Serial Communication with USART

USART (**U**niversal **S**ynchronous/**A**synchronous **R**eceiver **T**ransmitter) is a ***serial*** hardware communication protocol built into AVR µC's. It is characterized by the following:

- **Clock Operation Modes**:

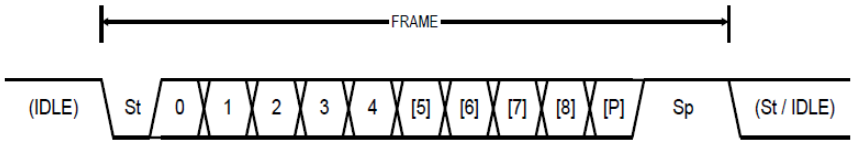
Modes	Asynchronous Normal Speed	Asynchronous Double Speed	Synchronous Master	Synchronous Slave
Sampling Rate	16	8	2	2

- ***Baud Rate and Oscillator Related Error***: Describes the number of ***bits*** transmitted ***per second***. In ***asynchronous*** mode, both the receiver and transmitter must function at the same ***baud rate*** for adequate communication. The generated ***baud rate*** depends on the frequency of the oscillator. If the wanted baud rate can't be obtained directly by division of the oscillator, the closest match for that baud rate will be chosen, and there will be an error between the wanted baud rate and the actual baud rate. (***See baud rate tables starting on page 235 of the ATmega324pb datasheet***)

$$\boldsymbol{BAUD = \frac{f_{osc}}{SR \times (UBRRn + 1)}}$$

  For good quality reception, the ***baud rate error*** should not surpass certain values depending on the data frame, ***the specification are given on page 232 of the ATmega324pb datasheet***. The maximum ***baud rate*** for the USART hardware protocol is $\frac{f_{osc}}{SR}$ (***sampling rate*** (**SR**) is **16** in normal speed asynchronous mode and 8 in double speed asynchronous mode and **2** in synchronous master mode).

- ***Frame***: The data frame to be transmitted or checked upon reception has the following form:



- ***Start Bit***: The ***start bit*** send by the transmitter signals to the receiver that a ***symbol*** is about to be transmitted. There is ***only one*** start bit, and it is always **low**.

- ***Number of Data Bits***: We can determine the number of data bits to be transmitted or received. This is done through the $\boldsymbol{UCSRnB}$ register.

- ***Parity***: A bit at the end of the data frame that is generated by the transmitter and checked by the receiver for data validity. The parity bit is calculated using the "***exclusive or***" operator on the transmitted bits for ***even*** parity. For ***odd*** parity, the result of the "***exclusive or***" is inverted. (**See pages 222 and 223 of the ATmega324pb datasheet**) The ***parity mode*** of the frame is set by the $\boldsymbol{UCSRnC}$ register.

- ***Stop Bits***: The ***stop bit(s)*** indicate(s) that the transmission of a ***symbol*** ended. There can be 1 or 2 ***stop bits***, and they are always **high**. It is better to use 2 ***stop bits*** at higher ***baud rates*** to give the µC time to process the received information and is useful in high noise environments.

Asynchronous Data Reception: The USART's asynchronous mode doesn't have a clock, this it needs a ***Clock and Data Recovery*** unit.

- ***Clock Recovery***: The ***USART*** receiver uses the ***falling edge*** of the ***start bit*** to synchronize its internal clock with the rest of the upcoming serial frame, thus synchronizing with the internal clock of the transmitters ***USART's*** clock. This process is done at each ***start bit*** with relatively good accuracy. For more information about how the first ***start bit*** is detected and the synchronization margin, ***see page 230 of the ATmega324pb datasheet***.
- ***Data Recovery***: The ***data recovery*** unit uses a ***state machine*** that has 16 states for each bit in ***normal mode***, and 8 states for each bit in ***double speed mode***. The bigger number of states helps with the ***clock synchronization*** accuracy. For both operation modes, the ***middle 3 bits*** determine by a ***"majority vote"*** if the current bit is high or low.

Synchronous Mode: The ***synchronous*** mode is triggered by ***setting*** the bit $\boldsymbol{UMSELn0}$ in the $\boldsymbol{UCSRnC}$ register. The $\boldsymbol{XCKn}$ pin is used as a clock and is only active when ***synchronous*** mode is enabled. The associated $\boldsymbol{DDRx}$ register determines whether the clock source is ***internal*** (***Master Mode, Output***) or ***external*** (***Slave Mode, Input***). That pin is only active when using ***synchronous*** mode. The maximum baud rate for synchronous mode is $\frac{f_{osc}}{4}$. The $\boldsymbol{UCPOL}$ bit in

UCSRnC selects which ***edge*** (falling or rising) is used for data sampling and which is used for data change. (***See page 221 of the ATmega324PB datasheet***)

MPCM: **MPCM** (Multi-Processor Communication Mode) is used to transmit information from one ***master*** device to multiple ***slave*** devices, with the introduction of an ***address*** referring to the device for which the information is meant. For an 8-bit data frame, we must introduce a ***ninth bit*** transferred through the ***RXB8*** bit in ***UCSRnB*** and the data frame must be redefined of course. The ninth bit must be written by the transmitter and read by the receiver before writing or reading to ***UDRn***. (***See page 234 of the ATmega324PB datasheet***)

Error Detection

- ***Parity Error***: A parity error indicates that the data has been modified along the way on the channel due to noise, or bad detection by the receiver (***see baud rate errors***). The bit ***UPE*** in the register ***UCSRnA*** is set whenever a parity error is detected in the ***data buffer***.
- ***Frame Error***: A frame error indicates that the received frame doesn't correspond to the one we are expecting to receive. The bit ***FE*** in the register ***UCSRnA*** is set when a frame error is detected in the ***data buffer***. A frame error is only detected when the first stop bit is low instead of high.
- ***Data Overrun***: The data overrun flag indicates data loss due to new data overrunning unread old data in the ***UDRn***.

Interrupts: The ***global interrupt*** flag should be ***cleared*** when initializing the USART peripheral. There are three interrupts for each USART peripheral. (***See interrupts table on pages 73 and 74 of the ATmega324PB µC***)

- ***Transmitter Interrupts***: ***UDRE*** (USART Data Register Empty) and ***TXC*** (Transmit Complete) in ***UCSRnA*** are two transmission flags that can be set as ***interrupts*** by setting the bits ***TXCIE*** and ***UDRIE*** in the ***UCSRnB***.
- ***Receiver Interrupt***: ***RXC*** (Receive Complete) in ***UCSRnA*** is a flag that can be se as an ***interrupt*** by setting the bit ***RXCIE*** in ***UCSRnB***.

Important Registers

UBRRnH & ***UBRRnL*** ***UCSRnA*** ***UCSRnB*** ***UCSRnC*** ***UCSRnD***