Université d'Ottawa
Faculté de génie

École d'ingénierie et de
technologie de l'information

uOttawa

L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Information
Technology and Engineering

CEG4161/CSI4131 **Sample Midterm Examination**

Time allowed: 75 minutes

Attempt all questions
Questions carry the weights indicated
The total number of points for the examination is 50

Answer the questions in the spaces provided
Use both sides of these sheets if necessary

Name: ______________________ **Student Number:** ____________

Part 1:	
Part 2:	
Part 3:	
Total:	

Part 1 – Short Answer Questions

Please provide short answers to each question.

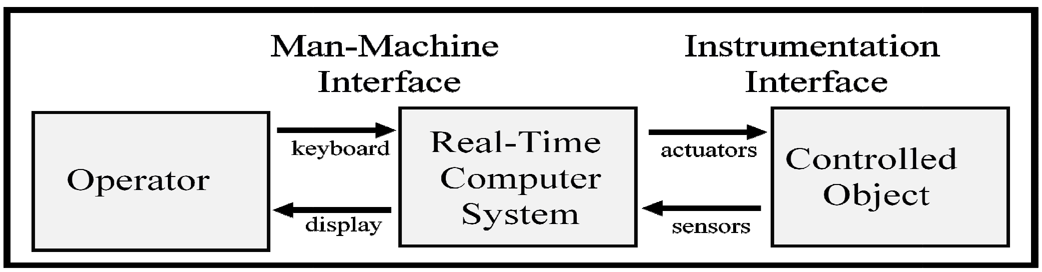
(a) Provide a definition of hard real time system.

Systems where it is absolutely imperative that responses occur within the required deadline

(b) What is the main difference between the typical operating system configuration of a real-time system and an embedded system configuration?

The OS components are integrated with the user program in an embedded system configuration, whereas they are separate in a typical OS configuration.

(c) Draw the composition diagram for a real-time system (involving the operator).



(d) What is the difference between a soft real time system and a firm real time system? Give an example to show the differences.

Firm real-time systems are systems which are soft real-time but in which there is no benefit from late delivery of service. Typical example is the forecast system.

(e) Explain the following statement: "The error detection/correction in hard real time system should be autonomous".

Since the system does not have enough time for user intervention (the user delay is indeterminate and should be avoided).

(f) What is a fail-safe real time system? Give an example of such system.

A system is fail-safe if a safe state can be quickly reached after occurrence of a failure. An example is the real-time service level agreement (SLA) verification.

(g) What is signal conditioning used for? Give an example in the context of real time systems.

Signal conditioning is used to adapt the raw input signal to suit the computer world (filters and amplifiers are examples of signal conditioning).

(h) What is the main difference – from the real-time systems design perspective – between UML and SDL?

SDL is a formal language, UML is not!

(i) Explain the concept of exception propagation in less than three lines.

It is performed by looking up the chain of invokers for handlers.

(j) Provide two advantages for using a centrally controlled clock (for clock synchronization) over using a distributed clock system.

1. No need to implement synchronization of distributed clocks.
2. Highly precise.

(k) Explain the concept of “busy waiting” for process synchronization in a real-time system. What is the main limitation of this method? Propose a solution to overcome this limitation.

Busy waiting means that a process sets a shared variable that is acting as a flag. Another process is waiting for the flag and will proceed whenever the flag is set.
Limitation: low efficiency
Proposed solution: suspend and resume the waiting process.

Part 2 – Analysis Questions (40 points)

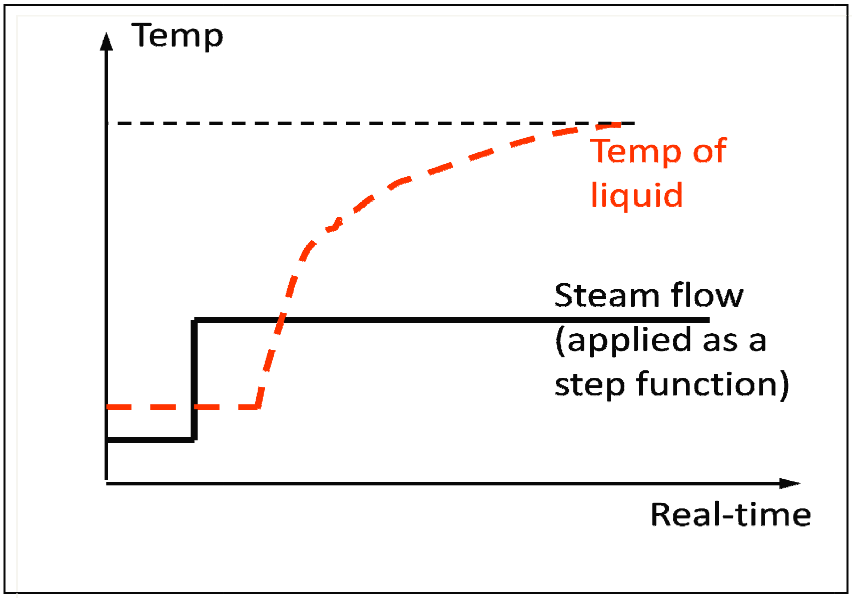
1. Explain the parameters that describe the temporal characteristics of a real-time system (parameters of the control loop)?

Symbol	**Parameter**	**Sphere of Control**	**Description**
d^{object}			
d^{rise}			
d^{sample}			
$d^{computer}$			
$\Delta d^{computer}$			
$d^{deadtime}$			

Answer:

Symbol	**Parameter**	**Sphere of Control**	**Description**
d^{object}	Controlled object delay	Controlled object	Delay after which the measured variable begins to rise
d^{rise}	Rise time of step response	Controlled object	Time until the new equilibrium has been reached
d^{sample}	Sampling period	Computer	length of time between two sample points
$d^{computer}$	Computer delay	Computer	length of time between a sample being taken and the computer generating the control output for the actuator
$\Delta d^{computer}$	Jitter of the computer delay	Computer	difference between maximum and minimum computer delay
$d^{deadtime}$	Dead time	Computer and Controlled object	time interval between the observation of the RT entity and the start of a reaction of the controlled object

2. Consider the following diagram. Show on the diagram how to compute the object delay (d^{object}) and the Rise time *(d^{rise}):*



Please check the lecture notes for the solution of this problem.

3. Consider a system with two lamps. This system has two inputs (from the user), one for each lamp. Each input is meant to turn on the corresponding lamp. When the user turns on a lamp, the other one is turned off automatically. If the user tries to turn on a lamp which is already turned on, then the input is simply ignored.

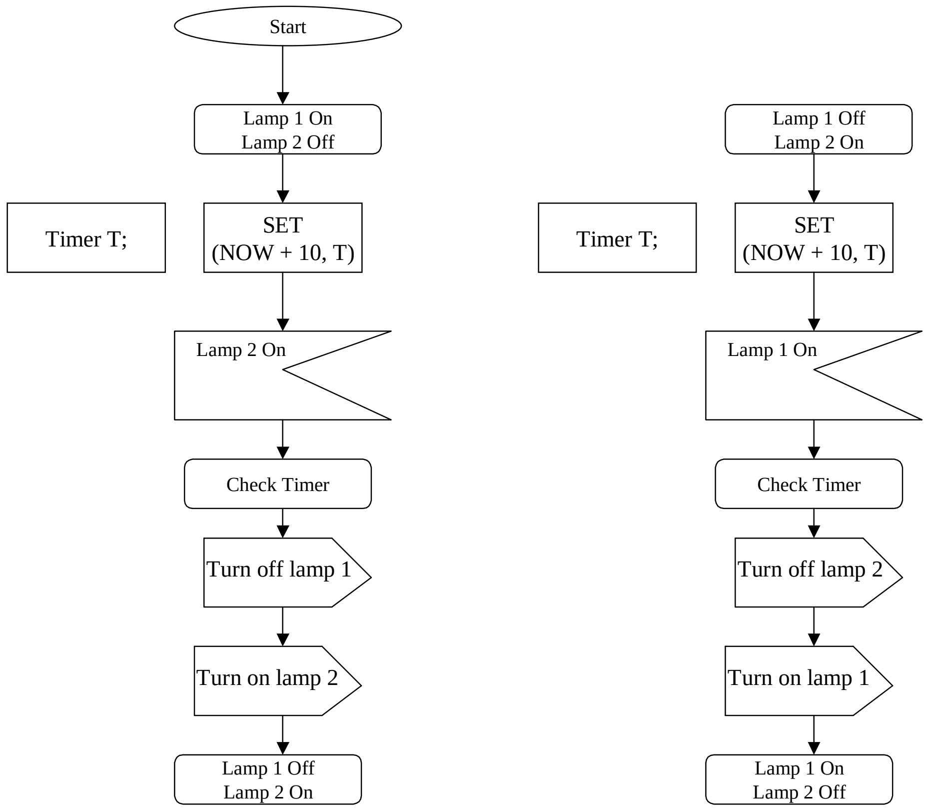
Initially, one of the lamps is turned on and the other lamp is turned off.

(*HINT*: Think first how you would design this system.)

Now we take into account an additional constraint: when the user turns on a lamp, it must stay turned on for at least 10 seconds. If the user requests to turn on the other lamp before this timeout, then the user's request is delayed until the timeout expires and then performed immediately.

Design this system using SDL.

3.


Start
Lamp 1 On
Lamp 2 Off
Timer T;
SET
(NOW + 10, T)
Lamp 2 On
Check Timer
Turn off lamp 1
Turn on lamp 2
Lamp 1 Off
Lamp 2 On
Lamp 1 Off
Lamp 2 On
Timer T;
SET
(NOW + 10, T)
Lamp 1 On
Check Timer
Turn off lamp 2
Turn on lamp 1
Lamp 1 On
Lamp 2 Off

Part 3 – Design Questions

We are planning to design a new light-voice controlling system. The voices will basically coordinate the crossing of a road by blind people. Suppose the crossing will be operated by two coordinated systems: a light system and a voice system. The light controlling system is an ordinary set of traffic lights intended for vehicles, but with the amber light flashing before the lights change over to green. These lights operate in cycles and display the colors in the order: green, continuous amber, red, flashing amber, green. The set of voices is intended for blind people and makes alternately the voices ‘You can cross’ and ‘Stop’. Making the voice ‘You can cross' goes through two phases; a first voice phase ‘You can cross’ followed by a second voice phase ‘Prepare to stop’ before the sound change to the ‘Stop’ voice. The voice ‘Stop’ lasts through exactly the period from the onset of the green light in traffic lights to the end of the continuous display of the amber light. The periods of playing the voice ‘Prepare to stop’ and the flashing amber traffic light are also the same. No two lights or two voices are to be displayed or played together at any time in either set when taken in isolation. To simplify the design, we will assume the crossing is not controlled by blind people and that the lights are operated continuously in approximately fixed cycles. Timing parameters of the display of lights or playing of voices are as follows.

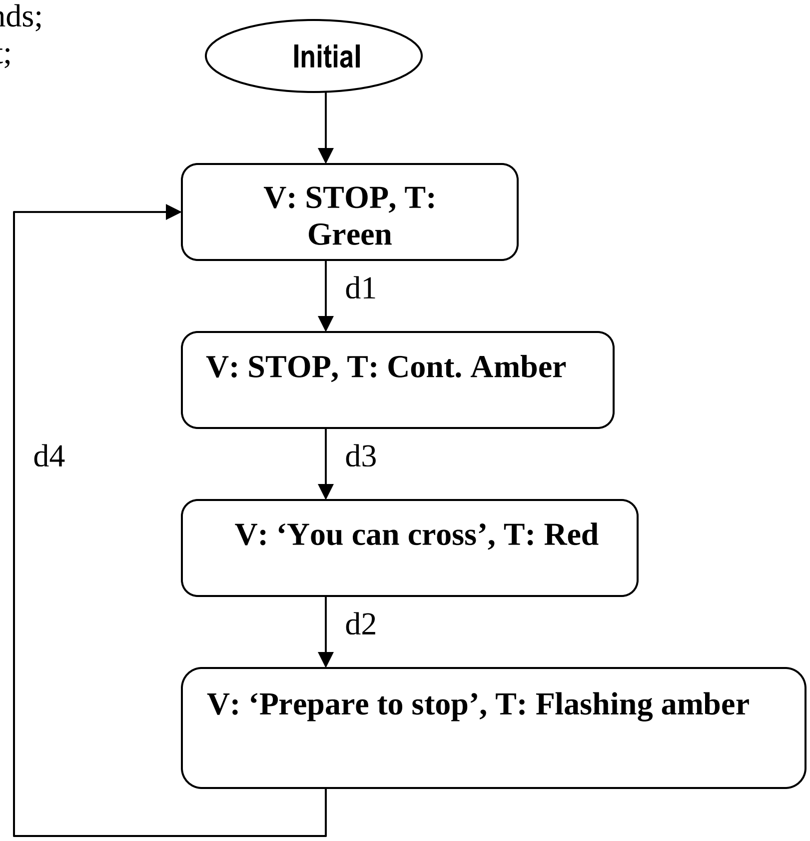
Phase	Delay of phase
Green light	d1
Red light	d2
Continuous amber light	d3
Flashing amber light	d4
Voice 'You can cross'	d5
Voice 'Prepare to stop'	d6
Voice 'Stop'	d7

1) Identify all the components of this new crossing system;

1) System Components:
2) Voice controller
3) traffic light
4) lights controller
5) timer

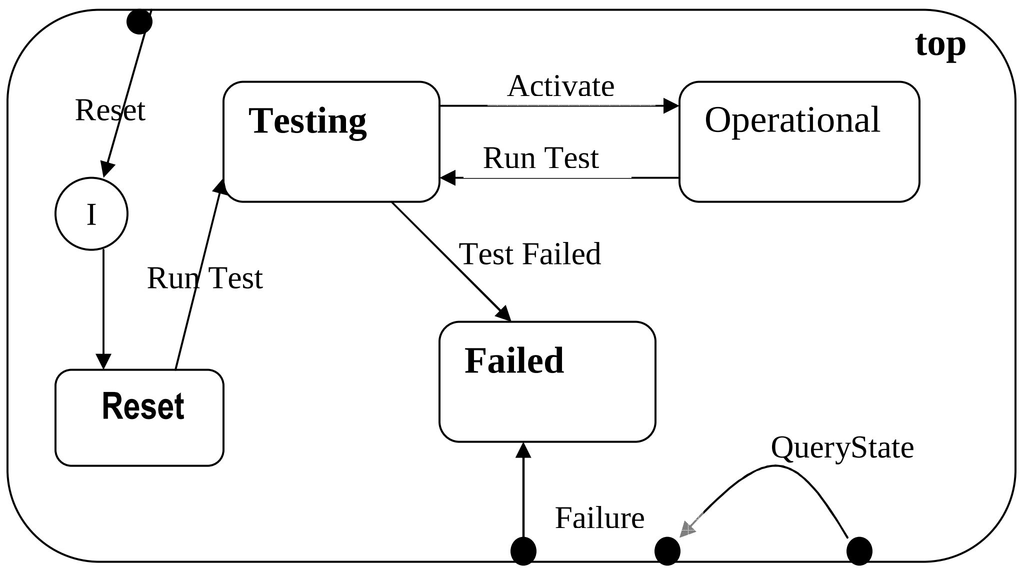
2) Give a global state transition diagram for this system;

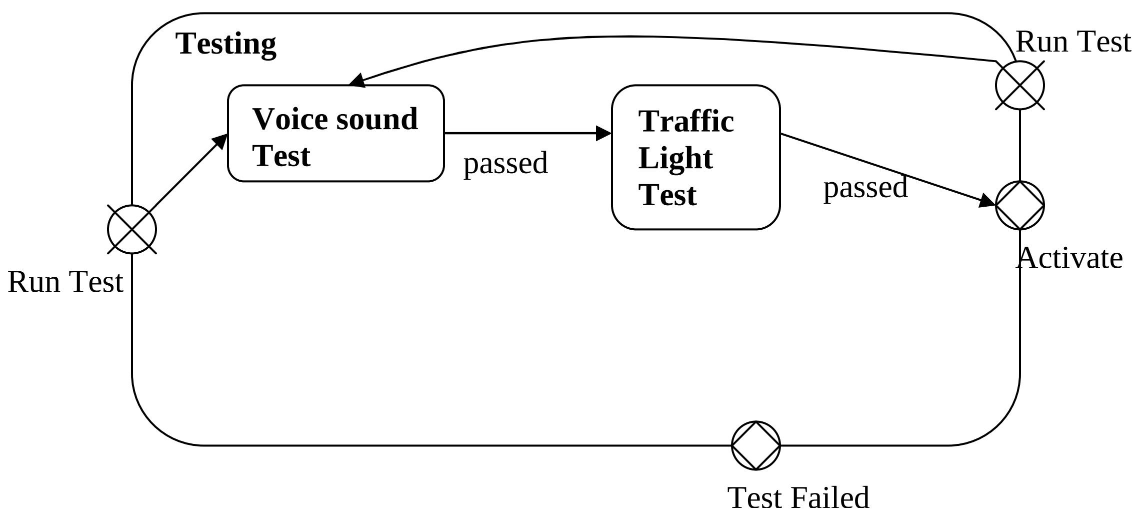
Global state transition diagram:
V: Voice sounds;
T: traffic light;

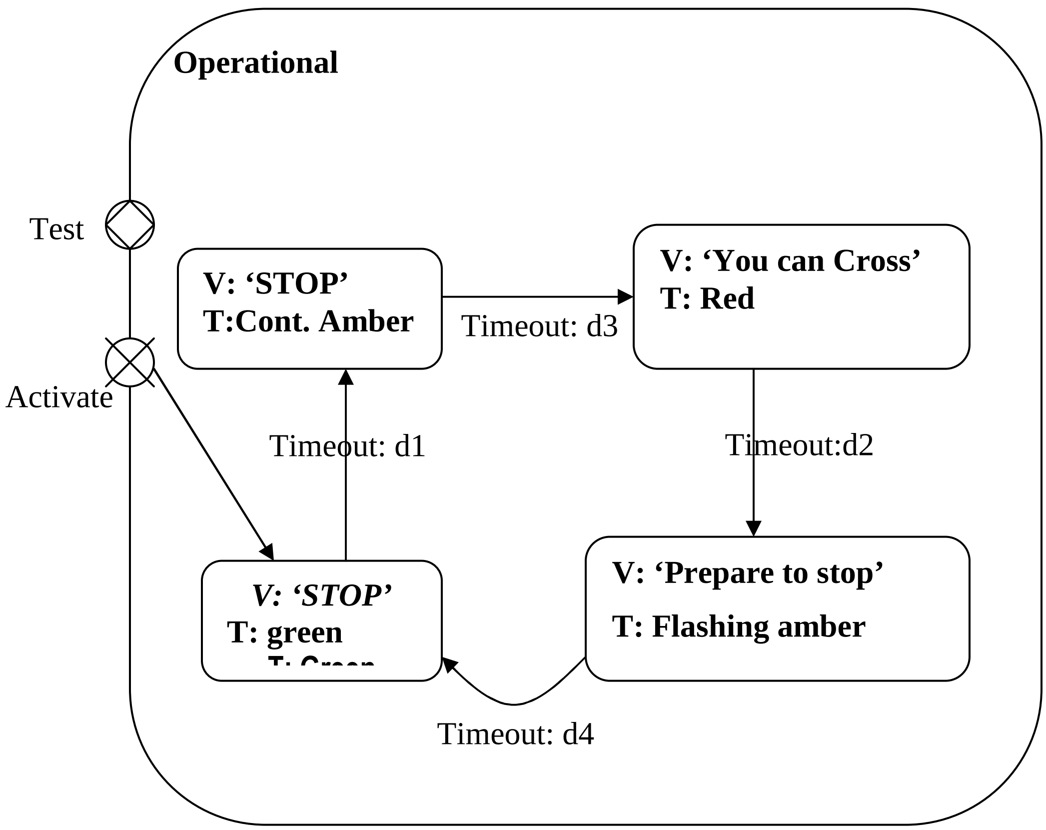


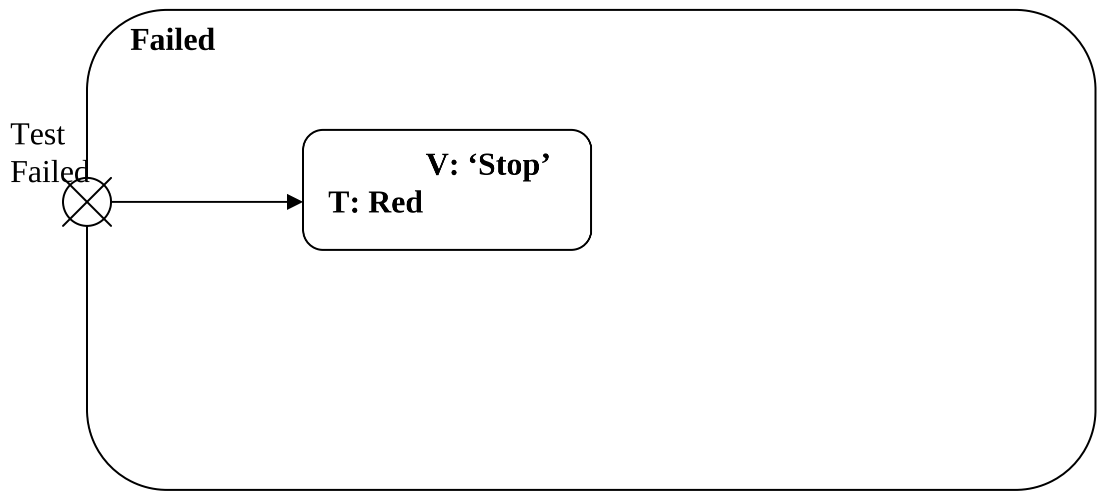
d7 = d1 + d3
d4 = d6
d2 = d5

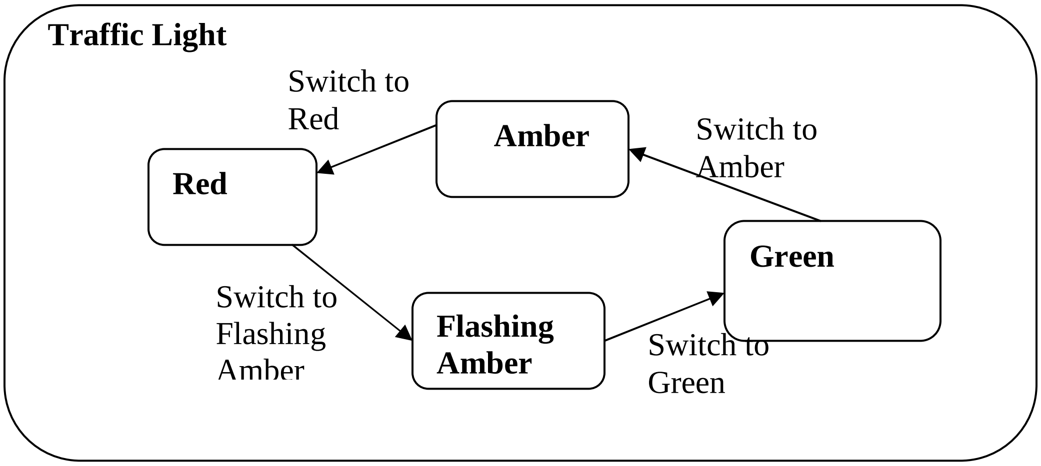
3) Considering the functionality of each part of the system give also the hierarchical ROOM state charts.






Operational
Test
Activate
V: ‘STOP’
T:Cont. Amber
Timeout: d3
V: ‘You can Cross’
T: Red
Timeout: d1
Timeout:d2
V: ‘STOP’
T: green
V: ‘Prepare to stop’
T: Flashing amber
Timeout: d4


Failed
Test
Failed
V: ‘Stop’
T: Red


Traffic Light
Switch to
Red
Amber
Switch to
Amber
Red
Green
Switch to
Flashing
Amber
Flashing
Amber
Switch to
Green