

## ECGR 4101/5101, Fall 2006: Lab 7

### Timers and Interrupts

### Learning Objectives

This lab will introduce you to using Round Robin Scheduling on the 30626P-SKP board to light LEDs and send text to a PC via a serial cable.

### General Information

The general steps for this lab are:

1. Generate a new project. Name your new project Lab7.
2. Modify the main.c file and include the appropriate files. Include commenting along the way.
3. Program the lab. Don't forget the necessary include files to get the correct functionality.
4. Compile the code into an .x30 file, and load onto the board.
5. Test the program and repeat steps 3 and 4 until the program works as required.
6. Write your lab report.
7. Demonstrate for a TA and turn in your report.

### Prelab Activity

You may use the PCs in Woodward 203 or your own PC to do this lab experiment. The machines in Woodward 203 already have the software tools loaded. Answer these questions in the lab report:

1. How many timers should you use for the lab?
2. How many interrupts must you use for the lab?
3. What are the values needed in one or two 16-bit timer SFRs to get a time of 500 microsecond?

### Laboratory Assignments

In this lab you will be programming the timers and using interrupts. The LED's will be used for signaling and the LCD can be used to display debugging information. This lab must be demonstrated to the TA.

1. The program should rely on Round Robin scheduling and use interrupts for the scheduler and UART only.

### Steps

1. Follow the steps given in lab 1 and 2 for generating a new project.
2. Create the main.c file and include the appropriate files.
3. Build your program slowly, testing along the way. Perform compiles and solve each requirement one at a time.

4. Continue to build and test the program until all of the requirements have been met. Did we mention you should write your comments as you progress, not at the end?
5. If you run into problems, use the break point functionality of KD30 to step through the code until you find the problem.
6. Once all the requirements have been met, ensure that everything works.
7. Finish lab write-up and demonstrate for a TA.

## Requirements

- Req. 1 – The code generated is written in C for the SKP16C62P.
- Req. 2 – The code is well commented and easy to follow
- Req. 3 – This board must use Round Robin Scheduling (RRS) for the LEDs and starting the serial communications.
- Req. 4 – Toggle the red LED every 1 second (one second on, one second off).
- Req. 5 – Toggle the yellow LED every 5 seconds.
- Req. 6 – Toggle the green LED every 10 seconds.
- Req. 7 – Schedule a character string transmission to the PC every 1 second via UART0.
- Req. 8 – The string should be of the format "Red=ON Yellow=ON Green=ON "  
or "Red=OFF Yellow=OFF Green=OFF" (or any combination, whatever represents the true state).
- Req. 9 – The RRS timer tick is 500 micro seconds.
- Req. 10 – The RRS Priority, high to low, is Red, Yellow, Green, UART.
- Req. 11 – The UART communications is via queues and interrupts, but the string is built and the first character sent due to RRS.
- Req. 12 – The UART speed and configuration is any speed or configuration.
- Req. 13 – Scoring: The lowest 20% of the lab group results with respect to speed will lose 4 points. The upper 20% of the lab group results with respect to speed will earn 4 points Extra Credit.
- Req. 14 – Scoring: The lowest 20% of the lab group results with respect to code size will lose 4 points. The upper 20% of the lab group results with respect to code size will earn 4 points Extra Credit.

## Lab Report

Turn in a hard copy of the code you wrote and a printout of the map file. Also 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. *. . . .*

*Code generated for this lab...*

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

*In this lab we learned . . . .*