

## STM32L433 在 STOP 模式 USART 不能工作的问题解决

关键字: STM32L433, STOP, USART

### 1. 引言

在很多 STM32 系列中，USART 是不能在 STOP 模式下工作的。但是少数的系列中可以支持 USART 工作在 STOP0/1 模式下并可将芯片唤醒，比如 STM32L0 系列和 STM32L4 系列。那么，如果要在 STOP0/1 模式下使用 USART，有什么需要注意的地方呢？

### 2. 问题

#### 2.1. 问题起源

某客户在其产品的设计中，使用了 STM32L433VCT6。由于是低功耗产品应用，客户在阅读数据手册的时候，发现 STM32L4 可以支持 USART 工作在 STOP 模式下，如图 1:

图1. 各个工作模式下的功能列表（部分）

Table 5. Functionalities depending on the working mode<sup>(1)</sup> (continued)

Peripheral	Run	Sleep	Low-power run	Low-power sleep	Stop 0/1		Stop 2		Standby		Shutdown		VBAT
					-	Wakeup capability	-	Wakeup capability	-	Wakeup capability	-	Wakeup capability	
USB FS	○ <sup>(8)</sup>	○ <sup>(8)</sup>	-	-	-	○	-	-	-	-	-	-	-
USARTx (x=1,2,3)	○	○	○	○	○ <sup>(6)</sup>	○ <sup>(6)</sup>	-	-	-	-	-	-	-
Low-power UART (LPUART)	○	○	○	○	○ <sup>(6)</sup>	○ <sup>(6)</sup>	○ <sup>(6)</sup>	○ <sup>(6)</sup>	-	-	-	-	-

客户看到 USARTx 可以在 STOP 0/1 模式下工作，很开心，打算把它用起来。需要注意的是，这里打了个上标“(6)”，要先看一下写的是什么。

这个标注 6 的内容是这样的：“UART and LPUART reception is functional in Stop mode, and generates a wakeup interrupt on Start, address match or received frame event.”意思是说，UART 和 LPUART 接收可以在 STOP 模式下工作，并可以在起始位、地址匹配或接收帧事件上产生唤醒中断。嗯，在此，再次确认确实是在 STOP 模

式下工作的。同样的表格也出现在 STM32L433 的参考手册 RM0394 中的第 5 章 Power Control (PWR) 中。

因为是接收，所以客户也在参考手册的“38.5.3 USART receiver”一节中关于时钟源的选择中看到了“Choosing LSE or HSI16 as clock source may allow the USART to receive data while the MCU is in low-power mode. Depending on the received data and wakeup mode selection, the USART wakes up the MCU, when needed, in order to transfer the received data by software reading the USART\_RDR register or by DMA.”意思是说，当 MCU 工作在低功耗模式下时，选择 LSE 或者 HSI16 作为时钟源可以允许 USART 接收数据。根据接收的数据和唤醒模式的选择，USART 在需要时唤醒 MCU，以便通过软件读取 USART\_RDR 寄存器或 DMA 传输接收到的数据。

## 2.2. 问题描述

客户确定可以使用之后，客户开始其软件设计，使用了 USART3，选择采用了 LSE (32.768KHz) 作为 USART3 的时钟源，使其可以工作在 STOP 1 模式，波特率为 9600bps。在软件设计上，客户采用了 STM32L4Cube 的 LL 库来实现。但是，当其写完软件之后，却发现 USART3 无法正常建立通信。

客户分享了其时钟配置代码，如图 2 所示。

图2. 用户时钟配置代码

```
LL_RCC_SetAHBPrescaler(LL_RCC_SYSCLK_DIV_1);
LL_RCC_SetAPB1Prescaler(LL_RCC_APB1_DIV_1);
LL_RCC_SetAPB2Prescaler(LL_RCC_APB2_DIV_1);
LL_SetSystemCoreClock(16000000);
LL_RCC_SetUSARTClockSource(LL_RCC_USART1_CLKSOURCE_PCLK2);
LL_RCC_SetUSARTClockSource(LL_RCC_USART3_CLKSOURCE_LSE);
LL_RCC_SetLPTIMClockSource(LL_RCC_LPTIM2_CLKSOURCE_LSE);
LL_RCC_SetLPTIMClockSource(LL_RCC_LPTIM1_CLKSOURCE_LSE);
LL_RCC_SetADCClockSource(LL_RCC_ADC_CLKSOURCE_SYSCLK);
```

## 2.3. 问题分析

猜测问题来自于软件设计。

检查客户发来的时钟配置代码，没有发现什么问题。

所以，回到参考手册，需要仔细阅读才能猜测客户代码的问题可能会出现在哪里。打开参考手册 RM0394，研读“38.5.17 Wakeup from Stop mode using USART”一节。读完这一小节，认为客户所遇到的问题很大可能性来自于 UESM 位的设置。

在这一节的开头就已经写到：“The USART is able to wake up the MCU from Stop mode when the UESM bit is set and the USART clock is set to HSI16 or LSE (refer to Section Reset and clock control (RCC)).”意思是说，当 UESM 置位并且 USART 时钟设置为 HSI16 或 LSE 时，USART 能够从停止模式唤醒 MCU（参见 Reset and clock control (RCC) 章节）。也就是说，光把时钟设置为 HSI16 或 LSE 还不够，UESM 必须被置位才能实现其功能。

后面又强调了一次，“In order to be able to wake up the MCU from Stop mode, the UESM bit in the USART\_CR1 control register must be set prior to entering Stop mode.”

即为了能够从停止模式唤醒 MCU，必须在进入停止模式之前设置 USART\_CR1 控制寄存器中的 UESM 位。

### 3. 问题解决

跟客户确认他的应用代码中是否使能了 UESM 位，客户检查后,的确没有使能该位。由于其使用的是 STM32CubeL4 的 LL 库，于是建议他加上以下函数以使能 UESM 位，如下：

```
LL_USART_EnableInStopMode();
```

加上以上函数后问题解决。

### 4. 结论

即使客户已经将 USART3 的时钟源配置为 LSE，但由于在使用上漏掉了对 UESM 位进行置位，导致 USART3 无法在 STOP 模式下工作。所以，在应用程序上需要对 UESM 位置位来确保其工作正常。

### 5. 建议

其实这个问题，客户很容易去查证。虽然我们在 \STM32Cube\_FW\_L4\_V1.16.0\Projects\NUCLEO-L433RC-P\Examples 中并没有 UART 的例程，但是在同系列中，可以找到关于使用 USART 从 STOP 模式下唤醒的例程来作参考。比如，NUCLEO-L476RG 这个 Nucleo 板的例程下就分别有基于 HAL 库和基于 LL 库的例程，如下：

基于 HAL 库：\STM32Cube\_FW\_L4\_V1.16.0\Projects\NUCLEO-L476RG\Examples\UART\UART\_WakeUpFromStop。可以在例程中看到在 main()函数里边，在执行 HAL\_PWREx\_EnterSTOP1Mode(PWR\_STOPENTRY\_WFI);进入 STOP 1 模式之前，先执行了 HAL\_UARTEx\_EnableStopMode(&UartHandle);语句。这个 HAL\_UARTEx\_EnableStopMode()函数的功能就是将 UESM 位置位。

基于 LL 库：\STM32Cube\_FW\_L4\_V1.16.0\Projects\NUCLEO-L476RG\Examples\_LL\USART\USART\_WakeUpFromStop1。可以在例程中看到在 main()函数里边，在执行 EnterStop1Mode();进入 STOP 1 模式之前，先执行 PrepareUSARTToStopMode();语句。这个 PrepareUSARTToStopMode()函数其实就调用了 LL\_USART\_EnableInStopMode(USART1);置位了 USART1\_CR1 中的 UESM 位。

除此以外，最重要的还是要去阅读参考手册的内容。不仅仅是 UESM 位需要注意，其实里边还讲了其他相关的一些需要注意的地方，比如静默模式，还有 USART 允许的最大波特率的计算等等，都需要仔细了解一下。

---

## 参考文献

文件编号	文件标题	版本号	发布日期
DS11449	Ultra-low-power Arm® Cortex®-M4 32-bit MCU+FPU, 100DMIPS, up to 256KB Flash, 64KB SRAM, USB FS, LCD, ext. SMPS	Rev 6	21-May-2021
RM0394	STM32L41xxx/42xxx/43xxx/44xxx/45xxx/46xxx advanced Arm®-based 32-bit MCUs	Rev 4	03-Oct-2018

## 版本历史

日期	版本	变更
2021 年 12 月 15 日	1.0	首版发布

### 重要通知 - 请仔细阅读

意法半导体公司及其子公司 (“ST”) 保留随时对 ST 产品和 / 或本文档进行变更的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。若需 ST 商标的更多信息，请参考 [www.st.com/trademarks](http://www.st.com/trademarks)。所有其他产品或服务名称均为其各自所有者的财产。

本文档是 ST 中国本地团队的技术性文章，旨在交流与分享，并期望借此给予客户产品应用上足够的帮助或提醒。若文中内容存有局限或与 ST 官网资料不一致，请以实际应用验证结果和 ST 官网最新发布的内容为准。您拥有完全自主权是否采纳本文档（包括代码，电路图 etc）信息，我们也不承担因使用或采纳本文档内容而导致的任何风险。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2020 STMicroelectronics - 保留所有权利