

[Área personal](#) / [Mis cursos](#) / [SIS_BAS_MIC](#) / [CONVOCATORIA EXTRAORDINARIA JULIO 2022](#) / [Test Bloque 2](#)

Comenzado el jueves, 30 de junio de 2022, 15:46

Estado Finalizado

Finalizado en jueves, 30 de junio de 2022, 16:00

Tiempo empleado 14 minutos 45 segundos


Calificación 8.00 de 10.00 (80%)

Pregunta **1**

Correcta

Se puntúa 1.00 sobre 1.00

Tras la ejecución de la instrucción SPIDrv->Send es necesario realizar una de las siguientes funciones, para que su comportamiento sea el esperado:

- ☐ Retardo de 10 microsegundos
- ☐ Retardo de 1 microsegundo
- ☒ Esperar a que la función Callback asociada al driver señalice que ha recibido el evento
ARM_SPI_EVENT_TRANSFER_COMPLETE 
- ☐ Consultar el estado del pin asociado a la funcionalidad MOSI para decidir si ha terminado la transferencia de información

La respuesta correcta es:

Esperar a que la función Callback asociada al driver señalice que ha recibido el evento
ARM_SPI_EVENT_TRANSFER_COMPLETE

Pregunta **2**

Correcta

Se puntúa 1.00 sobre 1.00

¿Qué valor lógico deben tener las siguientes líneas del controlador del LCD para escribir un comando?

- ☒ A0 = 0 /RD = 1 /WR = 0 
- ☐ A0 = 1 /RD = 1 /WR = 0
- ☐ A0 = 0 /RD = 0 /WR = 1
- ☐ A0 = 1 /RD = 0 /WR = 1

La respuesta correcta es: A0 = 0 /RD = 1 /WR = 0



Pregunta 3

Correcta

Se puntúa 1.00 sobre 1.00

Se desean utilizar las siguientes instrucciones dentro de una rutina de atención a las interrupciones:

- 1.- `osMessageQueuePut(id_Q1, &msg, 0, 0);`
- 2.- `osMessageQueuePut(id_Q1, &msg, 0, 1);`
- 3.- `osMessageQueuePut(id_Q1, &msg, 1, 1);`
- 4.- `osThreadFlagSet(id_Th1, id_flag);`

¿Qué afirmación es correcta?

- ☒ Se pueden utilizar las opciones 1 y 4
- ☐ Se puede utilizar únicamente la opción 1
- ☐ Se pueden utilizar las opciones 2 y 3
- ☐ Se puede utilizar únicamente la opción 4



La respuesta correcta es: Se pueden utilizar las opciones 1 y 4

Pregunta 4

Correcta

Se puntúa 1.00 sobre 1.00

Indique cuál de las siguientes afirmaciones sobre CMSIS-Driver **ES FALSA**:

- ☒ Permite inicializar y gestionar el uso del Timer 1 del microcontrolador
- ☐ Permite inicializar y gestionar el uso del interface I2C del microcontrolador
- ☐ La función Initialize permite asociar una función "callback" que se ejecuta cuando se producen distintos eventos asociados al periférico
- ☐ Puede utilizarse al mismo tiempo junto a funciones de la capa de bajo nivel (LL) y junto a funciones de la capa de alto nivel (HAL) de STMicroelectronics



La respuesta correcta es:
Permite inicializar y gestionar el uso del Timer 1 del microcontrolador



Pregunta 5

Correcta

Se puntúa 1.00 sobre 1.00

Los pines asociados al interface SPI deben configurarse en el fichero:

- ☒ RTE_Device.h
- ☐ RTX_Config.h
- ☐ SPI_STM32F4xx.c
- ☐ stm32f4xx_hal_conf.h



La respuesta correcta es:

RTE_Device.h

Pregunta 6

Sin contestar

Puntúa como 1.00

Con la configuración del fichero RTX_Config.h del entorno de Keil mostrada en la figura, puede afirmarse que:

Option	Value
<input checked="" type="checkbox"/> System Configuration	
Global Dynamic Memory size [bytes]	32768
Kernel Tick Frequency [Hz]	2000
<input checked="" type="checkbox"/> Round-Robin Thread switching	<input checked="" type="checkbox"/>
Round-Robin Timeout	5
ISR FIFO Queue	16 entries
Object Memory usage counters	<input type="checkbox"/>
<input checked="" type="checkbox"/> Thread Configuration	
<input checked="" type="checkbox"/> Object specific Memory allocation	<input type="checkbox"/>
Number of user Threads	1
Number of user Threads with default Stack size	0
Total Stack size [bytes] for user Threads with user-provided Stack ...	0
Default Thread Stack size [bytes]	3072
Idle Thread Stack size [bytes]	512
Idle Thread TrustZone Module Identifier	0
Stack overrun checking	<input checked="" type="checkbox"/>
Stack usage watermark	<input type="checkbox"/>
Processor mode for Thread execution	Privileged mode

- ☐ El tiempo de conmutación entre tareas del sistema operativo es de 2,5 ms
- ☐ El tick del sistema operativo es de 1 ms
- ☐ El tick del sistema operativo es de 500 ms
- ☐ El tiempo de conmutación entre tareas del sistema operativo es de 5 ms

La respuesta correcta es:

El tiempo de conmutación entre tareas del sistema operativo es de 2,5 ms



Pregunta 7

Correcta

Se puntúa 1.00 sobre 1.00

Se tiene una aplicación desarrollada con CMSIS-RTOSv2, configurado por defecto.

La aplicación está compuesta, entre otros elementos, por un timer virtual, cuya inicialización y callback se muestran a continuación.

```
osTimerId_t tim_id1;
static uint32_t exec1;
uint16_t contador=0;

static void Timer1_Callback (void const *arg) {
    contador++;
}

int Init_Timers (void) {
    osStatus_t status;
    exec1 = 1U;
    tim_id1 = osTimerNew((osTimerFunc_t)&Timer1_Callback, osTimerPeriodic, &exec1, NULL);
    if (tim_id1 != NULL) {
        if (status != osOK) {
            return -1;
        }
    }
    osTimerStart(tim_id1, 50U);
    return 0;
}
```

En una parte del código de dicha aplicación se invoca la función `Init_Timers`, y se ejecuta sin devolver ningún error. Una vez transcurridos 15 segundos desde esta llamada la variable `contador` tendrá el valor:

- ☐ 15000
- ☒ 300
- ☐ 0
- ☐ 44



La respuesta correcta es: 300



Pregunta 8

Correcta

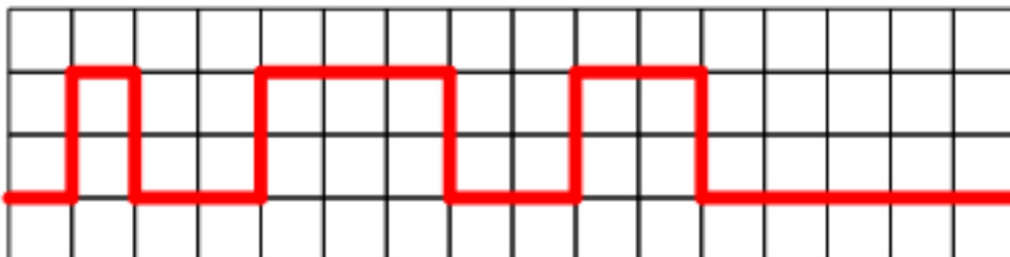
Se puntúa 1.00 sobre 1.00

Se tiene una aplicación desarrollada con CMSIS-RTOSv2, configurado por defecto.

La aplicación está compuesta, entre otros elementos, por una rutina de atención a las interrupciones de una línea GPIO (**configurada para generar interrupciones por flanco de bajada**) y un thread sincronizado mediante un flag con dicha rutina, tal y como se muestra en el siguiente fragmento de código.

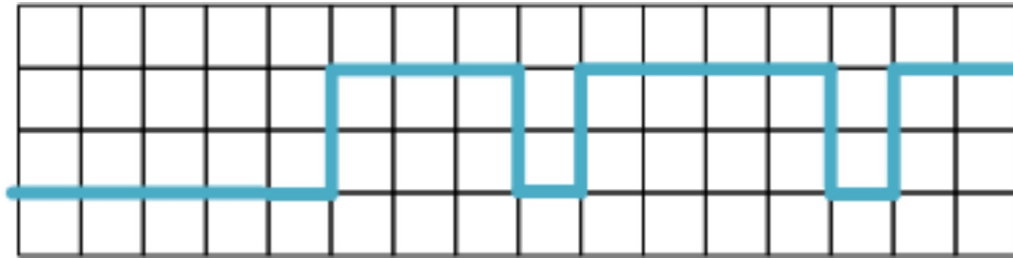
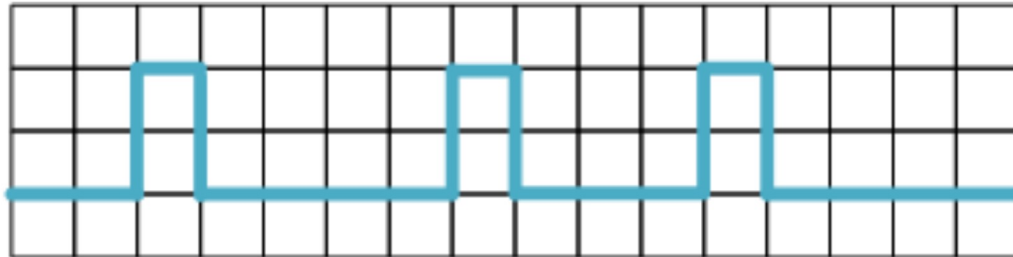
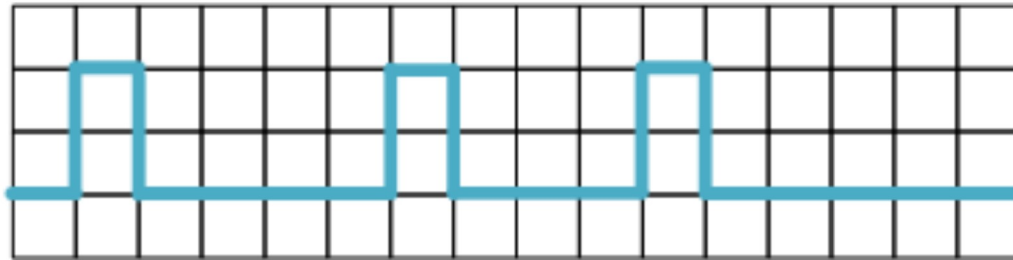
```
void EXTI15_10_IRQHandler(void)
{
    HAL_GPIO_EXTI_IRQHandler(GPIO_PIN_13);
}
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    osThreadFlagsSet(tid_Thread, sig_INT);
    HAL_GPIO_TogglePin(GPIO_B, GPIO_PIN_3);
}
void Thread (void *argument) {
    while (1) {
        osThreadFlagsWait(sig_INT, osFlagsWaitAny, osWaitForever);
        osDelay(100);
        HAL_GPIO_WritePin(GPIO_B, GPIO_PIN_3, GPIO_PIN_RESET);
        osThreadYield();
    }
}
```

Si la forma de onda de la señal externa conectada a la línea de interrupción (GPIO_PIN_13) es la siguiente:



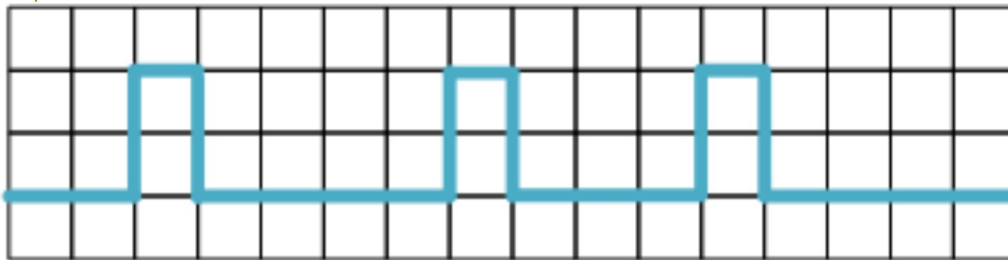
Indique qué opción representa la evolución temporal del pin 3 del puerto B (considere que cada división de la cuadrícula equivale a 100 ms y que antes de comenzar a ejecutarse el código mostrado, **el pin 3 del puerto B se ha puesto a cero**)

☐

☐☒☐

✓

La respuesta correcta es:



Pregunta 9

Correcta

Se puntúa 1.00 sobre 1.00

En la ejecución de la siguiente instrucción:

```
status = osMessageQueueGet(mid_MsgQueue, &msg, NULL, 25U);
```

¿Qué sucede si no hay ningún elemento en la cola?

- ☒ El thread pasa al estado BLOCKED si no hay una elemento que recuperar y espera 25 ticks del sistema
- ☐ El thread espera a que exista un elemento, lo recupera de la cola y espera 25 ticks del sistema
- ☐ El thread pasa al estado BLOCKED indefinidamente
- ☐ El thread está en el estado BLOCKED hasta que lee 25 elementos de la cola

✓

La respuesta correcta es: El thread pasa al estado BLOCKED si no hay una elemento que recuperar y espera 25 ticks del sistema



Pregunta **10**

Sin contestar

Puntúa como 1.00

Para la API CMSIS RTOSv2, utilizada con el dispositivo STM32F429ZI, indique qué afirmación es correcta:

- ☐ La ejecución de los threads se va asignando en función de la prioridad de los mismos y de un mecanismo de conmutación basado en una cola FIFO
- ☐ Los estados en los que puede encontrarse un thread del sistema operativo son Running, Ready, Blocked e Inactive
- ☐ En cada instante sólo puede haber un thread en el estado Inactive
- ☐ El tiempo de conmutación entre tareas depende del Timer 0

La respuesta correcta es:

Los estados en los que puede encontrarse un thread del sistema operativo son Running, Ready, Blocked e Inactive

[◀ Test Bloque 1](#)[Notas Julio 2022 ▶](#)