PROJECT 4

REAL TIME ANALYSIS REPORT

Gautam Sadarangani

Three Tasks

The three tasks completed are:

1. Reading from the sensor (inclusive of i2c communication delay) and posting updated value to the user space(Kernel Task)
2. Filtering and processing of the data on the user side along
3. Updating the display, inclusive of SPI communication delay

The three tasks, as mentioned above, have to be completed in 100Hz, i.e. 100 times in one second.
The 3 tasks can have a period of maximum of 10ms.

Task	Period	Execution Time
Reading & updating values to user space.	10ms	2.9ms
Reading from event & filtering	10ms	0.056ms
Updating Display	10ms	5.9ms

Either Rate Monotonic or Earliest Deadline first can be used to determine schedule.
Let's consider the Earliest Deadline First as it tends to be better at higher loading.[1]

Determining if a valid schedule exists:
Task 1 = (10, 3.5)
Task 2 = (10, 0.056)
Task 3 = (10, 7.1)

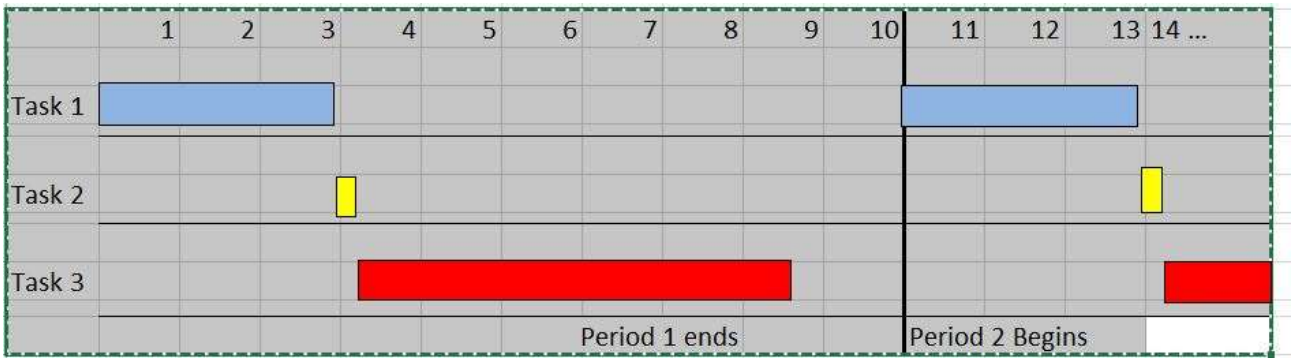
$$\Sigma U_i = \frac{2.9}{10} + \frac{0.056}{10} + \frac{5.9}{10} = 0.8856 < 1 \text{ but } > 0.779$$

$$\text{As } \Sigma U_{i=} 3(2^{1/3} - 1) = 0.779$$

Therefore the task is not guaranteed to be schedulable under rate monotonic scheme, hence we use the EDF scheme.
Since $\Sigma U_i < 1$, the three tasks are schedulable, under EDF, as shown below.

Schedule

The following diagram shows the valid schedule:



Task	Start Time(ms)	End Time(ms)
Task 1	0	2.9
Task 2	2.9	2.956
Task 3	2.956	8.856

Issues/Conditions that may delay the deadline:

- If the SPI runs at a slower rate than it may update the display at a slow rate causing the display to lag

Generic Delays

- Processing an interrupt would pre-empt the current task & require CPU time which may cause a deadline miss.
- Considering any other thread(s) that might be running on the system; it (they) could take up more time or it (they) may enter a critical section that would cause a deadline miss.

None of these delays or interruptions are critical, the system will still work.