EE4930 Advanced Embedded Systems
Section 011/021, Winter 2022/23

Professor: Kerry R. Widder, Ph.D.
Electrical Engineering and Computer Science Department
Milwaukee School of Engineering

Laboratory 6: "RTOS System – Phase I"

Pre-lab Submittal:	None.
Demonstration and Submittal:	Due by the end of the day of the Week 9 lab session. Late submittals will be penalized per the course syllabus.

Objectives

- Configure a basic RTOS system implementing tasks, events, hardware interrupts, and software interrupts.
- Setup the A/D to run on a periodic basis using the system timer and a software interrupt.
- Output information on the LCD display.

This is an individual lab. Each student must independently complete this assignment. While discussing ideas and potential solutions with your classmates is permitted, sharing code is prohibited.

All software must be written in C using the CCS IDE.

For this laboratory, you will interface a voltage signal from a potentiometer to the microcontroller board using an analog input and the A/D converter and write software that displays the conversion information on the LCD display. These actions will be implemented using a Real Time Operating System (RTOS).

Specifications:

- The A/D input will be the voltage from a simple voltage divider circuit that uses a potentiometer (pot) connected to 3.3 V and ground, with the pot wiper connected to one of the A/D inputs.
- This A/D input will represent a setpoint temperature for a temperature control device. The input needs to be scaled to represent user selectable values in the range of 50 – 90 deg. F. Put the scaled reading on the display as follows:

  `"Setpt: xx F"`

  Where xx is replaced by the converted temperature value. Your temperature output will be a rounded version of the actual temperature (Integer value).
- Update the setpoint value on the display only when the value changes.
- Setup two tasks, one for managing the LCD display, and one for managing inputs. Choose priority levels for these tasks that make sense in the context of the overall system**. Include a discussion of how you made this decision in your report.**
- Setup the A/D for generating an interrupt when a conversion is complete. Setup the corresponding Hwi to handle this interrupt (be sure to clear the flag) and have it call an Swi to actually do the work of reading the conversion value and performing the scaling to a temperature value. **Note: the RTOS remaps the interrupt numbers – the A/D in the RTOS is interrupt number 40, not number 24**.
- Setup a clock object to run at an appropriate period for reading the setpoint pot. **Include a discussion of how you made this decision in your report.** Consider how often this parameter really needs to be updated when setting the period. The function called each time the clock period expires needs to start another A/D conversion to read the setpoint pot.
- Use an event to let the input task know when a new pot reading is available. The task should block on this event. Once the event occurs, the task should check to see if the value changed. If it did change, post a

different event to indicate to the display task that a new setpoint value is ready for display, and then wait for the next A/D reading event.

- Use an event to let the display task know that a different value is available and needs to be put on the display. After this is done, the display task should wait for the next event. Note: with the RTOS, you may need to delay after you call the LCD init function for it to work.

Demo

Demonstrate correct operation of your RTOS system to the instructor.

Submittal

In your lab report, include:

- Cover page stuff as usual
- Printed copies of your source code
- Explanation of your decision process
    - Task priority assignment
    - Clock period for A/D conversion start
    - Other?
- Circuit diagram (generated electronically)