SD 192 | Digital Systems | June 14, 2001

Midterm Examination

Professor Paul Fieguth

Time:	3:30-5:00pm Sharp.
Aids Permitted:	*None!* – Calculators / notes / text not permitted
Advice:	Read problems carefully before jumping in.
	Don't expect to have *lots* of time for all questions. Answer what you know and move on! Come back later if you have time. Don't get hung up on one part of one question.
	The grade value for each question is indicated in brackets [ ] next to the question number. I *will* give part marks for relevant statements or insights. Tell me what you know!
	Always apply "sanity-checks" to your answers - do they make sense?

Bonus Question [1%] (**Don't waste your time here unless you're bored!**)

Prove or disprove:

$$a^{\log_d b^c} = b^{\log_d a^c}$$

for positive, real values a, b, c, d.

Problem 1 [43%]

The following Karnaugh maps define Boolean functions $f(x, y, z, w)$ and $g(x, y, z, w)$:

$f(x, y, z, w)$

wx \ yz	00	01	11	10
00	X	1	0	1
01	X	1	1	1
11	0	X	1	1
10	X	1	0	1

$g(x, y, z, w)$

xy \ wz	00	01	11	10
00	X	X	X	0
01	0	X	X	X
11	X	X	1	X
10	X	0	X	X

a) Identify all of the *essential* maxterm groupings for f.

b) Identify all of the *essential* minterm groupings for g.

c) Find a simplest sum-of-products Boolean equation for f. Is the simplest equation unique (yes/no)?

d) Find a simplest product-of-sums Boolean equation for g. Is the simplest equation unique (yes/no)?

For parts (e), (f), and (g), you can assume the availability of inverted inputs. That is, assume that $\bar{w}, \ldots, \bar{z}$ are available; you don't need to show the inverters.

e) Find the simplest NOR–NOR circuit realizations for f and g. Your NOR gates can have any number of inputs.

f) Implement f using an $8 \to 1$ multiplexer and any other required circuitry. Show your work (some sort of tabular approach is required as part of the circuit design).

g) Implement g using a $4 \to 1$ multiplexer and any other required circuitry. Show your work.

Problem 2 [32%]

Give brief answers to the following:

i) We know how to implement combinatorial circuits using decoders or multiplexers. Under what circumstances would you choose one approach as opposed to the other?

ii) Define fan-out.

iii) What is the purpose of a carry-look-ahead circuit? In what sort of circumstances might we want to use this kind of circuit?

iv) In terms of the Boolean operators '+' and '·', give the usual two equations to define each of commutativity, distributivity, and associativity. For each of these three axioms, name a standard mathematical operation (e.g., addition, multiplication, exponentiation, etc.) which satisfies the axiom, and another operation which doesn't.

v) In class we saw that a PLA is more limited in the circuits that it can implement than a ROM. So list reasons why you would ever want to use a PLA instead of a ROM in implementing a circuit.

vi) Using Boolean relations and identities from class (do not use truth tables), prove the following two expressions. Wherever possible, name the identity which you are using; do not skip steps.

$$x + yz = xx + xz + yx + yz$$

$$\left((xy)\,(z\bar{y})'\right)' = \bar{x} + \bar{y}$$

Problem 3 [25%]

In class, we discussed a modular approach to designing a comparator. In particular, we designed a sequence of tests to compare two n-bit unsigned numbers:

a_{n-1}	b_{n-1}	Result		a_{n-2}	b_{n-2}	Result		a_0	b_0	Result
0	0	?	$\Longrightarrow$	0	0	?	$\Longrightarrow \cdots$	0	0	$A = B$
0	1	$A < B$		0	1	$A < B$		0	1	$A < B$
1	0	$A > B$		1	0	$A > B$		1	0	$A > B$
1	1	?	$\Longrightarrow$	1	1	?	$\Longrightarrow \cdots$	1	1	$A = B$

i) Design a sequence of tests, similar to the above, but which compares two n-bit 2's-complement numbers. That is, we are now comparing *signed* numbers.

Show your work. That is, do not just write down your answer; give at least some sense of your rationale or thinking.

ii) Draw the circuit for your *signed* comparator for the case of $n = 2$; that is, to compare two 2-bit numbers.