The University of Western Australia
Department of Electrical and Electronic Engineering

FIRST SEMESTER EXAMINATIONS **JUNE 2001**

REAL-TIME DISTRIBUTED SYSTEMS 408 623.408

ONLY PAPER

Time Allowed: Three Hours

This papers contains: 12 questions; 7 pages.

Instructions to Candidates:
You are required to answer 10 questions. If you do more than 10 questions the highest marked 10 questions are used in the final mark. Each question carries equal marks and the mark associated with each part of a question is as shown.

QUESTION 1

a) Summarize the important features of at least two of the several case studies considered in this unit, of typical problems found in the design, development and deployment of real-time systems, and mention some of the important lessons that were learned. [5]

b) Identify what it is about distributed object-based infrastructures that makes them different to the traditional client/server model. Illustrate by sketching a diagram to show how this change is impacting the development of the WWW and the benefits this brings. [5]

QUESTION 2

Under a rate-monotonic real-time scheduling policy show via a complete derivation that for two concurrent tasks running on a single CPU the least upper bound CPU utilisation is 83%. What does this mean? [10]

QUESTION 3

a) With the aid of a simple diagram, explain how inter-task synchronisation and data transfer is achieved in an Ada language *rendezvous*. [4]

b) What is a *timed entry call* and how is it used in the Ada language? A code segment will help to illustrate. [3]

c) How are *threads* created in the Java language? A code segment will help to illustrate. [3]

QUESTION 4

a) It is common in real-time systems to assign an executive controller to assist in meeting safety or fault-tolerance objectives. Using a UML sequence diagram, illustrate a typical safety-executive pattern which includes a "watch-dog" timer and at least two sub-systems. [5]

b) The *priority ceiling* algorithm has a very useful property for real time scheduling theory. Describe this algorithm and illustrate its operation with an example to clearly identify this property. [5]

QUESTION 5

a) Consider a distributed system operating a master-slave clock synchronisation algorithm. Suppose the slave clock returns the time $C_j(T)$ and the master clock returns the time $C_i(T)$. The interprocessor communication delays are u_i^j (master to slave) and u_j^i (slave to master). Develop an expression for the clock skew (assuming the slave clock error can be modelled as a zero-mean random noise process) that can be used to correct the slave clock $C_j(T)$? Briefly mention what can be done to improve the clock skew estimation accuracy for the scheme. [5]

b) As an example, suppose a master clock had a time of 00:00:00.532000. After a master to slave communication time of 2 msec, the slave clock reads 00:00:00.578000. At a slave time of 00:00:00.671000, a communication delay of 6 msec is incurred, after which the master clock reads 00:00:00.685000. What clock update is applied to the slave clock after <u>one</u> cycle of the synchronisation algorithm? [2]

c) Consider the situation where the above clock synchronisation algorithm is operating with multiple slave processors. Suppose that all clock drift rates could be bounded at 0.01 sec/sec from specifications, and the clock update rate was at least fine times per second. In addition, the skew error derived for the data of part b) above can be assumed to bound all slave clock update errors. Under these conditions, what is the bound on the maximal clock difference between any slave clock in the distributed system? [3]

QUESTION 6

a) In a real-time distributed system, where time constraints imposed at one node must propagate to another, suppose we have the time constraints TC_1 and TC_2:

TC_1 (*begin*): $3.6 \rightarrow 5.2$	TC_1 (*end*): $4.0 \rightarrow 5.6$	C_{id1}: $0.4 \rightarrow 1.2$
TC_2 (*begin*): $0.2 \rightarrow 5.8$	TC_2 (*end*): $3.3 \rightarrow 8.0$	C_{id2}: $2.2 \rightarrow 4.0$

Draw a diagram that shows both time constraint laxity windows, show TC_1 propagating onto TC_2, and then identify the regions where $TC_1 \sqcap TC_2 \neq \varnothing$ and $TC_1 \prec TC_2$. [6]

b) Suppose we have an incoming time constraint imposed by a remote object of TC_{in} and the constraint imposed on the remote object is TC_{out}, and we have a specification using temporal object relations of the form:

$$TC_{in} \downarrow^{\mathrm{u}} \vee \varnothing \; TC_R \Uparrow TC_{out}$$

Show that the service to be invoked must succeed TC_{in} by less than γ time units, where $\|TC_R\| = \gamma$ by drawing appropriate inferences from the above relations. [4]

QUESTION 7

a) In the *DARTS* approach to the design of real-time systems, what are the main steps in the methodology, and to in common with many other approaches, what area of the methodology is attention specifically directed to. [5]

b) Identify what the *CORBA event service* is, and describe the two main ways in which event channels can be used. [5]

QUESTION 8

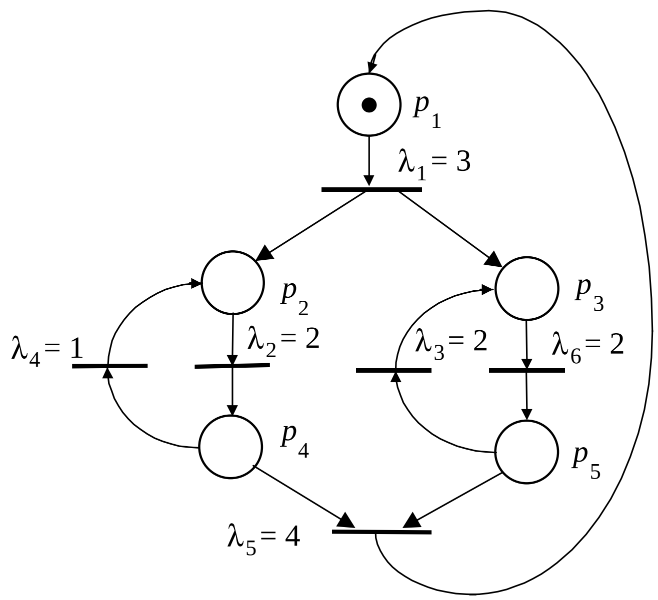
Suppose we have an information system module that processes *three* channels of data concurrently. The channels take T_2 = 4, T_4 = 3 and T_6 = 5 time units to input the data and then take T_3 = 1, T_5 = 3 and T_7 = 5 time units respectively to convert the data. After processing, the three channels are combined to produce a composite signal (taking T_8 = 2) which is passed through a moving average filter (taking T_9 = 3). The filter averages over several data points and eliminates older data points from the average (taking T_{10} = 1). The input data on only the first channel is processed quite differently in another process (taking T_{11} = 3), and combined with the averaged data after filtering but before recording. The recording process takes T_{12} = 3 time units. Input data is acquired synchronously by the system every T_1 = 6 time units.

For the system as described, produce a time-augmented Petri net graph to model the processes, and derive the constraints imposed on the process times using the notion of *safeness in the presence of time*. Can the module achieve the specified time constraints?

[10]

QUESTION 9

Given that the distributed control software for a flexible manufacturing system has the following simplified Stochastic Petri Net (SPN) model, determine the average cycle time of the system.



[10]

QUESTION 10

Consider the case of three periodic tasks (with context switch times included):

Task 1: $C_1 = 20$ ms; $T_1 = 120$ ms

Task 2: $C_2 = 40$ ms; $T_2 = 180$ ms

Task 3: $C_3 = 60$ ms; $T_3 = 360$ ms

a) Apply the *Utilization Bound Theorem* from Real-Time Scheduling Theory to determine if these tasks are schedulable using rate monotonic scheduling.. [2]

b) Suppose the computation time for task 2 increases to 120 msec. Now determine if the tasks are schedulable, and then apply the less conservative *Completion Time Theorem* if required. [4]

c) Suppose an additional interrupt driven aperiodic task is added to the first task set in a) above: Task A: $C_A = 10$ ms; $T_A = 300$ ms
Assuming that all the other tasks are likely to share access to common data at some time, use the *Generalized Utilization Bound Theorem* to determine if the new task set is schedulable. [4]

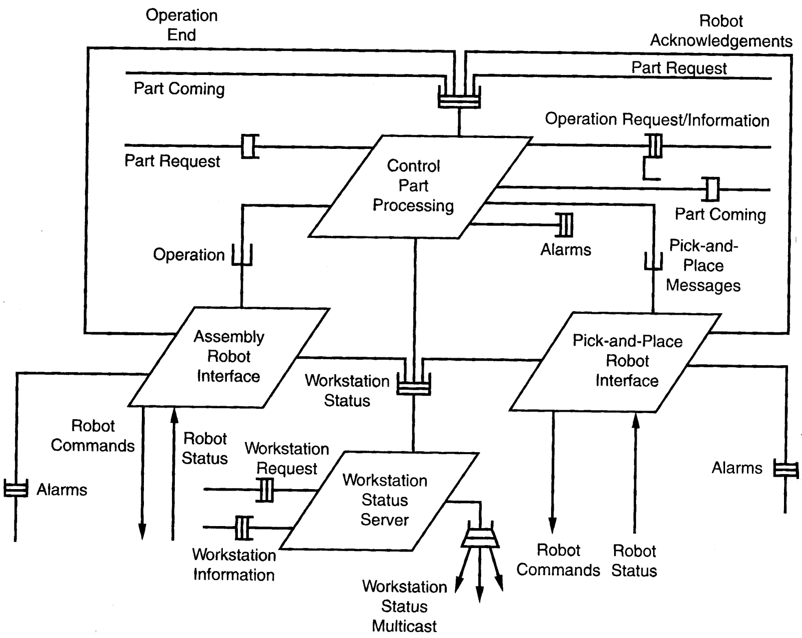
QUESTION 11

a) In a formal specification for the space shuttle launch control software, suppose the Main Engine Start (MES) process must be completed within 8 seconds of the LAUNCH command. And suppose the MES process must itself take at least 5 seconds. During the launch sequence the following safety property must be met: If the launch vehicle restraining clamps permitting LIFTOFF are released after MES completion but within 3 seconds of MES completion, then LIFTOFF can't occur before LAUNCH and the launch vehicle and shuttle will LIFTOFF within 10 seconds. Show how these time constraints could be represented as Real-Time Logic (RTL) axioms, draw the constraint graph and reduce it to show if the safety assertion is consistent with the system specification and can be met. [7]

QUESTION 11 continued over page

QUESTION 11 continued

b) Use any segment of the following task architecture diagram of a robot controller task architecture diagram to briefly discuss the purpose of *event sequence analysis* in real-time performance modelling:



[3]

QUESTION 12

a) In the SAP R/3 business orientated client/server architecture, explain what the BOR is and how is it interfaced with using either of two possible approaches (a simple diagram is helpful to illustrate)? [5]

b) In the *Rhealstone* benchmark, describe with the assistance of diagrams, the two resource based performance benchmarks of *semaphore shuffle* and *deadlock breaking*? [5]