

Judah Ben-Eliezer

112352727

3/25/2021

Prelab 7:

AVR128DB48 USART Module in Asynchronous Serial (RS232) Mode and
Saleae Logic Analyzer

```
/*
 * USART3_asynch_test.c
 *
 * Created: 3/25/2021 6:23:48 PM
 * Author : Judah Ben-Eliezer
 */

#define BAUD_RATE 9600UL
#define F_CPU 4000000UL

#include <avr/io.h>

char c;

int main(void)
{
    PORTB.DIRSET = PIN0_bm;                                     ↗
                                                                // enable output on PB0.

    USART3.BAUD = BAUD_RATE;                                   ↗
                                                                // set baud rate.
    USART3.CTRLA = USART_CMODE_ASYNCHRONOUS_gc | USART_PMODE_DISABLED_gc |
        USART_SBMODE_1BIT_gc | USART_CHSIZE_8BIT_gc;          ↗
                                                                // asynchronous mode, no parity
                                                                bit, 1 stop bit, 8 bits data.
    USART3.CTRLB = USART_TXEN_bm;                               ↗
                                                                // enable transmission.

    c = 'U';

    while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){} ↗
                                                                // wait for buffer to be empty.
    USART3.TXDATAL = c;

    while (1)
    {
        // do nothing
    }
}
```



```
/*
 * USART3_loopback.c
 *
 * Created: 3/25/2021 5:52:08 PM
 * Author : Judah Ben-Eliezer
 */

#define BAUD_RATE 9600UL
#define F_CPU 4000000UL

#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

char c;

int main(void)
{
    PORTB.DIRSET = PIN0_bm; // enable output on PB0.
    PORTB.DIRCLR = PIN1_bm; // enable input on PB1.

    USART3.BAUD = BAUD_RATE; // set baud rate.
    USART3.CTRLA = USART_CMODE_ASYNCHRONOUS_gc | USART_PMODE_DISABLED_gc |
        USART_SBMODE_1BIT_gc | USART_CHSIZE_8BIT_gc; // Asynchronous mode, no parity
        bits, single stop bit, 8 bits data.
    USART3.CTRLB = USART_TXEN_bm | USART_RXEN_bm; // enable transmission, enable receiving.
    USART3.CTRLA = USART_RXCIE_bm; // enable RX complete interrupt.

    sei();

    c = 'A';

    while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){} // wait till buffer is empty.
    USART3.TXDATAL = c; // transmit c

    while (1)
    {
    }
}

ISR (USART3_RXC_vect) {
    if ((USART3.STATUS & USART_RXCIF_bm) == USART_RXCIF_bm) {
        if (USART3.RXDATAH != 0x00) return;
    }
}
```

```
c = USART3.RXDATAL;

if (c >= 'A' && c <= 'Z') {
    while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){}
    // wait till buffer is empty.
    USART3.TXDATAL = c + 20;
    // send lowercase character.
} else if (c >= 'a' && c < 'z') {
    while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){}
    // wait till buffer is empty.
    USART3.TXDATAL = c - 19;
    // send next uppercase character.
} else if (c == 'z') {
    while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){}
    // wait till buffer is empty.
    USART3.TXDATAL = 'A';
    // send 'A'
}
}
}
```

```
/*
 * USART3_echo_rs232.c
 *
 * Created: 3/25/2021 8:00:21 PM
 * Author : jlb
 */

#define BAUD_RATE 9600UL
#define F_CPU 4000000UL

#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

char c;

int main(void)
{
    PORTB.DIRSET = PIN0_bm; // enable output on PB0.
    PORTB.DIRCLR = PIN1_bm; // enable input on PB1.

    USART3.BAUD = BAUD_RATE; // set baud rate.
    USART3.CTRLA = USART_CMODE_ASYNCHRONOUS_gc | USART_PMODE_DISABLED_gc |
        USART_SBMODE_1BIT_gc | USART_CHSIZE_8BIT_gc; // Asynchronous mode, no parity
        bits, single stop bit, 8 bits data.
    USART3.CTRLB = USART_TXEN_bm | USART_RXEN_bm; // enable transmission, enable receiving.
    USART3.CTRLA = USART_RXCIE_bm; // enable RX complete interrupt.

    sei();

    while (1)
    {
    }
}

ISR (USART3_RXC_vect) {
    if ((USART3.STATUS & USART_RXCIF_bm) == USART_RXCIF_bm) {
        if (USART3.RXDATAL != 0x00) return;
        c = USART3.RXDATAL;

        while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){}
        USART3.TXDATAL = c + 20;

    }
}
```

```
/*
 * USART3_echo_usb.c
 *
 * Created: 3/25/2021 7:56:05 PM
 * Author : Judah Ben-Eliezer
 */

#define BAUD_RATE 9600UL
#define F_CPU 4000000UL

#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

char c;

int main(void)
{
    PORTB.DIRSET = PIN0_bm; // enable output on PB0.
    PORTB.DIRCLR = PIN1_bm; // enable input on PB1.

    USART3.BAUD = BAUD_RATE; // set baud rate.
    USART3.CTRLA = USART_CMODE_ASYNCHRONOUS_gc | USART_PMODE_DISABLED_gc |
        USART_SBMODE_1BIT_gc | USART_CHSIZE_8BIT_gc; // Asynchronous mode, no parity
        bits, single stop bit, 8 bits data.
    USART3.CTRLB = USART_TXEN_bm | USART_RXEN_bm; // enable transmission, enable receiving.
    USART3.CTRLA = USART_RXCIE_bm; // enable RX complete interrupt.

    sei();

    while (1)
    {
    }
}

ISR (USART3_RXC_vect) {
    if ((USART3.STATUS & USART_RXCIF_bm) == USART_RXCIF_bm) {
        if (USART3.RXDATAH != 0x00) return;
        c = USART3.RXDATAL;

        while ((USART3.STATUS & USART_DREIF_bm) != USART_DREIF_bm){}
        USART3.TXDATAL = c + 20;

        // echo character.
    }
}
```