Concordia University - Department of Computer Science and Software Engineering
SOEN 422 Project Report Guideline - Fall 2020

Project Report

The Project Report is composed of five sections:

1. **Title Page:**

   The title page should include the team name, team members and students id numbers.

2. **Introduction: (1/4 page)**

   This section provides a short description of your project. Discuss what hardware and software were used.

3. **Goal of the Project: (1/2 page)**

   This section provides a quick overall summary of what your group's tasks have achieved during the port of the virtual machine on the Arduino Nano.

4. **Implementation of Tasks: (5 pages)**

   This section should summarize your implementation of the following main tasks on the target:

   Task 3 : BSL and HAL layers (maximum 1 1/2 page)
   Task 4 : VM Operand Stack and the VM core (maximum 1/2 page)
   Task 5 : Serial Loader on the target (maximum 2 pages)
   Task 6 : Interrupt functions[1] (maximum 1/2 page)
   Task 7 : I/O Register functions[2] (maximum 1/2 page)

   Theses subsections are expected to have the discussion of the implementations as well. You should include snippets of code that your group feels are relevant (and proud) to your project's overall functionality. Do not simply toss in code but rather, discuss what is relevant about the code of the specific task and where it belongs. Theses subsections serve also as a reflection of the tasks. Discuss which tasks were met, which weren't and why.

5. **Conclusion: (1/4 page)**

   The conclusion should tie everything together. Filling your conclusion with things such as "The project was fun and entertaining" does not add value. Make your conclusion meaningful.

Each team will be required to submit a final report on Moodle in PDF format by the due date. Naming convention for zip file is: `ProjectReport_TeamXX.zip`

Rick will contact each team by email to get the link for your team repository.

[1] Interrupt_Enable, Interrupt_Disable, Interrupt_SaveAndDisable, and Interrupt_Restore.
[2] IOReg_Read and IOReg_Write.