Embedded systems
Exercise session, 6/12
Scheduling problems

Problem 1

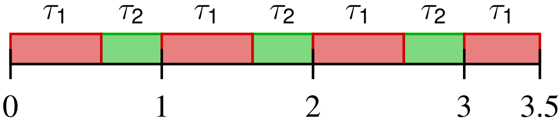
Two periodic tasks τ_1, τ_2 are characterized by their respective periods $T_1 = 1$ ms and $T_2 = 3.5$ ms. The execution time C_1 of τ_1 is equal to 0.6 ms.

For which value(s) of C_2 does this pair of tasks fully use the processor? (Justify all steps of your reasoning.)

T_1 : 1 ms $\quad$ C_1 : 0.6 ms

T_2 : 3.5 ms $\quad$ C_2 : ?

$$P_1 > P_2$$



$$C_2 = 3 \times 0.4 \text{ ms} = 1.2 \text{ ms}$$

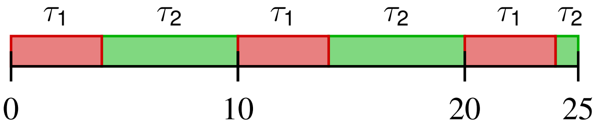
Problem 2

Let τ_1 and τ_2 be periodic tasks with the respective periods $T_1,\ T_2$ and execution times $C_1,\ C_2$, such that $T_1 = 10\ \mu$s, $T_2 = 25\ \mu$s, and $C_1 < 5\ \mu$s. The priority of τ_1 is higher than the one of τ_2. This pair of tasks fully uses the processor.

1. Represent this problem graphically.
2. Compute the value of C_2 as a function of C_1.

$$T_1 : \quad 10\ \mu s \qquad C_1 : \quad < 5\ \mu s$$
$$T_2 : \quad 25\ \mu s \qquad C_2 : \quad ?$$

$$P_1 > P_2$$



$$C_2 = 25\ \mu s - 3C_1$$

Problem 3

Consider the following set of periodic tasks $\tau_i = (C_i, T_i)$:

$$\{\tau_1 = (3, 13),\ \tau_2 = (1, 3),\ \tau_3 = (\alpha, 5)\},$$

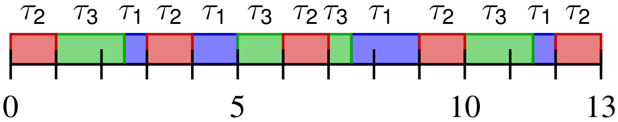
where α is a parameter.

1. Compute the maximum value of α for this set of tasks to be schedulable.
2. Verify your answer with a graphical simulation.

$$\begin{array}{llll} T_1: & 13 & C_1: & 3 \\ T_2: & 3 & C_2: & 1 \\ T_3: & 5 & C_3: & \alpha \end{array}$$

$$P_2 > P_3 > P_1$$

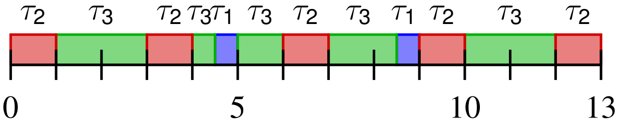
Case 1: τ_3 finishes before $t = 12$:



$$C_1 = 3 = 13 - 5 \times 1 - 3 \times \alpha$$

$$\Rightarrow \; \alpha = \frac{5}{3}$$

Case 2: τ_3 still active at $t = 12$:

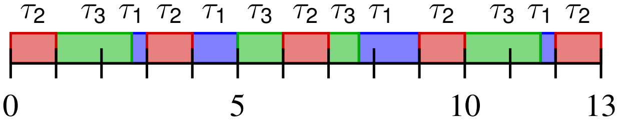


$$C_1 = 3 = 10 - 4 \times 1 - 2 \times \alpha$$

$$\Rightarrow \ \alpha = \frac{3}{2}$$

(Contradiction! α should be greater than $\frac{5}{3}$)

Simulation with $\alpha = \dfrac{5}{3}$:



$$\begin{aligned} C_1 &= \frac{1}{3} + 1 + \frac{4}{3} + \frac{1}{3} \\ &= 3 \quad \text{(OK!)} \end{aligned}$$

Problem 4

Consider the following set of periodic tasks $\tau_i = (C_i, T_i)$:

$$\{\tau_1 = (1, 4),\ \tau_2 = (3, 13),\ \tau_3 = (1, 10), \tau_4 = (\alpha, 7)\},$$

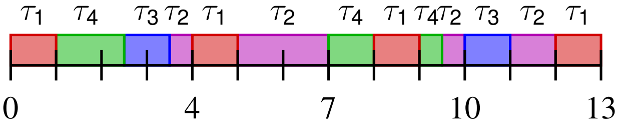
where α is a parameter.

Compute the largest value of α that makes this set of tasks schedulable.

$$\begin{array}{llll} T_1: & 4 & C_1: & 1 \\ T_2: & 13 & C_2: & 3 \\ T_3: & 10 & C_3: & 1 \\ T_4: & 7 & C_4: & \alpha \end{array}$$

$$P_1 > P_4 > P_3 > P_2$$

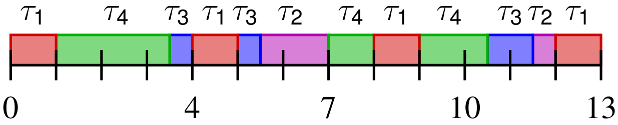
Case 1: τ_4 finishes before $t = 10$:



$$C_2 = 3 = 13 - 4 \times 1 - 2 \times \alpha - 2 \times 1$$

$$\Rightarrow \alpha = 2$$

Case 2: τ_4 still active at $t = 10$, and τ_3 finishes before $t = 12$:

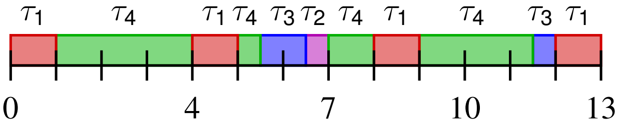


(Similar to previous case)

$$C_2 = 3 = 13 - 4 \times 1 - 2 \times \alpha - 2 \times 1$$

$$\Rightarrow \ \alpha = 2$$

Case 3: τ_3 still active at $t = 12$:



$$C_2 = 3 = 7 - 2 \times 1 - \alpha - 1$$

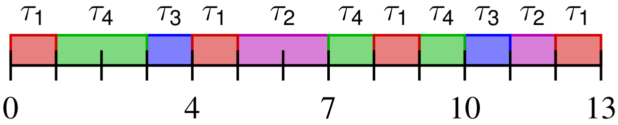
$$\Rightarrow \ \alpha = 1 \quad (\textit{contradiction}!)$$

Case 4: τ_4 still active at $t = 12$:

$\Rightarrow$ Impossible (does not leave any room for τ_2)!

Conclusion: $\alpha = 2$

Simulation:

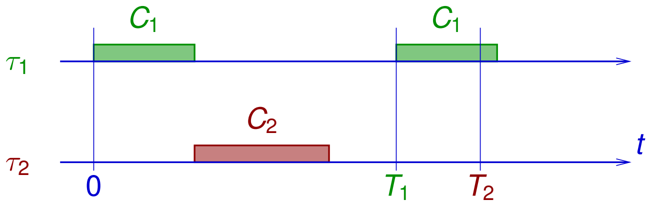


Problem 5

Let τ_1 and τ_2 be two periodic tasks with respective periods and execution times T_1, T_2 and C_1, C_2. We assume that these tasks satisfy $T_1 < T_2$, that they both initially start at $t = 0$, that they fully use the processor, that they are scheduled under the RMS policy, and that τ_1 is not idle (i.e., it is active) at $t = T_2$.

Under those hypotheses, express the processor load factor in terms of C_1, T_1 and T_2, carefully justifying your developments. Show graphically how this load factor varies with C_1, when T_1 and T_2 are assumed to be constant.

(Slides 147–148 of the course)



The following condition is satisfied:

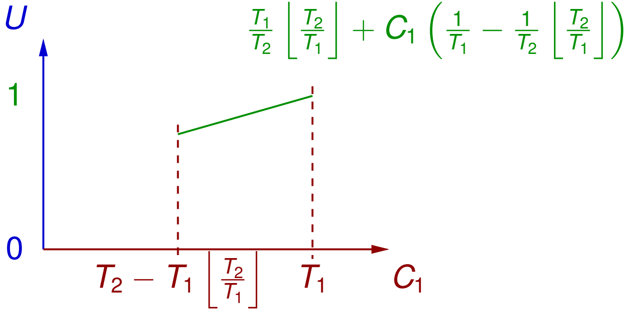
$$C_1 > T_2 - T_1 \left\lfloor \frac{T_2}{T_1} \right\rfloor .$$

For a given value of C_1, the largest possible value of C_2 is given by

$$C_2 = (T_1 - C_1) \left\lfloor \frac{T_2}{T_1} \right\rfloor .$$

For given values of T_1 and T_2, this expression increases with C_1, since

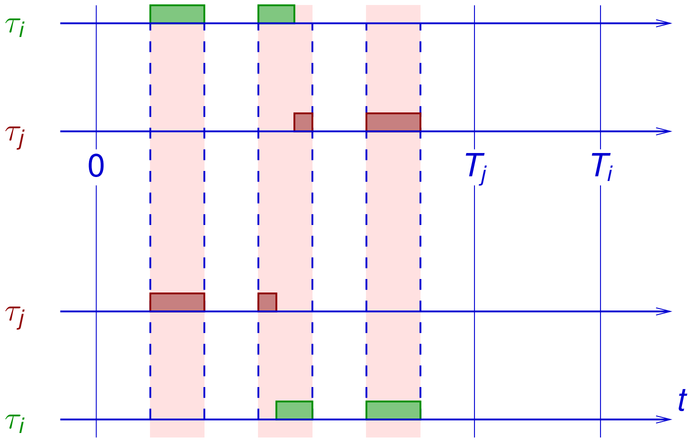
$$\frac{1}{T_1} - \frac{1}{T_2}\left\lfloor \frac{T_2}{T_1} \right\rfloor \geq 0.$$



Problem 6

Prove that every schedulable set of exactly three periodic tasks remains schedulable with a rate-monotonic assignement of priorities.

1. If two tasks τ_i and τ_j have priorities P_i and P_j that are adjacent and such that $P_i > P_j$, and periods T_1 and T_2 such that $T_1 > T_2$, then swapping their priorities preserves schedulability.



2. This is equivalent to saying that if two tasks τ_i and τ_j have priorities P_i and P_j that are adjacent and such that $P_i > P_j$, and periods T_1 and T_2 such that $T_1 > T_2$, then swapping their periods preserves schedulability.

3. Let us assume without loss of generality $P_1 < P_2 < P_3$.

If $T_1 < T_2$, we swap T_1 and T_2. Then, if $T_2 < T_3$, we swap T_2 and T_3. Afterwards, T_3 becomes equal to the smallest period.

If $T_1 < T_2$, we swap T_1 and T_2. One then has $T_1 > T_2 > T_3$, which corresponds to the RMS policy.

Every permutation operation in this procedure preserves schedulability.

Problem 7

If a set $\{\tau_1, \tau_2, \tau_3\}$ of three periodic tasks τ_1, τ_2 and τ_3 is schedulable, is the set $\{\tau_1, \tau_2\}$ always schedulable as well? (Justify your answer.)

1. Since $\{\tau_1, \tau_2, \tau_3\}$ is schedulable, if we simulate them in the critical zone of the task with the longest period, starting each task at $t = 0$, then τ_1 finishes at or before $t = T_1$, and τ_2 finishes at or before $t = T_2$.

2. Removing τ_3 can only advance the execution time of each instruction of τ_1 and τ_2, or leave it unchanged. The real-time constraints on τ_1 and τ_2 will thus remain satisfied, therefore $\{\tau_1, \tau_2\}$ is schedulable.