Carrera de Especialización en Sistemas Embebidos

Sistemas Operativos en Tiempo Real

Clase 4 (Práctica): Sincronización en FreeRTOS

ACSE
Asociación Civil para la Investigación, Promoción y Desarrollo de los Sistemas Electrónicos Embebidos
FACULTAD DE INGENIERIA
Universidad de Buenos Aires
CIM
CADIEEL

Resumen: Sincronización de tareas

- FreeRTOS nos ofrece, como solución para la sincronización de tareas: Semáforos binarios y contadores.

  - Semáforo Binario

    Solamente permite dos estados: Tomado o Disponible (Rojo o Verde). Al estar tomado, una tarea que intente tomarlo deberá esperar hasta que otra tarea lo libere o hasta que se cumpla un plazo.

  - Semáforo Contador

    Permite varias liberaciones. Cada liberación incrementa su estado en 1 hasta un número máximo establecido y cada vez que una tarea lo tome, decrementará su valor hasta llegar a 0.

    Una tarea que desee tomar un contador en 0, deberá esperar hasta que otra lo libere al menos una vez o hasta que se cumpla un plazo.

Resumen: Sincronización de tareas

- Asimismo, nos ofrece herramientas para la gestión de acceso concurrente a recursos:
  - Secciones críticas

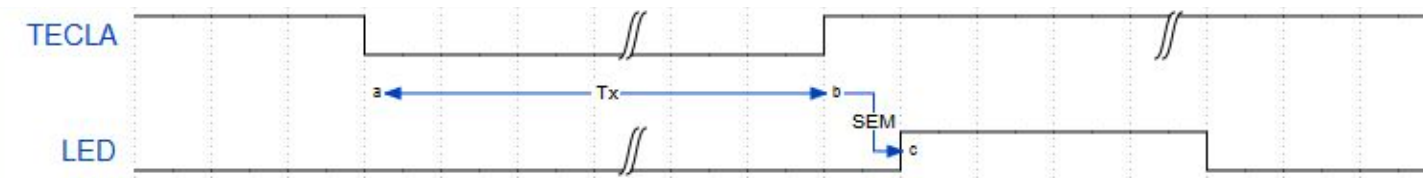
    Deshabilita los cambios de contexto del sistema operativo, lo que nos permite operar sin la posibilidad de que otra tarea tome el procesador en ese tiempo.
  - Mutex

    Funciona como un semáforo, impidiendo el acceso a un recurso hasta que haya sido liberado por la tarea que lo está manipulando. Tiene la ventaja de que no impide que tareas de mayor prioridad tomen el procesador y, además, eleva la prioridad de la tarea que tiene tomado el recurso, si otra tarea de mayor prioridad necesita tomarlo.

Ejercicio #1 - Semáforos binarios

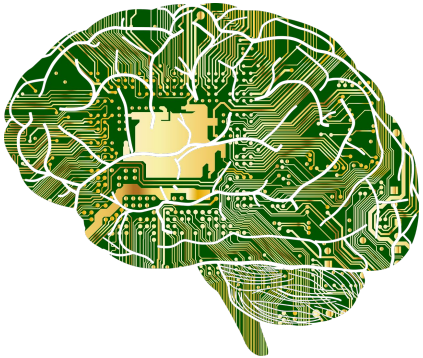
- Comenzaremos implementando dos tareas de las cuales una deberá **leer el tiempo de pulsación** de un botón, mediante antirrebote. Al completar la lectura, una segunda tarea, deberá destellar un led en señal de recepción.

  Tomar de referencia el siguiente gráfico:



Ejercicio #1 - Semáforos binarios

Refresquemos la memoria



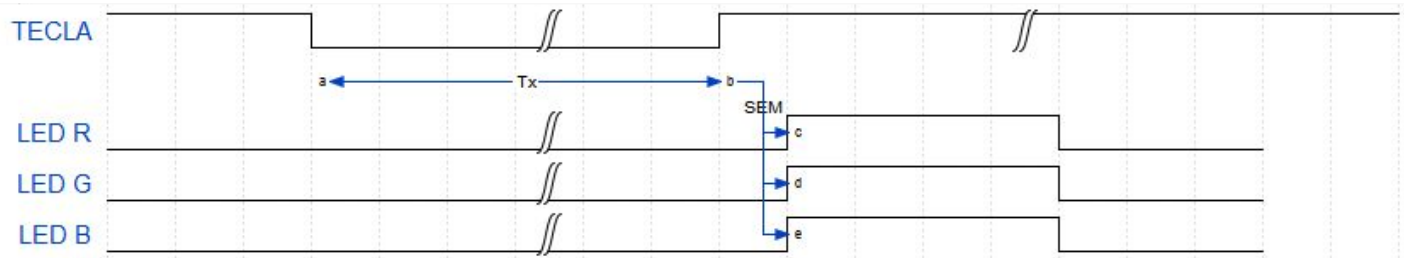
- **¿Cómo deberían asignarse las prioridades de cada tarea?**
- **¿Qué pasaría si utilizamos la misma prioridad para ambas tareas?**
- **¿Qué pasaría si seteamos el preemption a 0 en FreeRTOSConfig.h?**

¡PROBEMOS!

Ejercicio #2 - Semáforos y contadores

- Ahora repitamos el ejercicio anterior para que **tres tareas distintas** destellen, cada una, un led distinto de la EDU-CIAA, tras completar la lectura la primera tarea.

  Tomar de referencia el siguiente gráfico:



Ejercicio #2 - Semáforos y contadores

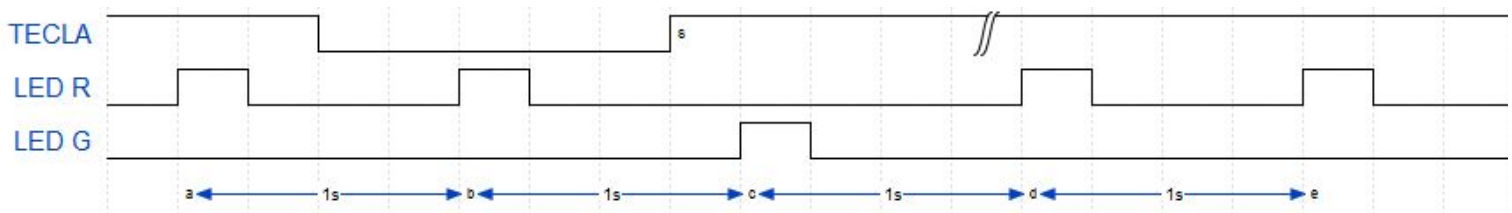
Démosle una vuelta de rosca

- ¿Qué herramientas tengo para compartir el valor medido a alguna(s) de las tres tareas?
- ¿Qué precauciones debo tomar con la primera tarea si quiero mostrar el valor medido en las otras 3?
- ¿Qué precauciones debo tomar si quiero que cada tarea utilice la UART para indicar el estado de los leds?

¡Manos a la obra!

Ejercicio #3 - Integrador

- Mantengamos la primera tarea midiendo el tiempo de pulsación de un botón. Ahora vamos a sincronizarla con una tarea que **muestre**, en el led Azul, **el tiempo de pulsación** medido por la primera tarea y, **cuando no hayan mediciones**, muestre un **Heartbeat** de período 1s y duty cicle 50% en el led Rojo.



Bibliografia

- RTOS 1 - Clase 3 - Sincronización entre tareas (Partes 1 y 2), Franco Bucafusco, 2018
- Sistemas Operativos de Tiempo Real - Guía de Ejercicios, Franco Bucafusco, 2016