Universiti Tun Hussein Onn Malaysia

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER I
SESSION 2018/2019

COURSE NAME	:	REAL-TIME EMBEDDED SYSTEM
COURSE CODE	:	BEH 42003
PROGRAMME CODE	:	BEJ
EXAMINATION DATE	:	DECEMBER 2018 / JANUARY 2019
DURATION	:	2 HOURS 30 MINUTES
INSTRUCTION	:	ANSWER **ALL** QUESTIONS

THIS QUESTION PAPER CONSISTS OF **FIVE (5)** PAGES

Q1 (a) Define real-time system.

(2 marks)

(b) Differentiate periodic and aperiodic tasks in real time systems.

(2 marks)

(c) Real-time systems can be classified as hard or soft real-time systems. Discuss hard real time and soft real time systems with one example each.

(3 marks)

(d) One of the main characteristics of a real time system is to achieve predictable behaviour. Relate this characteristic to the real time embedded systems that discussed in **Q1 (c).**

(3 marks)

Q2 (a) LED 1 and LED 2 are directly connected to the digital pin 6 and pin 7 of an Arduino Uno microcontroller that based on sinking mode, respectively. A push button switch is connected to the digital pin 8 of the microcontroller that based on pull-down resistor concept so that the logic signal is low when the push button swich is released. When the push button switch is released, the LED 1 will be turned off, and vice versa. Besides, both LEDs are always in opposite states, i.e. when LED 1 is on, the LED 2 will be switched off, and vice versa.

(i) Sketch a schematic for the microcontroller that connected to the LED and the push button.

(5 marks)

(ii) Determine the logic state for digital pin 6 and pin 7 to switch ON both of the LEDs, and the logis state for digital pin 8 when the push button is pressed.

(2 marks)

(iii) Construct a complete C-statement code including *void setup ( )* and *void loop ( )* subroutines for controlling the LED based on the push button switch state, i.e. when the push button switch is released, the LED will be turned off, and vice versa.

(6 marks)

(b) A Bluetooth module is connected to the digital pin 0 and digital pin 1 of an Arduino (UNO) microcontroller as a transceiver; while an accelerometer sensor is connected to the analog pin A4 and A5 of the microcontroller.

(i) Distinguish synchronous and asynchronous serial communication methods.

(3 marks)

(ii) Discuss the communication methods and protocols for the Bluetooth module and the accelerometer sensor (from **Q2 (b)**) with a concise explanation.

(4 marks)

Q3 (a) Distinguish real time operating system (RTOS) and general purpose operating system (GPOS).

(3 marks)

(b) An Arduino microcontroller has been programmed to implement real time tasks using to handle three threads using FreeRTOS. The code of the operation is given as follows:

```
#include < Arduino_FreeRTOS.h >
unsigned char BTN;

//Task 1

void Task1(void *pvParameters __attribute__((unused))) {
  while (1){
    BTN = digitalRead(3);
    vTaskDelay(25/portTICK_PERIOD_MS);
  }
}

//Task 2
void Task2(void *pvParameters __attribute__((unused))) {
  while (1){
    digitalWrite(4,BTN);
    vTaskDelay(50/portTICK_PERIOD_MS);
  }
}

//Task 3
void Task3(void *pvParameters __attribute__((unused))) {
  while (1){
    Serial.print(BTN);
    vTaskDelay(100/portTICK_PERIOD_MS);
  }
}

void setup()
{
  Serial.begin(9600);
  pinMode(3,INPUT);
  pinMode(4,OUTPUT);
  xTaskCreate(Task1,"T1",64,NULL,2,NULL);
  xTaskCreate(Task2,"T2",64,NULL,3,NULL);
  xTaskCreate(Task3,"T3",64,NULL,1,NULL);
}
void loop() { }
```

(i) Analyse the operation of the Task 1, Task 2, Task 3, and setup functions with real time embedded system prespectives.

(7 marks)

(ii) Rewrite the *void setup( )* rountine so that the priority will be Task 1 > Task 2 > Task 3.

(2 marks)

(c) Propose a real time solution in controlling the LEDs based on the push button switch state that stated in **Q2(a)** by proposing a suitable period and a priority for each proposed task with a concise justification for the proposed period and priority.

(8 marks)

Q4 (a) Differentiate the binary semaphore and mutex in terms of their operation and purposes.

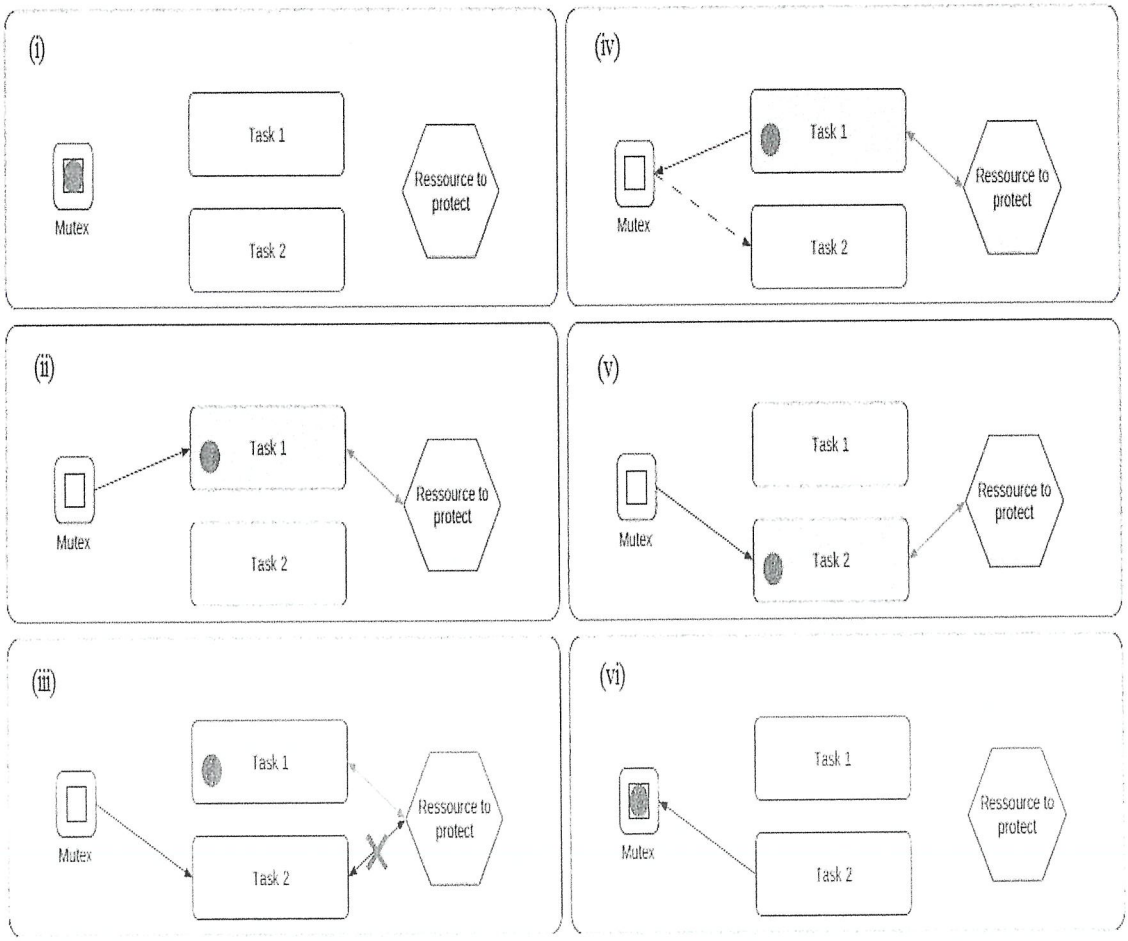
(4 marks)

(b) Differentiate the xSemaphoreTake() and xSemaphoreTakeFromISR() in terms of their operation.

(3 marks)

(c) Discuss the following illustration with related functions of xSemaphoreGive() and xSemaphoreTake() if applicable for a mutex application.

(7 marks)



(d) Propose a solution to make interrupts handlers' execution shorter with a concise discussion.

(6 marks)

Q5 (a) Define the deadline and the maximum elapsed time (Max_E) of a task in temporal scope.

(2 marks)

(b) Differentiate Round-Robin, Cyclic Code, Rate-Monotonic, and Earliest Deadline First Approach scheduling strategies.

(4 marks)

(c) Assume a system has three independent tasks A, B, and C as given in **Table Q5**.

Table Q5: Task Specification

Task	Period (ms)	CPU resources (ms)
A	40	5
B	20	5
C	30	10

(i) If the priority level of Task A > Task B > Task C, draw a task activition diagram for the first 100ms of system operation.

(6 marks)

(ii) Construct a table to state the start delay, elapse time, and completion time for each task.

(6 marks)

(d) Another way to analyze the schedulability of the tasks in the **Table Q5** is by using full test of rate monotonic schedulability (RMS).

(i) Re-arrange the new priority level for each task that based on RMS concept.

(2 marks)

(ii) Calculate the worst-case completion time for each task.

(8 marks)

(iii) Comment on the ability of each task to meet its deadline and schedulability.

(3 marks)

– END OF QUESTIONS –