UNIVERSITY OF NEVADA LAS VEGAS. DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING LABORATORIES.

Class:	**CPE301L Digital Systems Architecture and Design 1001**	Semester:	**Spring 2025**
Points		Document author:	**Narek Kalikian**
		Author's email:	**kalikn1@unlv.nevada.edu**
		Document topic:	**Postlab 5**
Instructor's comments:			

1. Introduction / Theory of Operation

In this lab, we will be learning about Universal asynchronous receiver/transmitter modes (USART) with the goal of transmitting data to a computer using the RS232 communication standard. RS232 is a set of standards used for serial communication between Data Terminal Equipment (DTE) and Data Circuit-terminating Equipment (DCE). The RS232 limits speed and large voltage swings so the standard connection was replaced several times to account for certain limitations. In this lab we will be using the DB-9 connector, which precedes USB connections, and using a DTE-to-DTE (null modem) connection. This report will include the USART experiments using the ATmega328p microcontroller, DB-9 connector cable, and MobaExterm software for printing to the terminal.

2. Prelab Content

1. UCSR0C Value:

- Bit 0: UCPOL0
- Bit 1: UCSZ00
- Bit 2: UCSZ01
- Bit 3: USBS0
- Bit 4: UPM00
- Bit 5: UPM01
- Bit 6: UMSEL00
- Bit 7: UMSEL01

- Asynchronous:
    - UMSEL00 = 0, UMSEL01 = 0
- Disabled Parity:
    - UPM00 = 0, UPM01 = 0
- One Stop Bit:
    - USBS0 = 0

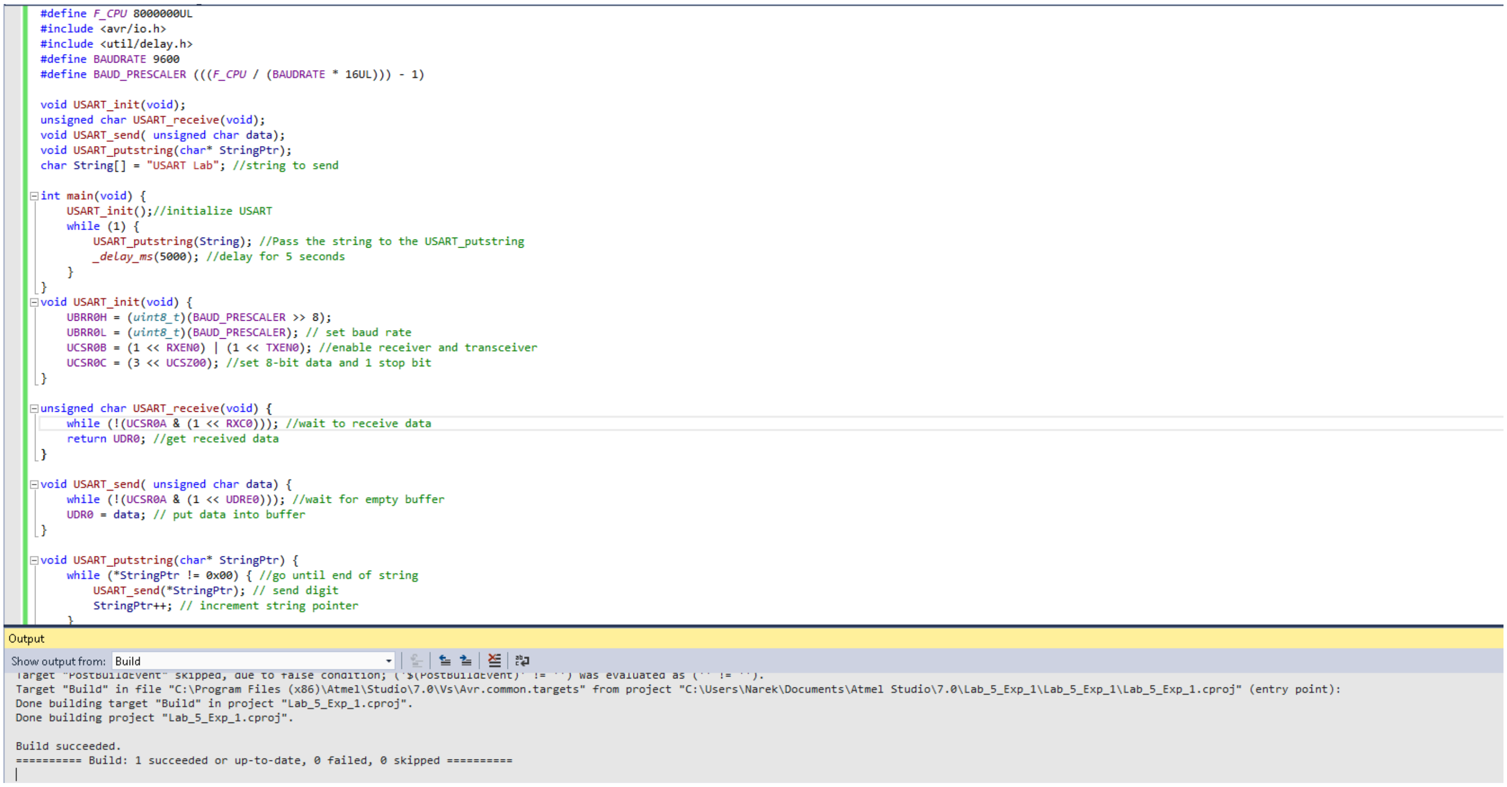
- 8 Data Bits:
    - UCSZ00 = 1, UCSZ01 = 1 → So, UCSR0C = **00000110**

2. UBRR0L and UBRR0H for 9600 baud rates (8 MHz Clock):

- CPU Frequency = 8 MHz
- Baud Rate = 9600
- U2X0 = 0
- UBBR = ((CPU Frequency)/((16)*(Baud))) - 1 → (8*10^6)/((16)(9600)) - 1 = 51 (0x33)
- UBBR0H (High Byte) = **0x00**
- UBBR0L (Low Byte) = **0x33**

3. Description of Experiments

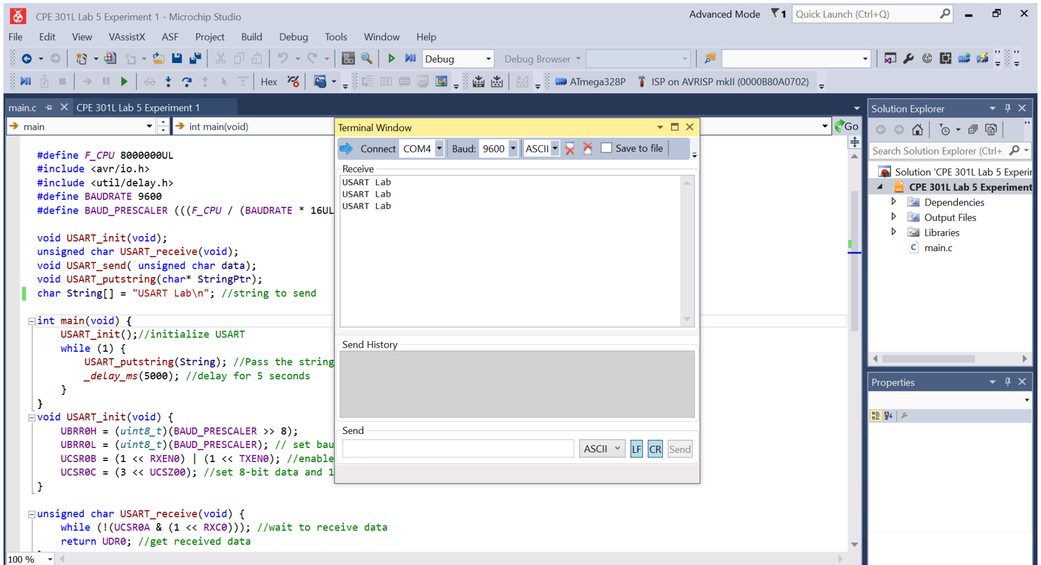
Experiment 1 - Code:



```
#define F_CPU 8000000UL
#include <avr/io.h>
#include <util/delay.h>
#define BAUDRATE 9600
#define BAUD_PRESCALER (((F_CPU / (BAUDRATE * 16UL))) - 1)

void USART_init(void);
unsigned char USART_receive(void);
void USART_send( unsigned char data);
void USART_putstring(char* StringPtr);
char String[] = "USART Lab"; //string to send

int main(void) {
    USART_init();//initialize USART
    while (1) {
        USART_putstring(String); //Pass the string to the USART_putstring
        _delay_ms(5000); //delay for 5 seconds
    }
}
void USART_init(void) {
    UBRR0H = (uint8_t)(BAUD_PRESCALER >> 8);
    UBRR0L = (uint8_t)(BAUD_PRESCALER); // set baud rate
    UCSR0B = (1 << RXEN0) | (1 << TXEN0); //enable receiver and transceiver
    UCSR0C = (3 << UCSZ00); //set 8-bit data and 1 stop bit
}

unsigned char USART_receive(void) {
    while (!(UCSR0A & (1 << RXC0))); //wait to receive data
    return UDR0; //get received data
}

void USART_send( unsigned char data) {
    while (!(UCSR0A & (1 << UDRE0))); //wait for empty buffer
    UDR0 = data; // put data into buffer
}

void USART_putstring(char* StringPtr) {
    while (*StringPtr != 0x00) { //go until end of string
        USART_send(*StringPtr); // send digit
        StringPtr++; // increment string pointer
    }
```

```
Target "PostBuildEvent" skipped, due to false condition; ('$(PostBuildEvent)' != '') was evaluated as ('' != '').
Target "Build" in file "C:\Program Files (x86)\Atmel\Studio\7.0\Vs\Avr.common.targets" from project "C:\Users\Narek\Documents\Atmel Studio\7.0\Lab_5_Exp_1\Lab_5_Exp_1\Lab_5_Exp_1.cproj" (entry point):
Done building target "Build" in project "Lab_5_Exp_1.cproj".
Done building project "Lab_5_Exp_1.cproj".

Build succeeded.
========== Build: 1 succeeded or up-to-date, 0 failed, 0 skipped ==========
```

Experiment 1 - Terminal:



COM4 Baud: 9600 ASCII

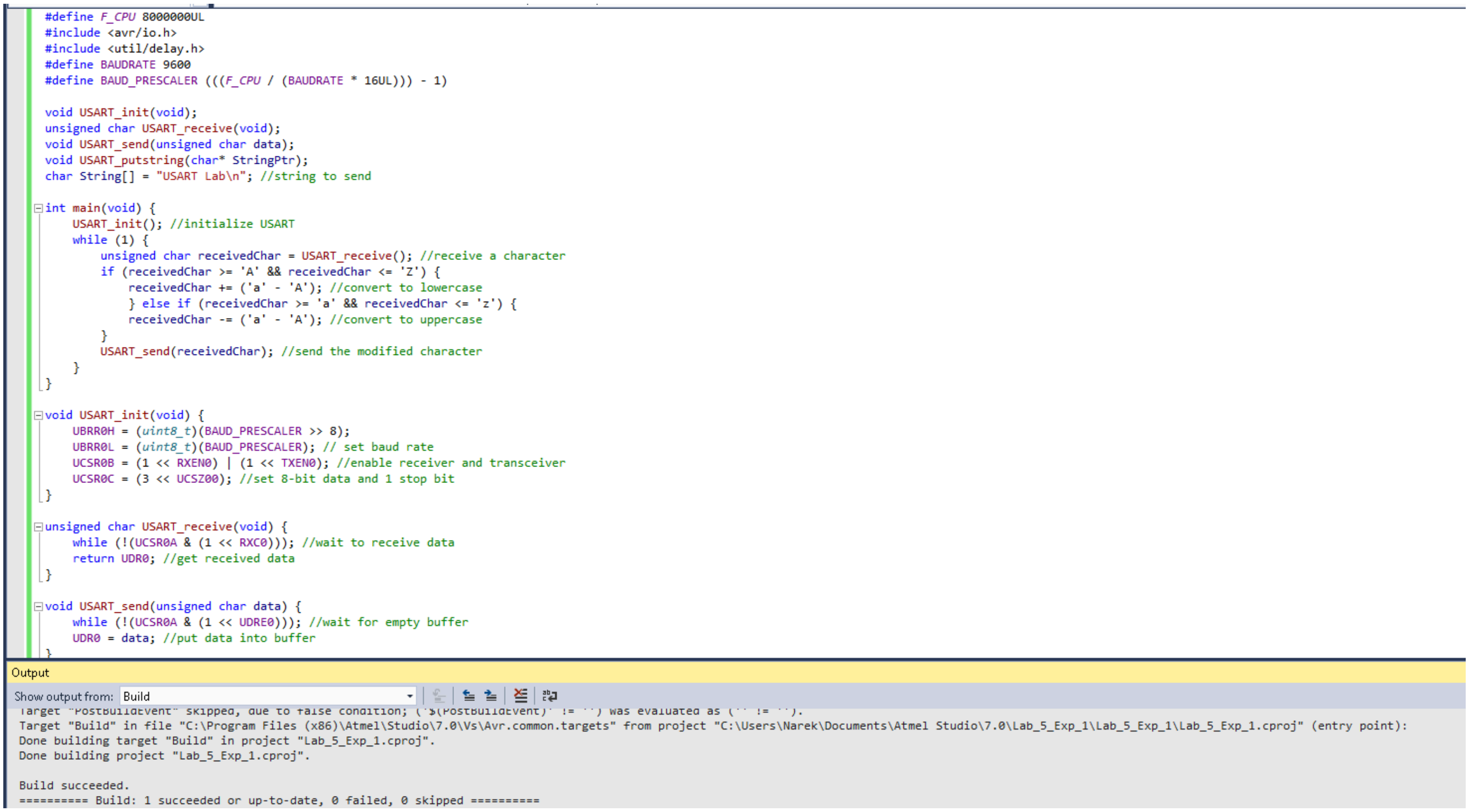
```
USART Lab
USART Lab
USART Lab
```

```
#define F_CPU 8000000UL
#include <avr/io.h>
#include <util/delay.h>
#define BAUDRATE 9600
#define BAUD_PRESCALER (((F_CPU / (BAUDRATE * 16UL

void USART_init(void);
unsigned char USART_receive(void);
void USART_send( unsigned char data);
void USART_putstring(char* StringPtr);
char String[] = "USART Lab\n"; //string to send

int main(void) {
    USART_init();//initialize USART
    while (1) {
        USART_putstring(String); //Pass the string
        _delay_ms(5000); //delay for 5 seconds
    }
}
void USART_init(void) {
    UBRR0H = (uint8_t)(BAUD_PRESCALER >> 8);
    UBRR0L = (uint8_t)(BAUD_PRESCALER); // set bau
    UCSR0B = (1 << RXEN0) | (1 << TXEN0); //enable
    UCSR0C = (3 << UCSZ00); //set 8-bit data and 1
}

unsigned char USART_receive(void) {
    while (!(UCSR0A & (1 << RXC0))); //wait to receive data
    return UDR0; //get received data
```

Experiment 1 - Video:

https://www.loom.com/share/4d831e6976fe47f69f29962e384b9cac?sid=b9e28956-85ab-482b-8ed7-fd6b8a4dcd00

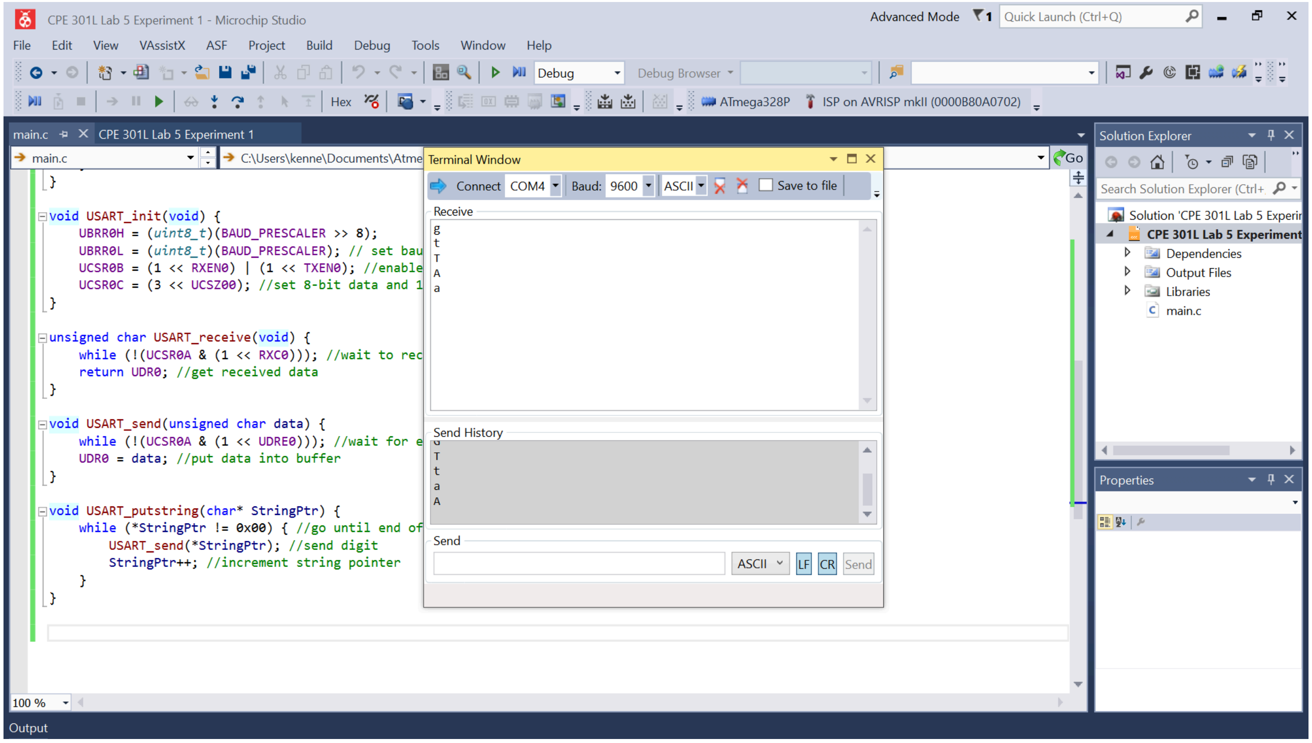
Experiment 2 - Code:



```
#define F_CPU 8000000UL
#include <avr/io.h>
#include <util/delay.h>
#define BAUDRATE 9600
#define BAUD_PRESCALER (((F_CPU / (BAUDRATE * 16UL))) - 1)

void USART_init(void);
unsigned char USART_receive(void);
void USART_send(unsigned char data);
void USART_putstring(char* StringPtr);
char String[] = "USART Lab\n"; //string to send

int main(void) {
    USART_init(); //initialize USART
    while (1) {
        unsigned char receivedChar = USART_receive(); //receive a character
        if (receivedChar >= 'A' && receivedChar <= 'Z') {
            receivedChar += ('a' - 'A'); //convert to lowercase
        } else if (receivedChar >= 'a' && receivedChar <= 'z') {
            receivedChar -= ('a' - 'A'); //convert to uppercase
        }
        USART_send(receivedChar); //send the modified character
    }
}

void USART_init(void) {
    UBRR0H = (uint8_t)(BAUD_PRESCALER >> 8);
    UBRR0L = (uint8_t)(BAUD_PRESCALER); // set baud rate
    UCSR0B = (1 << RXEN0) | (1 << TXEN0); //enable receiver and transceiver
    UCSR0C = (3 << UCSZ00); //set 8-bit data and 1 stop bit
}

unsigned char USART_receive(void) {
    while (!(UCSR0A & (1 << RXC0))); //wait to receive data
    return UDR0; //get received data
}

void USART_send(unsigned char data) {
    while (!(UCSR0A & (1 << UDRE0))); //wait for empty buffer
    UDR0 = data; //put data into buffer
}
```

```
Target "PostBuildEvent" skipped, due to false condition; ('$(PostBuildEvent)' != '') was evaluated as ('' != '').
Target "Build" in file "C:\Program Files (x86)\Atmel\Studio\7.0\Vs\Avr.common.targets" from project "C:\Users\Narek\Documents\Atmel Studio\7.0\Lab_5_Exp_1\Lab_5_Exp_1\Lab_5_Exp_1.cproj" (entry point):
Done building target "Build" in project "Lab_5_Exp_1.cproj".
Done building project "Lab_5_Exp_1.cproj".

Build succeeded.
========== Build: 1 succeeded or up-to-date, 0 failed, 0 skipped ==========
```

Experiment 2 - Terminal:



CPE 301L Lab 5 Experiment 1 - Microchip Studio
Advanced Mode
Quick Launch (Ctrl+Q)
File Edit View VAssistX ASF Project Build Debug Tools Window Help
Debug
Debug Browser
Hex
ATmega328P
ISP on AVRISP mkII (0000B80A0702)
main.c
CPE 301L Lab 5 Experiment 1
main.c
C:\Users\kenne\Documents\Atme
Go
Solution Explorer
Search Solution Explorer (Ctrl+
Solution 'CPE 301L Lab 5 Experir
CPE 301L Lab 5 Experiment
Dependencies
Output Files
Libraries
main.c
Properties
100 %
Output

```
}

void USART_init(void) {
    UBRR0H = (uint8_t)(BAUD_PRESCALER >> 8);
    UBRR0L = (uint8_t)(BAUD_PRESCALER); // set bau
    UCSR0B = (1 << RXEN0) | (1 << TXEN0); //enable
    UCSR0C = (3 << UCSZ00); //set 8-bit data and 1
}

unsigned char USART_receive(void) {
    while (!(UCSR0A & (1 << RXC0))); //wait to rec
    return UDR0; //get received data
}

void USART_send(unsigned char data) {
    while (!(UCSR0A & (1 << UDRE0))); //wait for e
    UDR0 = data; //put data into buffer
}

void USART_putstring(char* StringPtr) {
    while (*StringPtr != 0x00) { //go until end of
        USART_send(*StringPtr); //send digit
        StringPtr++; //increment string pointer
    }
}
```

Terminal Window

Connect COM4 Baud: 9600 ASCII □ Save to file

Receive

```
g
t
T
A
a
```

Send History

```
g
T
t
a
A
```

Send

ASCII LF CR Send

Experiment 2 - Video:
https://www.loom.com/share/3193e8c83273435aacc9cdde2a998c0f?sid=6d5c4086-7f47-45a6-814e-1320967d1c99

4. Questions and Answers

1 - Baud Rate: Baud rate is a variable that represents the number of signal changes per second. The baud rate may be higher or lower than the bit rate. Bit rate (bits per second) is the measure of how many bits are transmitted per second.

2 - Asynchronous vs Synchronous Transmission: Asynchronous and synchronous are two types of data transmission. Asynchronous uses one byte at a time with start and stop bits. Synchronous sends data as a continuous stream of bits. Synchronous also has a shared clock signal between the send and receiver.

3 - Parity Bit: The parity bit is important in USART because it detects errors that may occur when data is transmitted. It checks the total number of bits that are 1 (high) to provide simple error detection.

5. Conclusions

We were able to transmit data using serial communication with our ATmega328p board, DB-9 connector, and RS232 chip. In the experiments, we connected our microcontroller to the 232 chip and Microchip Studio software on our computer. We faced quite a few challenges in this lab. Firstly, we were using the wrong cable to begin. After switching to a serial-to-USB cable, we were able to successfully connect to the correct serial port (COM 3). Even after making this switch, our code wasn’t working. We discovered this was due to our RS232 chip being faulty because one of the pins was broken. After replacing our RS232, we were able to get the experiments working successfully.