# ECE 443 – Project 2

## Objective

Use a Change Notice interrupt to unblock a handler task that operates LEDC in a push on/push off manner in response to BTN1 and performs button debouncing.

## Specifications

* Create an Idle Task hook function that indicates the state of BTN1 on LEDA. (i.e., LEDA lit when the button is pressed)
* Create a task that toggles LEDB every millisecond.
* Create a handler task that manages LEDC in response to BTN1 as described in the objective.
* Assert LEDD while the C portion of the Change Notice ISR is active.

## Deliverables

* “Zip-n-ship” your **entire** project **folder** via OneDrive by 10 pm PT on the specified due date, **maintaining** the proper directory structure and using **relative** paths. Make sure your Tracealyzer bin file is located immediately under the project folder so that it is included in the zip. (Use MPLAB to “clean” the project prior to sharing in order to exclude object files from the archive.)
* No Doxygen requirement, but look at Appendix A and B of the Barr Group coding standard under Handouts\Programming\Coding Standards for commenting your **own** header and source files. (Your code, not FreeRTOS or Tracealyzer supplied files.)
* Use Tracealyzer to capture system behavior. Start trace capture immediately, but establish a breakpoint in the “Toggle Led B” task that is reached only **after** the task has run twice with LEDC lit. Add at least two User Event channels to document behavior.
* Submit via Canvas before the next class period a 2-3 page report in PDF format that discusses how you implemented the debouncing and push on/push off behavior, your choice of task priorities, and any challenges you faced. Include in your report a screenshot showing the LED behavior during a button press and release, as well as a screenshot of your Tracealyzer window.