

## Practica 2 RTOS

Jose Luis Rocabado Rocha

### 1. Ejercicio 1

Se comprueba el flujo de ejecución del código así como la correcta ejecución del *status* de las tareas.

```
*****
Hello From MicroC/OS-II Running on NIOS II. Here is the status:

The number of messages sent by the send_task:      147

The number of messages received by the receive_task1: 91

The number of messages received by the receive_task2: 30

The shared resource is owned by: getsem_task1

The Number of times getsem_task1 acquired the semaphore 300

The Number of times getsem_task2 acquired the semaphore 231

*****

*****
Hello From MicroC/OS-II Running on NIOS II. Here is the status:

The number of messages sent by the send_task:      159

The number of messages received by the receive_task1: 100

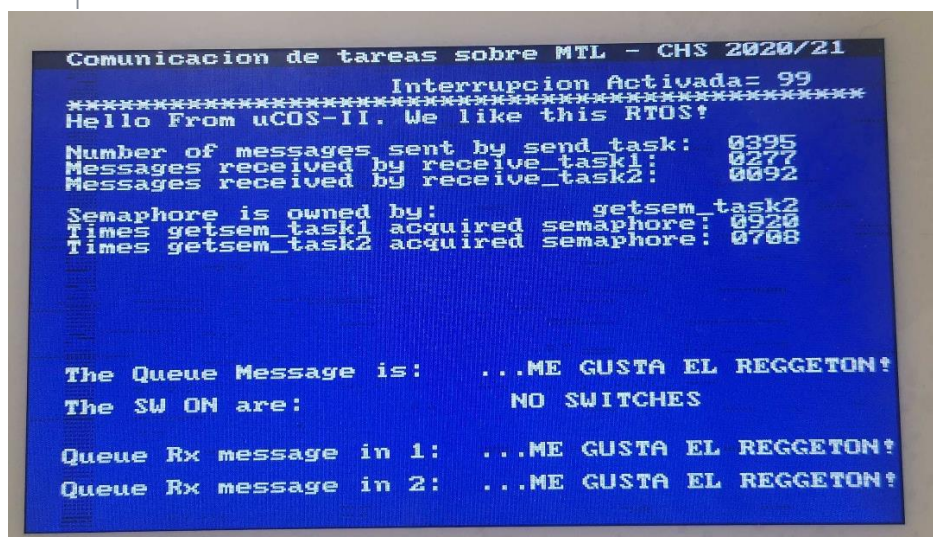
The number of messages received by the receive_task2: 33

The shared resource is owned by: getsem_task1

The Number of times getsem_task1 acquired the semaphore 330

The Number of times getsem_task2 acquired the semaphore 254

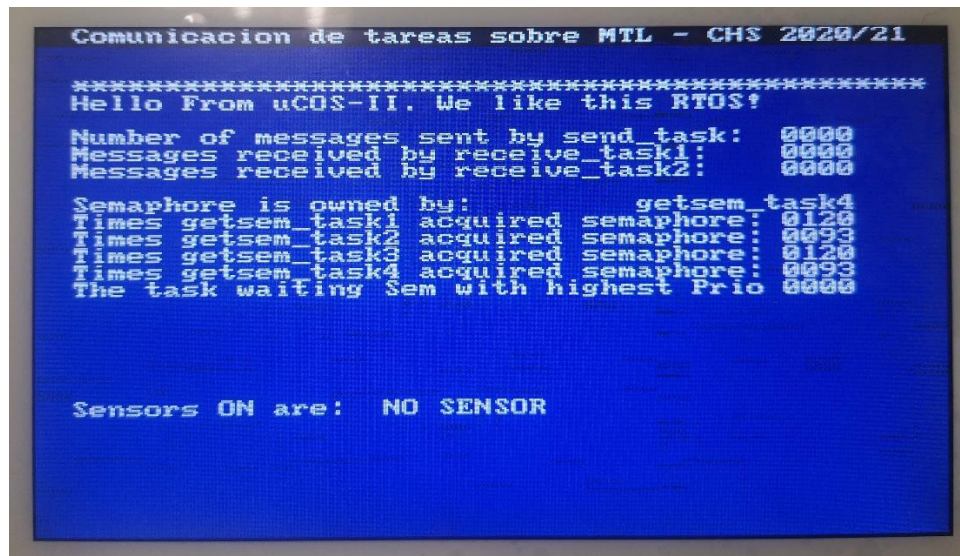
*****
```



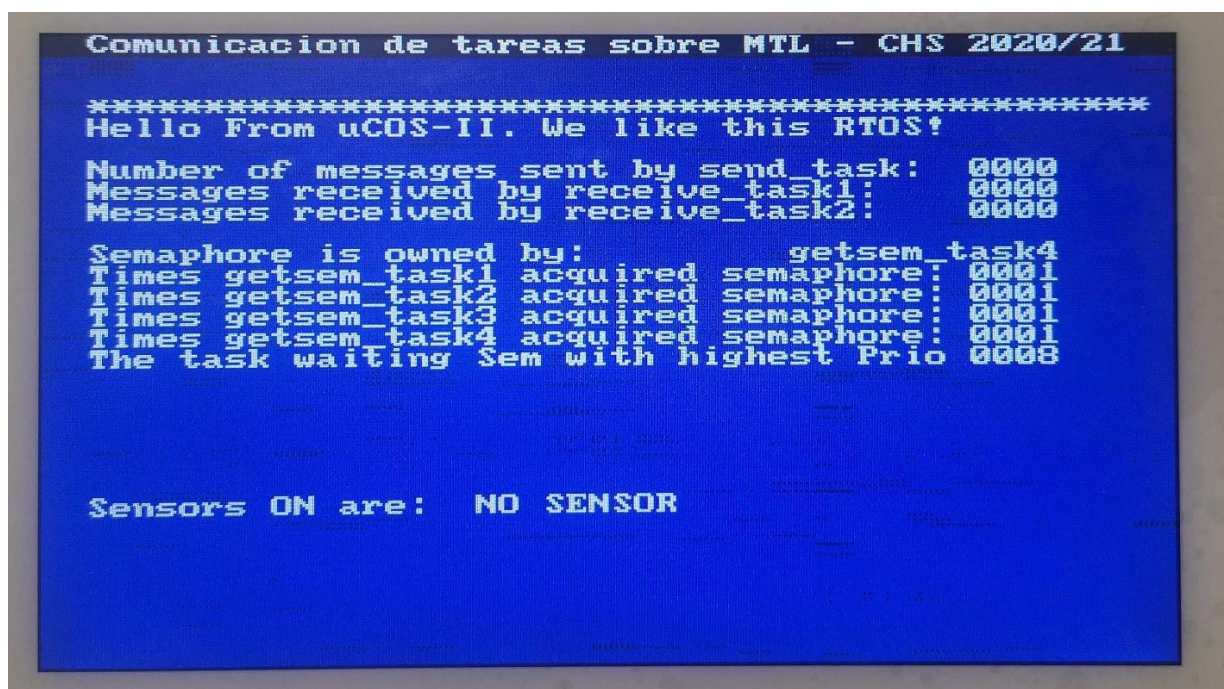
## 2. Ejercicio 2

Se comprueba que se representa el estatus de las tareas en la pantalla MTL.

a-c) Funciona todo correctamente, no hay ninguna tarea que esté esperando al semáforo



d) Al evitar que la task4 postee el semáforo, los recursos se quedan bloqueados y podemos observar que la tarea de mayor prioridad que está esperando a los recursos es la tarea task1 con prioridad 8. Se podría considerar un problema de inversión de prioridades por que la tarea de menos prioridad está utilizando completamente los recursos.





e) Funciona correctamente, no se retienen recursos

```
Comunicación de tareas sobre MTL - CHS 2020/21
*****
Hello From uCOS-II. We like this RTOS!
Number of messages sent by send_task: 0000
Messages received by receive_task1: 0000
Messages received by receive_task2: 0000
Semaphore is owned by: getsem_task3
Times getsem_task1 acquired semaphore: 0090
Times getsem_task2 acquired semaphore: 0077
Times getsem_task3 acquired semaphore: 0090
Times getsem_task4 acquired semaphore: 0077
The task waiting Sem with highest Prio 0000

Sensors ON are: NO SENSOR
```

f) Se ejecutan las tareas hasta que la tarea 4 consume los 3 recursos obteniendo el mensaje siguiente por consola y haciendo que se pare la ejecución del programa

```
Hello from Task Init
El motor no esta preparado ...
El motor no esta preparado ...
El motor no esta preparado ...
La tarea 4 se ha comido todos los recursos
```

```
Comunicación de tareas sobre MTL - CHS 2020/21
*****
Hello From uCOS-II. We like this RTOS!
Number of messages sent by send_task: 0000
Messages received by receive_task1: 0000
Messages received by receive_task2: 0000
Semaphore is owned by: getsem_task3
Times getsem_task1 acquired semaphore: 0101
Times getsem_task2 acquired semaphore: 0077
Times getsem_task3 acquired semaphore: 0101
Times getsem_task4 acquired semaphore: 0002
The task waiting Sem with highest Prio 0000

Sensors ON are: NO SENSOR
```

g) Por último se ejecuta todo correctamente hasta que la tarea 4 ocupa todos los recursos, por lo que genera que la tarea 3 genere un error de *timeout* ya que a los 100 *ticks* no ha recibido los recursos.

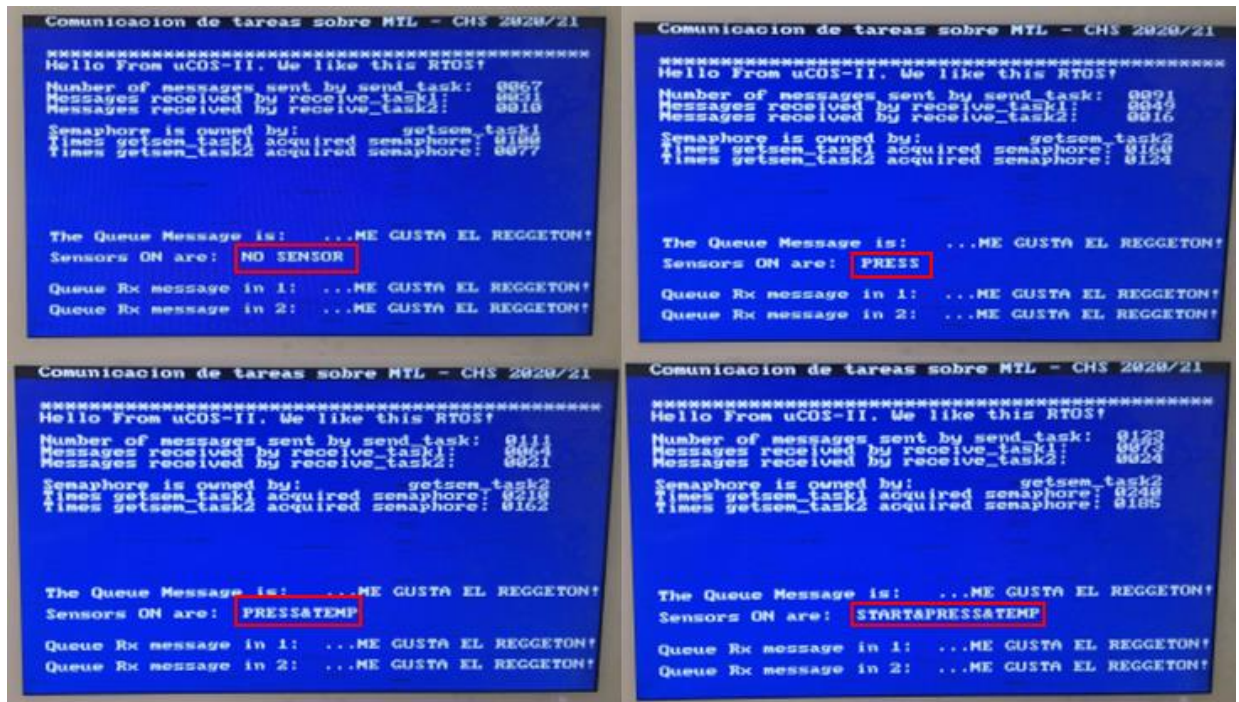
---

```
Hello from Task Init
El motor no esta preparado ...
El motor no esta preparado ...
El motor no esta preparado ...
La tarea 4 se ha comido todos los recursos
[Program Error]: See STDERR
Resource not received in specified time
Program is terminating.
```

### 3. Ejercicio 4

Se observa como las flags van cambiando y en el momento que se activa el swith (Solo se capturan los sets, no los clears) y cuando se cumple el pend se enciende el motor.

```
El valor actual de los switches es 04
El valor actual de los switches es 04
El valor actual de los switches es 04
El valor actual de los switches es 04
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El motor no esta preparado ...
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 06
El valor actual de los switches es 14
El Motor ha arrancado yupiiii...los flags registrados estan a 14
El valor actual de los switches es 14
```



#### 4. Ejercicio 5

Se obtiene el contenido de la cola y su tamaño dentro de las tareas consumidoras tal y como se muestra en la siguiente figura

```
Hello from Task Init
El mensaje es The Queue Message is:
La cola de mensajes se inicia en la posicion de memoria 190324
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254028
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246636
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254028
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246636
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254028
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246636
Receive_task1 ha tomado el mensaje ESTE ES EL MENSAJE SECRETO desde la posicion de memoria 254028
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246636
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254028
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246636
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254028
```

Si añadimos el análisis de la cola, obtenemos la siguiente figura, donde se observa como los consumidores reciben los mensajes generados por el productor. También se aprecia cómo se generan más mensajes que los que se reciben.



```

La cola tiene 7 mensajes y el tamaño de la cola es 30
La cola tiene 8 mensajes y el tamaño de la cola es 30
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254924
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246732
La cola tiene 7 mensajes y el tamaño de la cola es 30
La cola tiene 8 mensajes y el tamaño de la cola es 30
La cola tiene 9 mensajes y el tamaño de la cola es 30
La cola tiene 10 mensajes y el tamaño de la cola es 30
La cola tiene 11 mensajes y el tamaño de la cola es 30
La cola tiene 12 mensajes y el tamaño de la cola es 30
Receive_task1 ha tomado el mensaje ESTE ES EL MENSAJE SECRETO desde la posicion de memoria 254924
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246732
La cola tiene 11 mensajes y el tamaño de la cola es 30
La cola tiene 12 mensajes y el tamaño de la cola es 30
La cola tiene 13 mensajes y el tamaño de la cola es 30
La cola tiene 14 mensajes y el tamaño de la cola es 30
La cola tiene 15 mensajes y el tamaño de la cola es 30
La cola tiene 16 mensajes y el tamaño de la cola es 30
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 254924
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246732
La cola tiene 15 mensajes y el tamaño de la cola es 30
La cola tiene 16 mensajes y el tamaño de la cola es 30

```

Por último en el apartado extra, se comprueba el correcto funcionamiento de la cola en estilo LIFO y del *flush* de la cola en caso de que se detecten fallos como un *overflow* (cola con más de 9 mensajes genera un error que se pone delante de la cola para que pueda ser leído por la primera tarea que pueda procesar el error y continúe con la ejecución).

```

Hello from Task Init
El mensaje es The Queue Message is:
La cola de mensajes se inicia en la posicion de memoria 190496
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 255000
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246808
La cola tiene 0 mensajes y el tamaño de la cola es 30
La cola tiene 0 mensajes y el tamaño de la cola es 30
La cola tiene 1 mensajes y el tamaño de la cola es 30
La cola tiene 2 mensajes y el tamaño de la cola es 30
La cola tiene 3 mensajes y el tamaño de la cola es 30
La cola tiene 4 mensajes y el tamaño de la cola es 30
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 255000
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246808
La cola tiene 3 mensajes y el tamaño de la cola es 30
La cola tiene 4 mensajes y el tamaño de la cola es 30
La cola tiene 5 mensajes y el tamaño de la cola es 30
La cola tiene 6 mensajes y el tamaño de la cola es 30
La cola tiene 7 mensajes y el tamaño de la cola es 30
La cola tiene 8 mensajes y el tamaño de la cola es 30
Receive_task1 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 255000
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246808
La cola tiene 7 mensajes y el tamaño de la cola es 30
La cola tiene 8 mensajes y el tamaño de la cola es 30
La cola tiene 9 mensajes y el tamaño de la cola es 30
La cola tiene 10 mensajes y el tamaño de la cola es 30
La cola tiene 11 mensajes y el tamaño de la cola es 30
La cola tiene 12 mensajes y el tamaño de la cola es 30
Receive_task1 ha tomado el mensaje ESTE ES EL MENSAJE SECRETO desde la posicion de memoria 255000
La cola se ha vaciado
Receive_task2 ha tomado el mensaje ESTE ES EL MALDITO MENSAJE desde la posicion de memoria 246808
La cola tiene 0 mensajes y el tamaño de la cola es 30
La cola tiene 0 mensajes y el tamaño de la cola es 30
La cola tiene 1 mensajes y el tamaño de la cola es 30
La cola tiene 2 mensajes y el tamaño de la cola es 30
La cola tiene 3 mensajes y el tamaño de la cola es 30
La cola tiene 4 mensajes y el tamaño de la cola es 30

```