AN4991
Application note

How to wake up an STM32xx Series microcontroller from low-power mode with the USART or the LPUART

Introduction

The universal synchronous asynchronous receiver transmitter (USART) and the low-power universal asynchronous receive transmitter (LPUART) feature advanced low-power mode functions, which allow receiving data properly even when the STM32xx Series MCU is in low-power mode and the APB clock is disabled.

In this document, STM32xx Series refer only to the product Series listed in *Table 1*.

Table 1. Applicable products

Type	Product Series
Microcontroller	STM32F0, STM32F3, STM32G0, STM32G4, STM32L0, STM32L4, STM32L4+, STM32L5 and STM32WB

Contents

List of tables

List of figures

1 General information

The STM32xx Series microcontrollers are Arm®[a]-based devices.

arm

2 Low-power modes from which the STM32xx Series MCU can be woken up with the USART/LPUART

The USART and the LPUART can wake up the STM32xx Series MCU from low-power mode. *Table 2* gives a summary of the low-power modes depending on the MCU Series.

Table 2. Low-power modes versus STM32xx Series

Product Series	The USART can wake up the MCU from	The LPUART can wake up the MCU from
STM32F0/F3	Stop mode (with main regulator in Run mode or in low-power mode)	N.A
STM32L0	Stop mode (with main regulator in Run mode or in low-power mode, range 1/2/3)	Stop mode (with main regulator in Run mode or in low-power mode, range 1/2/3)
STM32L4/L4+/L5/ WB	Stop mode 0 Stop mode 1	Stop mode 0 Stop mode 1 Stop mode 2
STM32G0/G4	Stop mode 0 Stop mode 1	Stop mode 0 Stop mode 1

For a detailed description of the above low-power modes, refer to the *Power control* section of the corresponding reference manual.

a. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

3 USART/LPUART wakeup features

3.1 Dual clock domain

The USART/LPUART is able to wake up the MCU from low-power mode only when the peripheral supports the dual clock domain. This means that the USART/LPUART can be clocked by a clock independent from the APB clock. This clock can be either the HSI or the LSE clock. The USART/LPUART is then able to receive data even if the USART/LPUART clock is disabled and the MCU is in low-power mode.

3.2 USART/LPUART wakeup sources

Different USART/LPUART wakeup sources are available to wake up the MCU from low-power mode:

- A specific event selected through the WUS bits field of the USART/LPUART_CR3 register.

  00: Wakeup on address match (as defined by ADD[7:0] and ADDM7 of USART/LPUART_CR2 register)

  01: Reserved.

  10: Wakeup on Start bit detection

  11: Wakeup each time data are received (i.e. RXNE set in USART/LPUART_ISR register)

  When the wakeup event is verified, the WUF flag in the USART/LPUART_ISR register is set by hardware independently of whether the MCU is in low-power mode or in Run mode. It generates a wakeup interrupt if the corresponding interrupt enable bit (WUFIE) is set in the USART/LPUART_CR3 register.
- RXNE interrupt

  The RXNE interrupt must be enabled by setting the RXNEIE bit in the USART/LPUART_CR1 register before entering low-power mode.

To enable the USART/LPUART to wake up the MCU from low-power mode, the UESM bit in the USART/LPUART_CR1 control register must be set prior to entering low-power mode.

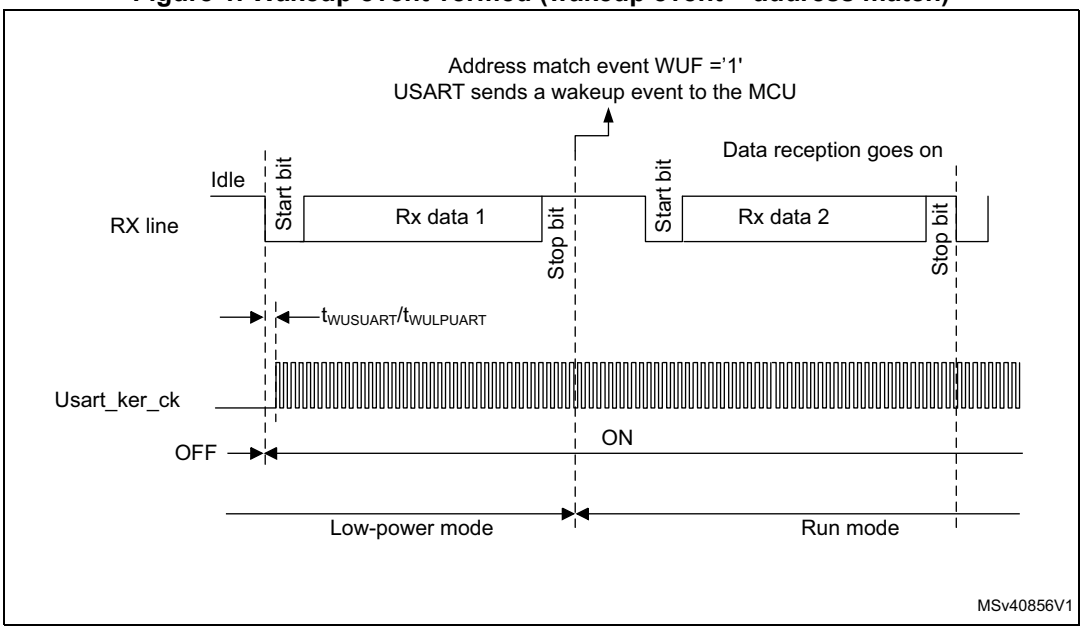
4 How the USART/LPUART wakes up the STM32xx Series MCU from low-power mode when the HSI clock is OFF

If the STM32xx Series MCU is in low-power mode and the HSI clock that is used as the USART/LPUART kernel clock is switched OFF when a falling edge on the USART/LPUART receive line is detected, the USART/LPUART interface requests the HSI clock to be switched ON again. The HSI clock is then used for frame reception.

If the wakeup event is verified, the MCU wakes up from low-power mode and data reception goes on normally.

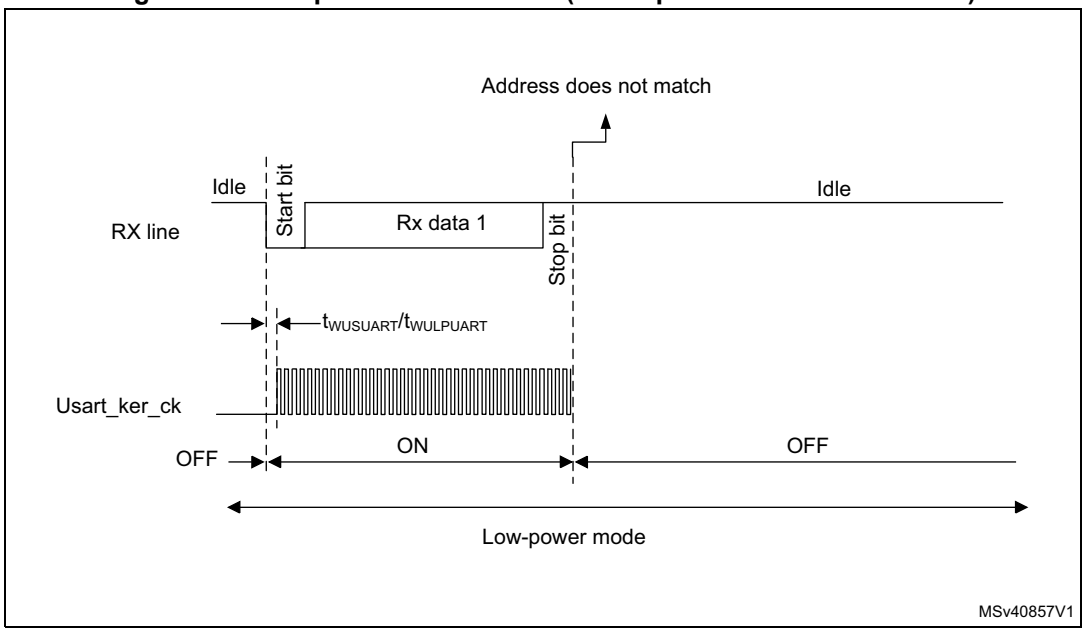
If the wakeup event is not verified, the HSI clock is switched OFF again, the MCU is not woken up and remains in low-power mode, and the kernel clock request is released. *Figure 1* and *Figure 2* show an example of wakeup event programmed to “address match detection”.

Figure 1. Wakeup event verified (wakeup event = address match)



1. Refer to *Section 5.2* for details on $t_{WUUSART}$ and $t_{WULPUART}$.

Figure 2. Wakeup event not verified (wakeup event = address match)



1. Refer to *Section 5.2* for details on $t_{WUUSART}$ and $t_{WULPUART}$.

5 How to determine the USART/LPUART baud rate that allows a correct wake up from low-power mode

The maximum baud rate that allows waking up correctly from low-power mode depends on whether the kernel clock is switched ON or OFF when the STM32xx Series MCU is in low-power mode.

5.1 USART/LPUART kernel clock switched ON in low-power mode

If the USART/LPUART kernel clock is switched ON during low-power mode, there is no constraint on the maximum baud rate that allows waking up from low-power mode. It is the same as in Run mode.

5.1.1 HSI clock used as USART/LPUART clock source

In STM32L0/L4 Series, there are two ways to keep the HSI clock ON during low-power mode:

- Set the HSIKERON bit in the RCC_CR register.
- or set the UCESM bit in the USART/LPUART_CR3 register. This bit allows the USART/LPUART to request the clock all the time and not only on Start bit falling edge.

Note: The UCESM bit is not available in the STM32G0/G4/L4+/L5/WB Series. So, in these series, only setting the HSIKERON bit allows keeping the HSI ON during low-power mode.

This section is not valid for STM32F0/F3 Series where the HSI clock is always OFF during Stop mode and is switched ON only when a falling edge is detected on the USART/LPUART receive line.

5.1.2 LSE clock used as LPUART clock source

When the LSE clock is used as LPUART clock source, the maximum baud rate that can be reached is 9600 bauds.

The LSE clock remains ON in low-power mode but in the STM32L0/L4 Series it is not propagated to the LPUART if the LPUART does not request this kernel clock. To correctly receive data at 9600 baud during low-power mode, the UCESM bit must be set in the LPUART_CR3 register. This bit allows the LPUART to request the clock all the time not only on Start bit falling edge.

In the other STM32xx Series, the LSE clock remains ON in low-power mode and is propagated to the LPUART, so there is no constraint to receive data at 9600 bauds.

5.1.3 LSE clock used as USART clock source

When the LSE clock is used as USART clock source, the maximum baud rate that can be reached is 4096 bauds in case of oversampling by 8, and 2048 bauds in case of oversampling by 16.

5.2 USART/LPUART HSI kernel clock is OFF in low-power mode

If the HSI clock is switched OFF during low-power mode, the maximum baud rate allowing to correctly wake up the STM32xx Series MCU from low-power mode depends on the following criteria:

- The wakeup time parameter ($t_{WUUSART}$ or $t_{WULPUART}$)

  In STM32F0/F3/L0 MCUs, $t_{WUUSART}$ (or $t_{WULPUART}$) equals t_{WUSTOP}, as specified in the device datasheets.

  In the other STM32xx Series, $t_{WUUSART}$ (or $t_{WULPUART}$) is specified in the device datasheets.
- The USART receiver tolerance which in turn depends on the following parameters:
  - 9-, 10- or 11-bit character length configured through the M bits of the USART_CR1 register
  - Oversampling by 8 or 16 configured through the OVER8 bit in the USART_CR1 register
  - BRR[3:0] bits of USART_BRR register equal to or different from 0000.
  - Use of one or three sample bits to sample data, depending on ONEBIT bit in the USART_CR3 register

  Table 3 and *Table 4* summarize the USART receiver tolerance according to the values of the above parameters.

Table 3. Tolerance of the USART receiver when BRR[3:0] = 0000

M bits	OVER8 bit = 0		OVER8 bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.75 %	4.375 %	2.50 %	3.75 %
01	3.41 %	3.97 %	2.27 %	3.41 %
10	4.16 %	4.86 %	2.77 %	4.16 %

Table 4. Tolerance of the USART receiver when BRR[3:0] # 0000

M bits	OVER8 bit = 0		OVER8 bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.33 %	3.88 %	2 %	3 %
01	3.03 %	3.53 %	1.82 %	2.73 %
10	3.7 %	4.31 %	2.22 %	3.33 %

- The LPUART receiver tolerance, which in turn depends on the following parameters:
  - Number of Stop bits configured through STOP[1:0] bits in the LPUART_CR2 register.
  - LPUART_BRR register value

  Table 5 summarizes the USART receiver tolerance according to the values of the above parameters.

Table 5. Tolerance of the LPUART receiver

M bits	OVER8 bit = 0		OVER8 bit = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
8 bits (M = 00), 1 Stop bit	1.82 %	2.56 %	3.90 %	4.42 %
9 bits (M = 01), 1 Stop bit	1.69 %	2.33 %	2.53 %	4.14 %
7 bits (M = 10), 1 Stop bit	2.08 %	2.86 %	4.35 %	4.42 %
8 bits (M = 00), 2 Stop bits	2.08 %	2.86 %	4.35 %	4.42 %
9 bits (M = 01), 2 Stop bits	1.82 %	2.56 %	3.90 %	4.42 %
7 bits (M = 10), 2 Stop bits	2.34 %	3.23 %	4.92 %	4.42 %

The USART/LPUART asynchronous receiver works correctly only if the total clock system deviation is less than the tolerance of the USART/LPUART receiver. The causes which contribute to the total deviation are:

- DTRA: deviation due to the transmitter error (which also includes the deviation of the transmitter local oscillator)
- DQUANT: error due to the baud rate quantization of the receiver
- DREC: deviation of the receiver local oscillator
- DTCL: deviation due to the transmission line (generally due to the transceivers which can introduce an asymmetry between the low-to-high transition timing and the high-to low transition timing)

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART/LPUART receiver tolerance}$$

where DWU is the is the error due to sampling point deviation when the wakeup from low-power mode is used.

The maximum baud rate that allows a correct wake up from low-power mode can be computed as follows:

- Case of USART/LPUART receiver with 9-bit data length, M bits = 01
  DWU max = $t_{WUUSART/WULPUART}$ / (11 x $T_{bit\ min}$)
  Baud rate max = (11 x DWU max)/ $t_{WUUSART/WULPUART}$
  where $T_{bit\ Min}$ is the minimum bit duration
- Case of USART/LPUART receiver with 8-bit data length, M bits = 00
  DWU max = $t_{WUUSART/WULPUART}$ / (10 x $T_{bit\ min}$)
  Baud rate max = (10 x DWU max)/ $t_{WUUSART/WULPUART}$
- Case of USART/LPUART receiver with 7-bit data length, M bits = 10
  DWU max = $t_{WUUSART/WULPUART}$ / (9 x $T_{bit\ min}$)
  Baud rate max = (9 x DWU max)/ $t_{WUUSART/WULPUART}$

Example of an STM32L4 USART receiver with OVER8 = 0, M bits = 10, ONEBIT = 1 and BRR [3:0] = 0000.

In these conditions, according to *Table 3: Tolerance of the USART receiver when BRR[3:0] = 0000*, the USART receiver tolerance is 4.86 %.

If we consider an ideal case where the DTRA, DQUANT, DREC and DTCL parameters are at 0%, DWU max is 4.86%.

In reality, we need to take into account at least the HSI inaccuracy.

Let us consider an HSI inaccuracy of 1 %, $t_{WUUSART}$ = 8.5 µs (for Stop mode 1/2)

DWU max = 4.86 % - 1 % = 3.86 %

$T_{bit\ min}$ = 8.5 µs / (9 x 3.86 %) = 24.4 µs.

In these conditions, the maximum baud rate allowing to wakeup correctly from low-power mode is

1/23.31 µs = ~ 40 Kbauds.

6 Conclusion

This application note explains how the USART/LPUART wakes up the STM32xx Series MCU from Stop mode. It also provides guidelines to approximately determine the USART/LPUART maximum baud rate that allows a proper wakeup from low-power mode.

7 Revision history

Table 6. Document revision history

Date	Revision	Changes
08-Mar-2017	1	Initial release.
17-Dec-2019	2	Updated microcontroller list in: – Title of the document – *Table 1: Applicable products* – *Section 5.2: USART/LPUART HSI kernel clock is OFF in low-power mode* Added *Section 1: General information* (Arm logo added) Updated *Table 2: Low-power modes versus STM32xx Series* Updated *Section 5.1.1: HSI clock used as USART/LPUART clock source* and *Section 5.1.2: LSE clock used as LPUART clock source*