Linnaeus University

Faculty of Technology – Department of Computer Science

1DT301 - Computer Technology
Lab 3

Student: Hussam Aldeen ALKHAFAJI

Student ID: ha223cz@student.lnu.se

1. What is baud rate? How does it differ from bit

rate?

Baud rate is expressed in the number of times a signal can change on transmission line per second. Usually, the transmission line uses only two signal states, and makes the baud rate equal to the number of bits per second that can be transferred.
It differs from bitrate in the fact that it can determine the required bandwidth to send the signal.
And while the baud rate determines how many times the state of a signal is changing, bitrate determines the number of bits traveled per second.
Their relation is highlighted by the equation.
Baud rate = bit rate / the number of bits per signal unit
[1].

2. What is the difference between synchronous and asynchronous communication?

In synchronous communication, one side of the communication transmits a clock signal along with the data signal, this clock signal is to be used to synchronize the communication between the two components. One of them will be called the master the other is the slave.
In asynchronous communication, both sides send the data signal only, they're meant to be in agreement beforehand on the speed of transmission.
[2]

3. Describe how USART initiates communication and sends data.

For the communication to be set up, we need to handle 5 different registers.
First of all, we need to calculate and set the right baud rate using the formula
BAUD = fOSC / 16UBRRn + 1.
We usually know what BAUD rate we want to work with and we figure out the UBRR which is then given to UBBRNL. (UBBRNH is set to zero by default)
After that we can configure configure the frame format of the data however we wish it to be using the UCSR0C register:

- We can set the data to be send/receive to minimum 5 bits [0 - 4] and maximum 9 bits
- We can set either one stop bit at the end of the frame format (it’s the case by default) or two bits.
- We can set the parity bit which transmits information about the number of highs in our data, this can be used for error checking by the receiver.
- We can also choose the mode we’re going to work with in this register, which is either asynchronous or synchronous Master mode.

Next, we can configure UCSR0A to enable error detection, double speed, and multiprocessor mode.

Most importantly in the UCSR0B, we must enable the receiver bit and/or the transmitter bit (TXEN0 and RXEN0), without this, the USART will ignore anything it receives or is stored in it’s UBBR0 register.

Optionally, you can also enable the interrupt related bits in the same register

Now that our USART is initialized and correctly set up, we can start communicating with another device.
To initiate communication,
To transmit, all we have to do is to store the data in the UDRN. this will automatically start the transmission. The USART will start the communication starting with a low voltage (0) then it’ll proceed to send the data bit by bit according to the frame format until it sends the stop bit which is always a high voltage (1).
To receive, all we need to do is to take the data from UDRN.
However, it’s good practice to check that the UDR is free for transmission and reception before manipulating the UDR by checking the corresponding bits in the UCSRNA register.
[3]

Bibliography

[1]https://techdifferences.com/difference-between-bit-rate-and-baud-rate.html#KeyDifferences

[2]http://electronics.stackexchange.com/questions/64772/ddg#64776

[3] https://www.youtube.com/watch?v=KnyyQujvcBo

Embedded C Programming and the Atmel AVR, 2nd Edition.