AIN SHAMS UNIVERSITY
FACULTY OF ENGINEERING
ICHEP

Spring, 2023 **Course Code:** ***CSE 411*** ***Time allowed: 1:00 Hr.***

Real-Time and Embedded Systems Design- Midterm

The Exam Consists of **Three** Questions in **Three** Pages. Maximum Marks: 20 Marks 1 / 3

Question (1): Fill in the spaces below (5 marks = 1 mark for each):

1- **To disable all the interrupts before a critical section we write "__asm(_ _ _ _ _ _ _ _ _ _ _ _);"**

2- **In Tiva C Board and on Exception Entry, The Stack grows towards _ _ _ _ _ _ _ _ addresses.**

3- **On Exception entry, _ _ _ _ _ _ _ registers are pushed to the stack when FPU is not working.**
(Complete with a number)

4- **On Exception return, the 6th register that is popped from the stack is _ _ _ _ _ _ _ _ _ _ _**

5- **In FreeRTOS, A task can delete itself by passing _ _ _ _ _ _to vTaskDelete() in place of a valid task handle.**

Question (2): (4 marks = 2 marks for each):

1-	What is the difference between **vTaskDelay()** and **vTaskDelayUntil()**?
2-	What are the arguments of the below API.s: a- uxTaskPriorityGet b- vTaskPrioritySet
	uxTaskPriorityGet( )
	vTaskPrioritySet( )

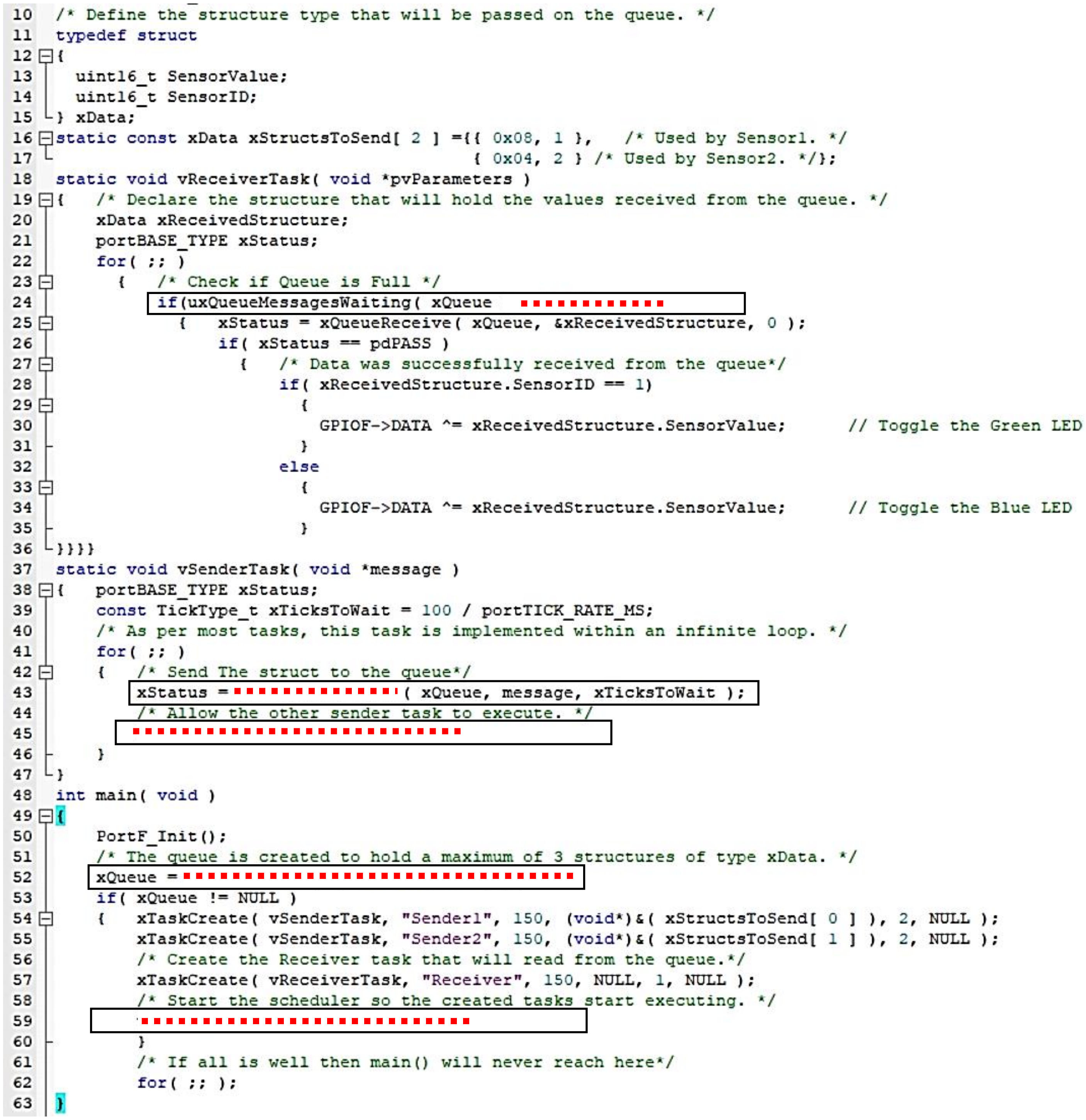
AIN SHAMS UNIVERSITY
FACULTY OF ENGINEERING
ICHEP

Spring, 2023 | **Course Code: *CSE 411*** | ***Time allowed: 1:00 Hr.***

Real-Time and Embedded Systems Design- Midterm

The Exam Consists of **Three** Questions in **Three** Pages. Maximum Marks: 20 Marks

Question (3): a) Read the comments and fill in the spaces (7.5 marks = 1.5 mark for each):



```c
/* Define the structure type that will be passed on the queue. */
typedef struct
{
  uint16_t SensorValue;
  uint16_t SensorID;
} xData;
static const xData xStructsToSend[ 2 ] ={{ 0x08, 1 },   /* Used by Sensor1. */
                                        { 0x04, 2 } /* Used by Sensor2. */};
static void vReceiverTask( void *pvParameters )
{   /* Declare the structure that will hold the values received from the queue. */
    xData xReceivedStructure;
    portBASE_TYPE xStatus;
    for( ;; )
      {   /* Check if Queue is Full */
       if(uxQueueMessagesWaiting( xQueue   ▪▪▪▪▪▪▪▪▪▪▪▪▪
          {   xStatus = xQueueReceive( xQueue, &xReceivedStructure, 0 );
              if( xStatus == pdPASS )
                {   /* Data was successfully received from the queue*/
                    if( xReceivedStructure.SensorID == 1)
                      {
                        GPIOF->DATA ^= xReceivedStructure.SensorValue;      // Toggle the Green LED
                      }
                    else
                      {
                        GPIOF->DATA ^= xReceivedStructure.SensorValue;      // Toggle the Blue LED
                      }
}}}}
static void vSenderTask( void *message )
{   portBASE_TYPE xStatus;
    const TickType_t xTicksToWait = 100 / portTICK_RATE_MS;
    /* As per most tasks, this task is implemented within an infinite loop. */
    for( ;; )
    {   /* Send The struct to the queue*/
        xStatus =▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪( xQueue, message, xTicksToWait );
        /* Allow the other sender task to execute. */
        ▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪
    }
}
int main( void )
{
    PortF_Init();
    /* The queue is created to hold a maximum of 3 structures of type xData. */
    xQueue =▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪
    if( xQueue != NULL )
    {   xTaskCreate( vSenderTask, "Sender1", 150, (void*)&( xStructsToSend[ 0 ] ), 2, NULL );
        xTaskCreate( vSenderTask, "Sender2", 150, (void*)&( xStructsToSend[ 1 ] ), 2, NULL );
        /* Create the Receiver task that will read from the queue.*/
        xTaskCreate( vReceiverTask, "Receiver", 150, NULL, 1, NULL );
        /* Start the scheduler so the created tasks start executing. */
        ▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪▪
    }
    /* If all is well then main() will never reach here*/
    for( ;; );
}
```

AIN SHAMS UNIVERSITY
FACULTY OF ENGINEERING
ICHEP

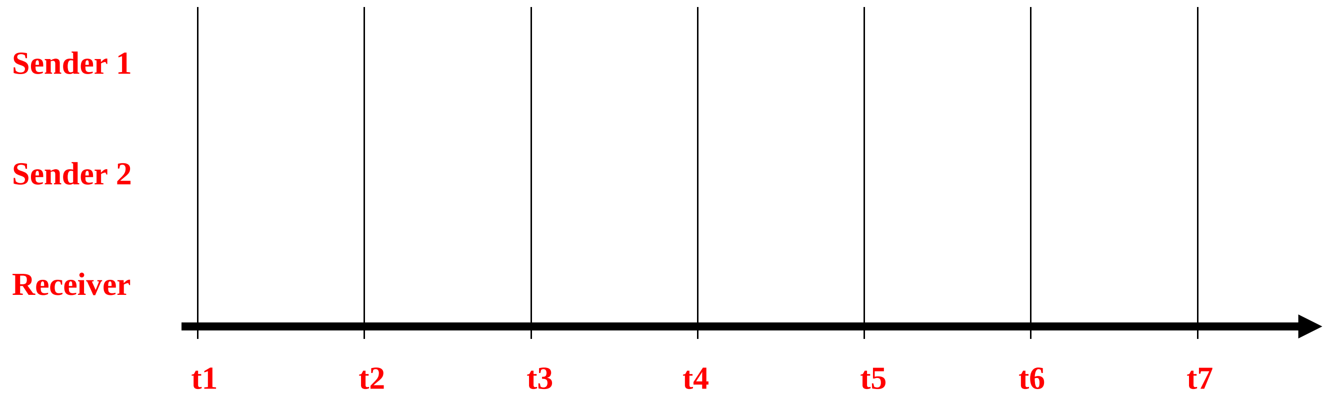
Spring, 2023 **Course Code: *CSE 411*** ***Time allowed: 1:00 Hr.***

Real-Time and Embedded Systems Design- Midterm

The Exam Consists of **Three** Questions in **Three** Pages. Maximum Marks: 20 Marks

Line 52	
Line 59	
Line 43	
Line 45	
Line 24	

b) Draw the Timing Diagram for the above snippet of code. **(3.5 marks)**



Exam Committee:
Prof. Dr. Sherif Hammad

Best Wishes