SYSC 4001 Assignment 1
Part II : Simulations

Stefan Martincevic, 101295641
Sadman Sajid, 101303949

06/10/2025

GitHub: https://github.com/stefan-martincevic/SYSC4001_A1

Overview of Simulations

A total of 30 simulations were performed. The first 15 simulations were done on three different context switch time values. It was performed by running five input files (trace_1 – trace_5.txt) under three different context switching time for each file. It was also supplied with device_table.txt. For each context switch time value, the time spent on each event category was composed as an average from the five output files. The next 15 simulations were done to observe ISR activity time. A new device table was created as device_table_40_200.txt. This device table contained ISR activity times which ranged from 40 to 200 ms, offering a smaller variance and average ISR activity time compared to device_table.txt. Similarly to the first 15 simulations, it was performed on five input files (trace_1 – trace_5.txt) under three different context switching times for each file. For each context switch time value, the time spent on each event category was composed as an average from the five output files.

Altering Save/Restore Context Time Value

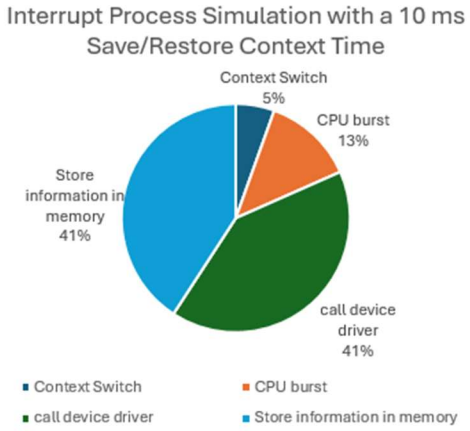

Figure 1: Setting Context Switch Time to 10 ms

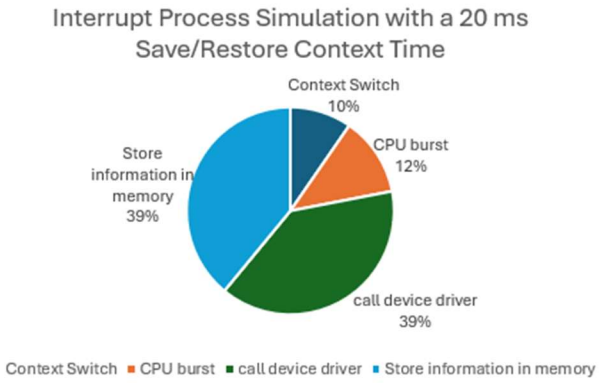

Figure 2: Setting Context Switch Time to 20 ms

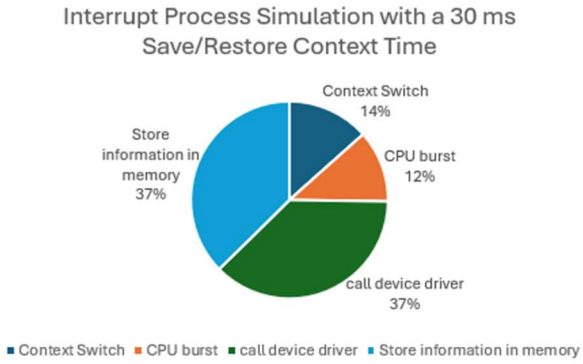

Figure 1: Setting Context Switch Time to 30 ms

Figures 1 – 3 demonstrate how the change in context switch time impact the interrupt process. As the context switch time increases, there is a relative decrease in time for all other events the CPU executes. This is the case as whenever an interrupt occurs the CPU must stop and save its current task and spend a certain amount of time to switch to the task of executing the interrupt service routine. For example, the total execution time was 97,029 ms for Figure 1, 101,484 for Figure 2, and 105,924 for Figure 3. This demonstrates that as the context switch time increases, the execution of the entire program also increases. Therefore, as the context switch time increases, more time is spent switching between the CPU's current task and the ISR, which increases the total time it takes for the CPU to execute the program and decreases the percentage of time the CPU spends executing processes or ISRs.

Setting ISR Activity Time between 20 ms and 400 ms

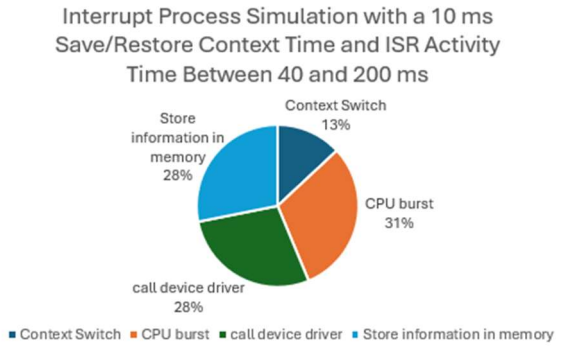

Figure 4: Setting ISR Activity Time Between 40 - 200 ms and Context Switch Time to 10 ms

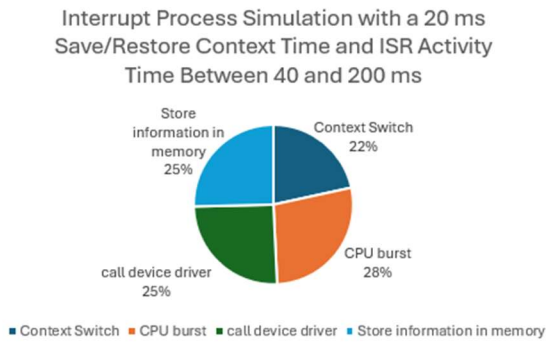

Figure 5: Setting ISR Activity Time Between 40 - 200 ms and Context Switch Time to 20 ms

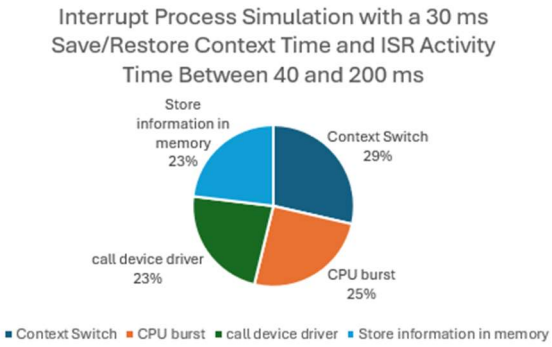

Figure 2: Setting ISR Activity Time Between 40 - 200 ms and Context Switch Time to 30 ms

Setting the ISR activity time between 40 and 200 ms offers a smaller variance and average ISR activity time in comparison to the original device table. The given device table has an average ISR activity time of 352.6 ms. The device table with ISR activity times between 40 and 200 ms has an average time of 105.8 ms. The total execution time was 40,802 for Figure 4, 45,242 for Figure 5, and 49,682 for Figure 6. These program execution times are approximately half the time compared to those from the original device table. This shows that ISR activity times can drastically impact the entire speed of a program but also takes up the resources of the CPU. As the time it takes to service an interrupt is a lot smaller, far more time is spent performing the fetch - execute cycle in the CPU burst event category. The percentage of time spent in the CPU burst event category is approximately 3 times greater than when using the original device table. This is justified, as the amount of time it takes to service an interrupt is more than three times less. Another observation is that even in these simulations as the context switch time increases, the percentage of time spent performing a CPU burst decreases, further validating the context switch time observations. This significantly reduces the available time for CPU processing. Observing Figure 3, and Figure 6, more time is spent on switching tasks from a CPU process to an interrupt and back to CPU bursts as opposed to the other figures with lower context switch times. This is very costly as more time is spent on saving and restoring the context of the process.