

## STM32 UART 整理说明

该接口通过 3 个引脚连接到另外的外部设备上。

任何 USART 双向通信都至少需要两个引脚：接收数据输入 RX 和发送数据输出 TX

当发送器禁能时输出引脚恢复到 I/O 端口配置。当发送器使能时且无数据发送，TX 引脚为高电平。

字长可以通过设置 USART\_CR1 寄存器中的 M 位来选择是 8 位还是 9 位

TX 引脚在起始位期间为低，停止位期间为高

空闲符被认为是一个全“1”的帧，其后紧跟着包含数据的下一个帧的起始位（“1”的数目包含了停止位的数目）

间隙符被认为是一个帧周期都接收到“0”。在间隙帧之后，发送器插入 1 个或者 2 个的停止位（逻辑“1”）来应答起始位

### 发送器

发送器可以发送 8 或者 9 位的数据字，这取决于 M 位的状态。相关时钟脉冲在 SCLK 引脚输出

#### 1、字符发送

USART 发送期间，TX 引脚先出现最低有效位。这种模式下，USART\_DR 寄存器包含了一个内部总线和发送移位寄存器之间的缓冲区 TDR。每个字符之前都有一位逻辑低电平的起始位，以可设置数目的停止位结束。

TE 位使能之后将发送一个空闲帧

#### 2、可配置的停止位

1 个停止位：这是陌生人的停止位数目

2 个停止位：常规 USART，单线和调制解调器模式下支持

0.5 个停止位：当处于智能卡模式下接收数据时使用

1.5 个停止位：当处于智能卡模式下发送数据时使用

空闲帧的发送包含了停止位。

间隙帧是 10（11）个低位之后跟着配置的停止位

配置流程：通过把 USART\_CR1 寄存器中的 UE 位写 1 来使能 USART→配置 USART\_CR1 寄存器中的 M 位来定义字长→配置 USART\_CR2 寄存器中的停止位数目→若采用多缓冲通信选择 USART\_CR3 寄存器中的 DMA 使能位（DMAT），按照多缓冲通信中解释的配置 DMA 寄存器→设置 USART\_CR1 寄存器中的 TE 位来发送一个空闲帧来作为第一次发送→通过 USART\_BRR 寄存器选择期望的波特率→往 USART\_DR 寄存器中写入要发送的数据，这也将清除 TXE 位。

#### 3、单字节通信

清除 TXE 位一般都是通过往数据寄存器中写入数据完成的

TXE 是由硬件设置的，它表明：数据已经从 TDR 中转移到移位寄存器了，数据发送已经开始；TDR 寄存器是空的；下一个数据可以写入 USART\_DR 寄存器，而不会覆盖前面的数据

当发送在进行时，一个对 USART\_DR 寄存器的写命令将数据保存到 TDR 寄存器中，并且当前传输完成之后，TDR 寄存器中的数据将被复制到移位寄存器中。

当没有进行发送时，往 USART\_DR 寄存器中写入一个数据，数据将直接被放入移位寄存器，发送开始，TXE 位将被立即置 1

当一个帧发送完成时（结束位之后），TC 位被置 1

清除 TC 位是通过下面的软件操作完成的：(1) 读一次 USART\_SR 寄存器 (2) 写一次 USART\_DR 寄存器（TC 位也可以通过对它写 0 清除，这个清除序列只建议在多缓存通信中使用）

#### 4、间隙字符

设置 SBK 位将发送一个间隙字符。若 SBK 位被置 1，在完成当前的发送之后将在 TX 线路上发送一个间隙字符。这一位在间隙字符发送完成时由硬件复位。USART 在最后一个间隙帧的末端插入一个逻辑 1，从而保证下一个帧的起始位能被识别

软件在间隙符发送之前复位 SBK，间隙符将不会被发送

## 5、空闲符

设置 TE 位将驱动 USART 在第一个数据帧之前发送一个空闲帧

## 接收器

接收器可以接收 8 位或 9 位的数据字，这取决于 USART\_CR1 寄存器中的 M 位

### 1、字符接收

在一次 USART 接收期间，RX 引脚最先接收到最低有效位。这种模式下，USART\_DR 寄存器由一个内部总线和接收移位寄存器之间的缓冲区（RDR）构成

配置流程：通过把 USART\_CR1 寄存器中的 UE 位写 1 来使能 USART→配置 USART\_CR1 寄存器中的 M 位来定义字长→配置 USART\_CR2 寄存器中的停止位数→若发生多缓冲通信，选择 USART\_CR3 寄存器中的 DMA 使能位（DMAT）→通过波特率寄存器 USART\_BRR 来选择期望的波特率→置位 USART\_CR1 寄存器中的 RE，这将使能接收器开始寻找起始位。

当接收到一个字符时：

RXNE 位被置 1，表明移位寄存器的内容被转移到 RDR

如果 RXNEIE 位被置 1，将产生一个中断

接收期间若发现帧错误，噪音或者溢出错误标志将会被置 1

多缓冲接收中，RXNE 在每接收到一个字节都会被置 1 并通过 DMA 读取数据寄存器来清除

在单缓冲模式，清除 RXNE 位是由软件读取 USART\_DR 寄存器清除。RXNE 标志也可以通过对它写 0 清除。RXNE 位必须在下一个字符接收完成前被清除，否则将产生溢出错误

### 2、间隙符：

当接收到间隙符时，USART 把它当做帧错误处理

### 3、空闲符：

当接收到空闲帧时，将和接收到数据一样处理，此外如果 IDLEIE 位被置 1 的话将产生一个中断

### 4、溢出错误

当接收到一个字符，而 RXNE 位还没有被复位，这时候将出现错误。在 RXNE 位被清除之前数据不能从移位寄存器转移到 RDR 寄存器

出现溢出错误时，ORE 位被置 1，通过在读 USART\_SR 寄存器之后读 USART\_DR 寄存器，ORE 位被复位

ORE 位被置 1 时，表明至少 1 个数据已经丢失：若 RXNE=1，上一个有效数据存放在接收寄存器 RDR，并且可读；RXNE=0，上一个有效数据已被读出，RDR 中无可读数据

### 5、噪音错误

在帧内发现噪音：NE 在 RXNE 位的上升沿被置 1

无效的数据从移位寄存器转移到 USART\_DR 寄存器

若为单字节通信，将不产生中断；多缓冲通信下，若 USART\_CR3 寄存器中的 EIE 位准备置 1，将导致一个中断

NE 位通过依次读取 USART\_SR 寄存器和 USART\_DR 寄存器复位

### 6、帧错误

由于没有同步上或大量噪音的原因，停止位没有在与它的时间上接和识别出来

当发现帧错误时：FE 位被硬件置 1；无效的数据从移位寄存器转移到 USART\_DR 寄存器；若为单字节通信，将不会产生中断，但这一位将和自身产生中断的 RXNE 位一起上升，多缓冲通信中，若 USART\_CR3 寄存器中

的 EIE 位被置 1，将导致一个中断

NE 位通过一次读取 USART\_SR 寄存器和 USART\_DR 寄存器复位

#### 7、接收期间配置停止位

要接收的停止位的数目可以通过控制寄存器 2 中的控制为配置。普通模式下可以是 1 位或者 2 位。智能卡模式下可能是 0.5 位或 1.5 位

#### 分数波特率的产生

接收器和发送器 (RX 和 TX) 都是设置成 USARTDIV 整数和小数寄存器中配置的值。

$\text{TX/RX 波特率} = \text{Fck} / (16 * \text{USARTDIV})$

例子：从 BRR 寄存器的值计算得到 USARTDIV

如果 DIV\_Mantissa=27D, DIV\_Fraction=12D (BRR=1BCH)，那么

$\text{Mantissa}(\text{USARTDIV}) = 27\text{D}$

$\text{Fraciton}(\text{USARTDIV}) = 12/16 = 0.75\text{D}$

因此， $\text{USARTDIV} = 27.75\text{D}$

在写入 USART\_BRR 后，波特率计数器会被波特率寄存器中的新值更新，因此在处理期间不应改变波特率寄存器的值

只有 USART1 是由 PCK2（最大位 72MHZ）提供时钟，其他的都由 PCLK1 提供时钟（最大为 36MHZ）

#### 多处理器通信

利用 USART 可以进行多处理器通信（只需把多个 USART 连接成一个网络）。

未编址设备可以通过静默功能的方式置为静默模式。

静默模式下：所有接收状态位都不会被设置

所有的接收中断都被禁止

USART\_CR1 寄存器中的 RWU 位被置 1，RWU 可以硬件自动控制或者在某些条件下有软件写

USART 可以通过两种方式进入和退出静默模式：如果 WAKE 位被复位，采用空闲线路检测模式

如果 WAKE 位被置位，采用地址标记检测模式

##### 1、空闲线路检测模式 (WAKE=0)

当 RWU 位被写 1 时，USART 进入静默模式

当发现空闲帧时，USART 退出静默模式，RWU 位也将被硬件清除，但是 USART\_SR 寄存器中的 IDLE 位不会被置 1。RWU 也可以被软件清 0

##### 2、地址标记检测 (WAKE=1)

此模式下，MSB 为 1 的字节被认为是地址，否则被认为是数据。

当接收到一个和预先设置在 USART\_CR2 寄存器中 ADD 位中的地址不匹配的地址字符，USART 进入静默模式。

当接收到一个和设置的地址匹配的地址字符，USART 退出静默模式。RWU 被清除，后面的字节也将正常接收，RXNE 位会因为接收到地址字符被置 1。

当接收端缓冲区没有数据时 (USART\_SR 寄存器中的 RXNE=0)，RWU 位可以被写 0 或者 1，否则写操作会被忽略。

在选择静默模式之前（设置 RWU 位）。USART 必须先接收一个数据字节，否则它不能运行在通过空闲线路检测唤醒的静默模式。

在地址标志检测唤醒配置中 (WAKE=1)，RWU 位在 RXNE 位被置 1 时不能通过软件修改

## 奇偶控制

可以通过设置 USART\_CR1 寄存器中的 PCE 位来使能奇偶控制。

发送模式：若 USART\_CR1 的 PCE 位被置位，写进数据寄存器的数据 MSB 位被校验位替换后发送出去。

## LIN 模式

此模式通过 USART\_CR2 寄存器的 LINEN 位选择。LIN 模式下，CLKEN 位，STOP[1:0] 位，SCEN, HDSEL, IREN 必须保持清除状态

### 1、LIN 发送

与正常 USART 发送存在如下区别：清除 M 位来设置 8 位字长度；设置 LINEN 位进入 LIN 模式，此情况下，设置 SBK 位来发送 13 个“0”作为间隙符，然后发送一个“1”来开启其实检测

### 2、LIN 接收

当 LIN 模式被使能时，间隙检测电路被激活。检测和正常 USART 接收器完全独立。间隙不管是在空闲时或者接收帧期间发生都能被检测到

检测起始位的方法和寻找间隙符或者数据是一样的。发现起始位后，电路采样下面的位。若 10 (LBDL=0) 或者 11 位 (LBDL=1) 连续的位都是 0，且金钩一个分隔符，USART\_SR 寄存器的 LBD 标志被置 1

如果第 10 或者 11 次采样之前采样到 1，间隙检测电路取消当前的检测而重新查找一个起始位

LIN 模式被使能，一旦发生了帧错误，接收器不会停止直到间隙字没有完成时接收到一个“1”或检测到间隙后接收到一个分隔符

## USART 同步模式

同步模式是通过往 USART\_CR2 寄存器中的 CLKEN 位写 1 来选择。此模式下，下面这些位必须保持清除状态：LINEN, SCEN, HDSEL, IREN

USART 允许用户在主模式下控制双向同步串行通信。SCLK 引脚是 USART 发送者时钟的输出。起始位和停止位期间不会往 SCLK 发送时钟脉冲。

在空闲，实际数据到来前和发送间隙期间，外部时钟不会被激活

SCLK 和 TX 同步，TX 上的数据也是同步的。USART 接收器和异步模式采用不同的工作方式。若 RE=1，数据在 SCLK 上采样而没有任何过采样

SCLK 引脚和 TX 引脚一起工作，故只有在发送使能时且数据在发送时才会提供时钟，这就意味着不可能在不发送数据时接收到同步数据

LBCL, CPOL, CPHA 必须在发送器和接收器都禁能时选择，这些为在发送器或者接收器使能时不能改变  
建议在同一条指令中设置 TE 和 RE 位以保证接收器的建立时间和保持时间最小

## 单线半双工模式

此模式通过设置 USART\_CR3 寄存器中的 HDSEL 位来选择。此模式下必须保持下面这些位的清除状态：LINEN, CLKEN, SCEN, HDSEL, IREN

一旦 HDSEL 被写 1：RX 不再被使用；无数据传输时，TX 总是被释放的。因此，它在空闲状态或接收状态时表现为一个标准 I/O 口，该 I/O 口在不被 USART 驱动时，必须配置成悬空输入或开漏的输出高。

特别的是，发送永远都不会被硬件阻止，一旦 TE 位被置 1 并且数据写入数据寄存器，发送就会连续发生

## 智能卡

智能卡模式是通过设置 USART\_CR3 寄存器中的 SCEN 位来选择。此模式下，下面这些位必须保持清除状态：LINEN, HDSEL, IREN

CLKEN 位可能被设置，从而为智能卡提供时钟

智能卡接口设计是支持 ISO7816-3 标准中定义的一部协议的智能卡。USART 应做如下配置：8 位数据奇偶校验，USART\_CR1 寄存器中的 M=1, PCE=1，并且满足如下条件之一：

接收时 0.5 停止位：USART\_CR2 中的 STOP=01

发送时 1.5 停止位，USART\_CR2 中的 STOP=11

当与智能卡相连时，USART 的 TX 输出驱动一个智能卡也驱动的双向线（SW\_RX 和 TX 必须连接到相同的 I/O）。在发送起始位和数据字节时，TX\_EN 被置有效，而在停止位被置无效。这样接收器只能在出现奇偶错误时才能驱动这条线路。若没有使用 TX\_EN，在停止位期间 TX 被拉高，这样只要 TX 被配置成开漏，接受者也可以驱动这条线路

智能卡是一个单线半双工通信协议

通过发送移位寄存器的数据发送至少延迟 1/2 的波特时钟

如果在接收 1/2 停止位帧时检测到奇偶错误，发送线路在完成接收帧时拉低并保持一个波特时钟

置 TC 标记有效可以通过设置保护时间寄存器延迟

TC 标志的撤销不会受智能卡模式影响

如果发送器检测到帧错误，NACK 不会被发送器的接收模块当做起始位

在接收器端，若检测到奇偶错误并且发送了 NACK，接收器不会把 NACK 当做起始位

智能卡模式下，间隙符是没有意义的，带帧错误的 00H 数据被看做是数据而不是间隙符

当来回切换 TE 位时，不会发送空闲帧。ISO 协议没有定义空闲帧

USART 能够通过 SCLK 输出位智能卡提供时钟。在智能卡模式下，SCLK 和通信无关，而是先通过一个 5 位预分频器简单地用内部的外设输入时钟来驱动智能卡的时钟。

## IrDA SIR ENDEC 模块

IrDA 模式是通过设置 USART\_CR3 寄存器中的 IREN 位来选择的。此模式下，下面这些位必须保持清除状态：LINE, STOP, CLKEN, SCEN, HDSEL

SIR 发送编码器对从 USART 输出的 NRZ 比特流进行调制。正常模式下，发送的脉宽定义在 3/16 位周期

SIR 接收解码器借条来自红外检测器归零位流，且向 USART 输出 NRZ 串行比特流。在空闲状态里，解码器的输入就通常是高。发送器输出和解码器输入有相反的极性。

IrDA 是一个半双工通信协议，如果发送器忙，IrDA 解码器将忽略所有 IRDA 接收线路上的数据。如果接收器忙，TX 上从 USART 到 IrDA 的数据不会被 IrDA 编码。在接收数据时，应避免发送数据，否则要发送的数据可能被破坏

“0”是作为高脉冲发送，而“1”是作为“0”发送

SIR 解码器把 IrDA 兼容的接收信号转变成 USART 的比特流

SIR 接收逻辑把高状态逻辑“1”，而低脉冲看做逻辑“0”

发送编码器输出和解码器输入有相反的极性。空闲时 SIR 的输出是低电平

IrDA 规范要求可接受的脉冲大于 1.41 微秒，可接受的脉冲宽度是可设置的。

接收器可以和低功耗发送器通信

在 IrDA 模式下，USART\_CR2 寄存器中的 STOP 位必须设置成 1 停止位

IrDA 低功耗模式叙述如下：

发送器：脉宽 3 倍于低功耗波特率。低功耗模式下可设置预分频值对系统时钟分频

接收器：与正常模式下接收类似。USART 应忽略宽度小于 1PSC 的脉冲

## 使用 DMA 的连续通信

USART 可以利用 DMA 进行连续通信。RX 和 TX 缓冲器可以独立产生 DMA 请求

### 1、使用 DMA 发送

DMA 模式发送可以通过设置 USART\_CR3 寄存器中的 DMAT 位使能。只要 TXE 位被置 1，数据就可以通过 DMA 外设从配置好的 SARM 区域导入到 USART\_DR 寄存器。使用下面的流程映射一个用于发送的 DMA 通道

把 USART\_DR 寄存器的地址写到 DMA 控制寄存器，配置成传输的目标地址，每次 TXE 事件发生时，数据将从存储器转移到这个地址

把存储器的地址写到 DMA 控制寄存器，配置成传输的源地址，每次 TXE 时间发生时，数据将从这个存储器区域转移到 USART\_DR 寄存器

把要发送的字节总数写入 DMA 控制寄存器

在 DMA 寄存器中设置通道的优先级

根据应用需要，设置半/全传输的 DMA 中断

利用 DMA 寄存器激活通道

当传输的数目达到 DMA 控制寄存器中设置的值时，DMA 控制寄存器在 DMA 通道中断向量上产生一个中断。

若要使用 DMA 来发送，不要使能 TXEIE

### 2、使用 DMA 接收

DMA 模式接收可以通过设置 USART\_CR3 寄存器中的 DMAR 位使。只要接收到一个数据字节，数据就可以通过 DMA 外设从 USART\_DR 寄存器导入到配置好的 SARM 区域。使用下面的流程映射一个用于 USART 接收的 DMA 通道

把 USART\_DR 寄存器的地址写到 DMA 控制寄存器，配置成传输的目标地址，每次 RXNE 事件发生时，数据将从这个地址转移到存储器

把存储器的地址写到 DMA 控制寄存器，配置成传输的目标地址，每次 RXNE 事件发生时，数据将从 USART\_DR 寄存器转移到这个存储器区域

把要发送的字节总数写入 DMA 控制寄存器

在 DMA 寄存器中设置通道的优先级

根据应用需要，设置半/全传输的 DMA 中断

当传输的数目达到 DMA 控制寄存器中设置的值时，DMA 控制寄存器在 DMA 通道中断向量上产生一个中断。

若使用 DMA 来接收，不要使能 RXNEIE 位

### 3、多缓冲通信中的错误标志和中断产生

在多缓冲通信情况下，传输过程中发生任何错误，错误标志都将在当前字节之后置有效。在单字节接收中，和 RXNE 一起置有效的帧错误，溢出错误和噪音错误，它们有独立的错误标志中断使能位，若被使能，初相任何一个错误，都会在当前字节传输之后产生中断

## 硬件流控制

可以通过 nCTS 输入和 nRTS 输出来控制两个设备之间的串行数据流

RTS 和 CTS 流控制可以分别通过 USART\_CR3 寄存器中的 RTSE 和 CTSE 位来使能

### 1、RTS 流控制

若 RTS 流控制被使能，那么只要 USART 接收器准备好了接收新数据，nRTS 有效。当接收寄存器为空时，nRTS 无效，表明希望在发送当前帧结束后停止传输

### 2、CTS 流控制

若 CTS 流控制被使能，那么发送器在发送下一个帧之前检查 nCTS 输入。若 nCTS 有效，那么下一个数据将被发送，否则发送不会发生。若 nCTS 在发送期间变为无效，当前的传输完成之后停止发送

当 CTSE=1 时，一旦 nCTS 输入翻转，CTSIF 状态位自动被硬件置位，这表明接收器是否准备好了通信。若 USART\_CR3 寄存器中的 CTSIE 位被置 1，将产生一个中断。