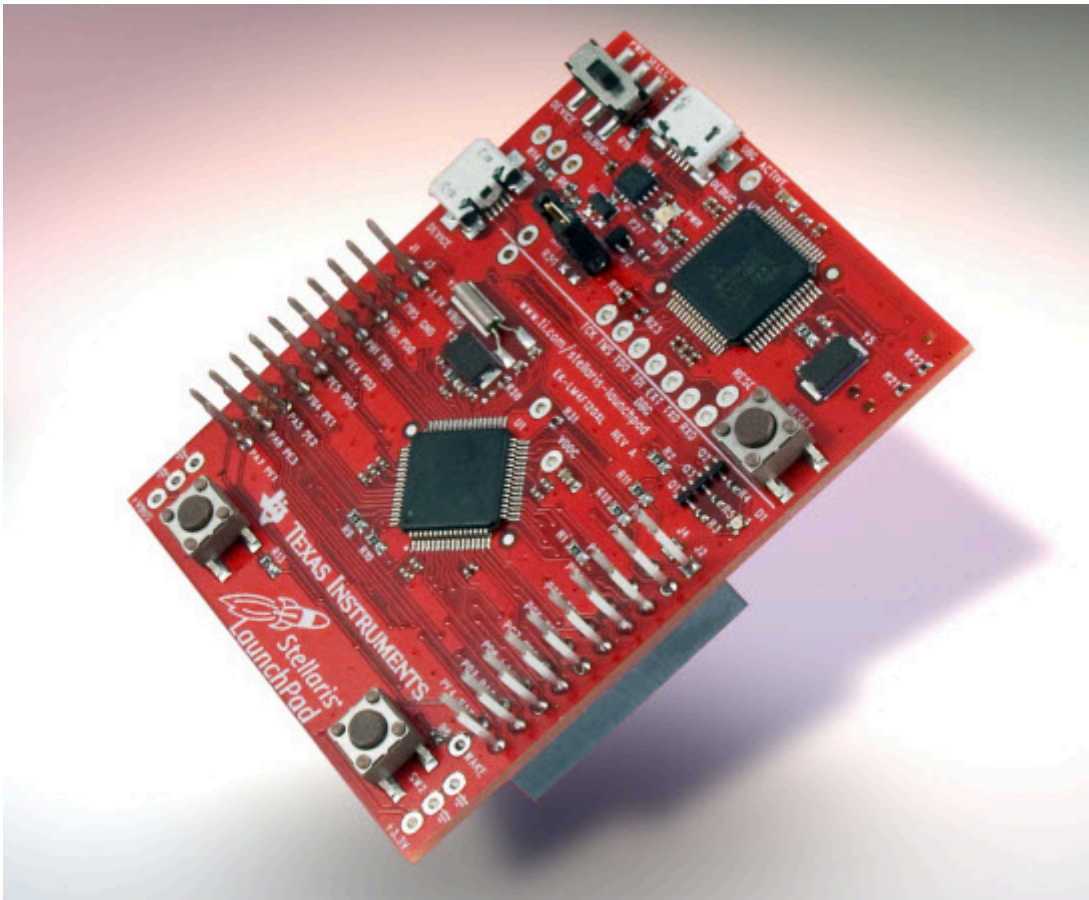


# Práctica 2

## SYS/BIOS



## Sistemas Empotrados II

Héctor Lacueva Sacristán (869637)

18/11/2025

<b>Apartado 1.....</b>	<b>3</b>
Apartado 1.1.....	3
Añadir una Swi que atienda al evento y que sea activada por la interrupción hardware (flanco de subida en el PIN PD3). Deben unirse los PINES PD2 y PD3 entre sí. Comprobar sobre el gráfico de ejecución el comportamiento del sistema. ¿Pueden garantizarse plazos en estas circunstancias?.....	3
<b>Apartado 2.....</b>	<b>4</b>
Apartado 2.1.....	4
Programar sobre el programa comun el manejo de los eventos mediante un servidor esporádico. Una interrupción hardware activa una interrupción software que recoge los eventos y los envía a un mailbox. Los eventos serán procesados por el servidor esporádico que es el responsable de ejecutar la acción asociada a cada evento.....	4
Apartado 2.2.....	5
Realizar el análisis DM (demostrando que se van a cumplir, o no, los plazos correspondientes) y asignar prioridades.....	5
T1.....	5
E1.....	6
T2.....	6
T3.....	6
Apartado 2.3.....	7
Verificar el cumplimiento de los plazos sobre un gráfico de ejecución del programa..	7

## Apartado 1

Partimos de una situación como la siguiente:

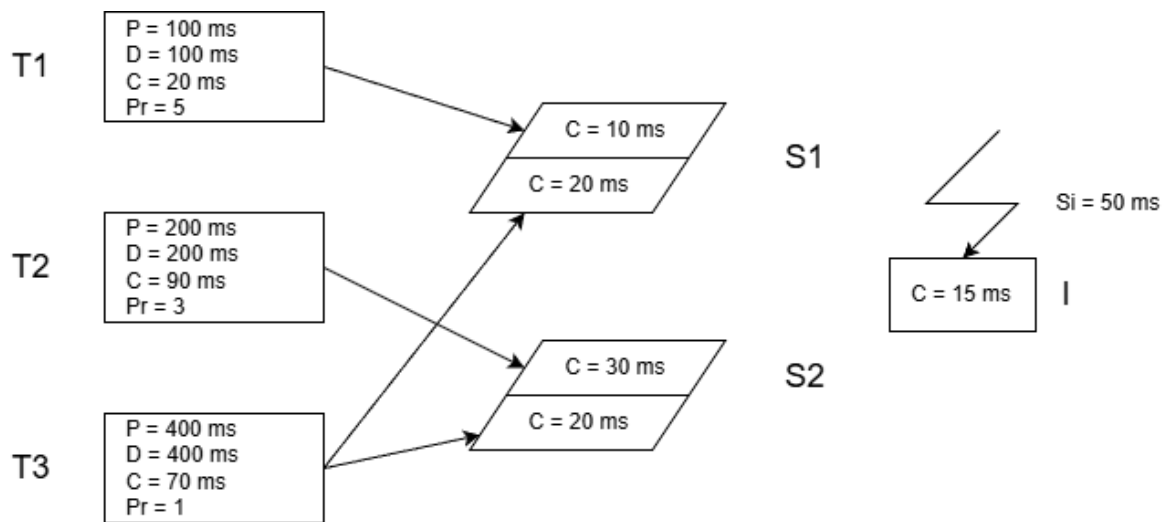


Figura 1: Diagrama de la situación de partida de la práctica. Tres tareas periódicas que acceden a dos servidores y una interrupción cada 50 ms con tiempo de cómputo de 15 ms.

### Apartado 1.1

Añadir una Swi que atienda al evento y que sea activada por la interrupción hardware (flanco de subida en el PIN PD3). Deben unirse los PINES PD2 y PD3 entre sí. Comprobar sobre el gráfico de ejecución el comportamiento del sistema. ¿Pueden garantizarse plazos en estas circunstancias?

El diagrama de la nueva situación es el siguiente:

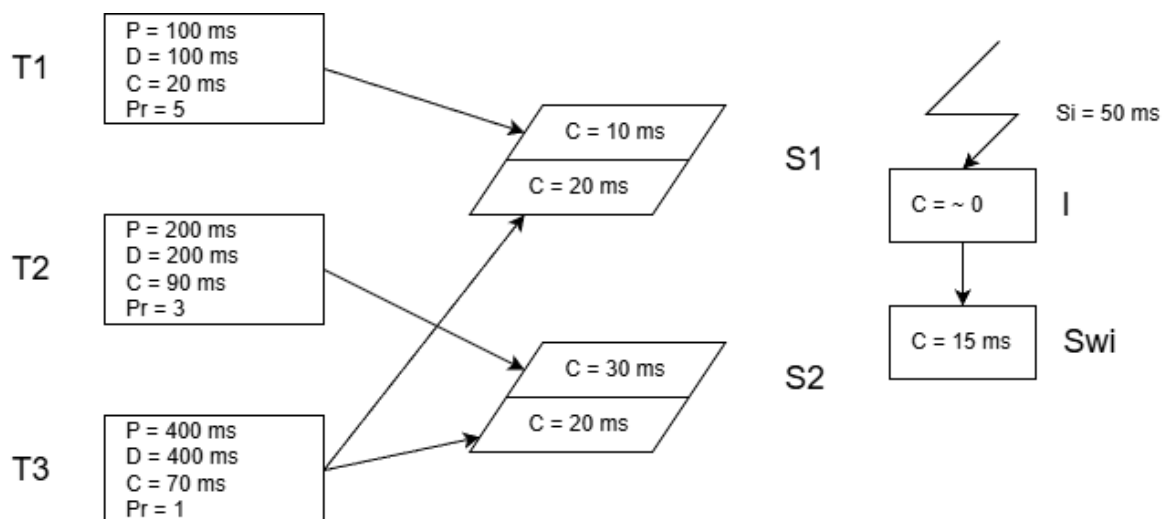


Figura 2: Diagrama de la situación expuesta en el apartado 1. En vez de gestionar la interrupción en la ISR, se gestiona en una SWi.

Ejecutamos el programa y analizamos el Execution Graph:

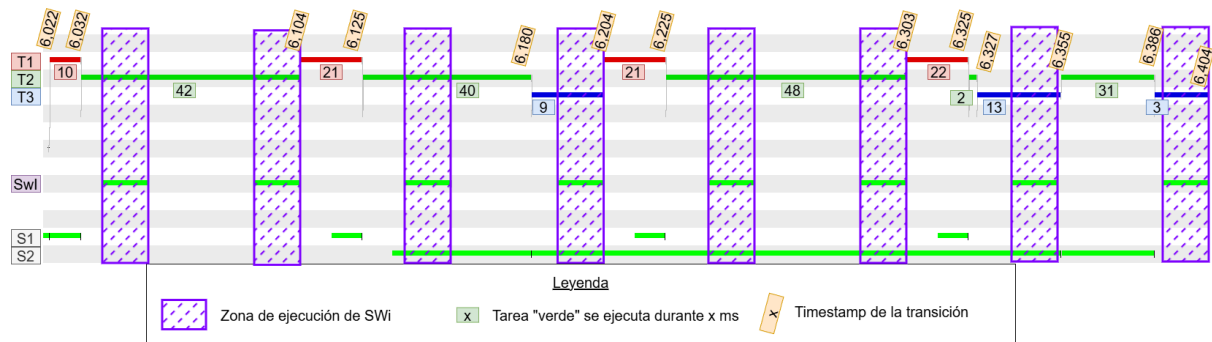


Figura 3: Captura de pantalla de la traza de Execution Graph con la situación de la [Figura 2](#). Se pueden ver los timestamps, los tiempos de las diferentes tareas ejecutándose en la CPU.

En la imagen se ve que no se cumplen los plazos para todas las tareas, sobre todo se puede ver en la tarea T3, que a lo largo de toda la traza sólo ha podido ejecutarse 25 ms de los 70 ms que deberían haberse ejecutado desde el comienzo hasta el final. La tarea T2 tampoco tiene por qué cumplir, hay un punto en el que se ve que ha acabado la ejecución (6,327), su activación podría ocurrir en el intervalo (6,104 - 6,125). En ninguno de los casos cumpliría. La tarea 1 parece cumplir los plazos pero tampoco está garantizada solo viendo la traza.

## Apartado 2

### Apartado 2.1

Programar sobre el programa comun el manejo de los eventos mediante un servidor esporádico. Una interrupción hardware activa una interrupción software que recoge los eventos y los envía a un mailbox. Los eventos serán procesados por el servidor esporádico que es el responsable de ejecutar la acción asociada a cada evento.

La nueva situación es la siguiente:

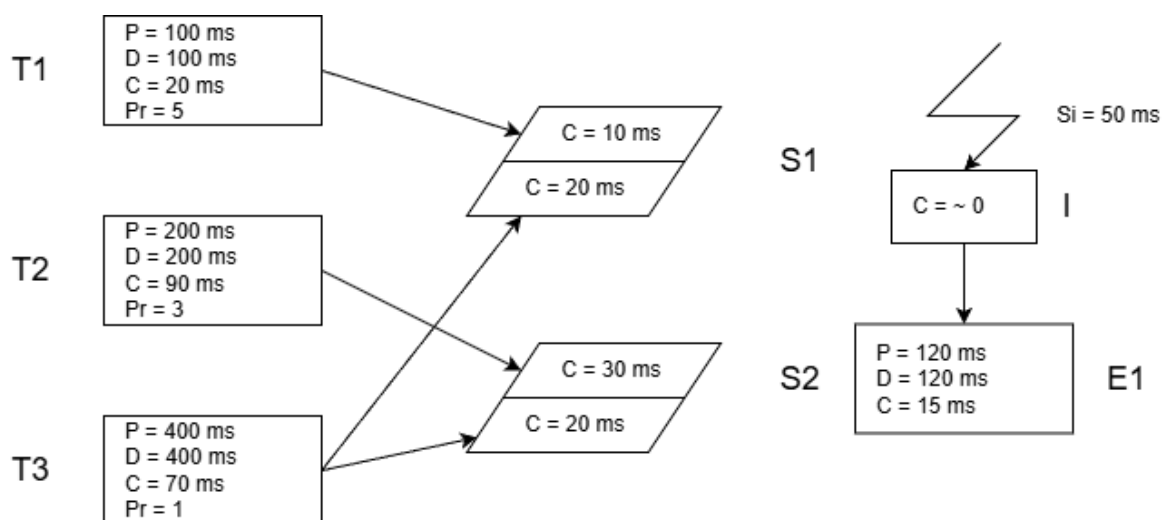


Figura 4: Diagrama de la situación expuesta en el apartado 2. Las interrupciones se gestionan en un servidor esporádico, que, cada 120 ms requiere de 15 ms de cómputo.

## Apartado 2.2

Realizar el análisis DM (demostrando que se van a cumplir, o no, los plazos correspondientes) y asignar prioridades.

Tarea	Periodo	Deadline	Tiempo de cómputo	Prioridad	S1	S2
T1	100 ms	100 ms	20 ms	5	10 ms	
E1	120 ms	120 ms	15 ms	4		
T2	200 ms	200 ms	90 ms	3		30 ms
T3	400 ms	400 ms	70 ms	1	20 ms	20ms
Techo de prioridad					5	3

Primero calculamos los **tiempos de bloqueo** teniendo en cuenta el protocolo de herencia de prioridad.

Tarea	Tiempo de bloqueo (herencia de prioridad)
T1	$b_1 = \min(20 \text{ ms}, 20 \text{ ms}) = 20 \text{ ms}$
E1	$b_{E1} = \min(20 \text{ ms}, 20 \text{ ms}) = 20 \text{ ms}$
T2	$b_2 = \min(20 \text{ ms}, 40 \text{ ms}) = 20 \text{ ms}$
T3	$b_3 = 0 \text{ ms}$

T1

$$T1 \Rightarrow C_1 + b_1 \leq D_1 \rightarrow 20 + 20 = 40 < 100$$

✓ T1 cumple plazos.

E1

$$E1 \Rightarrow \left\lceil \frac{D_E}{P_1} \right\rceil C_1 + C_E + b_E \leq D_E \Rightarrow \left\lceil \frac{120}{100} \right\rceil 20 + 15 + 20 = 40 + 15 + 20 = 75 \leq 120$$

✓ E1 cumple plazos.

T2

$$T2 \Rightarrow \left\lceil \frac{D_2}{P_1} \right\rceil C_1 + \left\lceil \frac{D_2}{P_E} \right\rceil C_E + C_2 + b_2 \leq D_2 \Rightarrow \\ \Rightarrow \left\lceil \frac{200}{100} \right\rceil 20 + \left\lceil \frac{200}{120} \right\rceil 15 + 90 + 20 \leq 200 \Rightarrow 40 + 30 + 90 + 20 = 180 \leq 200$$

✓ T2 cumple plazos.

T3

$$T3 \Rightarrow \left\lceil \frac{D_3}{P_1} \right\rceil C_1 + \left\lceil \frac{D_3}{P_E} \right\rceil C_E + \left\lceil \frac{D_3}{P_2} \right\rceil C_2 + C_3 + b_3 \leq D_3 \Rightarrow \\ \Rightarrow \left\lceil \frac{400}{100} \right\rceil 20 + \left\lceil \frac{400}{120} \right\rceil 15 + \left\lceil \frac{400}{200} \right\rceil 90 + 30 + 0 \leq 400 \Rightarrow \\ \Rightarrow 80 + 30 + 180 + 30 = 320 \leq 400$$

✓ T3 cumple plazos.

## Apartado 2.3

Verificar el cumplimiento de los plazos sobre un gráfico de ejecución del programa.

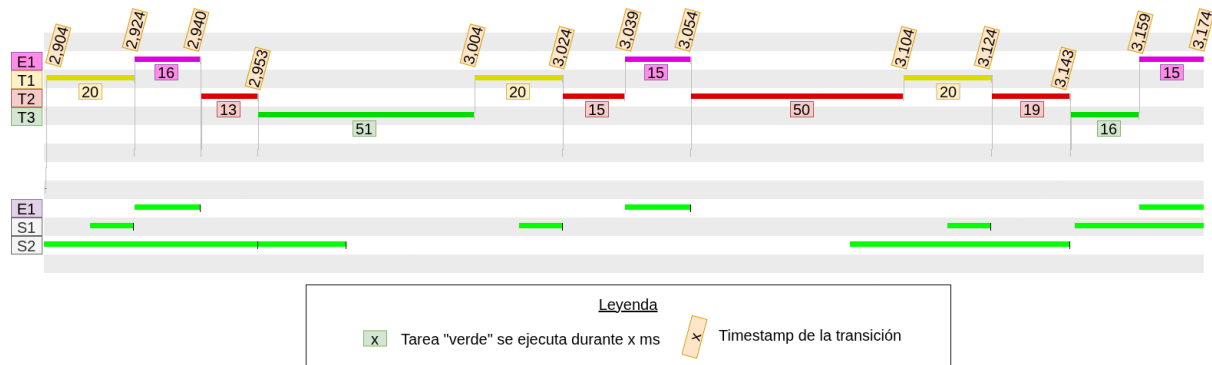


Figura 5: Captura de pantalla del Execution Graph de CCS tras ejecutar el programa.

Como se puede ver en la [Figura 5](#), la tarea 1 cumple plazos, el servidor esporádico también cumple plazos, la tarea 2 no se puede asegurar en la traza de la [Figura 5](#) pero cumple plazos y la tarea 3 lo mismo.