

ECGR6185, Spring 2010: Lab 4

Serial I/O with Timers and Interrupts

Learning Objectives

You are to write a program that runs on your QSK62P board that uses timers and interrupts to transmit serial data.

Laboratory Assignments

In this lab you will be programming the timers and using interrupts to perform serial communications. This lab will use the on-board UART to communicate between your board and a PC. The LCD can be used to display debugging information. This lab must be demonstrated to the TA using a PC to generate test data. The program must rely only on timers and interrupts. Note: Character transmission to the PC must be done using interrupts. You do not need to create a circular queue like the example in class.

Requirements

- Req. 1. The code generated is written in C for the QSK62P or QSK62P Plus.
- Req. 2. The code is well commented and easy to follow.
- Req. 3. You may not use the BNS functions.
- Req. 4. The Renesas board will be connected via the RS-232C port transmit and ground pins to a PC.
- Req. 5. You do not need to use the Renesas board LCD, but it may be helpful while you debug your code.
- Req. 6. The Renesas board will create an ASCII string representing a "stopwatch" of the format MM:SS<cr><lf> (MM means minutes, SS means seconds, followed by carriage return and line feed).
- Req. 7. Toggling the Renesas board switch S1 must pause and resume the time count.
- Req. 8. Pressing Renesas board switch S2 must reset the stopwatch to 00:00 and cause it to pause. Also send the number 00:00<cr><lf> to the PC.
- Req. 9. Generate an interrupt on the Renesas board every second and send ASCII characters to the PC with a time that has incremented by one second. These ASCII characters must display on the PC.
- Req. 10. Every 60 seconds send ASCII characters to the PC with a time that has incremented by one minute. These ASCII characters must display on the PC.
- Req. 11. Use another interrupt to send each character of the ASCII string to the PC using the RS 232 interface.
- Req. 12. Make sure the serial interrupt is set up in such a way that each line on the screen (PC) must have the updated value of the stop watch.
- Req. 13. The serial communications should operate at 9600 baud, odd parity, 8 data bits and one stop bit.
- Req. 14. The code should be as compact as possible.

Sample Test Procedure

Note: Hand the lab checkout sheet to the TA when you demonstrate your program. You will turn in your code and report electronically.

The result on the screen should be

00:01

00:02

.

.

00:59

01:00

.

.

.

59:59

00:00

.

.

Lab Report

Include in your lab report observations and procedure like the following:

The general learning objectives of this lab were . . .

The general steps needed to complete this lab were . . .

Some detailed steps to complete this lab were

1. *Step one*

2. *Step two*

3. *. . . .*

Some important observations while completing/testing this lab were . . .

In this lab we learned

Create a single pdf containing:

1. Your lab report
2. Your code (no need to include the sect30.inc, ncrct0.a30, LCD, or any .h files). Include all c files that have code that you wrote (but if you are smart, this should be one small file so that the code size is small). Ensure you use an 8 or 9 courier font so that most lines of code take one line of text.
3. The full map file. Ensure you use an 8 or 9 courier font so that most lines take one line of text.

Upload this pdf to moodle. Name the file xxxxxxxx_yyyyyyy_lab4.pdf, where xxxxxxxx is the last name of one lab partner, and yyyyyyy is the last name of the other lab partner.

ALSO, upload your C file as a separate file

FAILURE TO FOLLOW THESE SIMPLE INSTRUCTIONS COULD RESULT IN THE LOSS OF POINTS.