THE UNIVERSITY OF BRITISH COLUMBIA
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
EECE 485 : Digital Instrumentation for Mechanical Systems
2000/2001 Winter Session Term 1

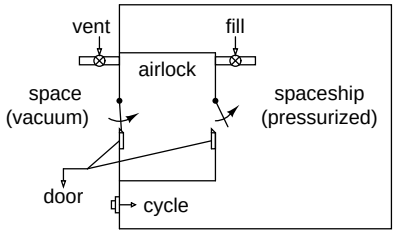
FINAL EXAMINATION
3:30 – 5:30 P.M.
December 7, 2000

This exam has five (5) questions on four (4) pages. The marks for each question are as indicated. There are a total of thirty-six (36) marks. Answer all questions. Write your answers in the exam book provided. Show your work. You may answer the questions in any order. Books, notes and calculators are allowed. You may keep this exam paper.

You can take up to one additional hour to finish this exam if you wish.

Question 1 (10 marks)

This question asks you to design a controller for a spaceship's airlock:



The controller has two outputs: `fill=1` opens a valve that lets air fill the airlock and `vent=1` opens a valve that lets the air out of the airlock.

The controller has two inputs: `door=1` indicates that one of the doors is open and `cycle=1` indicates that a control button is being pushed.

Normally only one of the two valves is turned on. If `fill=1` and `vent=0` the airlock is pressurized. If `fill=0` and `vent=1` the airlock is evacuated (unpressurized).

When the button is pushed (`cycle=1`) *and both doors are closed* the controller should close both valves (`fill=vent=0`). When the button is released (`cycle=0`) the airlock state should be changed: if the airlock was pressurized then it should be evacuated; if it was evacuated then it should be pressurized.

If a door is opened while the button is being pushed, the controller should abort the change of state and return to the previous state.

Design a state machine for the controller. List the inputs and outputs. Choose a sufficient number of states and give a name to each state. Write a table giving the output for each state. Write out a tabular description of the state machine with the following columns: starting state, input, next state.

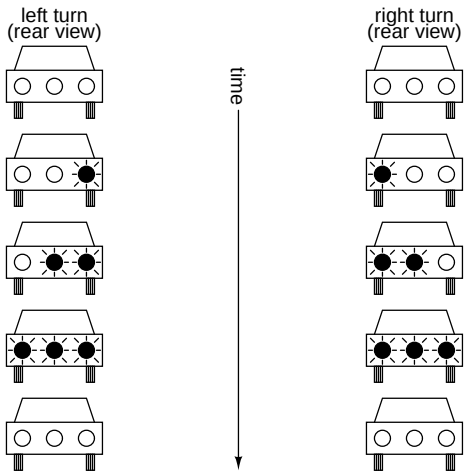
Your design must be unambiguous.

You may use an "X" to indicate that an *input* has no effect.

Hints: my solution uses one more than the minimum possible number of states. This question does not *ask for a state transition diagram.*

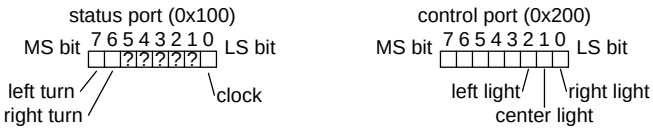
Question 2 (10 marks)

A turn signal uses three lights on the back of a car. The lights are turned on from right to left to indicate a left turn and from left to right to indicate a right turn:



This question asks you to write a C program to control this turn signal.

Your program can read an 8-bit status port at address 0x100, and write to an 8-bit control port at address 0x200:



Bit 0 (the least-significant bit) of the status port alternates between 1 and 0 periodically. Bit 7 of the port is 1 if the driver wants to signal a left turn and bit 6 is 1 if the driver wants to signal a right turn. Bits 7 and 6 are never 1 at the same time. The values of the other status port bits are *undefined*.

The least-significant three bits of the control port turn the lights on (if a bit is set to 1) or off (if set to 0). The other bits of the control port have no effect.

Use the `speek()` and `spoke()` functions from the labs to access the ports.

(a) Write a function, `wait()`, that takes no arguments and returns no value. The function should wait until bit 0 of the status port is 1, then wait until bit 0 is 0, and then return.

(b) Write a function, `signal()`, that takes one `int`eger argument, `left`, and returns no value. If `left` is non-zero the lights should be turned on in the correct sequence for a left turn, otherwise in the correct sequence for a right turn. To obtain the proper delays, `wait()` should be called each time the the lights change.

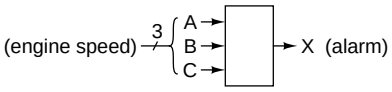
(c) Write a program, `main()`, that takes no arguments and returns no value. The program should run forever. Whenever the driver wants to signal a turn the program should call `signal()` with the appropriate argument.

You do *not* need to do any of the following: include header files, use symbolic constants, or add comments.

Question 3 (4 marks)

This question asks you to design a combinational logic circuit that turns on an alarm when an engine's speed falls outside a certain range.

The circuit has a three inputs (A, B, C) and a single output (X). The three inputs indicate the speed of the engine as a 3-bit binary value with A being the most-significant bit and C the least-significant bit. The output X causes an alarm to sound when it is set to 1:



As shown in the following truth table, if the input is between 2 (010) and 5 (101) then X should be set to 0. If the input is less than 2 or greater than 5 then X should be 1.

inputs			output
A	B	C	x
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Derive the boolean logic equation for the output. Draw a schematic diagram for the circuit using AND, OR and NOT gates. You need not simplify your solution.

Question 5 (12 marks)

(a) For each electric motor requirement in column "A" select the best matching entry in column "B". Write your answers in numerical order and show the number and the selected letter unambiguously.

A		B	
1	two positions	A	series DC motor
2	runs at fraction of supply frequency	B	solenoid
3	high start-up torque	C	stepper motor
4	positioning without feedback	D	synchronous motor

(b) For each electronic component listed in column "A" select the best matching description in column "B". Write your answers in numerical order and show the number and the selected letter unambiguously.

A		B	
1	voltage follower	A	LED plus photodiode
2	SCR	B	high current gain
3	MOSFET	C	conducts in two directions
4	triac	D	controlled by gate voltage
5	optoisolator	E	conducts in one direction
6	Darlington pair	F	high impedance, no gain

(c) You need to control a servo motor with the output from a D/A converter. The servo motor position varies linearly from 0 to 180 degrees for input voltages between 0 and 5 volts. If you need to set the position with an accuracy of 5 degrees, what resolution do you require from the D/A converter?

If the D/A output voltage is between 0 and 5 volts, what accuracy, in bits, would you specify for this D/A converter?