CS 207 Digital Logic - Spring 2019
Assignment 2

Deadline: Friday, Apr. 19, 2019

Digital Logic Theory

Write down your answer to the questions on a new sheet with detailed procedures.

1. (0.7 points) Consider the combinational circuit shown in Fig. 3.

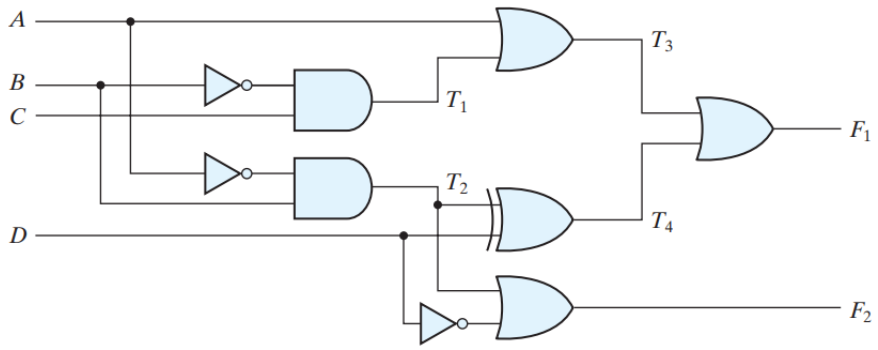

Figure 1: Combinational circuit for Question 1.

   (a) Derive the Boolean expressions for T_1 through T_4. Evaluate the outputs F_1 and F_2 as a function of the four inputs.

   (b) List the truth table with 16 binary combinations of the four input variables. Then list the binary values for T_1 through T_4 and outputs F_1 and F_2 in the table.

   (c) Plot the output Boolean functions obtained in part (b) on K-maps and show that the simplified Boolean expressions are equivalent to the ones obtained in part (a).

2. (0.7 points) Design a combinational circuit with three inputs, x, y, and z, and three outputs, A, B, and C. When the binary input is 0, 1, or 2, the binary output is two greater than the input. When the binary input is 3, 4, 5, 6, or 7, the binary output is one less than the input.

3. (0.7 points) An ABCD-to-seven-segment decoder is a combinational circuit that converts a decimal digit in BCD to an appropriate code for the selection of segments in an indicator used to display the decimal digit in a familiar form. The seven outputs of the decoder (a, b, c, d, e, f, g) select the corresponding segments in the display, as shown in Fig. 2. The numeric display chosen to represent the decimal digit is also shown in Fig. 2. Using a truth table and Karnaugh maps, design the BCD-to-seven-segment decoder using a minimum number of gates. The six invalid combinations should result in a blank display.

Figure 2: Numeric display for Question 3.

4. (0.7 points) (a) Design a half-subtractor circuit with inputs x and y and outputs $Diff$ and B_{out}. The circuit subtracts the bits $y - x$ and places the difference in D and the borrow in B_{out}.

   (b) Design a full-subtractor circuit with three inputs x, y, B_{in} and two outputs $Diff$ and B_{out}. The circuit subtracts $y - x - B_{in}$, where B_{in} is the input borrow, B_{out} is the output borrow, and $Diff$ is the difference

5. (0.7 points) Implement a full adder with two 4×1 multiplexers.

6. (0.7 points) The D latch is constructed with four NAND gates and an inverter in the class. Consider the following three other ways for obtaining a D latch. In each case, draw the logic diagram and verify the circuit operation.

   (a) Use NOR gates for the SR latch part and AND gates for the other two. An inverter may be needed.

   (b) Use NOR gates for all four gates. Inverters may be needed.

   (c) Use four NAND gates only (without an inverter).

7. (0.7 points) A PN flip-flop has four operations: clear to 0, no change, complement, and set to 1, when inputs P and N are 00, 01, 10, and 11, respectively.

   (a) Tabulate the characteristic table.

   (b) Derive the characteristic equation.

   (c) Tabulate the excitation table.

   (d) Show how the PN flip-flop can be converted to a D flip-flop.

8. (0.7 points) A sequential circuit has two JK flip-flops A and B, two inputs x and y, and one output z. The flip-flop input equations and circuit output equation are $J_A = Bx + B'y'$, $K_A = B'x + y$, $J_B = A'x$, $K_B = A + xy'$, $z = Ax'y' + Bx'y'$.

   (a) Draw the logic diagram of the circuit.

   (b) Tabulate the state table.

   (c) Derive the state equations for A and B.

9. (0.7 points) For the following state table

   (a) Draw the corresponding state diagram

   (b) Tabulate the reduced state table.

   (c) Draw the state diagram corresponding to the reduced state table.

   (d) Determine the output sequence for input sequence 01010010111 with the original state table and the reduced state table.

10. (0.7 points) Design a one-input, one-output serial 2's complementer. The circuit accepts a string of bits from the input, and outputs 0's until the first 1 is received. After that, the 2's complement of the input is generated at the output. The circuit can be reset asynchronously to reset the operation.

Present State	Next State		Output	
	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	f	b	0	0
b	d	c	0	0
c	f	e	0	0
d	g	a	1	0
e	d	c	0	0
f	f	b	1	1
g	g	h	0	1
h	g	a	1	0

Figure 3: State table for Question 9.

Digital Logic Experiment

Pack (`tarball`, `zip`, `7z`, etc.) the source and output files as indicated in the respective sections in lab sheets.

1. (0.3 points) Section 2.1 of Lab 5 (32-bit full adder).
2. (0.3 points) Section 3.1 of Lab 5 (parity checker).
3. (0.3 points) Section 2 of Lab 6 (4-bit adder-subtractor).
4. (0.3 points) Section 3 of Lab 6 (three decoders)
5. (0.3 points) Section 2 of Lab 7 (Boolean function with 4-to-16 decoder).
6. (0.3 points) Section 3 of Lab 7 (Boolean function with 8-to-1 multiplexer).
7. (0.3 points) Section 2 of Lab 8 (flip-flops).
8. (0.3 points) Section 3 of Lab 8 (JK flip-flop in UDP).
9. (0.3 points) Section 2 of Lab 9 (sequence detector).
10. (0.3 points) Section 3 of Lab 9 (4-bit binary counter).