



Project: Light controller using FreeRTOS

Introduction

The PSE Real-Time Module project builds up on the “Bare Bone” project, developed in the previous course module, consisting in replicating its original functionality while using the real-time computation model. More specifically, it will be used the FreeRTOS real-time kernel to implement the micro-controller firmware.

Specifications

Students are required to implement a periodic real-time control firmware composed at the least of the following tasks:

- **Acquisition:** periodic task, with 100 ms period, that samples the light sensor.
- **Decision:** takes as inputs the light sensor value produced by the “Acquisition” task and the user-selected mode and computes the value that should be applied to the light module.
- **Actuation:** takes as input the result of the “Decision” task and sets the micro-controller so that the desired light intensity is attained.

In addition to the control module, above described, the firmware should also include a task to handle the user-interface, i.e., select the operation mode (Manual, Dimmer, Automatic, adjust the intensity, ...).

Appropriate task synchronization and inter-process communication methods should be devised by the students.

System Design Aspects

It is important to remember that a real-time application must be correct both in the logic and time domains. As such, students have to attain the desired functionality and, in addition, provide evidence that the application performs timely. To this end, it is necessary to:

- Measure the WCRT of each one of the tasks.
- Determined the periods/minimum inter-arrival time of task, as well as eventual blocking and interference issues.
- Based on the above information demonstrate (formally) that the system is schedulable.

Deliverables

Students are required to provide the following deliverables:

- One compressed file (tar or zip format) with a self-contained MPLABX project with the developed application code. **The project is required to compile from the “Demo” FreeRTOS folder.**



- Report:
 - “Pdf” format, A4 size, up to 8 pages, single column, single space, 11pt font
 - Structure:
 - Introduction.
 - Presentation and discussion of the firmware architecture. Aspects to address include a description of the functionality of the tasks implemented, which synchronization IPC mechanism were used (and why), eventual data structures, etc.
 - Evaluation of the real-time behavior of the application. Aspect to address include the temporal characterization of tasks and the demonstration of the schedulability of the application.
 - Experimental results, tests and verification.
 - Conclusions.
- The final project shall be presented and demonstrated during the last class of the semester.
 - Each group has a slot of 15 minutes organized as follows:
 - Presentation: 5 min;
 - Discussion: 5 min;
 - Demonstration: 5 min.
- The report and code must be submitted until one week after the last class of the semester.

Important dates

Considering the amendment to the Academic Calendar (Despacho 69-Reit_2020), which allows a compensation day on January 20th, the calendar is as follows:

- Presentation: January 20th 2021, during the class (16H00-19H00);
- Report and code submission: eLearning, January 27th 2021, until 23H55m.