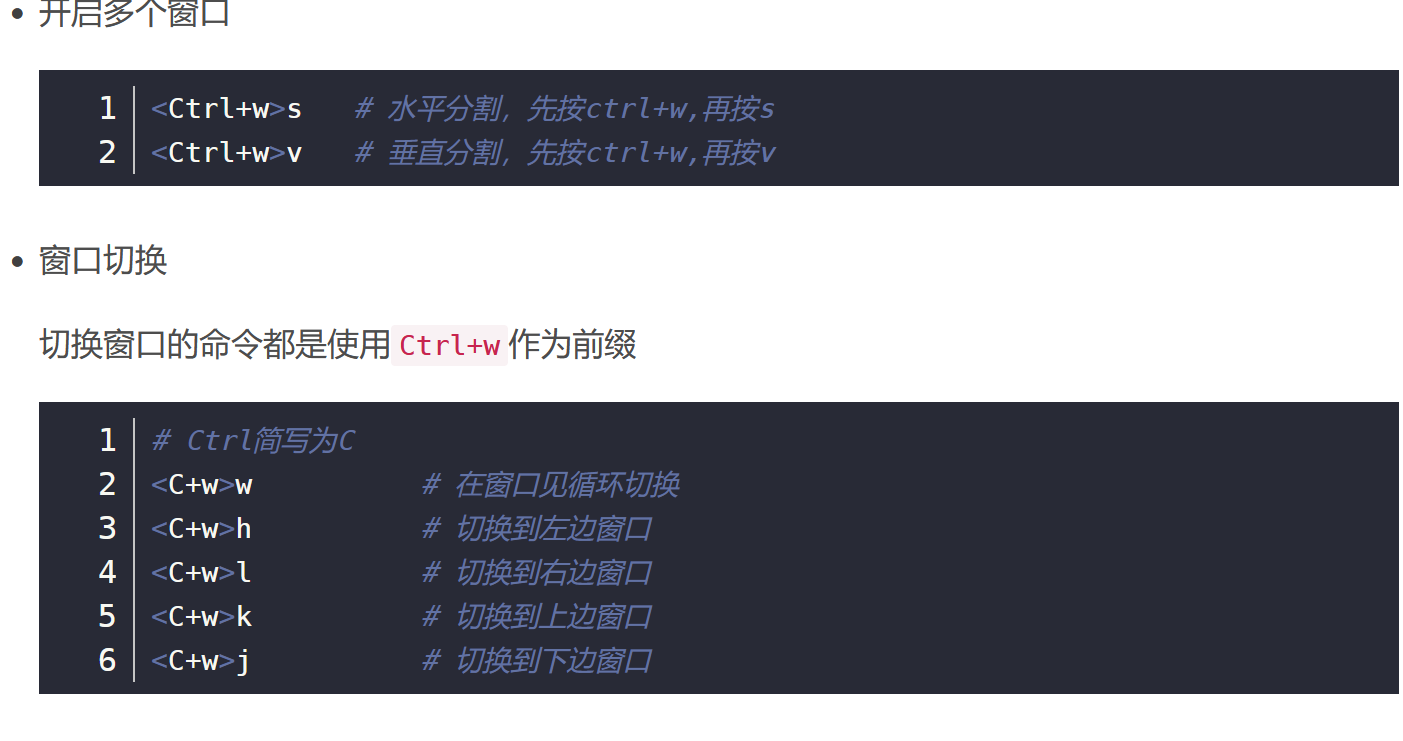


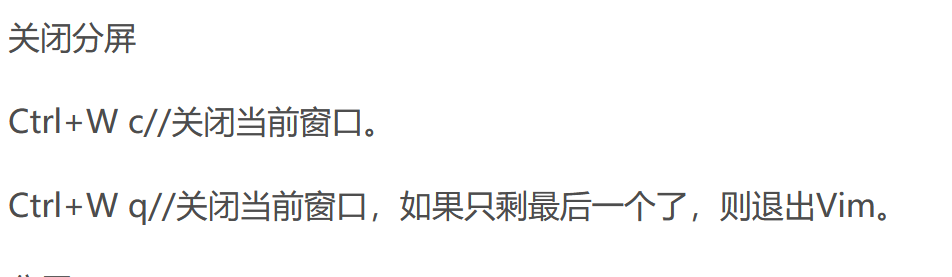
如果使用的时USART3的部分复用：

GPIO\_PinRemapConfig(GPIO\_PartialPemap\_USART3, ENABLE);// 使用USRT3的部分重映射

本开发板中USART1端口没有被复用

Vim





Shift + \*查找当前光标指定的单词出现位置