**PRÁCTICA 3. *Bitmap* de prioridades y *Ready List***

# OBJETIVOS

En esta práctica se trabajará con los dos procesos del sistema operativo en tiempo real que gestionan el conjunto de Tasks que precisan un cambio de estado *Ready* -> *Running*

# TAREAS PREVIAS

* ¿Qué es el *bitmap* de prioridades? Defina las características de la tabla que lo compone, entradas y tamaño.
* Tetsteo ia completa cosas :
* ¿Qué es la *Ready List*? Explique las características que definen la tabla que lo compone, entradas y tamaño.

La ready list es la estructura de datos donde se colocan las tareas que han cambiado su estado a ready y como su nombre indica están esperando para poder entrar a ejecución.

* ¿Qué parámetro debe retornar el bitmap?
* ¿Qué Task está asociada a la última casilla del bitmap?

Siempre ha de ser la idle task para estar disponible para ejecutarse siempre la ultima.

* Dada una tabla de bitmap compuesta por dos vectores de entrada: Bitmap[0] y Bitmap[1]. Cada vector que compone la tabla esta formada por una palabra de 8-bit. Suponga los siguientes valores decimales en cada vector:

Bitmap[0] = 82 (dec) Bitmap[1] = 14 (dec)

* 1. Determine el valor que retornará el bitmap de prioridades
  2. ¿Existe algún fallo en el valor decimal reportado en los vectores anteriores? Razone la respuesta

**APLICACIÓN**

Se deberán crear 6 Tasks de aplicación y 1 Task de monitorización. La configuración de cada Hilo de aplicación es la siguiente:

|  |  |  |  |
| --- | --- | --- | --- |
| **TASK** | **PRIO.** | **PERIODICIDAD** | **FUNCIÓN** |
| TASK\_1 | 2 | 1 seg | LED + delay\_ms(1000) |
| TASK\_2 | 2 | 500ms | LED |
| TASK\_3 | 3 | 1 seg | LED + delay\_ms(500) |
| TASK\_4 | 4 | 2 seg | LED + delay\_ms(2000) |
| TASK\_5 | 5 | 5 seg | LED + delay\_ms(500) |
| TASK\_6 | 6 | 8 seg | LED + delay\_ms(1000) |

La Task de monitorización debe disponer de:

* Nivel de prioridad: 1
* Periodicidad mínima/máxima: 10ms/5
* El usuario deberá monitorizar la tabla bitmap y reportar por UART, en valor binario y decimal, el estado dinámico de cada casilla que compone el vector. El vector dispondrá de un tamaño por defecto de 32-bit. De este modo, podrá visualizarse el estado actual de las Tasks en estado Ready.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Tenga en cuenta que el sistema operativo define el nombre de la tabla bitmap como
* OSPrioTbl
* La dirección del vector [0], por tanto, podrá accederse mediante
* &OSPrioTbl[0]

Aquí sacamos cada posición de la tabla a través de sus punteros, tenemos que aplicarle una máscara ya que

Tabla

Descripción generada automáticamente

Posteriormente imprimimos cada una de estas posiciones y aplicamos la temporización que se nos ha requerido en el apartado anterior, como hicimos en la práctica anterior.

Interfaz de usuario gráfica, Texto, Tabla

Descripción generada automáticamente

Posteriormente solo enviamos por la UART y ajustamos la temporización requerida.