

SJSU EE138
Introduction to Embedded Control System Design
Lab 7: Receiver & Transmitter (UART)

Readings:

Lecture Chapter 12: UART

TI MSP432 Technical Reference Manual

- Section 2.4.3: NVIC registers
- Section 24: eUSCI – UART Mode
-

TI MSP432P401r Data Sheet

- Section 6.7: Interrupts
- Section 6.12: Input/Output Diagrams

Materials Required:

USB-RS232 Converter Cable

Software/Drivers

- UART Terminal (Termite)
http://www.compuphase.com/software_termite.htm
- drivers for the USB-RS232/LC
<http://www.ftdichip.com/Drivers/VCP.htm>

Lab Description:

In this Lab, students will be required to transmit and receive a message through a USB-RS232 converter cable using UART communication. The lab will allow students to send and receive a message to a terminal window outside of CCS using ports **P3.3** (TX) and **P3.2** (RX). This will be done through MSP432's UART **interrupt method**, not the polling method. Upon completion, students will be able to apply the use of the terminal window for debugging, observing behaviors, user input, and countless other applications.

Theory of Operation:

On the MSP432, only eUSCI_A supports UART mode. The module we will be using in this lab is eUSCI_A2. P3.3 and P3.2 will be UCA2TXD and UCA2RXD, respectively, and their functional group should be configured according to MSP432401r data sheet. The steps to enable UART module is taken from technical reference manual:

NOTE: Initializing or reconfiguring the eUSCI_A module

The recommended eUSCI_A initialization/reconfiguration process is:

1. Set UCSWRST.
 2. Initialize all eUSCI_A registers with UCSWRST = 1 (including UCAxCTL1).
 3. Configure ports.
 4. Clear UCSWRST with software.
 5. Enable interrupts (optional) with UCRXIE or UCTXIE.
-

Figure 6.1: Step to configure UART

Note that even though the following figure says step number 5 is optional, in this lab we will be using interrupts rather than polling, so this step is required. UART transmission will start when data is written into the transmit buffer (UCA2TXBUF), and the receive interrupt flag UCRXIFG can be used to know when UART module has received a complete character, indicating a full RX buffer. There should also be a line to service the TX flag in your interrupt routine, because as soon as the UART module is enabled, it will trigger the TX flag because TX buffer is empty.

Baud Rate:

Upon initialization of the UART registers, one of the most important registers to fill out and calculate is the baud rate. For a given BRCLK clock source, the baud rate can be set using the steps below:

NOTE: Baud-rate settings quick set up

To calculate the correct settings for the baud-rate generation, perform these steps:

1. Calculate $N = f_{BRCLK} / \text{baud rate}$ [if $N > 16$ continue with step 3, otherwise with step 2]
 2. $OS16 = 0$, $UCBRx = \text{INT}(N)$ [continue with step 4]
 3. $OS16 = 1$, $UCBRx = \text{INT}(N/16)$, $UCBRFx = \text{INT}([(N/16) - \text{INT}(N/16)] \times 16)$
 4. $UCBRs_x$ can be found by looking up the fractional part of N ($= N - \text{INT}(N)$) in table [Table 24-4](#)
 5. If $OS16 = 0$ was chosen, a detailed error calculation is recommended to be performed
-

Figure 6.2: Terminate Terminal

Keep in mind that the baud rate needs to match the baud rate from the terminal window.

Termite (software)

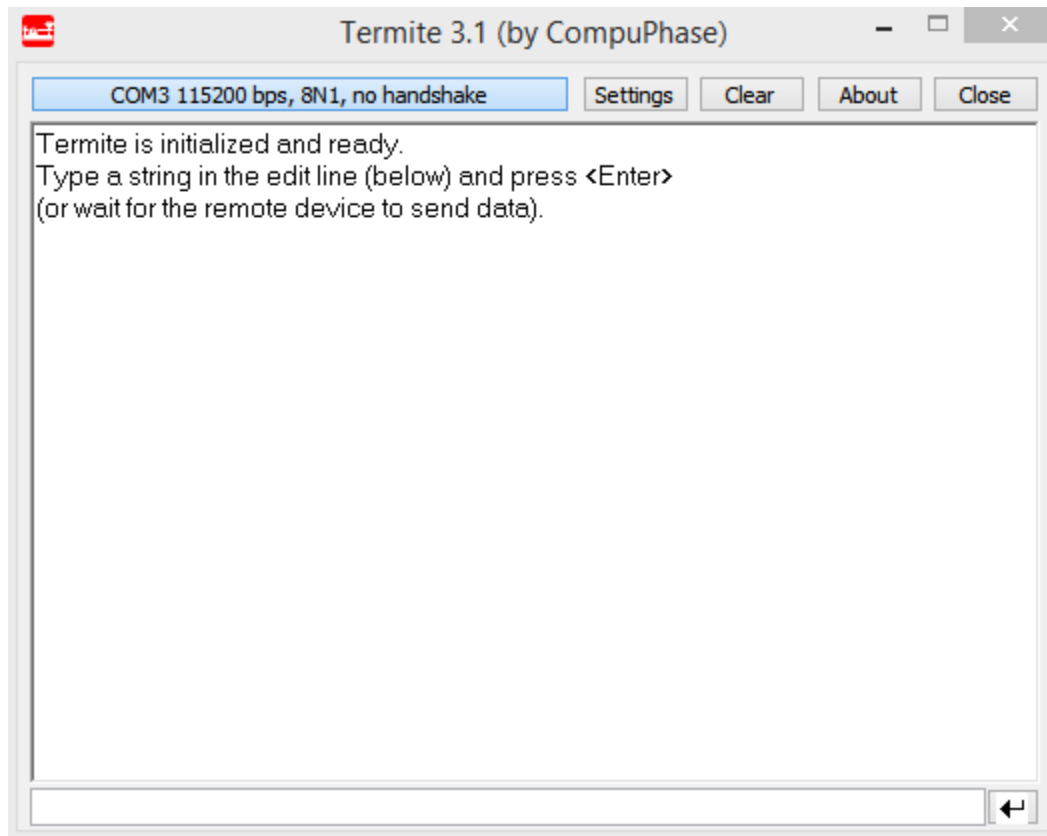


Figure 6.3: Termite Terminal

In the lab, you will be required to install and/or utilize the Termite software for your UART communication. Figure 6.3 show the terminal window used in the lab.

On the top left, there is a button that displays the communication port, baud rate, number of bits, stop bit, and if there is handshaking. This button is used to connect and disconnect the communication between your MSP432 and the terminal software.

The bottom of the terminal allows for user input. The user can type letters, words, numbers, and/or symbols into the terminal for the purpose of transmitting bits to the MSP432. Please note that pressing the "return" key does not count as another byte (ASCII) but rather just begins the transmission of data.

To edit the terminal window to a desired specification, the setting button next to the connection button should be pressed. Upon pressing the setting button, another window will appear, displayed in figure 6.4.

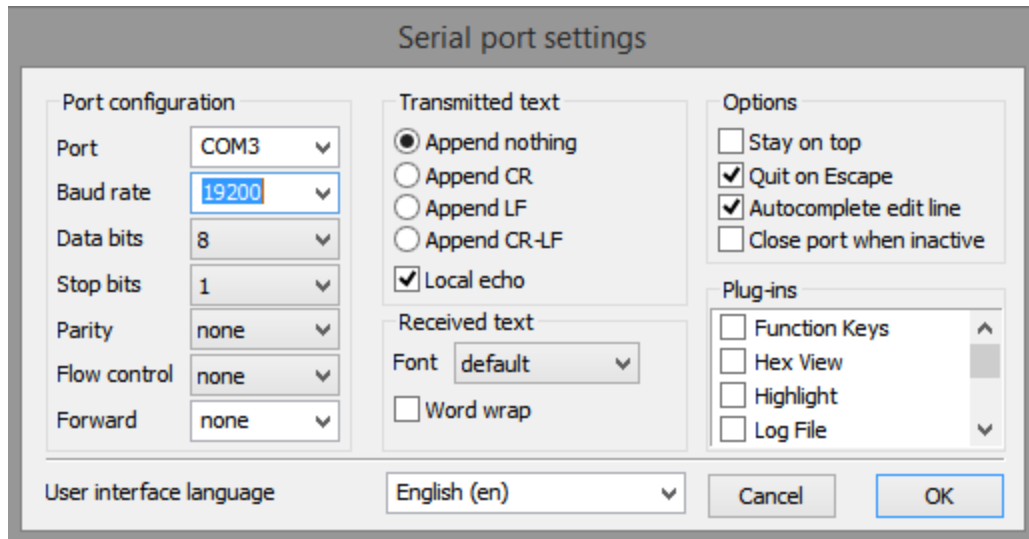


Figure 6.4: Termit Terminal's settings

There are multiple configurations and options given in the settings of the Termit software. One of the most important configurations is the port configuration. The port COM number will be based on which USB port was connected to the PC. When a USB is connected to the PC, there will be an auto-detection on which COM ports are available.

The other important configurations are below the "Port" configuration. The baud rate, data bits, and others should be set appropriately. Check the "local echo" option in order to see what you typed onto the terminal.

Peripheral and Coding:

Address	-	MSP432 Syntax Code	

0x40004C00	-	DIO_BASE	//definition address for GPIO functionality
offset 0x0020	-	P3	
For P3:			
offset 0x04	-	DIR	
offset 0x0A	-	SEL0	
offset 0x0C	-	SEL1	
0x40001800	-	eUSCI_A2	//serial communication interface - UART mode
offset 0x00	-	UCA2CTLW0	// control word register 0
offset 0x06	-	UCA2BRW	// baud rate control word register
offset 0x08	-	UCA2MCTLW	// modulation control word register
offset 0x0C	-	UCA2RXBUF	// receive buffer
offset 0x0E	-	UCA2TXBUF	// transmit buffer
offset 0x1A	-	UCA2IE	// interrupt enable register
offset 0x1C	-	UCA2IFG	// interrupt flag register
offset 0x1E	-	UCA2IV	// interrupt vector

Required Tasks:

***please comment your code**

Implement the UART using **interrupts** not by polling

Transmit a message using UART communication. Any baud rate is okay.

- Transmit A 4-digit number entered on the keypad from the MSP432 to the Terminate software.
- Use the keypad's 'D' button to transmit the number entered.
- You do not have to display the entered number on the PCB board during this task. Output can simply be on the Terminate window only.
- The number must be converted to an ASCII array of characters before sending to the Terminate program

Example:

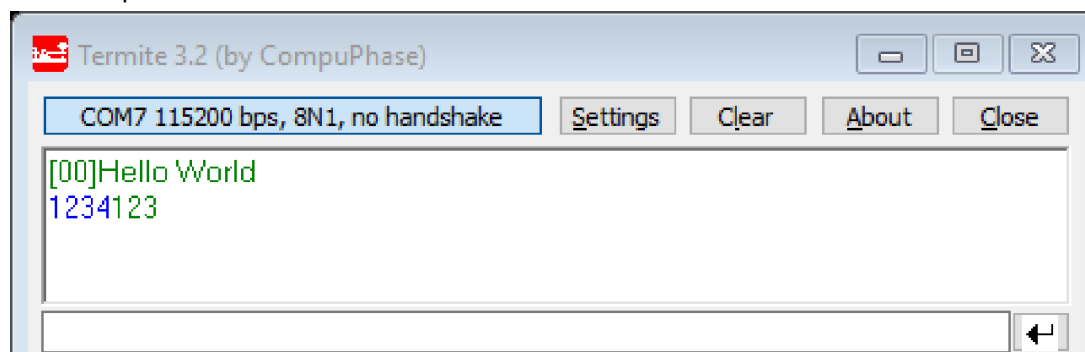


Figure 6.5: Terminate Terminal Example

Blue is what you typed (echo) and Green is the message sent from the MSP432.

Receive a message using UART communication.

- Utilize the code from Task 1 and modify it so that a number can be transmitted from the Terminate software to the 7-segment display. You will need to create a function that receives the data similar to the write() function.
- Write a function that converts an ASCII character array to an integer.
- Note that the program should also be able to transmit message at this point.