EET4340

Lab 1

The USART

Use usart.c as the starting point for this lab. Do as many of these as you can. Make your best attempt at the code even if you don’t think it is entirely correct.

1. Make a project with usart.c Set the simulator as the debug tool. Make sure it compiles. Enable UART1 in the simulator (we’ll go over this during the meeting) and view the output in the UART1 output window.
2. Calculate the appropriate value to set the baud rate to 19200 baud. Remember that we are assuming an Fosc of 32 MHz. Modify the code to reflect this new value. This will not change the simulator behavior.
3. Modify the firmware so the Hello World string is only transmitted when the button is pressed. There is an empty section of code for detecting the button in the main loop. You can try simulating the button if you want. (Hint – There is a bit that enables/disables transmitting.)
4. Modify the firmware so the received bytes (from the PC if we were connected) are echoed on the terminal. That means that as each character is received by the PIC it should be transmitted back to the PC. You will want to comment out the printf(“Hello… first so you can see the echoed text. I’ve included a stimulus file to get you started. I’ll cover this during the meeting as well.
5. Create a simple command processor. When the PIC receives a character of your choice is should execute a simple command. For example you could set it so an ‘a’ transmits a message to the PC and a ‘b’ transmits a different message or perhaps the current rxCount. You should have at least two or three commands. Use or modify the stimulus file.