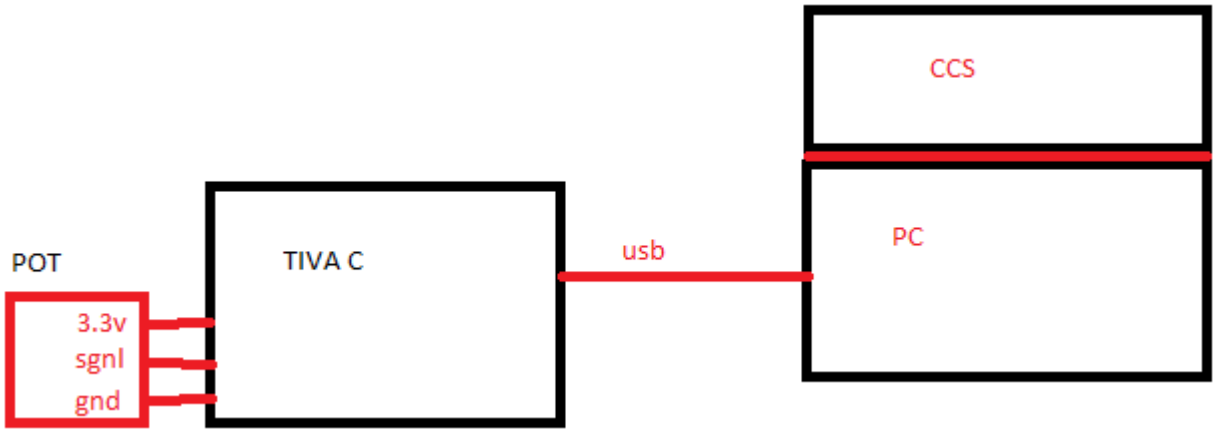
Saul Mendoza
Github root directory: https://github.com/mendos1/submission_L

Date Submitted: **December 13, 2019**

Task : Goal of this assignment is to create three tasks, 1) ADC task, 2) UART display task, and 3) Switch Read task. Each task will be executed in order specified above every 30 ms. Connect a potentiometer to the ADC pin. Use ADC1 CH3. Also initialize a PWM signal to a LED (PF1). Initial value of the PWM dutycycle is set to 0. Create a timer 0/1/2 HWI for every 1 ms, at 10th instance of HWI the task ADC is performed, at 20th instance of HWI the task UART displays the current value ADC in the terminal, and at 30th instance of HWI the task Switch Read is performed to check the status of the SW1/SW2 to update the current value of dutycycle based on the ADC value.

Schematics:



Video Link:
https://www.youtube.com/watch?v=_t3DiTvF3Dg

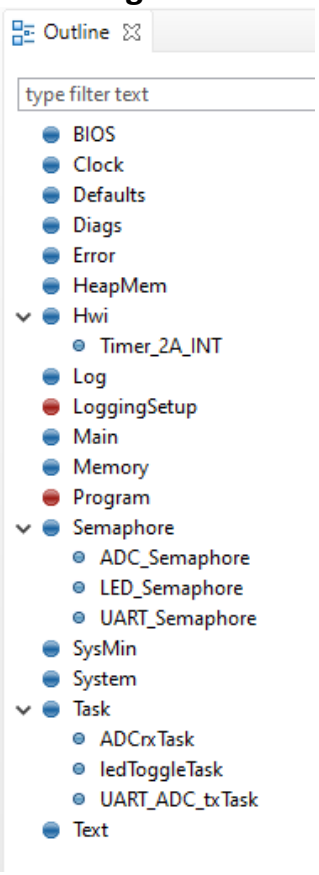
Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

Saul Mendoza
Github root directory: https://github.com/mendos1/submission_L

Implementation: In this assignment I used the CCS cfg tool to create 3 tasks that would read ADC values from the Tiva C Launch Pad, create a UART Tasks that will display the value of the ADC to the terminal using 115200 baud rate, and also an LED read Task that will change the rate of the LED based on the values of the switch. Task one will happen every second and a half starting at the first half second, task two will happen every second and a half starting at the first second, and task three will happen every second and a half starting at the first second and half. There is a counter that will count the instance of the Timer_ISR and once the counter reaches 10 counts it will post the semaphore, after it reaches 20 clicks it will post the UART semaphore, and at 30 clicks it will post the LED semaphore that will blink the LED and then it will reset the counter back to 0 and start the process all over again.

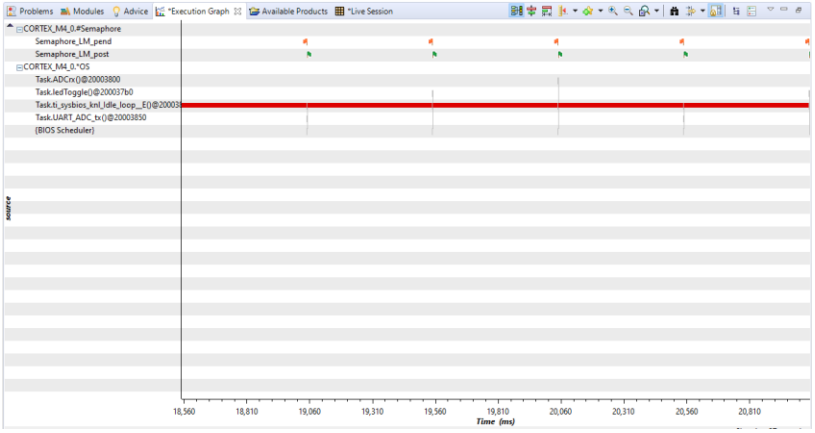
Screenshots:

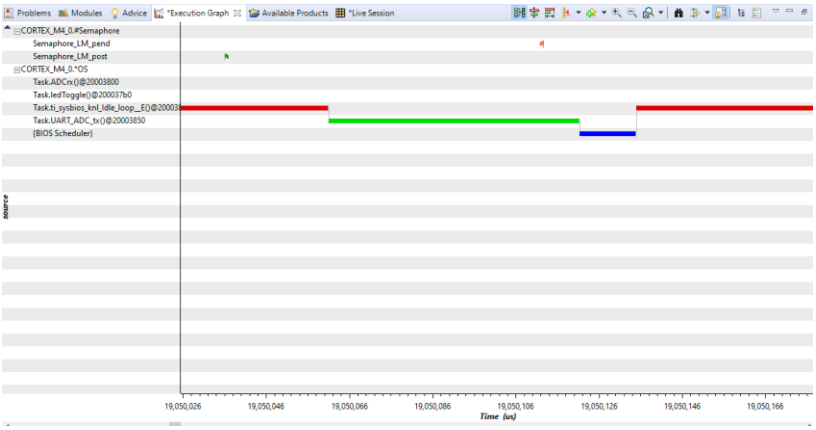
Configuration:

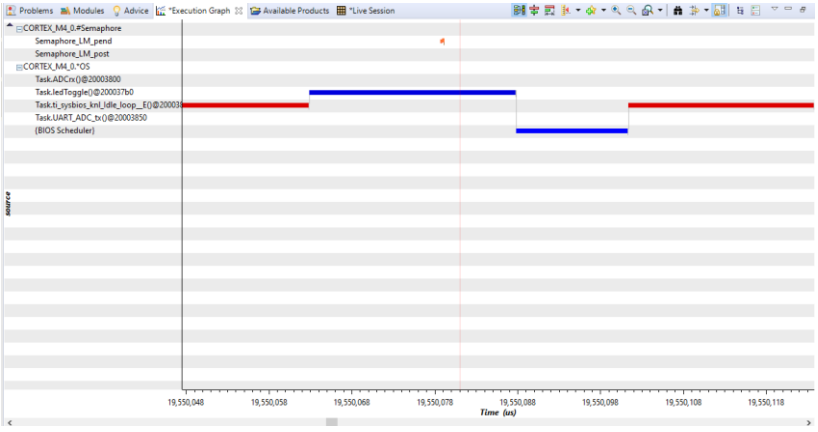


Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

Saul Mendoza
Github root directory: https://github.com/mendos1/submission_L







Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.