CSCI130: Mobile Robotic Devices Embedded Systems
Lab 3: USART Communication

Lab Overview:
This lab is an exercise in which participants use the Romeo Microcontroller kit and the IO Trainer Kit to implement an embedded system that can perform various USART and digital input functions.

Pre lab:
For this lab, participants will need a terminal program that interface with serial port. Windows XP systems will have HyperTerminal. For participants using Windows 7 and newer systems, there is a free, lightweight program called PuTTy (http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html). Another program with more feature would be RealTerm (http://sourceforge.net/projects/realterm/). There are many more. You can choose which ever ones you like best.

When using either HyperTerminal or PuTTy, the COM# and baud rate must be set according to your Romeo board and program.

Lab 3A
   i.   Write a C program that waits for input from the serial terminal, when an input is received, transmit the next character on the ASCII table on a new line. For example

      Input: A

      Output: B

      Input: s

      Output: t

   ii.   The program must have 3 functions, initUART, transmitByte, receiveByte from the lecture notes.

  iii.   The participant may choose any baud rate, however the initUART function must run the USART in double speed mode.

  iv.   Name the file lab3a.c

Lab 3B
   i.   Connect 2 of the 4 momentary switches (S1-S4) on the IO Trainer Board to 2 pins on the Rome Board

   ii.   Write a C program that replicate the functions of Lab 2B but use the serial terminal as a display. Like Lab 2B, the program will read two momentary switches and used them to increase or decrease a count by 1. The count can go from -9999 to 9999. The decimal representation of the count will be output on the serial command. The program will only display a new number when there is an increment or decrement in the count.

      The program must have a function defined as:

      `void printDec(int num)`

      The function takes in one integer as a parameter. The function would output the ASCII representation of the number through the UART. Both negative and positive number has to be presented properly.

iii. Create a connection chart in the comments above the program like in Lab 2B.

iv. Name the file lab3b.c

Lab 3C

i. Connect the 4 DIP switches (SW1_1-SW1_4) on the IO Trainer Board to 4 pins on the Romeo Board

v. Write a C program that read in the state of the 4 DIP switches as a binary number where SW1_1 is the most significant bit and SW1_4 is the least significant bit. The program will display both a binary and decimal representation of the state of the DIP switches. For example:

SW1_1 = on;   SW1_2 = on;   SW1_3 = off;   SW1_4 = on;

Serial Output:
00001101 13

The program will only display a new number when there is the state of the DIP switches have been changed.

ii. Include the printDec function from 3B. Add a new function defined as:

```
void printByte(uint8_t num)
```

The function takes in 8-bit unsigned integer as a parameter. The function would output the ASCII representation of the 8-bit number in binary through the UART. The output should have leading zeros.

iii. Name the file lab3c.c

Submission

Participant must submit a video demonstrating that the system works as described in 3A, 3B, and 3C. The video can be uploaded to the D2L or uploaded to Youtube (include link). Additionally, 3 files should be submitted through D2L: lab3a.c, lab3b.c, lab3c.c.