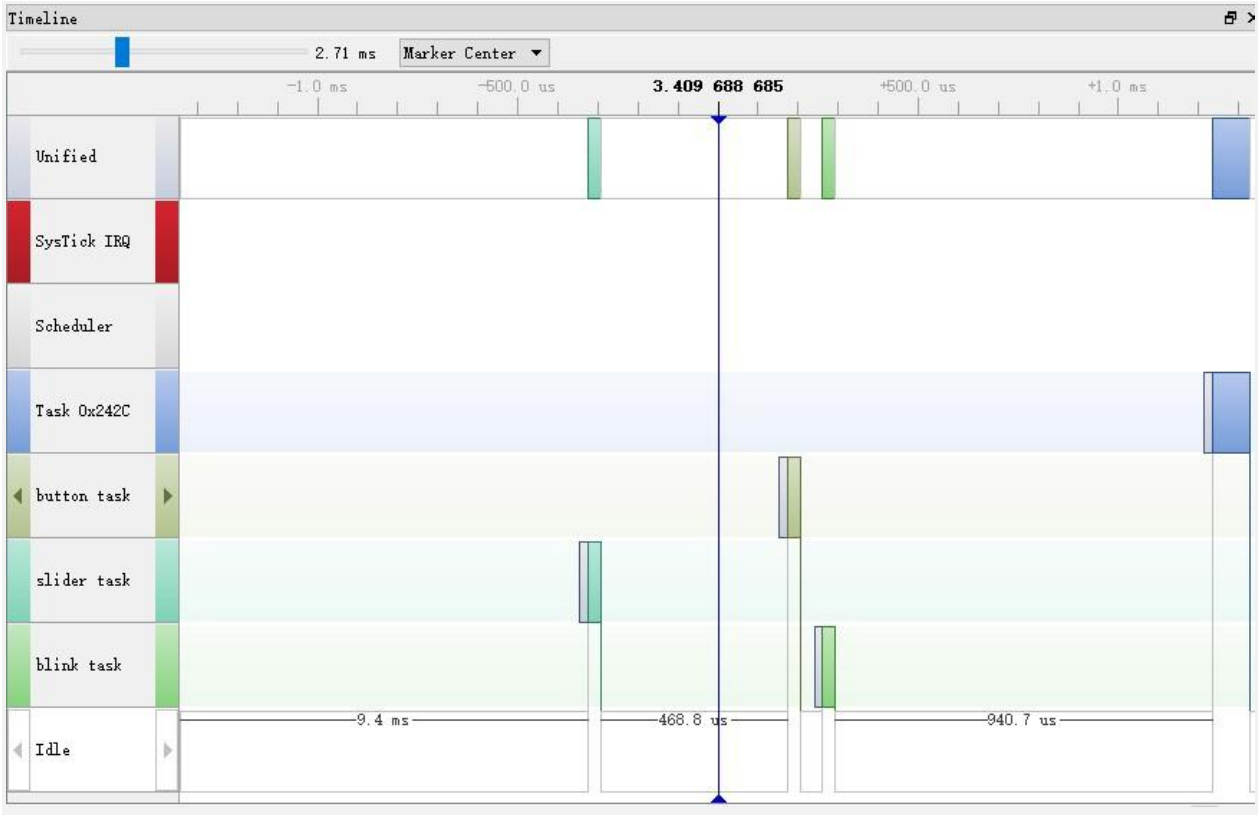
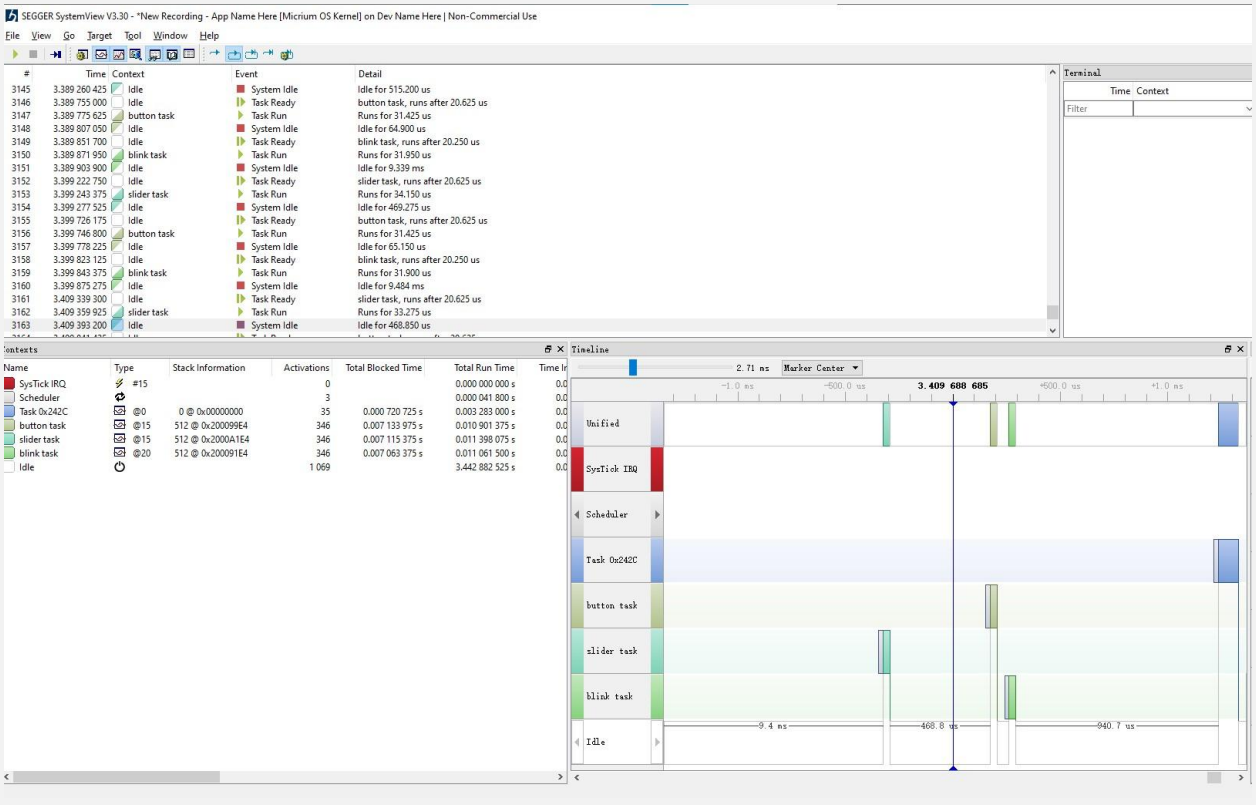
11. Save a screenshot of the SystemView application.



12. Expand the Timeline to zoom in on a section where which each of the application tasks is running. Analyze the events in the event log to observe when each task is made ready and run.



Each task happen in the following sequence: System Idle->Task_Ready(still idle)->Task Run
Each task has a period of idle time, or ready time , before running.
Based on the delay I set in the code as 10, Each task occurs in every 0.01s.

13. What do you observe about the scheduling of each task? How do the priority of other tasks affect scheduling? Does the scheduling of each of the application tasks that you created seem optimal?

Based on the priority I set for these task, slider and button have 15 and blink task has 20, Slider task goes first, then button goes after that and the blink task is the last.
Higher priority tasks will go first. But when bunch of task have same priority, the system will decide the priority for those tasks.
It seems optimal to me since the button and slider decide the behavior of the LED. So, button task and slider task should go before the blink task.

14. Record the frequency and min/max run time of each task

Button task:
Min:0.027850ms
Max:0.033850ms
frequency: 1/0.01 = 100Hz

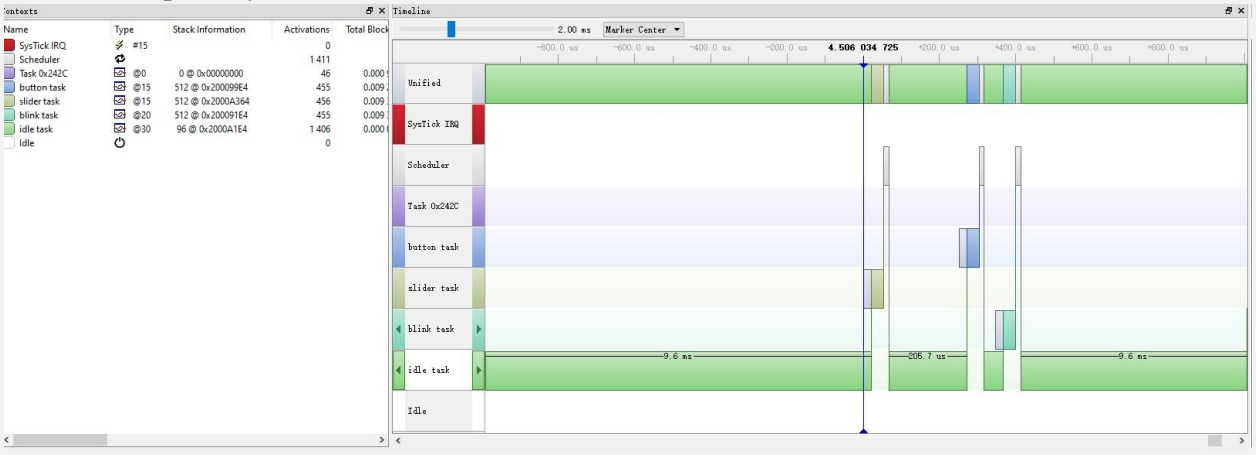
Slider task:

Min:0.032600ms
Max:0.059900ms
frequency: 1/0.01 = 100Hz

Blink task:
Min:0.031900ms
Max:0.034250ms
frequency: 1/0.01 = 100Hz

Average Power Consumed at nominal conditions:
18.45 mW
The average power consumption is higher than the consumption(either polling or interrupt method) in Lab2.

4.
Build and run the application. Verify with SystemView that the new Idle Task is running at the lowest priority of all other tasks.



5.
Start the Energy Profiler and record the average power consumed at nominal conditions (no buttons pressed, slider not touched). Compare this value with the value recorded in Step 1.

The average power consumption in this step is 12.5mW, which is lower than what I have in step 1. This consumption is due to the CPU going into Sleep mode when it is not being used.