



第 8 章

STM32串口USART及 应用

I STM32端口重映射 ·

II 串口通信基础

III USART功能描述

IV USART应用实例



课程学习知识点和能力点

知识点

- 1) STM32端口到USART重映射;
- 2) 串口通信基础及USART功能描述;
- 3) USART应用实践。

能力点

了解STM32端口到USART重映射

掌握USART功能、寄存器设置、波特率计算

掌握ASRT应用要点和编程



8.1 STM32端口重映射

■ GPIO端口重映射

STM32处理器上有很多的I/O口和片上外设，为了节省引脚，这些片上外设和I/O口共用引脚，统称为I/O引脚复用。很多复用还可以通过端口重映射，从不同的引脚引出，即处理器I/O引脚通过一个复用器连接到板载外设/模块，该复用器一次仅允许一个外设的复用功能(AF)连接到I/O引脚，以确保共用同一个I/O引脚的外设之间不会发生冲突。

复用功能即“端口重映射”——将GPIO 端口通过软件配置，变成复用功能：

- 最多有16个复用功能（AF0—AF15）
- 其中AF0 即“系统复用功能”（复位后的默认功能）
- 各种外围设备分别映射到 AF1—AF14
- 每个I/O 都可以作为“事件输出”（EVENTOUT），映射到AF15。但PA15、PA14、PA13和PB4、PB3例外
- 每个I/O都可以作为外中断输入，只需配置成输入（或复位默认状态）即可。

8.1 STM32端口重映射

■ GPIO端口重映射

- 通过两个复用功能寄存器 (GPIOx_AFRL、GPIOx_AFRH) 实现复用功能的映射

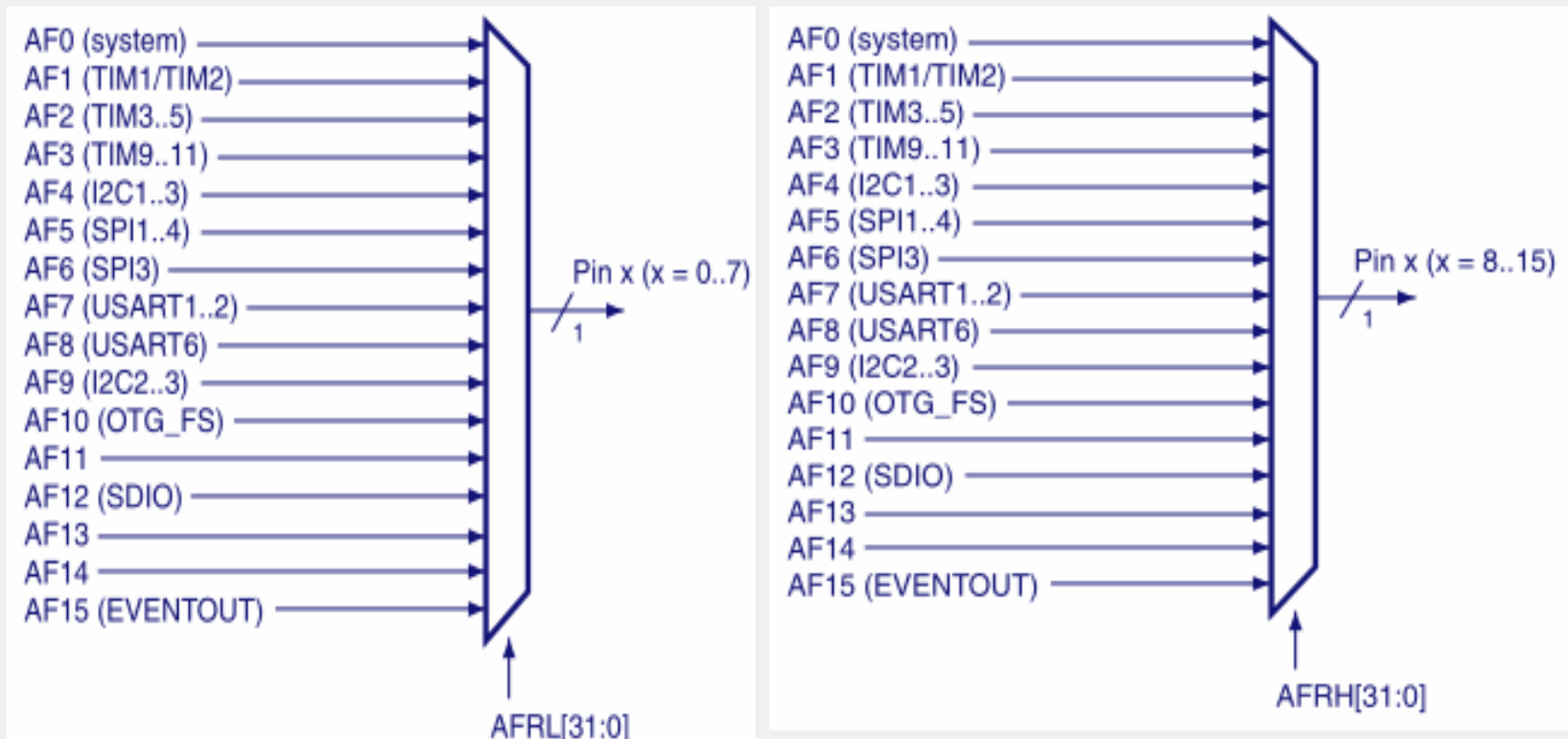


图8.1 STM32F407引脚复用



8.1 STM32端口重映射

■ GPIO端口重映射

STM32处理器的引脚复用图，如上图8.1所示。其中每个I/O引脚都有一个复用器，该复用器采用16路复用功能输入（AF0到AF15），可通过GPIOx_AFRL（针对引脚0到7）和GPIOx_AFRH（针对引脚8到15）寄存器对这些输入进行配置：完成复位后，所有I/O都会连接到系统的复用功能0（AF0）。

外设的复用功能映射到 AF1至AF13。

Cortex-M4 EVENTOUT映射到AF15。

RXD	PA9	101
TXD	PA10	102

图8.2 串口1原理图

以USART1为例，图8.2中使用的引脚为PA9、PA10即可以作为通用IO使用，又可以作为串口1（USART1）的发送数据（TXD）和接收数据（RXD）引脚。

第8章 STM32串口USART及应用

8.2 串行通信基础

– 单片机与外设通信的基本方式

• 1) 并行通信:

- 原理：数据的各个位同时传输
- 优点：效率高，速度快
- 缺点：占用引脚资源多

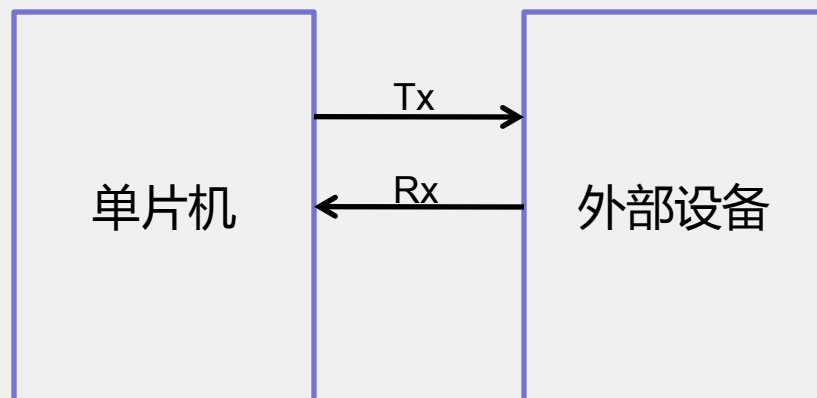
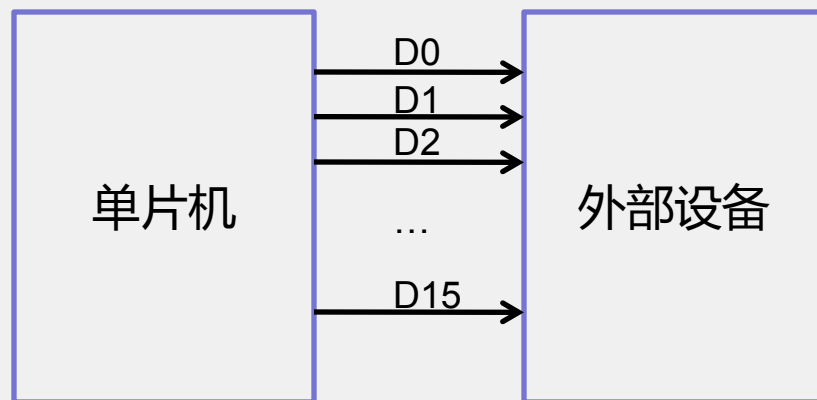
– 多用于芯片内部

• 2) 串行通信:

- 原理：数据一位一位按顺序传输
- 优点：占用引脚资源少
- 缺点：效率低，速度慢

– 多用于芯片间及数字设备之间

– 串行通信的种类的应用越来越广



• 8.2 串行通信基础

– 串行通信种类

- UART（通用异步收发器）传统串口
 - Universal Asynchronous Receiver and Transmitter
- USART（通用同步/异步收发器）现代串口
 - Universal Synchronous Asynchronous Receiver and Transmitter
- SPI（串行外设接口）
 - Serial Peripheral Interface
- I2C（IIC，集成电路接口总线）
 - Inter-Integrated Circuit
- I2S（IIS，集成电路音频总线）
 - Inter-IC Sound
- USB（通用串行总线）
 - Universal Serial Bus
- 其他……

• 8.2 串行通信基础

- 串行通信工作方式

• 1) 单工

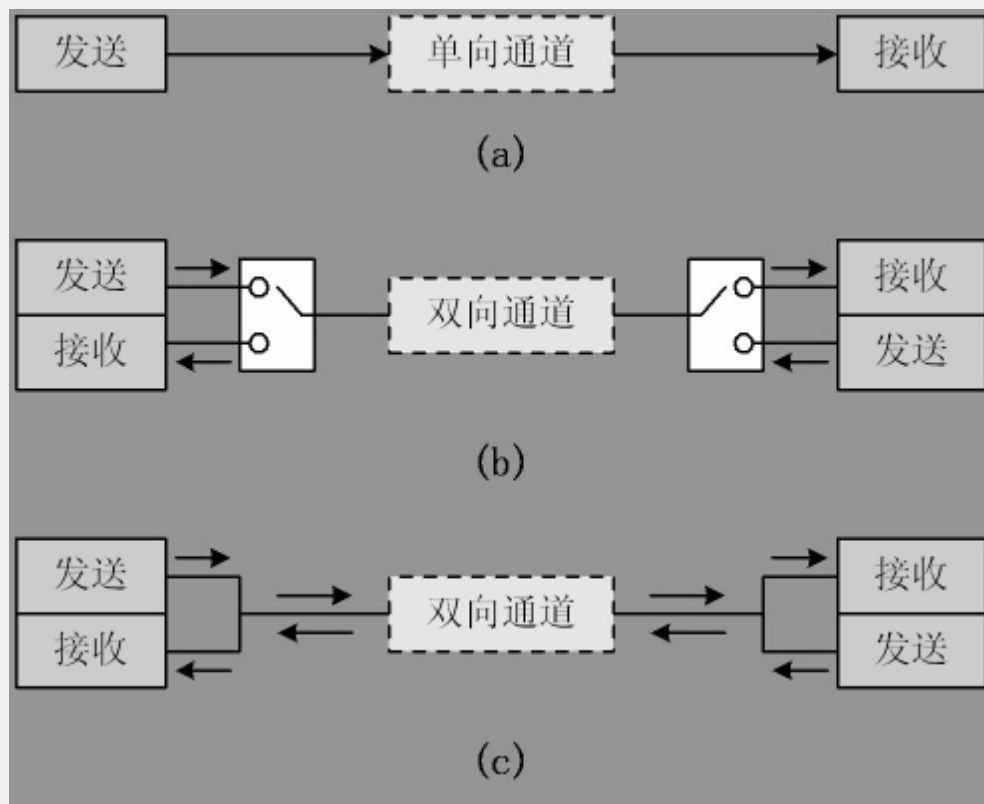
- 数据传输只能在一个方向上实现，即单向传输

• 2) 半双工

- 允许数据在两个方向上传输，但是不能同时实现，即在某一时刻，只允许一个方向传输
- 实际上是一种切换方向的单工通信

• 3) 全双工

- 允许数据同时在两个方向上传输，相当于两个单工通信方式的结合
- 要求双方都有独立的接收和发送能力



• 8.2 串行通信基础

- 异步与同步

• 异步通信

- 双方按约定速率（频率）工作，不能保证双方“步调一致”
- 节省引脚资源，简单易用
- 易产生时序误差，速度较低
- 传统串口（UART）、单总线等属于异步通信

• 同步通信

- 工作时，通过**时钟信号**保证双方步调一致（**同步**）
- 除了数据线，多了一根时钟线
- 时序精准，能够实现更高速度
- SPI、IIC等属于同步通信

• 8.2 串行通信基础

- 常见串行通信的工作方式

通信标准	引脚说明	通信方式	通信方向
UART	TXD : 数据发送端 RXD : 数据接收端 GND : 公共地	异步	全双工
1-wire (单总线)	DQ : 数据输入/输出端	异步	半双工
SPI	SCK : 同步时钟 MISO : 主机输入, 从机输出 MOSI : 主机输出, 从机输入	同步	全双工
I2C	SCL : 同步时钟 SDA : 数据输入/输出端	同步	半双工

• 8.2 串行通信基础

– STM32F401的串口通信

• STM32系列支持两种类型的串口：

- UART：通用异步收发器
- USART：通用同步/异步收发器

• STM32F4xx系列最多可支持8个串口，但STM32F401只实现了3个：

- USART1、USART2、USART6

• USART引脚

- RXD（或RX）：数据输入（接收）
- TXD（或TX）：数据输出（发送）
- CTS：清除发送 **硬件控制流**
- RTS：请求发送 **模式下使用**
- SCLK（或CLK/CK）：同步时钟信号

• USART三种工作模式：

	异步	控制流 (异步)	同步
RXD	√	√	√
TXD	√	√	√
CTS		√	
RTS		√	
SCLK			√
GND	√	√	√

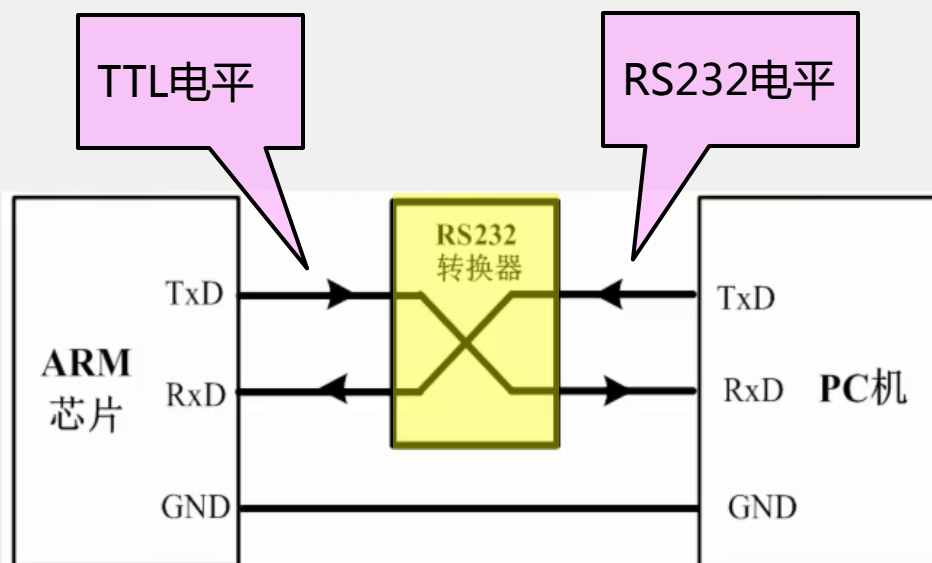
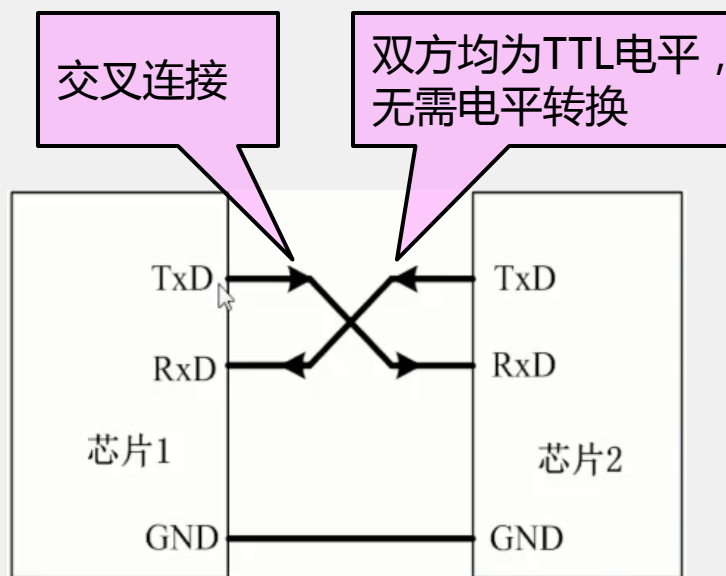
8.2 串行通信基础

通用异步收发器

UART引脚连接方法:

- 只需3个引脚：RXD、TXD、GND
- 信号线交叉连接

数字信号	TTL电平	RS232电平
0	0V (低)	+3~+15V
1	VDD(高)	-3~ -15V



• 8.2 串行通信基础

- 通用异步收发器

• RS232连接器

- PC机串口与外部连接时都是RS232电平，采用**DB9针式连接器**（俗称DB9公头）
- 另外，还有一种**DB9孔式连接器**（俗称DB9母头），也可用于RS232串口



- 针式接头和孔式连接器可以直接对接，也可以使用**直通线**连接。而针式与针式、孔式与孔式，则必须使用**交叉线**连接
- 建议开发板采用与PC相同的DB9针式接头，使板与板、板与PC连接方法相同，只需配制一根交叉线即可（两头均为孔式）

第8章 STM32串口USART及应用

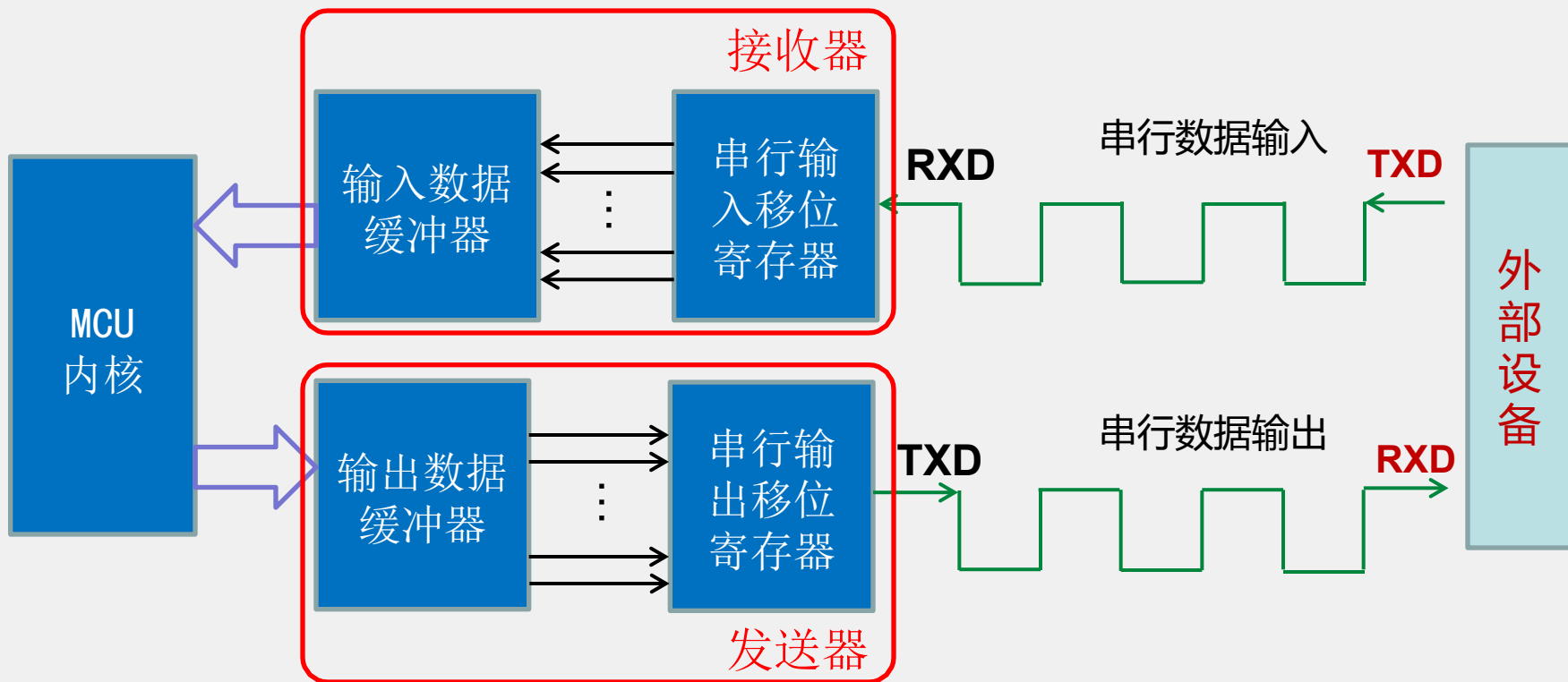
8.2 串行通信基础

- 通用异步收发器

• 串口通信过程

关键技术：移位寄存器

- ◆ 输入移位：串行 → 并行
- ◆ 输出移位：并行 → 串行



第8章 STM32串口USART及应用

• 8.2 串行通信基础

- 通用异步收发器

• 数据格式

- 要实现通信，收发双方不仅要约定速率，还要约定具体的数据格式，包括：

» 一个数据多少位——**字长**

数据的基本单位是字节（8bit），加上可能的校验位，所以串口通信的字长一般为8位或9位

» 谁打头谁断后——**发送顺序**

规定从低位开始发送，即LSB（bit0）打头，然后bit1，……最后是MSB（bit7 或 bit8）

» 如何表示开始——**起始位**

起始位为1位，其值默认为0（低电平）

» 如何表示结束——**停止位**

有0.5位、1位、1.5位、2位四种情况，适用于不同应用。停止位默认为1（高电平）。
①**1位：大多数情况下适用，也是默认值**
②2位：常规USART，单线模式及调制解调器
③0.5位和1.5位：多用于智能卡模式

» 是否需要核对——**校验位**

校验位为1位（可选），发送时排在数据后面。有奇校验、偶校验两种：
①奇校验：使数据+校验位中1的个数为奇数
②偶校验：使数据+校验位中1的个数为偶数

第8章 STM32串口USART及应用

8.2 串行通信基础

通用异步收发器

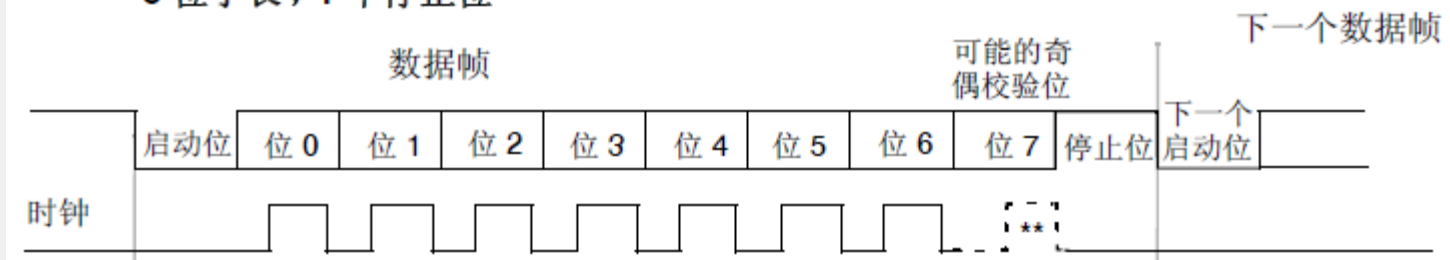
• 数据帧的概念：从起始位、数据到停止位的整个传输周期称为一“帧”

– 1 帧 = 起始位 + 数据位 (8或9位) + 停止位

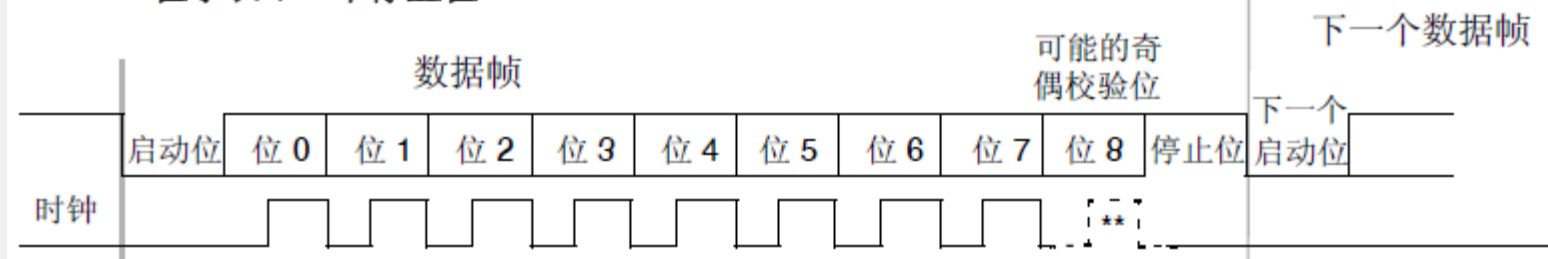
– USART传输时是以“帧”为单位的

每一位的发送接收由串口的内部时钟来控制

8 位字长，1 个停止位



9 位字长，1 个停止位



第8章 STM32串口USART及应用

• 8.2 串行通信基础

– 通用异步收发器

• 波特率 (Baud Rate)

- 波特率指单位时间传送的二进制位数（即1秒钟发送或接收多少位，单位是 bit/s或bps），也称“码速”
- 波特率是串口通信的重要指标，代表了数据传输的速度。接收器和发送器使用相同的波特率
- 由于历史的原因，形成了一些特定的波特率（bit/s）：
 - » 1200、2400、4800
 - » **9600**、19200、38400、57600
 - » **115200**、.....
- 随着技术的发展，波特率早已超出上述特定值，越来越高（如STM32F401最高可达10.5Mbit/s），但传统还是要遵守的
- 波特率由系统时钟分频得到，通过配置相关的寄存器来实现

第8章 STM32串口USART及应用

• 8.2 串行通信基础

– 通用异步收发器

• UART使用前需要定义的参数

- ①起始位（默认1位）
- ②数据位（8位或9位）
- ③奇偶校验位（第9位）
- ④停止位（0.5、1、1.5、2位）
- ⑤波特率
- ⑥硬件流控制（一般不用）

• 以上参数只是UART初始化的一部分，还有一些配置取决于具体芯片和应用，如：

- 时钟使能
- 引脚复用
- 工作模式
- 中断设置
-

第8章 STM32串口USART及应用

• 8.2 串行通信基础

– STM32F407串口通信

- STM32F401实现了3个通用同/异步串行口：
 - USART1、USART2、USART6
- 这些串口可以映射到不同的I/O端口上，功能也有所不同：
 - USART1：可以映射到PA、PB口，其中PB上只支持异步（3线）
 - USART2：可以映射到PA、PD口，均为全功能
 - USART6：只能映射到PC口，不支持硬件流控制

串口号	RXD	TXD	CTS	RTS	CLK
1	PA10	PA9	PA11	PA12	PA8
	PB7	PB6	--	--	--
2	PA3	PA2	PA0	PA1	PA4
	PD6	PD5	PD3	PD4	PD7
6	PC7	PC6	--	--	PC8

第8章 STM32串口USART及应用

8.2 串行通信基础

– STM32F407串口通信

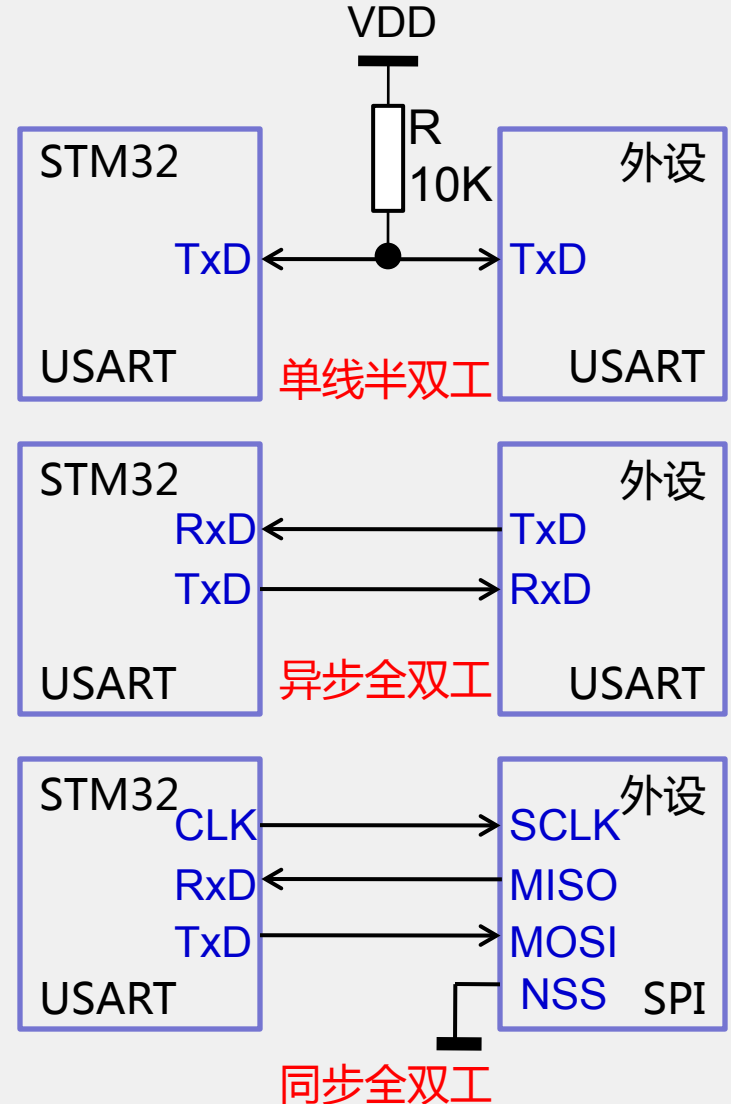
• 主要特性

– ①通信模式

- » 单线半双工（仅用TXD，需上拉）
- » 异步全双工
- » 同步全双工（仅可用于主模式）

– ②数据帧格式可编程

- » 数据字长：8位或9位
- » 停止位：1、0.5、2、1.5位
- » 奇偶校验控制：禁用，或使用（奇校验或偶校验）



第8章 STM32串口USART及应用

• 练习思考题

- 1、串行通信中，半双工和全双工有什么区别？
- 2、异步串行通信与同步通信有什么相同和不同之处？
- 3、说说UART数据帧的组成
- 4、STM32的USART有几种通信模式？

第8章 STM32串口USART及应用

• 8.2 串行通信基础

- STM32F407串口通信

• 主要特性

- ③容差性

» 可配置16倍过采样或8倍过采样，实现速度容差与时钟容差的灵活配置

- ④小数波特率发生器

» 通常分频系数都是整数，STM32支持小数分频，可以提供更精确的波特率

- ⑤三种传输检测标志

» 接收缓冲器已满

» 发送缓冲器为空

» 传输结束标志

- ⑥四个错误检测标志

» 溢出标志

» 噪声检测标志

» 帧错误

» 奇偶校验错误

第8章 STM32串口USART及应用

• 8.2 串行通信基础

– STM32F407串口通信

• 主要特性

– ⑦十个具有标志位的中断源

- » CTS变化
- » LIN断开检测
- » 发送数据寄存器为空
- » 发送完成
- » 接收数据寄存器已满
- » 检测到线路空闲
- » 溢出错误
- » 帧错误
- » 噪声错误
- » 奇偶校验错误

第8章 STM32串口USART及应用

8.2 串行通信基础

– STM32F407串口通信

• 主要特性

– ⑧高级应用特性

- » LIN：局域互联网络（是CAN的一种补充，一主多从）
- » 智能卡（ISO 7816），支持智能卡仿真功能
- » IrDA编解码器：串行红外线协议（SIR ENDEC）
- » 硬件数据流控制，用于调制解调器操作

串口号	标准特性	调制解调器 (RTS/CTS)	LIN	SPI 主机	irDA	智能卡 (ISO 7816)	最大波特率 Mbit/s (16倍过采样)	最大波特率 Mbit/s (8倍过采样)	APB 映射
1	✓	✓	✓	✓	✓	✓	5.25	10.5	APB2 (max. 84 MHz)
2	✓	✓	✓	✓	✓	✓	2.62	5.25	APB1 (max. 42 MHz)
6	✓	N.A	✓	✓	✓	✓	5.25	10.5	APB2 (max. 84 MHz)

第8章 STM32串口USART及应用

- 8.2 串行通信基础

- STM32F407串口通信

- 主要特性

- ⑨其它特性

- » DMA支持

- » 多机通信

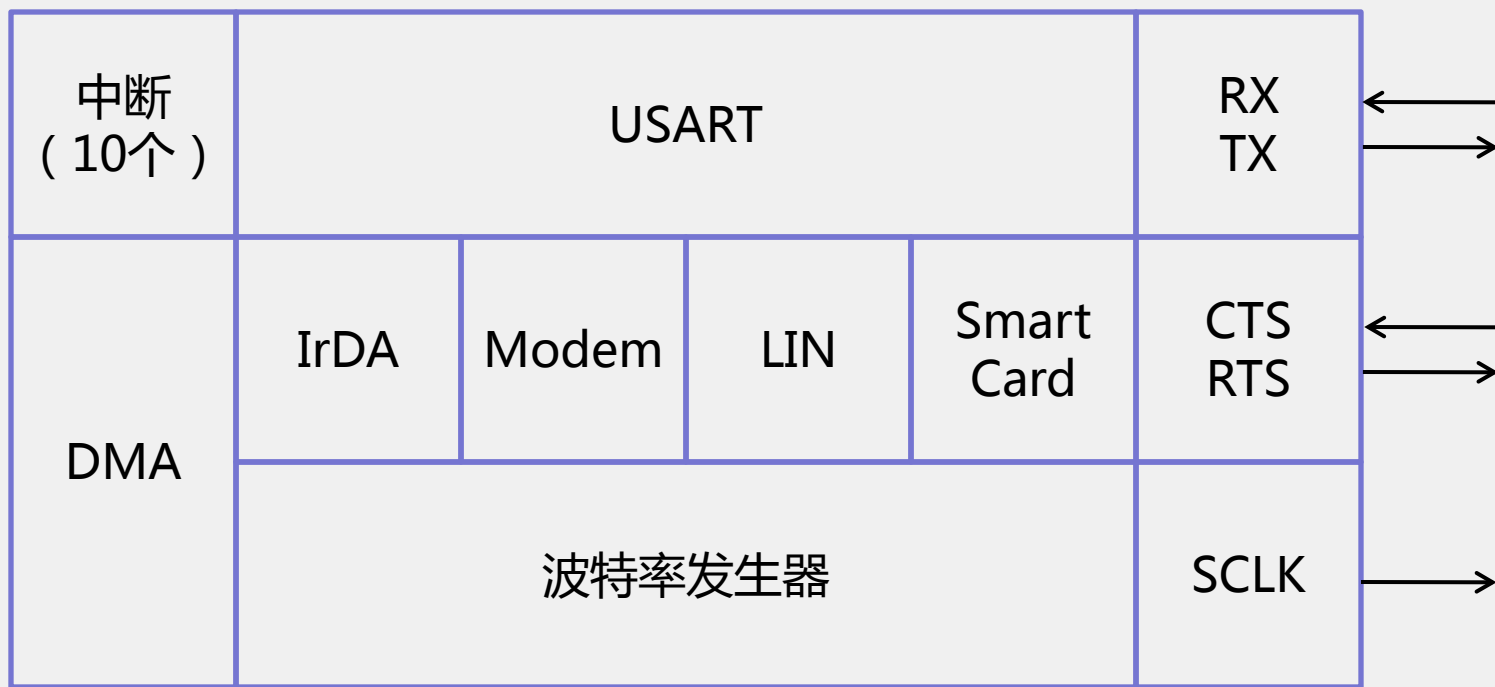
- » 唤醒功能，包括从静默模式唤醒和接收器唤醒

第8章 STM32串口USART及应用

- 8.3 USART功能描述

- USART结构

- 功能模块

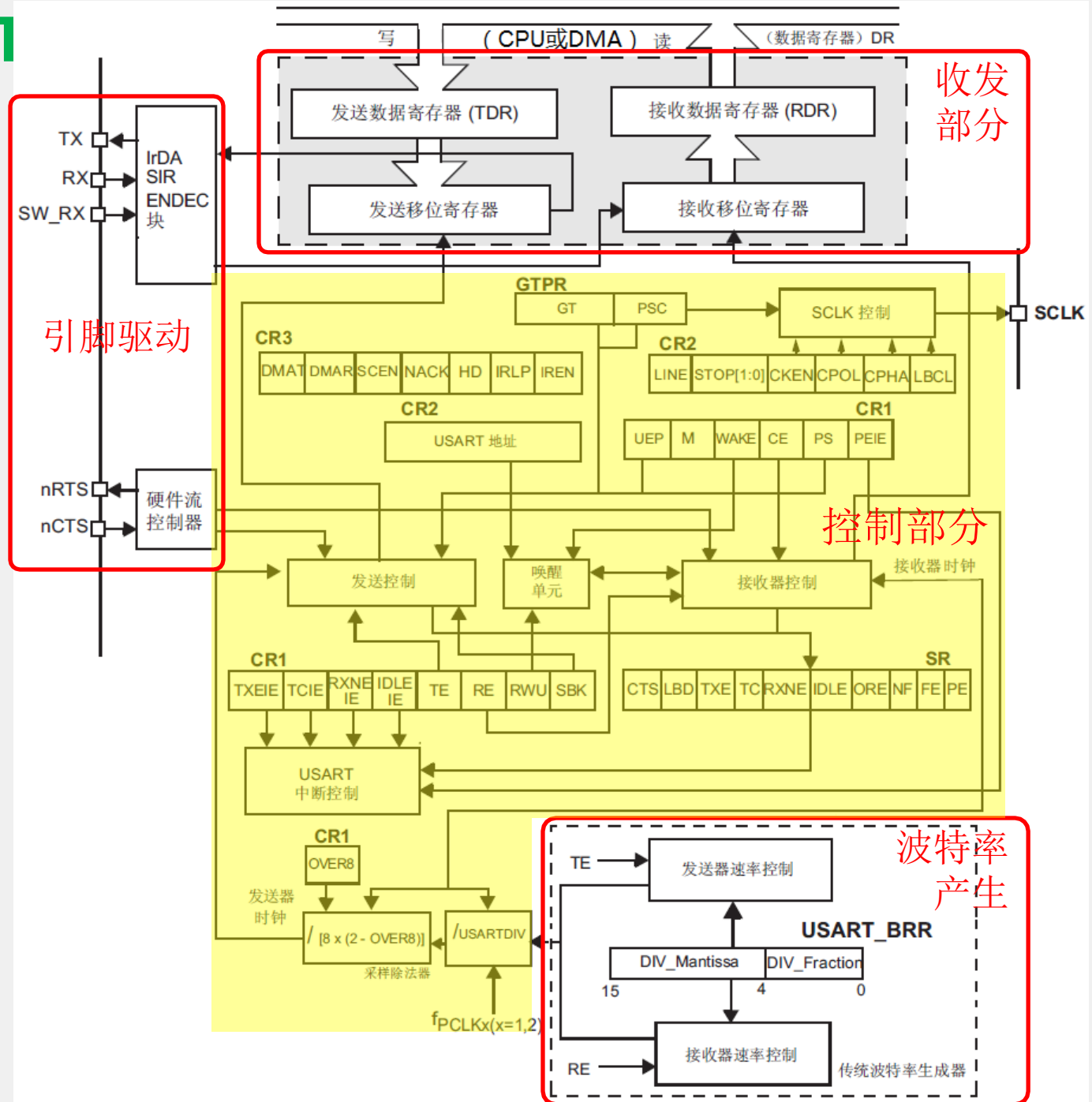


第8章 ST

8.3 USART功能描述

USART结构

- 1) 收发部分
 - 数据寄存器
 - 移位寄存器
- 2) 波特率产生
- 3) 控制部分
 - 数据帧参数
 - 时钟控制
 - 收发控制
 - 运行控制
 - 状态标志
 - 中断控制
 - 唤醒控制
 - 过采样
- 4) 引脚驱动



第8章 STM32串口USART及应用

8.3 USART功能描述

- 发送器

- 发送数据寄存器 (TDR)

- 存放要发送的数据

- 发送移位寄存器

- 负责一位一位发送

- 涉及控制寄存器CR1/CR2/CR3、状态寄存器SR、波特率寄存器BRR等

- 选择字长 (8位还是9位) : —— CR1中 **M**位

- 奇偶校验控制使能: —— CR1中 **PCE**位

- 奇偶校验选择位: —— CR1中 **PS**位

- 选择停止位: —— CR2中 **STOP[1:0]**位

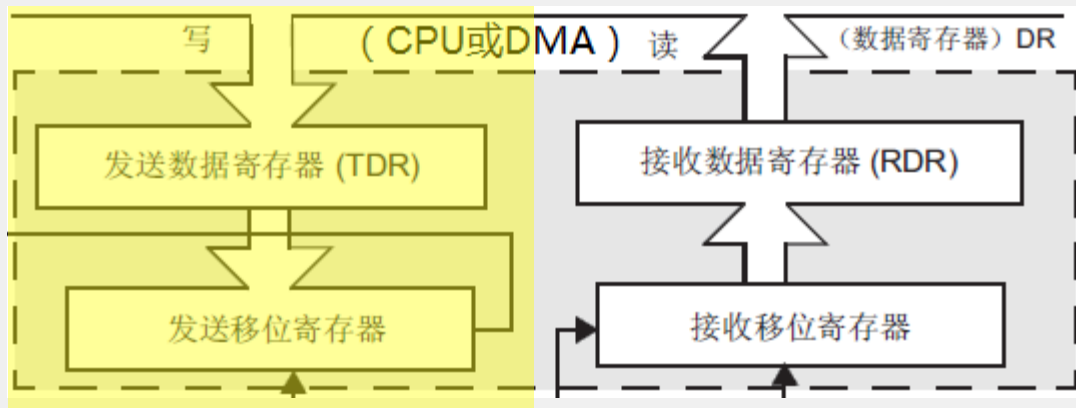
- USART使能: —— CR1中 **UE**位

- DMA使能 (如果需要) : —— CR3中 **DMAT**位

- 波特率设置: —— **BRR**寄存器

- 发送空闲帧 (启动发送) : —— CR2中 **TE**位

- 帧发送完成: —— SR中 **TC**位



注意：发送数据寄存器 (TDR) 和接受数据寄存器 (RDR) 共用一个名称DR，但物理上是分开的

第8章 STM32串口USART及应用

• 8.3 USART功能描述

- 发送器

• 发送配置步骤

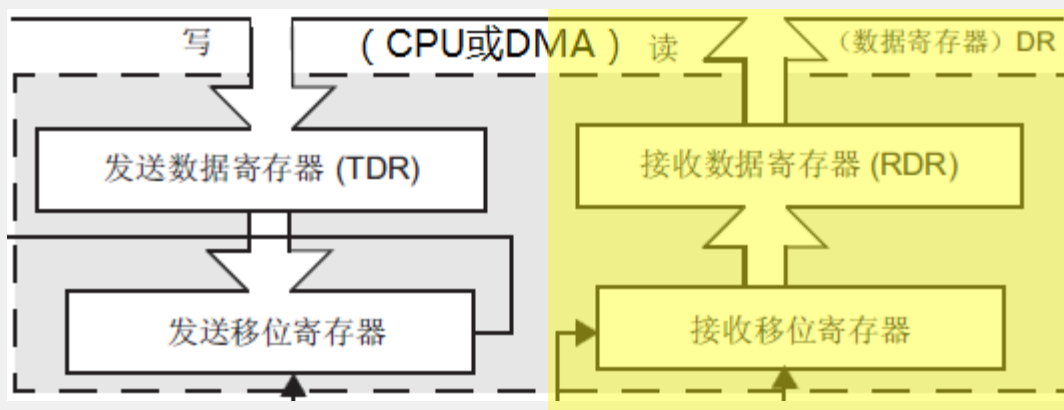
- ①使能USART
- ②定义字长，及奇偶校验位
- ③选择停止位
- ④如果需要，使能DMA，并按照多缓冲区通信中的解释说明配置DMA寄存器
- ⑤波特率设置
- ⑥发送一个空闲帧（启动发送）
- ⑦发送一个字（将要发送的数据写入数据寄存器DR）
- ⑧查询状态，等待发送完成

第8章 STM32串口USART及应用

• 8.3 USART功能描述

- 接收器

- 接收数据寄存器 (RDR)
 - 存放移位输入的数据
- 接收移位寄存器
 - 负责移位接收
- 接收器可以接收8位或9位数据，但要先解决一个问题：
 - 如何侦测起始位？



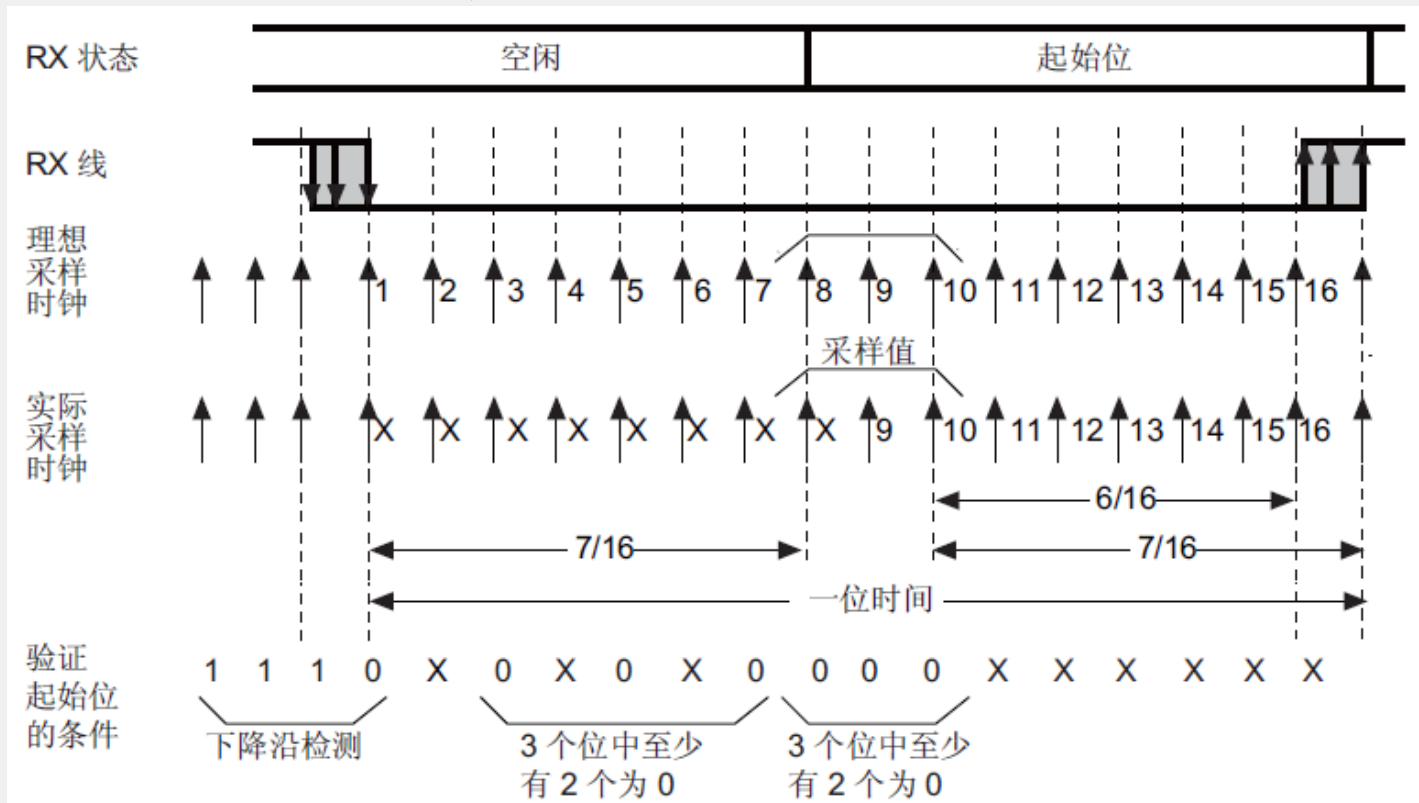
第8章 STM32串口USART及应用

8.3 USART功能描述

接收器

起始位侦测

- 在USART中，识别出特定序列的采样，可认为检测到了起始位。
- 该序列为：1110X0X0X0000
- 过采样率设成16 或 8，不影响起始位侦测的顺序



第8章 STM32串口USART及应用

• 8.3 USART功能描述

- 接收器

- 接收器同样有许多控制、状态和设置，涉及控制寄存器CR1/CR2/CR3、状态寄存器SR、波特率寄存器BRR等

- 选择字长（8位还是9位）：—— CR1中 **M**位 同发送器
- 奇偶校验控制使能：—— CR1中 **PCE**位
- 奇偶校验选择位：—— CR1中 **PS**位
- 选择停止位：—— CR2中 **STOP[1:0]**位
- USART使能：—— CR1中 **UE**位
- DMA使能（如果需要）：—— CR3中 **DMAR**位
- 波特率设置：—— **BRR**寄存器
- 选择适当的过采样率：—— CR1中 **OVER8**位
- 侦测起始位（启动接收）：—— CR2中 **RE**位
- 帧接收完成：—— SR中 **RXNE**位

第8章 STM32串口USART及应用

• 8.3 USART功能描述

- 接收器

• 接收配置步骤

- ①使能USART 同发送器
- ②定义字长，及奇偶校验位
- ③选择停止位
- ④如果需要，使能DMA，并按照多缓冲区通信中的解释说明配置DMA寄存器
- ⑤波特率设置
- ⑥侦测起始位（启动接收）
- ⑦查询RDR非空标志（为1时表示一帧数据接收完成）
- 如果接收到了数据，有可能：
- ⑧产生中断
- ⑨错误标志（帧错误、噪声、上溢）

第8章 STM32串口USART及应用

• 8.3 USART功能描述

- 波特率计算

• STM32支持所谓的“小数分频”，生成更精确的波特率

- 接收器和发送器采用相同的波特率，所以一个串口波特率只需设置一次
- 波特率一经设置立即生效，因此不允许在通信过程中更改波特率
- 1) 标准USART（包括SPI模式下）波特率计算公式如下：

$$\text{波特率} = \frac{f_{CK}}{8 \times (2 - OVER8) \times USARTDIV}$$

公式①

- 式中：

- » f_{CK} 为串口外设的总线时钟频率（USART1/6是APB2总线，USART2是APB1总线）
- » USARTDIV为分频系数，其值为小数，转换后就是波特率寄存器BRR的值。

- USARTDIV转换方法：

- » BRR高16位不用。将低16位分成2部分：
- » 其中高12位（[15:4]）保存UASRTDIV的整数部分，低4位（[3:0]）保存小数部分
- » 如果设置8倍过采样（OVER8=1），则小数部分只取3位（保持bit3=0），整数部分仍为12位

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/226052030134010134>