University of Toronto
Faculty of Applied Science and Engineering

Midterm Test
October 24, 2013

ECE253 – Digital and Computer Systems

Examiner – Prof. Stephen Brown

Print:

First Name . . . . . Last Name . . . . . .

Student Number . . . . . .

1. There are **6** questions and **12** pages. Do **all** questions. The duration of the test is 1.5 hours.
2. **ALL WORK IS TO BE DONE ON THESE SHEETS.** Use the back of the pages if you need more space. Be sure to indicate clearly if your work continues elsewhere.
3. Closed book. No aids are permitted.
4. No calculators are permitted.

1 [8]	
2 [10]	
3 [10]	
4 [10]	
5 [12]	
6 [5]	
Total [55]	

[8 marks] 1. Number bases:

(a) Convert the following decimal numbers to binary.

i. 511

Answer . . . . . .

ii. 1023

Answer . . . . . .

iii. 65535

Answer . . . . . .

(b) Convert the following decimal numbers to hexadecimal.

i. 1023

Answer . . . . . .

ii. 65535

Answer . . . . . .

(c) Convert the following decimal number to octal (base 8).

i. 123

Answer . . . . . .

(d) Convert the following decimal number to base 5.

i. 126

Answer . . . . . .

2. Boolean Algebra:

[10 marks] (a) Write the number of the most relevant Boolean identity (or identities) that can been used to produce the following relations. If the relation is incorrect, meaning that no identity can be used for it, then put an X in the box. Identity numbers are shown at the end of this test.

Identity

	$x + y = x + yz + xz + y$
	$xy + (\overline{x} + \overline{y}) \cdot z = xy + z$
	$x + y = x + yz$
	$((\overline{w \oplus x}) + \overline{y}) \cdot ((\overline{w \oplus x}) + z) = (\overline{w \oplus x}) + \overline{y}z$

[2 marks] (b) Use Boolean algebra to derive a minimal sum-of-products expression for the function f, below. In the boxes on the left specify the number of any identity that you used in that step. Identity numbers are shown at the end of this test. Use as few steps as possible.

Identity

$f = (a\overline{b} + \overline{a}b) \cdot z + (\overline{a}\overline{b} + ab) \cdot z$

	.....
	.....
	.....
	.....

[2 marks] (c) Use Boolean algebra to show that the following minimization is correct. In the boxes on the left specify the number of any identity that you used in that step. Identity numbers are shown at the end of this test. Use as few steps as possible.

Identity

$\overline{k} \cdot x_1x_0 + k \cdot (x_1 + x_0) = x_1x_0 + k \cdot (x_1 + x_0)$

	.....
	.....
	.....
	.....
	.....
	.....

3. Karnaugh Maps:

[10 marks] Consider the function f shown in the Karnaugh map below.

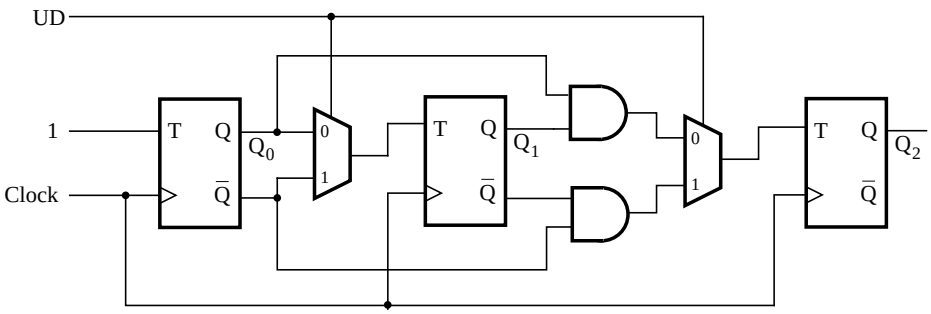
x_3x_4 \ x_1x_2	00	01	11	10
00	1	0	0	1
01	0	d	d	0
11	1	1	d	d
10	1	1	0	1

For each of the expressions below, place a ✓check mark in the box on the left if the expression represents a valid cover for f. You may wish to use the blank space on the opposite page for rough work.

	$\overline{x}_2\overline{x}_4 + \overline{x}_1x_3$
	$(x_2 + \overline{x}_3) \cdot (\overline{x}_3 + x_4) \cdot (x_1 + x_2)$
	$(\overline{x}_2 + x_3) \cdot (x_3 + \overline{x}_4) \cdot (\overline{x}_1 + \overline{x}_2)$
	$\overline{x_2\overline{x}_3 + \overline{x}_3x_4 + x_1x_2}$
	$x_3x_4 + \overline{x}_2\overline{x}_4 + \overline{x}_1x_2x_3\overline{x}_4$
	$\overline{x}_2x_3 + x_2x_4 + \overline{x}_2\overline{x}_3\overline{x}_4$
	$x_2x_4 + x_1\overline{x}_3$
	$\overline{x}_2(\overline{\overline{x}_3x_4}) + \overline{x}_1x_2(x_3 + x_4)$

4. Verilog Code:

[8 marks] (a) Consider the circuit shown below.



Write complete Verilog code that represents this circuit. Your code has to be hierarchical, and has to include separate modules called *Tff* and *mux*2 for the T flip-flops and the 2-to-1 multiplexers. Instantiate these modules along with any other needed Verilog code in a top-level module called *question*4. The ports of this top-level module must be *KEY*[0], *SW*[0], and *LEDR*[2:0], so that the circuit could be implemented on the DE2 board. Connect *KEY*[0] as the clock for the circuit, connect *SW*[0] to the *UD* input, and connect *LEDR*[2:0] to the Q_2, Q_1, Q_0 flip-flop outputs.

Write the Verilog code for the *Tff* and/or *mux*2 in the space below, and write the top-level Verilog module on the following page.

Answer:

Answer for Question 4 Verilog code continued...

[2 marks] (b) State in as few words as possible what the circuit "does". **Answer:**

.....

.....

5. Lab Exercise Question:

[8 marks] You are to design a circuit that can display the letters *b*, *A*, and *d* on the DE2 board 7-segment display called *HEX*0. The inputs to your circuit are the two-bit vector x_1x_0, which should be connected to switches SW_1 and SW_0 on the board. The display should show *b* when $x_1x_0 = 00$, *A* when $x_1x_0 = 01$, and *d* when $x_1x_0 = 10$. Recall that on the 7-segment displays on the DE2 board that the segments are indexed starting from 0 at the top of the display and then incrementing clockwise to 5, and ending with 6 in the middle of the display.

(a) In the space below derive minimum sum-of-products expressions for all seven segments in the display, making use of any don't-care minterms.

Answer:

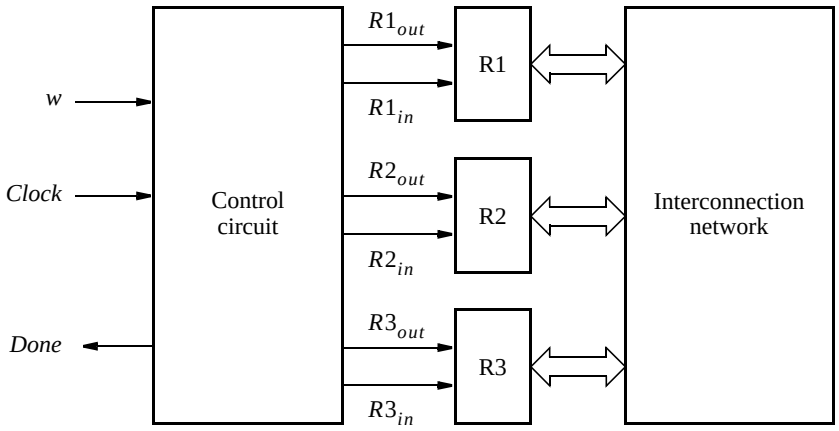
[4 marks] (b) In the space below, write complete Verilog code for your circuit that could be downloaded into the DE2 board.

Answer:

6. Circuit Design:

[5 marks] A computer system usually contains a number of registers that hold data during various operations. Sometimes it is necessary to swap the contents of two registers. Typically, this is done by using a temporary location, which is usually a third register. For example, suppose that we want to swap the contents of registers $R1$ and $R2$. We can achieve this by first transferring the contents of $R2$ into the third register, say $R3$. Next, we transfer the contents of $R1$ into $R2$. Finally, we transfer the contents of $R3$ into $R1$.

Registers in a computer system are connected via an interconnection network, as shown in the figure below. In addition to the wires that connect to the network, each register has two control signals. The Rk_{out} signal causes the contents of register Rk to be placed into the interconnection network. The Rk_{in} signal causes the data from the network to be loaded into Rk. Only the content of one register can be placed on the interconnection network at a time. The Rk_{out} and Rk_{in} signals are generated by a control circuit, which you are to design. Your control circuit should be implemented using D flip-flops and any other necessary logic gates.



Your control circuit has an input called w. Normally, $w = 0$ and your circuit does not need to do anything. But when your circuit needs to start the swap operation, then w will be 1 for exactly one clock cycle, and then w will be 0 again. You are to design a sequential circuit that performs the swap operation over multiple clock cycles starting when $w = 1$. When the swap operation is complete, you should set an output signal called *Done* to 1 for exactly one clock cycle. Draw a schematic diagram of your designed control circuit, showing any flip-flops and logic gates that you used. Label on your circuit the output signals Rk_{out} and Rk_{in} that control the registers, and also label the *Done* output signal.

Provide your answer on the following page.

Answer for Question 6:

Extra answer space for any question on the test, if needed:

Boolean Identities

12*a*.	$x \cdot (y + z) = x \cdot y + x \cdot z$	*Distributive*
13*a*.	$x + x \cdot y = x$	*Absorption*
14*a*.	$x \cdot y + x \cdot \overline{y} = x$	*Combining*
15*a*.	$\overline{x \cdot y} = \overline{x} + \overline{y}$	*DeMorgan's theorem*
16*a*.	$x + \overline{x} \cdot y = x + y$	
17*a*.	$x \cdot y + y \cdot z + \overline{x} \cdot z = x \cdot y + \overline{x} \cdot z$	*Consensus*