Advanced Embedded Software Development

Homework 5 (30 Pts) – Due Wednesday April 10th (midnight)

Revision 2019.04.02

1   Guidelines

For this homework, both Canvas and Github online materials need to be submitted. **Turn in a *.pdf for your report to this assignment to Canvas. Please neatly format submissions with your name, date, homework # and a link to your git repo at the top, and enumerate your answers using headings indicating each problem number, captions explaining images and screen captures, etc.**

2   Reading & Resources TBD

These are reading assignments that are good to complete the homework for the week as well as review materials.

- FreeRTOS API Reference https://www.freertos.org/a00106.html

Resources:

These sections are not required but may help with doing the homework assignments.

- http://processors.wiki.ti.com/index.php/Creating_IoT_Solutions_with_the_TM4C1294XL_Connected_LaunchPad_Workshop
- http://software-dl.ti.com/trainingTTO/trainingTTO_public_sw/Tiva_CLP/CLP_Workbook.pdf
- Link to download required software http://www.ti.com/tool/SW-TM4C

3   Problem Set

[Problem 1 - 30 Pts] Task creation and I2C on TI TIVA
For this problem you need to get FreeRTOS working on the TI TIVA board and be able to read temperature values from the TMP102 sensor used in project 1.

The requirements are

- Create three tasks in FreeRTOS - Temperature Task, LED Task, Logger Task
- The temperature task should read temperature values from the TMP102 sensor at a frequency of 1Hz.
- The temperature value read along with a timestamp should be passed as a structure to the logger task using FreeRTOS IPC mechanism.
- The LED Task should toggle LEDs D1 and D2 on the TIVA board at a frequency of 10Hz, and pass a structure containing a time stamp, toggle count and your name to the Logger task using a FreeRTOS mechanism.
- The logger task should receive the messages from both the above tasks and log these values to UART so that the logs can be viewed on the HOST machine.

- You can use either TIVA hardware timers or FreeRTOS soft timers to get the required timing for the tasks and timestamps.
- The timestamps should have millisecond resolution.

BONUS: Create an alert task which would be notified only when temperature goes above some threshold (preferably using some FreeRTOS API). This task must log some kind of alert statements to UART until temperature is out of the normal range. (3 points)

Include in your report:
1) **A link to your project's git repo.**
2) **A scope shot of the LED pin switching at the required frequency.**
3) **Screenshots of UART terminal on HOST machine showing log output of both tasks**
4) **(BONUS) If attempting, Screenshot of UART terminal showing alert.**

Submit the report to Canvas and your complete CCS Workspace/Project and code to your git repo.