UNIVERSITY OF EAST ANGLIA

School of Computing Sciences

May/June UG Examination 2010/2011

Embedded Systems

CMPE3D01

Time allowed: 3 hours

Answer 4 questions

Notes are not permitted in this examination.

DO NOT TURN OVER UNTIL YOU ARE TOLD TO DO SO BY THE INVIGILATOR

© Copyright of the University of East Anglia

1. (a) Define the terms priority inversion, priority inheritance and priority ceiling inheritance. *[6 marks]*

   (b) Consider the task set shown in Table 1.

Task	Computation Time	Period	Deadline
T1	7	40	18
T2	6	80	68
T3	6	90	72
T4	7	50	42
T5	8	25	11

Table 1 Task Characteristics (milliseconds)

There are three resources:A, B and C. The time required to access each resource is A: 2 ms; B: 3 ms; and C: 1 ms. The tasks access the resources once on each release according to Table 2. You may assume that there

Task	A	B	C
T1	X	X	
T2		X	
T3		X	X
T4	X	X	
T5	X	X	X

Table 2 Resource requirements

are no nested resource accesses (that is, each task can only access one resource at a time).

(i) Using Deadline Monotonic Priority Assignment, assign priorities to T1, T2, T3, T4 and T5. Use larger numbers to represent higher priorities. *[5 marks]*

(ii) Assuming simple priority inheritance, determine the blocking time of each task and compute its worst-case response time using response time analysis. Do any of the tasks miss their deadlines? *[12 marks]*

(iii) Assuming priority ceiling inheritance, determine the blocking time of

each task and compute its worst-case response time using response time analysis. Do any of the tasks miss their deadlines? . *[7 marks]*

2. (a) Briefly explain what needs to be done before a program can use interrupts and why (assume the system uses a Nested Vectored Interrupt Controller). *[8 marks]*

   (b) A European government is considering introducing charges for the use of motorways. One possible mechanism is that detector stations can be set up at regular intervals along all motorways; when a vehicle passes the detector station, its details are logged and a road charge is recorded. At the end of each month the vehicle owner can be sent a bill for his/her motorway usage. Each vehicle will require an interface device called a Motorway Toll Unit (MTU) which interrupts an on-board computer when a detector station requests its details. The computer has a word size of 16 bits and has memory mapped I/O, with all I/O registers 16 bits in length. Several registers are used for the interface to the detection station. The registers comprise

   - a "control and status" register (CSR),
   - an "input data buffer" register (IDBR), and
   - an "output data buffer" register (ODBR)

   The structure of the CSR register is shown in Table 3 and the register resides at hex address `0xFFF0`.

   Interrupts are vectored, and the address of the interrupt vector associated with the detector station is hex `0x60`. The hardware priority of the interrupt is 4. After an interrupt has been received, the read-only input register (IDBR), which is located at hex address `0xFFF2` contains the basic cost (in an integer number of Euros) of using the current stretch of motorway. The computer software must respond to the interrupt and pass its ownership details to the detector station via the write-only output data buffer register located at hex address `0xFFF4`. The ownership details can be read from a read-only register located at hex address `0xFFF6`. A dash board display informs the driver of the cost of using the motorway.

Bits	Meaning
0	enable device
1	when set, the value found in the IDBR is used as the current charging rate
2-5	not used
6	interrupt enable
7-11	not used
12	error bits (0 = no error)
13-15	not used

Table 3 CSR Structure

It can be accessed by a register located at hex address `0xFFF8`.

(i) Define a C code data structure for the interface device such that each register can be accessed using the syntax:

```
MTU->CSR; MTU->IDBR; etc.
```

Explain why the type qualifier **volatile** should be used in the definition. *[10 marks]*

(ii) Write a device handler for the Motorway Toll Unit. The handler should respond to interrupts from the interface device and be responsible for sending the correct vehicle details. It should also read the data register containing the current cost of the road usage and display the current total cost of the journey on the vehicle's dashboard. *[12 marks]*

3. (a) Discuss the advantages and disadvantages of developing in assembly language. Which parts of applications are usually written in assembly language, and why? *[6 marks]*

(b) Describe the ARM Cortex-M3 Register structure. *[8 marks]*

(c) The ARM Cortex provides two operating modes and two privilege levels. Explain why this is useful. *[4 marks]*

(d) The assembly code subroutine in Listing 1 can be invoked as

```
wait();
```

to produce a fixed time delay. An application requires an assembly language routine that will produce a variable delay. The value of this delay is to be determined by passing a parameter to the assembly language subroutine.

```
                AREA delay, CODE, READONLY
                ENTRY
                EXPORT  wait            ;
wait            LDR     R3, val         ;
loop            SUB     R3,R3,#1        ;
                CMP     R3,#0           ;
                BNE     loop            ;
                BX lr                   ;
val DCD         1000000                 ;
                END
```

Listing 1 Fixed Delay

(i) When a C program calls a ARM assembly language subroutine how are parameters and return results passed between the calling program and the function being called? *[6 marks]*

(ii) Modify the assembly language subroutine so that it accepts a unsigned short giving the delay value. *[6 marks]*

4. (a) What is an embedded system? Give a definition and list six example systems. . *[5 marks]*

(b) Microcontroller architectures are used in many embedded applications. Describe typical features of a microcontroller and give examples of the peripherals that it might provide. *[4 marks]*

(c) Many embedded systems target real-time applications. What characterises a real-time system? Give three examples of embedded real-time systems. *[5 marks]*

(d) Real-time operating systems play an important role in the design of real-time systems.

(i) How do real-time operating systems differ from general purpose operating systems? *[2 marks]*

(ii) Describe the features you might expect to be provided by a real-time operating system kernel and explain why they are useful. *[6 marks]*

(iii) If a procedure or function is described as *reentrant* what does this mean? Why are reentrant functions particularly important when a real-time kernel is used? *[6 marks]*

5. (a) Describe the structure of a parallel I/O port by drawing a suitable diagram. *[12 marks]*

(b) A number of control registers are provided to configure individual Port I/O pins. Describe possible configurations for I/O bits and give an example of a typical control register structure which enables the I/O to be configured. *[12 marks]*

(c) Parallel ports used by microcontrollers can sometimes be configured to provide so-called *alternate functions*. What does this mean and why is it useful? *[6 marks]*

END OF PAPER