STM32L476 Wake Up from Stop Mode

Application Note: AN4991

# Introduction

The following document aims to provide insight into the design choices and considerations for configuring the device in low power mode. In this section, the device is configured in a mode most suitable for standby while waiting for data reception. The micro controller will typically enter this mode while waiting for data from I2c or USART (aka GPS, Iridium and IMU) with long sampling periods and acquisition times. Running the microcontroller in run mode will result in unnecessary power dissipation however it will allow for the fastest response time for incoming data. As mentioned in previous documents, Standby and shut down mode are not suitable for this application since wake up from these modes results in power on reset. For this application, Stop Mode 1 is chosen, and the wake-up source is chosen to be an interrupt on either the UART4/5 or I2C2 peripherals

# USART Wake Up

A Wake up from stop mode can occur from a variety of sources. Each USART peripheral has the capability to wake up the device on the following conditions:

1. Specific wake up event by setting the WUS bits
   1. 00 – Wake up on address match
   2. 10 Wake up on start bit detection
   3. 11 Wake up on RXNE
2. RXNE interrupt

For these to work properly, The UE bit must be set in CR1 prior to entering LP mode. USART has the capability to wake the device up from stop modes 0 - 1 and sleep mode while LPUART has the capability to wake the device up from stop modes 0 -2.

Functionality of the wake-up feature is dependent on the USART clock source. If the MCU is switched off and the HSI (as the kernel clock) is off, the clock is turned on when a falling edge is detected on the USART Rx line is detected. The clock is then turned on and the USART peripheral detects a frame. If the triggering event corresponds to the event selected, the device will wake up otherwise, the device will go back to sleep

## Baud Rate Selection

Once again, the kernel clock determines the maximum baud rate that allows for wake up event detection. No baud rate constraints are imposed when the Kernel Clock is switched on in low power mode and the device is considered the same as if it were in run mode. If the HSI is used, there are 2 ways to keep it on: setting the HSIKON bit in the RCC\_CR or by setting UCESM bit in USART\_CR3. If the LSE is used, the maximum baud rate achievable is 9600 on LPUART and 4096 on USART this is way too low for the communication speeds we require.

If the HSI clock is switched off, the maximum baud rate is dependent on

1. The wake-up time parameter

For the STM32L4 This is notified by the paramters twuusart and twulpusart the values of this parameter are shown in the following table:

|  |  |  |
| --- | --- | --- |
|  | Stop Mode 0 wake up time | Stop Mode 1,2 wake up time |
| **LPUSART** | 1.7 uS | 8.5 uS |
| **USART** | 1.7 uS | 8.5 uS |

1. The USART receiver tolerance

This depends on the Mbit sample length as well as the oversample rate. If we consider the following peripherals: USART 2 (USB output) UART 4 (GPS) UART 5 (Iridium) The table shows the following parameters

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **M bit length** | **Oversample** | **1 bit enabled** | **Baud Rate** | **BRR value [0:3]** | **Tolerance** |
| USART2 | 9 | 16 | disabled | 115200 | 001 | 3.33% |
| UART 4 | 9 | 16 | disabled | 19200 | 010 | 3.33% |
| UART 5 | 9 | 16 | disabled | 115200 | 001 | 3.33% |

Thus USART asynchronous works correctly if the total system clock deviation < tolerance of USART receiver. Clock deviation can come from the following sources:

1. Transmitter Error (DTRA)
2. Reciever Baud Rate Quantization (DQUANT)
3. Local Oscilator Reciever (DREC)
4. Transmission line (DTCL)
5. Wake up sampling point deviation (DWU)

DTRA +DQUANT +DREC +DTCL +DWU < Tolerance of USART

Therefore we get the following equations

This allows us to calculate the deviation from wake up samling point for a device with 8 data bits and 1 stop bit

THUS The max baud rate can be calculated as:

Given the parameters in the table above and assuming clock from HIS\_16 oscilator with a tolerance of 1.5% for -40 to 85 degrees, this corresponds to:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | HSI 16 Oscilator | | MSI LSE PLL Clock | |
|  | DWU Max | BaudRate | DWU Max | BaudRate |
| Stop Mode 0 | 1.83% | 107647 bit/s | 3.3275% | 195735 bit/s |
| Stop Mode 1/2 | 1.83% | 21529 bit/s | 3.3275% | 39147 bit/s |

The table above shows maximum baud rates achievable for wake up from stop mode. As one can see the wake up from stop mode 1,2 is only possible with a PLL oscillator and can achieve a maximum wake up baudrate of 39 147 bits/s This is not suitable for debugging or GPS application however, it is suitable for Iridium. Therefore, it is recommended to keep the kernel clock on while device is in stop mode.

## Conclusions:

The USART peripherals, low power mode, baud rate and clock source will be set as follows

|  |  |  |  |
| --- | --- | --- | --- |
|  | Low Power Mode | Baud Rate | Clock Source |
| USART 2 | Stop Mode 1 WFI | 115200 | HIS (Kernal Off) |
| UART 4 | Stop Mode 1 WFI | 115200 | HSI (Kernal Off) |
| UART 5 | Stop Mode 1 WFI | 19200 | HIS (Kernal On) |