University of Toronto, Faculty of Applied Science and Engineering
Department of Electrical and Computer Engineering

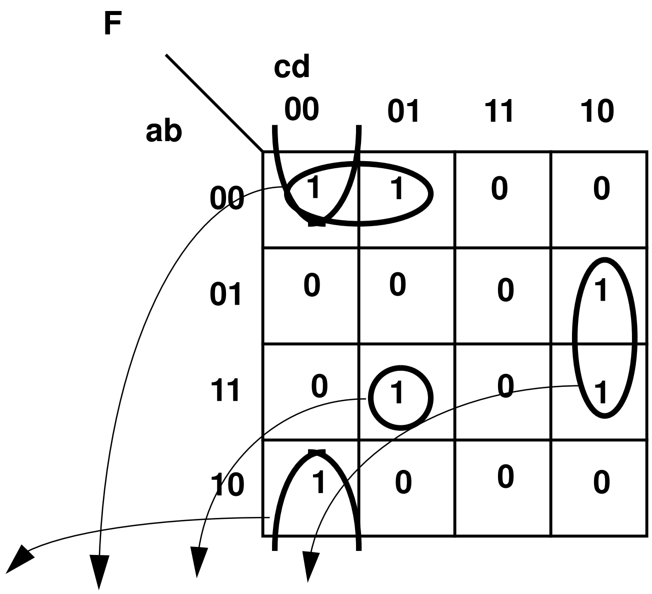
**ELE 241F - Computer Organization** **November 6, 1996**

# Mid-Term Examination Solutions

1. **[10 marks]**

This question deals with the function F(a,b,c,d) = Σ (0,1,6, 8,13,14).

1a. Determine the minimized two-level sum-of-products boolean expression for this function using the Karnaugh map below. The minimal expression is one that has the fewest gates + gate inputs.




Boolean Expression: $f = \overline{b}\,\overline{c}\,\overline{d} + \overline{a}\,\overline{b}\,\overline{c} + a\,b\,\overline{c}\,d + b\,c\,\overline{d}$

- *4 marks, one for each product term; -1 for each incorrect term*
- *if more gates or inputs are used, reduce mark by every additional gate and literal, up to max of 2*

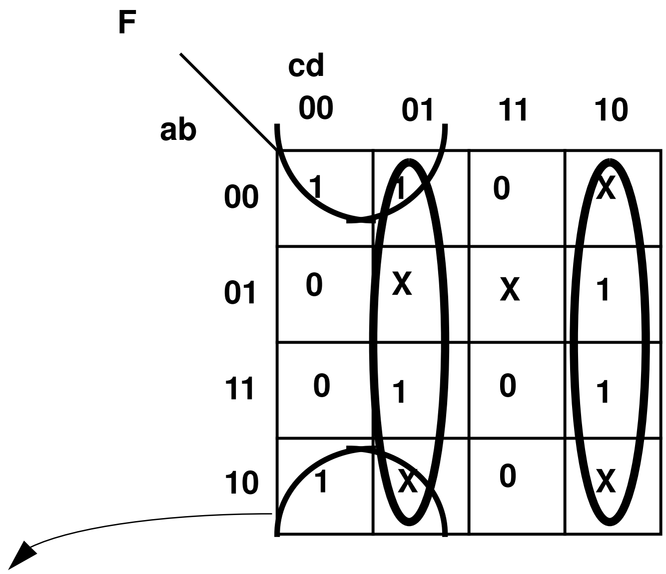
1b. Give the logic gate design for the circuit using AND and OR gates. You may use gates with "bubbles" on the inputs to indicate inversion.

- *2 marks, -1 for each mistake*

Picture not available.

1c. Given that the input combinations abcd = 0010, 0101, 0111, 1001, and 1010 will never occur, give a new minimized two-level sum of products expression for the function, using the same minimization criterion as in question 1a.

Use the following Karnaugh Map:




Boolean Expression: $f = \overline{b}\,\overline{c} + \overline{c}\,d + c\,\overline{d}$

- *1 mark for correct placement of don't cares into Karnaugh Map*
- *3 marks, 1 for each correct product term*
- *if more gates or inputs are used, reduce mark by every additional gate and literal, up to max of 4 (note difference from question 1a in this last marking standard)*

2. **[7 marks]** Consider the following two-level boolean expression:

$$\mathbf{f = a\,b\,c + a\,\overline{b}\,\overline{c}\,\overline{d} + \overline{a}\,\overline{c}\,d + \overline{a}\,b\,\overline{c}}\text{ - \textit{\textbf{13 literals}}}$$

2a. Determine a multi-level boolean expression (using only the Boolean operators AND, OR and NOT) for the function **f** that uses the fewest literals. How many literals were saved compared to the two-level expression?

Expression: $\mathbf{f = a\,(b\,c + \overline{b}\,\overline{c}\,\overline{d}) + \overline{a}\,\overline{c}\,(d + b)}$ - ***10 literals***

- *3 marks, -1 for each additional literal*

Number of Literals Saved: 23

- *1 mark (if calculation correct, regardless of whether the expression is correct; so if the expression is correct this mark can be obtained by correctly determining the difference)*

2b. Give the logic gate design that corresponds to this multi-level expression, **under the following restriction:** the fanin (i.e. the maximum number of inputs) to the AND and OR gates is limited to two inputs. You may use "bubbles" on the inputs of gates to indicate inversion.

Circuit

- *3 marks, -1 for each mistake.*

Picture not available.

3. **[10 marks]** Recall that a binary-coded decimal (BCD) is a four bit number that is used to represent a single base 10 digit - the numbers from 0 to 9. Design and optimize a two-level combinational circuit that determines if a BCD digit is **greater than 8** or **less than 3.** Be sure to make maximum use of don't cares. Answer the questions below to do this.

3a. Identify the inputs and outputs.

Inputs: $b_3b_2b_1b_0$
Outputs: F

- *1 mark, doesn't matter what variables used, as long as is there are four inputs and 1 output, and are used throughout the answer*

3b. Create the truth table and Karnaugh map representation of the function:

| $b_3$ | $b_2$ | $b_1$ | $b_0$ | F |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | X |
| 1 | 0 | 1 | 1 | X |
| 1 | 1 | 0 | 0 | X |
| 1 | 1 | 0 | 1 | X |
| 1 | 1 | 1 | 0 | X |
| 1 | 1 | 1 | 1 | X |

| F: $b_3b_2$ \ $b_1b_0$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 1 | 1 | 0 | 1 |
| 01 | 0 | 0 | 0 | 0 |
| 11 | X | X | X | X |
| 10 | 0 | 1 | X | X |

- *4 marks for the correct construction of the truth table*

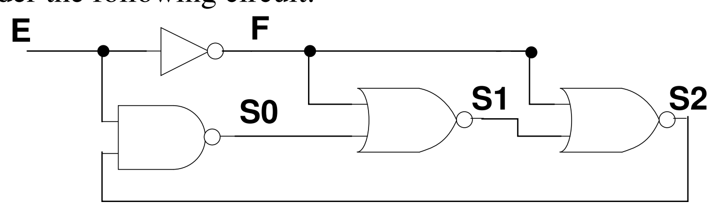
- *1 marks for the correct translation over to the K-map*

3c. Give the minimum cost two-level sum of products boolean expression for the function, where cost is defined as the number of gates + number of gate inputs.

$F = \overline{b}_2\, \overline{b}_1\, b_0 + \overline{b}_3\, \overline{b}_2\, \overline{b}_0$

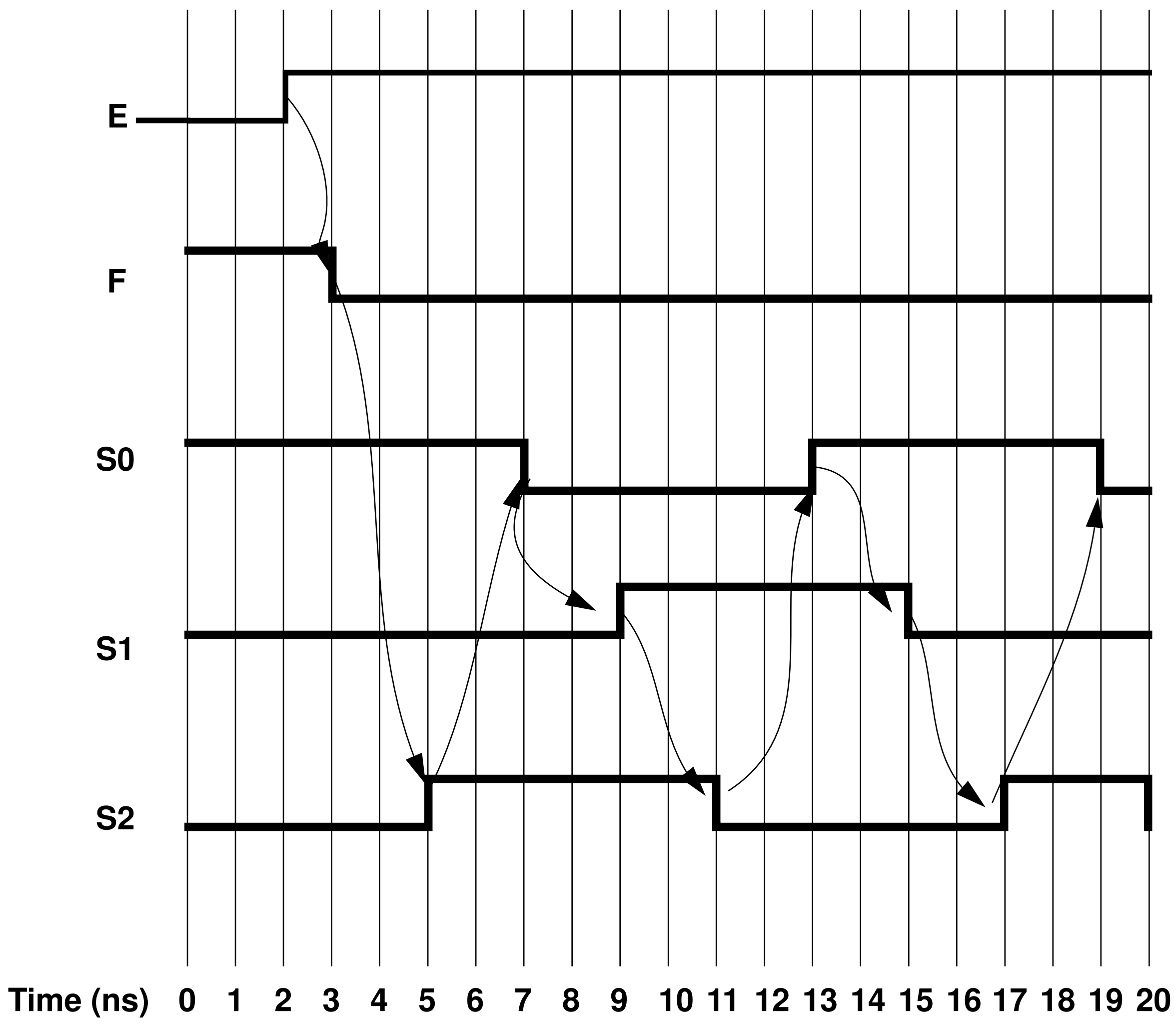
- *4 marks - -1 for each additional gate and input.*

4. **[15 marks]** Consider the following circuit:




Assume each type of gate has the following propagation delay:NOT: 1ns, NAND: 2ns, NOR: 2ns.

4a. Complete the following timing diagram. For each transition (edge), indicate by means of an arrow, which previous transition caused it.




- *[11. marks] For signal F: 1 mark, for signals S0 S1 and S2 - 2 marks each, for each transition arrow 0.5 marks, excluding the one from E to F.*

4b. What is the *period* of oscillation of the signal S2?

Period of Oscillation: = 6 x 2 = 12ns - *1 mar*k

4c. Suppose that the above circuit has **N** NOR gates in a row (rather than two as illustrated in the picture) in addition to the one NAND. For each of the following values of **N**, indicate if the circuit will oscillate or not, and if so give the *period* of the oscillation.

| **N** | 4 | 9 | 10 | 25 |
|---|---|---|---|---|
| Oscillates (Yes or No) | YES | NO | YES | NO |
| Period of Oscillation | 20ns | n/a | 44ns | n/a |

Period = (N+1) x 4

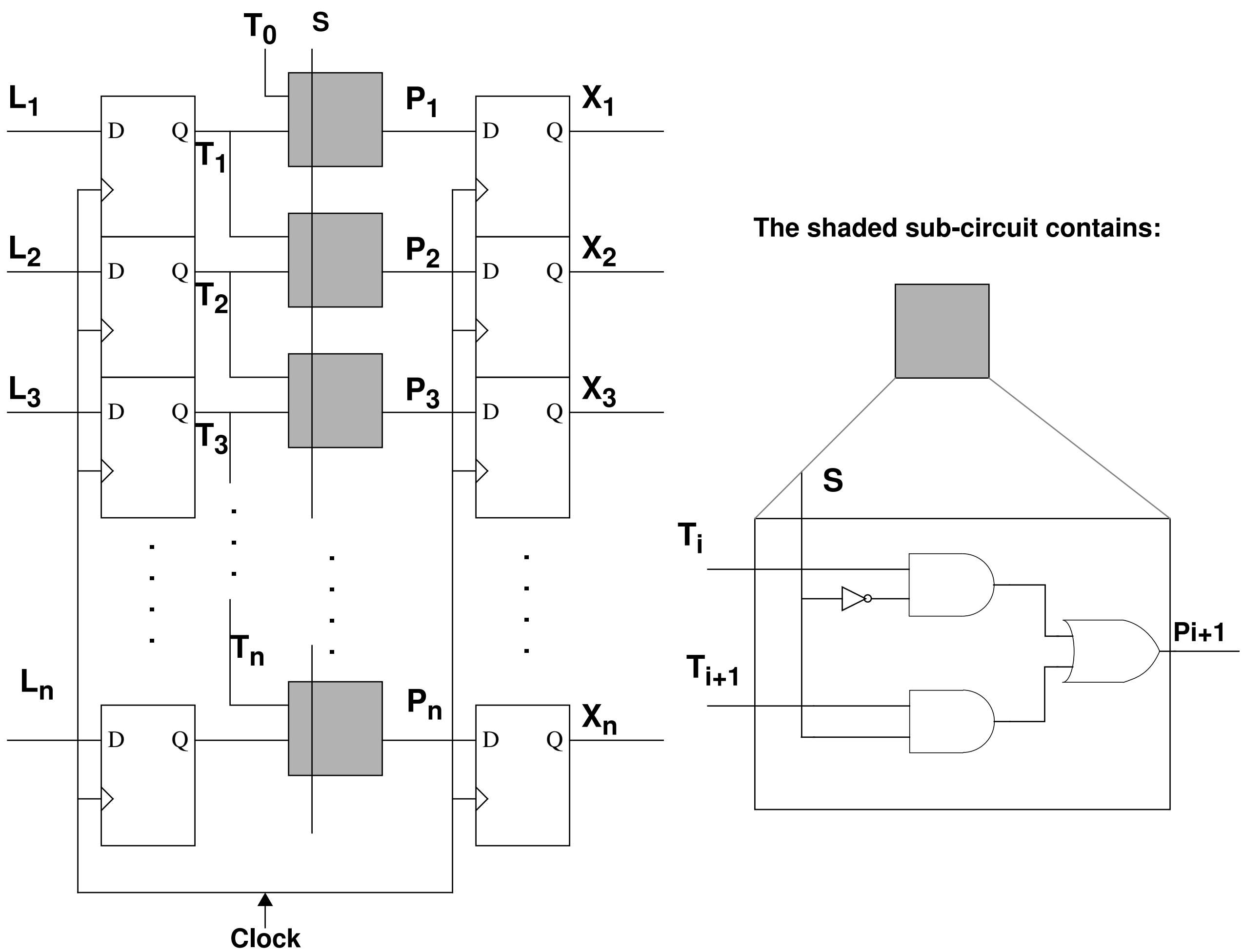
- *2 marks for the top row of table (0.5 marks each)*
- *1 mark for the bottom row (0.5 marks for each of the two answers).*

5. **[8 marks]** Give the complete ABEL-HDL program for the circuit of question 4, using the P18CV8 PAL, including just one test vectors for when E = 0.

```
module OSCILLATOR
title 'Oscillator ABEL-HDL Program'
declarations
OSC device 'P18CV8';
E pin 2;
F pin 18;
S0 pin 17;
S1 pin 16;
S2 pin 15;
equations
@ALTERNATE
F = /E;
S0 = /(E * S2);
S1 = /(F + S0);
S2 = /(F + S1);
test_vectors
([E] -> [F, S0, S1, S2])
[0] -> [1, 1, 0, 0];
end
```

- *8 marks, -1 for each mistake. e.g. must use proper output pins for outputs, input pins for inputs. all parts of the program must be present.*

6. **[10 marks]** For the sequential circuit:




6a. State, in words, what happens to the values of $\mathbf{X_i}$ (i = 1...n), as a function of the $\mathbf{T_i}$ (i= 0...n) after a positive edge on the clock signal arrives, under the following two circumstances:

i. S = 0

The values of $X_i$ becomes what $T_{i-1}$ was prior to the clock edge, i.e. $X_i = T_{i-1}$.

ii. S = 1

The values of $X_i$ becomes what $T_i$ was prior to the clock edge, i.e. $X_i = T_i$.

- *4 marks, (2 marks each.)(labelling of circuits could make these answers different)*

6b. Given the timing values in the table below, determine the minimum *period* that the clock can have for which the system will still operate correctly. Assume that the signals S and $T_0$ arrive (i.e. are stable) 1.5ns after the positive clock edge.

| **Gate/Attribute** | **Delay** |
|---|---|
| AND Gate Delay ($T_{and}$) | 3ns |
| OR Gate Delay ($T_{or}$) | 4ns |
| NOT Gate Delay (Tnot) | 1ns |
| Set-up Time ($T_s$) of D Flip-Flop | 2ns |
| Hold Time ($T_h$) of D Flip-Flop | 1ns |
| Clock-to-Q Time ($T_{cq}$) of D Flip-Flop | 2.5ns |

For signal $T_i$, (i not 0) in each clock period, must wait for the data to arrive at Q (clock to Q), propagate through the worst path in the logic ( AND + OR), and leave enough time to obey the setup time)

$T_{min} = T_{cq} + T_{and} + T_{or} + T_s = 2.5 + 3 + 4 + 2 = 11.5\text{ns}$

For the signals S and $T_0$, must propagate through NOT, AND, OR after the 1.5ns delay, and then meet the set-up time.

$T_{min} = \text{Stable} + T_{not} + T_{and} + T_{or} + T_s = 1.5 + 1 + 3 + 4 + 2 = 11.5\text{ns}$

The minimum clock period must be the greater of these two (which are equal), and so the minimum clock is 11.5ns.

- *6 marks - must show work. part marks if only include parts of the critical path.*