18-748. Lab1. Assignment 3.

Kedar Amladi // kamladi. Daniel Santoro // ddsantor. Adam Selevan // aselevan.

Initial investigation.

By running the source code provided with original priority settings (Task1 = priority 1, Task2 = priority 2) the following worst case response time was observed:

- Task1 - 607.4ms
- Task2 - 303.7ms

When the priorities were switched, however (Task1 = priority 2, Task2 = priority 1) the worst case response time paradigm flipped. The results for worst case response time were:

- Task1 - 301.7ms
- Task2 - 608.3ms

The difference between these two runs is blocking time. In the first instance, Task2 blocks Task1 from running initially because it has a higher priority. Similarly, in the second instance, Task1 has the higher priority and blocks Task2 from running for a short time. Afterwards, however, the blocked task is permitted to run to completion.

Bonus - Part 1.

Determining a proper reserve value involved a slight edit to the source code was necessary. An additional printf statement was placed at the beginning of Task1 and Task2 to give the ability to calculate the runtime of each task. Through this procedure it was determined that the worst case execution time of each task (assuming no preemption) is 301.75ms. Theoretically, this value could be used as the reserve value. Initial testing proved, however, that 302ms was not sufficient. Through intelligent trial-and-error, it was determined that 304ms was sufficient to to run each task without resetting the watchdog.

Bonus - Part 2.