

**Task 3**                      **Due: 10/18/2008****Task Purpose:**      **Develop serial I/O routines**

The microcontroller board has an RS-232 interface. This task involves adding library routines to the EESDlib.c code to use this interface.

1. Based on the microcontroller board schematic, list the microcontroller signals involved with the serial port and describe the function of each signal.
2. One of the signals involved with the serial port is an enable signal to the MAX3323. Based on the MAX3323 spec, determine how the microcontroller needs to set this signal to enable serial communications through the serial port.
3. For the signal identified in part 2, list all the registers and bits with those registers that need to be set to use this bit to enable the MAX3323. What are the appropriate settings for these bits?
4. Write a routine to initialize the serial port. This should include setting the bits discussed in part 3, enabling the USART on the microcontroller, and setting the baud rate. (Use 57600 for the baud rate. Note that the correct setting of some of the bits and registers will depend on the oscillator frequency.)
5. Write routines `putc()` and `getc()` that will send a character from the microcontroller to the serial port and receive a character from the serial port. Appropriate function prototypes for these functions are:

```
void putc(char value); // send ascii character to terminal
char getc(void);      // get ascii character from terminal
```

6. Write a main program that receives characters from HyperTerminal, echoes them back to the HyperTerminal screen, and also displays them on the LCD screen. (This is the same function that was in the original code in your microcontroller.)
7. Write a function that is the equivalent of the LCD routine `LCD_printf()`, but that sends a character string to the terminal via the serial port.
8. Write a function that will print on the HyperTerminal screen the value of an unsigned short variable. Note that this is the equivalent of the routine (found in the set of library routines you were provided) `void LCD_dec(unsigned short dat)`.
9. You will find three different routines in the library code you were provided that are called `LCD_dec()`. What's up with that?
10. Write a function that will read an unsigned short (16 bit) number from the terminal and store it in a variable in the microcontroller. The number should be stored in

binary. The function should be able to accept a variable number to digits, terminated by the enter key (which sends a carriage return character). NOTE: What you will be getting from the terminal is the ASCII for the digits of the number. Verify that your program works by using `LCD_dec()` to display the number on the LCD screen.

**Report:**

In addition to the answers to the questions posed in this task, please include a listing of the software you have written as part of this task in your task report.