CMPUT 329 - Computer Organization and Architecture II
Midterm Exam — Fall 2003

Prof. José Nelson Amaral
Computing Science Department
University of Alberta

Name:

CMPUT 329 Honor Code

By turning in the exam solution for grading, I certify that I have worked all the solutions on my own, that I have not copied or transcribed solutions from a classmate, someone outside the class, or from any other source. I also certify that I have not facilitated or allowed any of my classmates to copy my own solutions. I am aware that the violation of this honor code constitutes a breach of the trust granted me by the teaching staff, compromises my reputation, and subjects me to the penalties prescribed in Section 26.1 of the University of Alberta 2003/2004 Calendar.

Edmonton, October 20, 2003.

____

Question 1	/25
Question 2	/25
Question 3	/20
Question 4	/10
Question 5	/20
Total	/100
Curving	/100
Rank	/

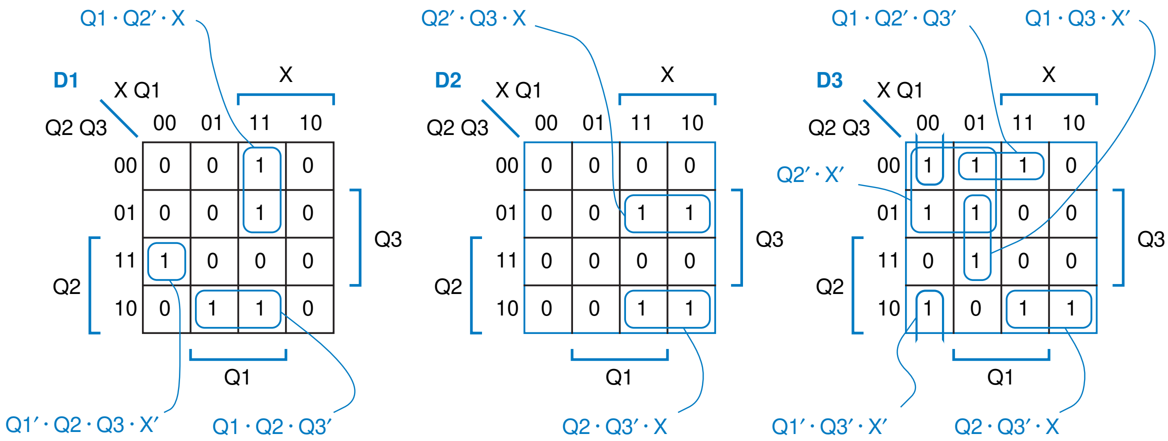

Figure 1: Implementation of functions `D1(C,Q1,Q2,Q3)`, `D2(C,Q1,Q2,Q3)`, `D3(C,Q1,Q2,Q3)`

Question 1 (25 points):

Figure 1 shows the minimal sum-of-products implementation of three logical functions: `D1(C,Q1,Q2,Q3)`, `D2(C,Q1,Q2,Q3)`, `D3(C,Q1,Q2,Q3)`.

a. (10 points) The cost of implementing a circuit is directly correlated to the total number of inputs to all gates used in the circuit. Determine the cost of building a circuit that has `X`, `X'` `Q1`, `Q1'`, `Q2`, `Q2'`, `Q2` and `Q2'` as inputs, and `D1`, `D2`, and `D3` as outputs.

b. (15 points) You are asked to generate minimal implementations for `D1`, `D2`, and `D3` that are static-hazard free. In Table 1 you should list additional terms, if any, that are required to ensure that the logic functions implemented with the Karnaugh maps of Figure 1 are hazard free.

Logic Function	**Additional Terms for Hazard-Free Implementation**
`D1(C,Q1,Q2,Q3)`	
`D2(C,Q1,Q2,Q3)`	
`D3(C,Q1,Q2,Q3)`	

Table 1: Additional terms to make the logic functions of Figure 1 hazard free.

Column I			Column II			Column III	
	wxyz			wxyz			wxyz
0	0000		5,7	01-1	√	5,7,13,15	-1-1
5	0101	√	5,13	-101	√	6,7,14,15	-11-
6	0110	√	6,7	011-	√		
9	1001	√	6,14	-110	√		
10	1010	√	9,13	1-01			
7	0111	√	10,14	1-10			
13	1101	√	7,15	-111	√		
14	1110	√	13,15	11-1	√		
15	1111	√	14,15	111-	√		

Table 2: Prime implicant computation for function `f(wxyz)` using Quine-McCluskey algorithm.

Question 2 (25 points): Table 2 shows the application fo the Quine-McCluskey algorithm to compute the prime implicants of the function $f(w,x,y,z) = \sum_{w,x,y,z}(0,5,6,7,9,10,13,14,15)$.

a. (5 points) How many pairwise comparisons of minterms are necessary to generate Column II of Table 2 from Column I?

b. (5 points) Complete the prime implicant chart of Table 3 by listing all the prime implicants on the first column of the table and marking in each cell of the table the minterm coverage of each prime implicant.

Prime Implicant	**minterm**								
	0	5	6	7	9	10	13	14	15

Table 3: Prime implicant chart for the function `f(wxyz)` of Table 2.

c. (5 points) Which prime implicants are essential?

d. (10 points) Write a minimum sum-of-products for the function $f(w,x,y,z)$.

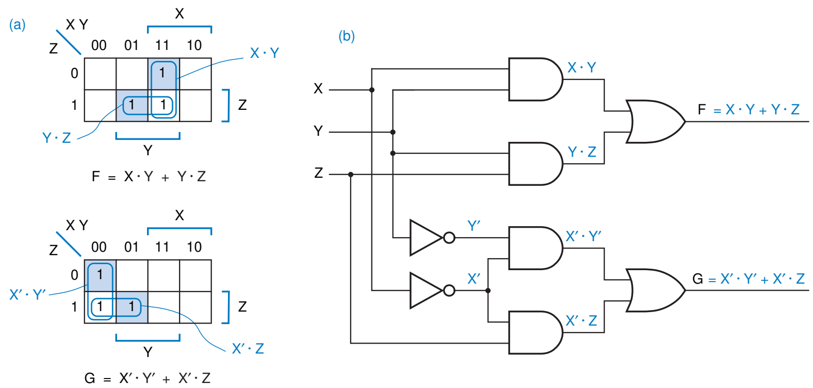

Figure 2: First Implementation for functions `F(X,Y,Z)` and `G(X,Y,Z)`.

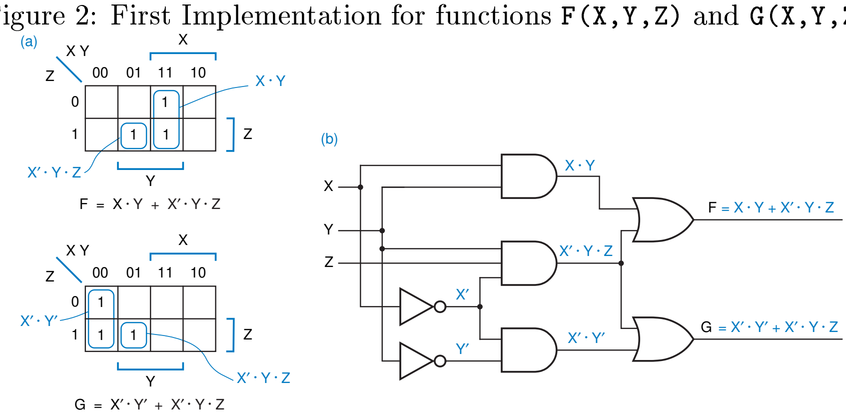

Figure 3: Second Implementation for functions `F(X,Y,Z)` and `G(X,Y,Z)`.

Question 3 (20 points points):

Two design constraints that may become relevant in the implementation of combinatorial circuits are the cost of the implementation and the existence of static hazards. The circuits of Figure 2 and Figure 3 implement the same logic functions `F(X,Y,Z)` and `G(X,Y,Z)`.

a. (10 points) Describe a design constraint that would lead you to choose the first implementation (Figure 2) over the second one (Figure 3) for your circuit.

b. (10 points) Describe a design constraint that would lead you to choose the second implementation(Figure 3) over the first one (Figure 2) for your circuit.

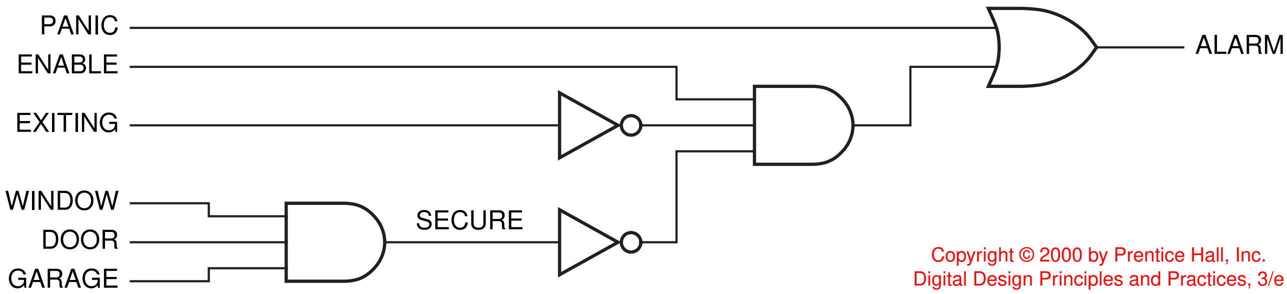

Figure 4: Alarm Circuit.

Question 4 (10 points): Write a minimum sum-of-product expression for the alarm circuit shown in Figure 4.

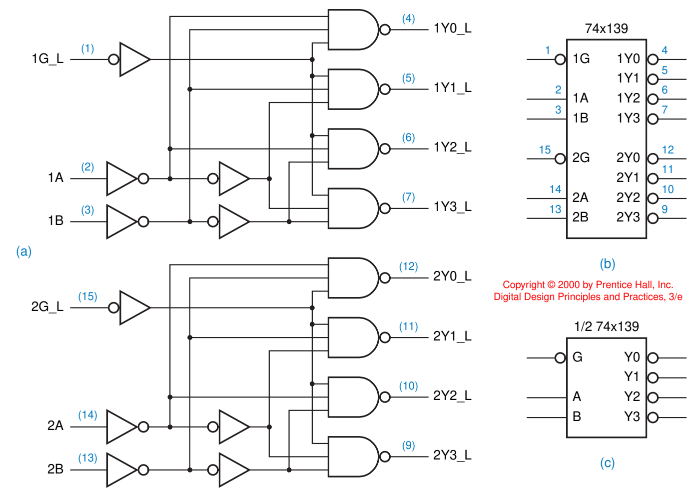

Figure 5: Circuit for the 74x139.

Question 5 (20 points): You are studying a digital system that has a 32-bit data bus,`X`, and a 16-bit address bus, `XA`. The most significant bit in the data bus is `X31`, the least significant bit in the data bus is `X0`. The most significant bit in the address bus is `XA15`, the least significant bit in the address bus is `XA0`. These buses are connected to a 32-to-1 multiplexer as shown in Figure 7.

- (10 points) What is the value of the output `XOUT` if `XEN_L = 0` and the values written in the buses are:
  `X = 0xF0DEFACE`, `XA = 0xFACA`?

- (10 points) Table 4 gives maximum propagation delays, expressed in nanoseconds, for the components used in the 32-to-1 multiplexer of Figure 7. Remember that t_{pHL} is the time between an input change and the corresponding output change when the output is changing from HIGH to LOW.
  What is the maximum propagation delay from any input change to the output for the 32-to-1 multiplexer of Figure 7? Show how you obtained the delay.

Part	From	To	t_{pLH}	t_{pHL}
20	any input	output	15	15
139	any select	output	29	38
	enable	output	24	32
151	any select	`Y`	43	30
	any select	`Y_L`	23	32
	any data	`Y`	32	26
	any data	`Y_L`	21	20
	enable	`Y`	42	32
	enable	`Y_L`	24	30

Table 4: Maximum propagation delays in nanoseconds (ns).

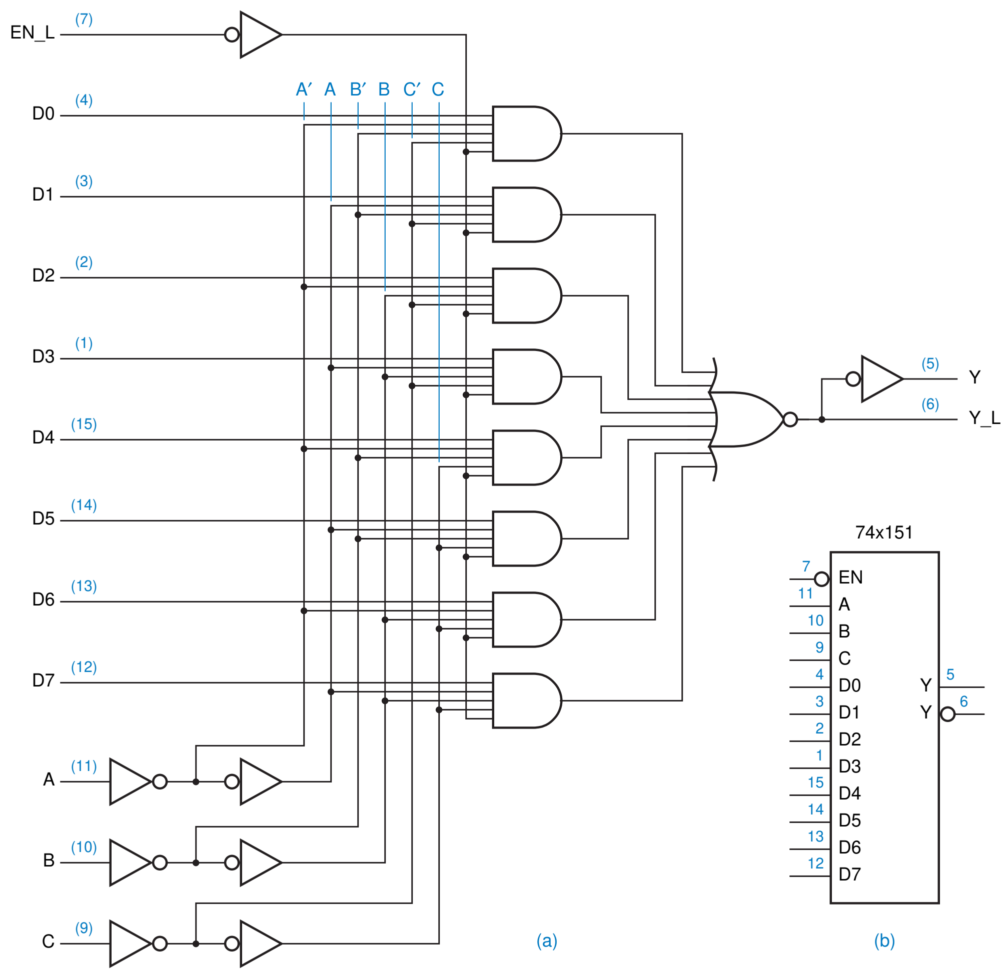

Figure 6: Circuit for the 74x151.

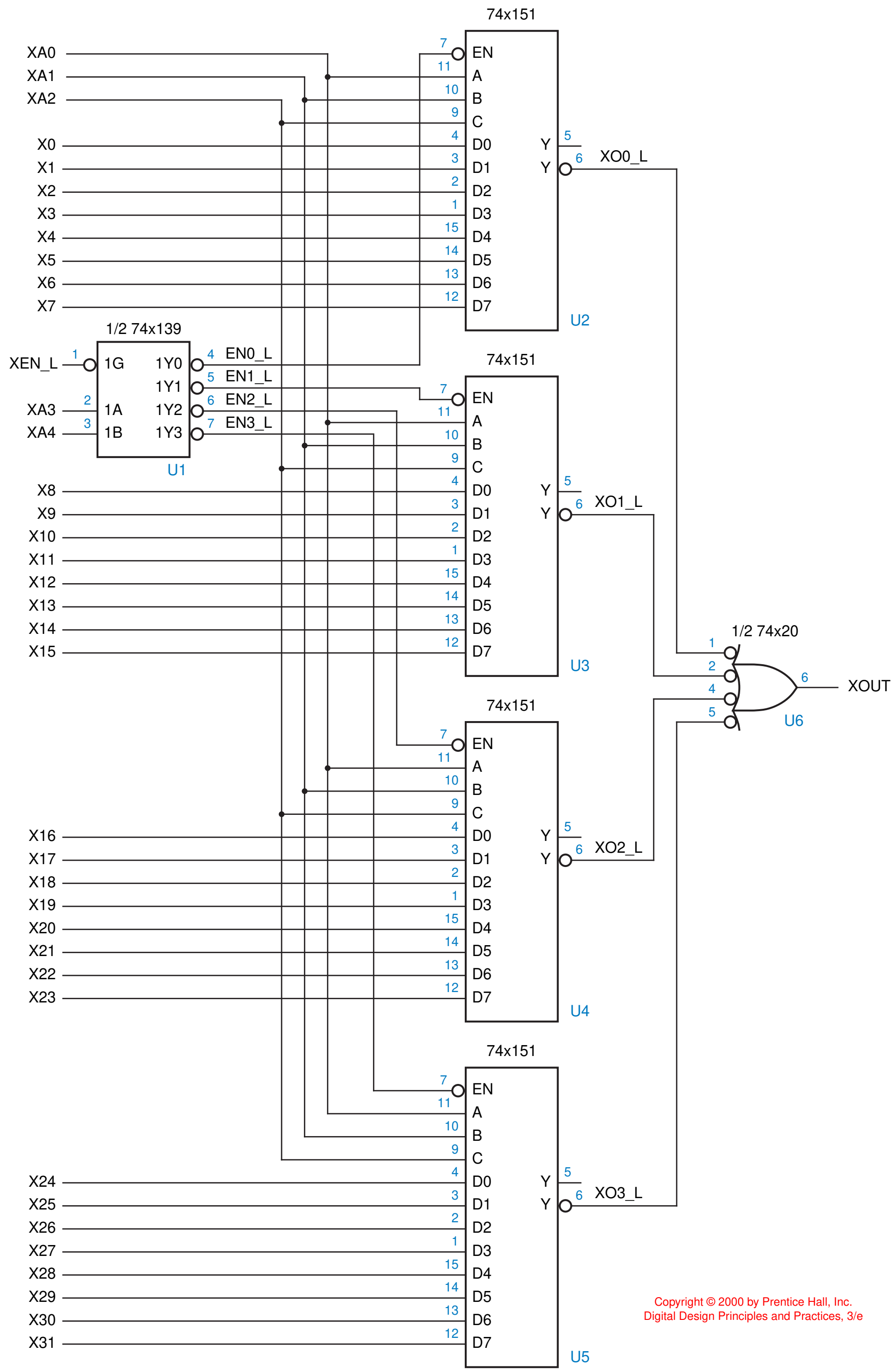

Figure 7: Circuit for a 32-to-1 multiplexer.