Lab-6: Serial Interface

In this lab, we shall learn how serial interfaces work with micro-controllers. In this set of experiments, we shall use asynchronous data transfer. The next lab will look at some synchronous data transfer protocols.

Please read the mini-tutorial on serial interfacing uploaded to moodle. (Notice that I say **read** and not browse! The tutorial contains many examples which will be important for these experiments and you should work those out before you begin.)

Implementations in this lab will use assembly language.

Homework

We want to use T1 in auto-reload 8 bit mode (mode 2). This mode puts no software load on the processor. All actions such as re-loading the timer etc. are done automatically in hardware. Therefore once the timer is configured, it goes merrily along, providing its output to the serial communications block.

Write a program which does the following:

- Configure the serial port for 8 bit data + Even Parity (11 bit frame) with baud rate adjusted to 1200 using T1. Serial port interrupts are to be enabled.
- Write an interrupt service routine for serial communication, which clears TF and transmits the character 'A' whenever the serial port interrupt occurs and TF is found set. Parity bit should not be hard coded for 'A'. It should be evaluated by adding 0 to the character being sent and checking the parity flag. (Then you will be able to use this routine for any character, not just 'A'). After writing the character, the ISR should increment a global counter (initialized by the main program) and create a software delay of about 10 ms. (This will be useful for triggering the oscilloscope properly when you observe the output during the lab).

  The main program initializes the global counter to 0. After that it goes in an endless loop, checking the counter. Every time it finds that the counter has reached 50 (decimal), it should reset the counter to 0 and toggle an on-board LED.

Assemble the program, download it to your kit and run it. Find the rate at which the LED is blinking by timing it for 10 cycles.

Lab Assignments

Before beginning the lab assignments, show the output of your homework problem to the TAs. Observe the frame being transmitted on an oscilloscope, and identify all bits of the frame on the waveform.

Write a program so that two kits can be connected to each other using serial communication. Transmit data line of one should be connected to receive data line of the other and *vice versa*. Each kit should send a 16 character string to the other. Each kit should display the string it received from the other on the first line of its LCD display and the string it sent on the second line.

How will you ensure that the other kit is listening when the first kit starts transmitting?

When the program on either kit starts, it should configure the serial port, enable interrupts and read the slide switches. It should then enter a loop which reads the slide switches and compares the recently read value with the previous one. It should quit the loop only when the value changes. Only when it quits the loop, it should transmit its 16 bit data character by character. Reception is in interrupt mode and is enabled before entering the wait loop. Thus both kits are ready to receive from the beginning but transmit only when permitted to do so by changing slide switches.