```c
/*
 * usart3_init_test.c
 *
 * Created: 3/3/2022 12:23:17 PM
 * Author : jason
 */

#include <avr/io.h>

#define F_CPU 4000000
#define USART3_BAUD_RATE(BAUD_RATE) ((float)(F_CPU * 64 / (16 *(float)BAUD_RATE)))  //Calculation of baud rate from data sheet
#include <avr/io.h>
#include <util/delay.h>

//Header function
void USART_sw_write(char c);
void USART3_init (uint16_t baud, uint8_t data_bits, unsigned char parity);


int main(void)
{
    /* Replace with your application code */

    uint16_t baudRate = 9600;   //For the baud rate of USART3
    uint8_t dataBits = USART_CHSIZE_8BIT_gc;    //For the (character size)  CHSIZE [2:0]
    unsigned char parity = 0x00;    //PMODE[1:0]

    USART3_init(baudRate, dataBits, parity);

    while (1)
    {
        //Send a character to the Tera Term (TX pin)
        USART_sw_write('U');
        _delay_ms(2);

    }
}

//In Laboratory 05 you were not required to organize your program using functions, even though
//this approach was discussed in class. A simple function to configure a USART might have a single parameter that specifies the desired baud rate.
//The function that you must write for this task goes further than that, it allows both baud rate and
//the frame format to be specified
void USART3_init (uint16_t baud, uint8_t data_bits, unsigned char parity){
    PORTB_DIR |= PIN0_bm;  //To transmit the data

    //Specify the baud rate value for the USART3
    USART3.BAUD = (uint16_t)USART3_BAUD_RATE(baud);
```

```
    //Initialize the data bits and the parity bits type
    USART3_CTRLC |= data_bits | parity;
    USART3.CTRLB |= USART_TXEN_bm;  //Enable USART transmitter
}


//Function to be able to transmit characters
//to the TX pin and display on the Tera Term
void USART_sw_write(char c)
{
    //Poll until the  transmit buffer register are empty
    //when they contain data that has not been moved to
    //transmit shift register
    while (!(USART3.STATUS & USART_DREIF_bm))
    {
        ;
    }

    //Load data to transmit shift register and
    //output each of the bits serially to the TXD pin
    USART3.TXDATAL = c;
}
```