Lab Assignment 4 **CSE/ECE 474** **Spring 2024**

Prof. Vikram Iyer[1] University of Washington

Learning Objectives

After completing this lab, students will be able to:

1. apply a real-time preemptive operating system: RT-OS
2. Integrate a solution-focused software and hardware project with novel sensors, displays, or actuators.

Due Date 6-Jun-2024

Note: The quarter ends on Friday 6/7 and so we cannot accept late submissions after this.

- Turn In Requirements:
- Lab report and demo.

Part I (4.1): Setting up Free RT-OS

1. Install FreeRT-OS operating system package for the Arduino. (use only the Canvas files linked below, they are tested on Arduino Mega board).
    a. Download the .zip file for the CSE/ECE474 Official FreeRTOS version [HERE].
    b. Unzip the folder and follow instructions to install the Arduino_FreeRTOS library [Instructions]. Use the "manual installation" instructions.

2. Verify operation of the initial "`blink_analogRead474.ino`" example under FreeRTOS. Download the CSE/ECE474 demo app [HERE].
    a. Plug in your analog thumbstick (or any potentiometer) to generate a varying 0-5VDC positive voltage on an analog input pin.
    b. Modify the demo app so that the analog input task uses the same pin you connect to the potentiometer/thumbstick.
    c. Check the baud rate of your Serial Monitor (in the Arduino IDE) to match what is in the demo code.
    d. Run the demo app and verify
        i. Light flashes
        ii. Analog values are printed to the Serial Monitor and they change between 0-1023 when you change the thumbstick/potentiometer.

3. In addition to these add the following tasks. Demonstrate all four running simultaneously.
    a. Task 3: Flash an OFF BOARD LED: ON for 100ms, OFF 200ms.
    b. Task 4: Configure Timer 4 and cause it to play the song you used in Lab 3 on your speaker (Mario theme or song of your choice). This task should play the theme three times (pause 1.5 sec between playbacks) and then stop itself.

Part II (4.2): Project

Using RT-OS,

1. Continue to run Task RT-1, defined above blinking an LED ON for 100ms, OFF 200ms.
2. Do not play sounds or compute FFTs anymore unless required by your project below.
3. Run additional tasks as required to perform a “Project”. This is your chance to get creative and challenge yourself. Figure out physical parts you want to use and functions you want to perform which are “stretch goals” beyond Lab 1 - 3 capabilities. Try to organize your project around an embedded system that solves a real world problem or use case (a game or entertainment device is ok too).

Alternative project ideas: If you have an idea for a project that does not meet the exact requirements below but involves other challenges (e.g. concepts like wireless communication, sensing, control that we have not discussed in class) check with the course staff in office hours.

Project Criteria:

A successful 474 Lab 4 project will meet the following **criteria**:

1. **Perform reliable time and digital I/O functions.** This means everything in your program runs smoothly without timing glitches or other bugs.

2. **Operate with high speed and high CPU load.** In terms of the parameters C, P, D, the projects should have at least one task which needs to run 50 times per second or faster (P <= 20ms). For example, a project which measures soil moisture and waters plants once a day would be too “slow”. However, we will retain the default FreeRTOS “tick time” of 15 ms so you are not required to have operations at faster time scales.

3. **External devices.** Interface at least one device to the Arduino Mega board which we have not used in prior labs.

4. **Make a measurement or control an actuator** (or do both) in a way that (at least conceptually) solves a defined problem. A wide variety of sensors, actuators, and displays are available in the Arduino parts kits (you are NOT limited to the kit if you have other parts you wish to use).

5. **Have a user-interface of some sort.** It does not have to be fancy, but it will not be acceptable to switch/demo functions by plugging jumpers into various breadboard pins or typing into the Arduino IDE serial communication tool. Acceptable examples include switches, keypads, buttons, LEDs, LCD 2-line display, 8x8 LED matrix, 4x7 segment display. You cannot double count a sensor/actuator from requirement 4 above for this.

6. **Demonstrate understanding of how to use FreeRTOS** features. Use a feature of FreeRTOs beyond Queues and Task creation we gave examples for. Eliminate holdover code from SRRI, DDS, or other simplistic schedulers from earlier labs.

Project Examples:

The following examples are listed to clarify the project scope and criteria. Note that we have reserved some points for creativity. If you choose to implement one of these examples instead of coming up with an original project, you will not get full points for project creativity (see rubric).

- Implement a security system with the ultrasonic sensor, PID sensor, PIN keypad, and RFID keyfob. Implement tasks that perform low-latency communications with the RFID and keypad. Allow a 10-second entry delay between tripping the sensor and disarming the system.

- Build a space-invaders game with the thumbstick and 8x8 display. Display # of aliens destroyed on the 7-seg display.

- Make a multimeter using Arduino Mega and the 4x7segment display. Implement as many standard DMM measurements as possible (try capacitance as well as Ohms, Volts, Amps). Measure its accuracy on all ranges.

Note: To encourage creativity some points are reserved for going beyond these examples. (see the Rubric)

CSE 474 Lab 4 Grading Rubric Spring 2024

Point value for each grading factor below will be graded based on both the report and the demo.

Factor	Poor	Fair	Excellent	Max Pts.
Factor I: RTOS (4.1 Demo)	Irregular, glitchy audio, timing issues, etc. 0-4	Incomplete functionality e.g. inoperable 5-8	All functions working with full RTOS integration 9-12	12
Factor II: Project Technical Merit (4.2 Demo, 8 pts each)	Fails 3 or more criteria 0-24	Fails 1-3 criteria 25-40	Fully meets all 6 criteria 48	48
Factor III: Project Creativity	Minimally implements an example project. 0-2	Adds 1 or more features to an example project 3-5	A new idea. 6-10	10
Factor IV: Project Solves a Problem	Implements devices without a defined purpose or problem. 0	User problem solution is impractical but makes some sense. 1-2	“I want one!”, “I need this.” 2-4	4
Factor V: Project Challenge Level	Easy. Similar to an earlier lab. 0	Adds at least one technical challenge beyond Labs 1-3 1-3	Ambitious project. Adds a challenging new peripheral, really pushes the ATMega2560 to the max. 3-6	6
Report: Professional Communications and documentation.	Sloppy, missing, or fails to address required parts or demos. Bad or absent commenting. 0	Basic points are present, but poor conventions, bad grammar, poor organization, little or no documentation. 0-7	Clear, complete, and well organized report. Rich use documentation and explanations 7-10	10
			TOTAL:	90