

1. The following Truth Table defines the function $H(A, B, C, D)$.

A	B	C	D	$H(A, B, C, D)$
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

- a. (7 points) Express $H(A, B, C, D)$ in sum-of-minterm form. That is, write it in the form of $H(A, B, C, D) = \sum m(\dots)$ where the appropriate minterm numbers are listed inside the parentheses.

$$H(A, B, C, D) = \sum m(1, 4, 5, 7, 9, 10, 15)$$

- b. (7 points) Draw the Karnaugh Map (K-Map) for $H(A, B, C, D)$.

AB \ CD	00	01	11	10
00		1		
01		1	1	1
11		1	1	1
10				1

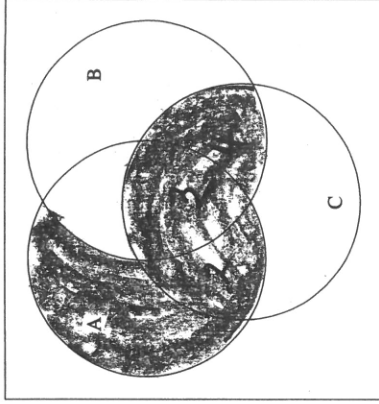
- c. (3 points) If the function $H(A, B, C, D)$ was synthesized/implemented in Canonical Sum-of-Products (CSOP) form, what would be the cost (as we have defined it in this class) of that synthesis/implementation?

$$\text{Cost}_{\text{CSOP}} = m(l+2) + 1 = 7(4+2) + 1 = 43$$

- d. (3 points) If the function $H(A, B, C, D)$ was implemented in Canonical Product-of-Sums (CPOS) form, what would be the cost (as we have defined it in this class) of that synthesis/implementation?

$$\text{Cost}_{\text{CPOS}} = M(l+2) + 1 = 9(4+2) + 1 = 55$$

2. (8 points) Consider the Venn Diagram shown below.



Write a Sum-of-Products (SOP) logic expression that this Venn Diagram represents. You must provide an explanation for how you arrived at your answer.

Regions 1 & 2 together is the intersection of $A \cap B : A \cdot B$

Regions 3 & 4 together " " " $B \cap C : B \cdot C$

The shaded area is the union of those two areas,

so the logic expression is $A \cdot B + B \cdot C$

Several other SOP answers possible.

3. Consider the following 3-input K-Map for function $J(A, B, C)$.

C \ AB	00 01 11 10			
	0	1	0	1
0	0	1	0	0
1	1	1	0	0

- a. (5 points) Find a minimum-cost Sum-of-Products (SoP) synthesis/implementation (just a logic expression) for $J(A, B, C)$ AND give the cost of that synthesis/implementation.

$$J = \bar{A} \cdot B + \bar{A} \cdot C$$

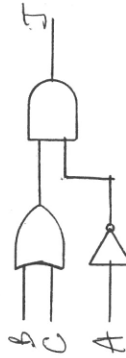
$$\text{Cost} = 2 \cdot 3 (2, 2\text{-input ANDs}) + 3 (2\text{-input OR}) = 9.$$

- b. (5 points) Find a minimum-cost Product-of-Sums (PoS) synthesis/implementation (just a logic expression) for $J(A, B, C)$ AND give the cost of that synthesis/implementation.

$$J = \bar{A} \cdot (B + C)$$

$$\text{Cost} = 3 (2\text{-input OR}) + 3 (2\text{-input AND}) = 6.$$

- c. (5 points) Using your logic template, draw the logic network/circuit for the Product-of-Sums (PoS) synthesis/implementation that you found in part (b), using only AND, OR, and NOT gates.

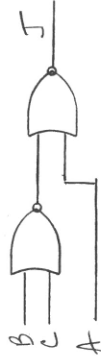
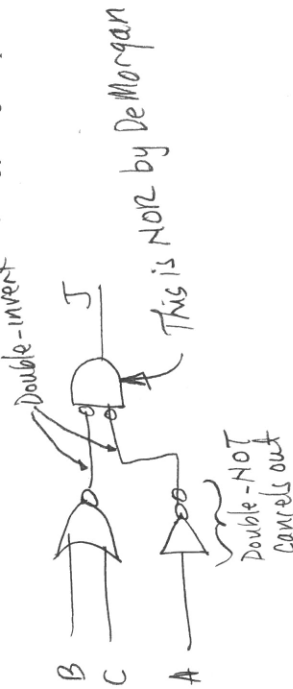


Prof. Petr

Copyright 2020 David W. Petr

Spring 2020

- d. (8 points) Using your logic template, draw the logic network/circuit for the Product-of-Sums (PoS) synthesis/implementation that you found in part (b), using only NOR gates. Be sure to simplify your network/circuit to use the fewest gates and show only gates that are actually needed to produce $J(A, B, C)$. You may do preliminary work freehand (without using your logic template), but your final logic network/circuit must be drawn neatly using your logic template.



- e. (5 points) If your final NOR-only logic network/circuit from the previous part was implemented with CMOS gates, how many total transistors would be required? You must give a brief explanation/calculation for your answer.

$$\text{NMOS NOR } 2 \text{ transistors} \Rightarrow 2 + 2 = 4 = T_{\text{NMOS}}$$

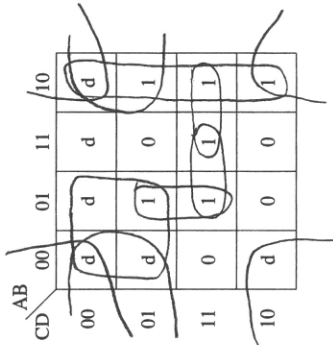
$$T_{\text{CMOS}} = 2 \cdot T_{\text{NMOS}} = 2 \cdot 4 = 8 \text{ transistors}$$

Prof. Petr

Copyright 2020 David W. Petr

Spring 2020

4. Consider the following Karnaugh Map (K-Map) for function $L(A, B, C, D)$, which we will be synthesizing/implementing with a Sum-of-Products (SoP) form. Each d in the K-Map indicates a *don't-care* output.



- a. (7 points) Identify all of the Prime Implicants (PIs) for $L(A, B, C, D)$, writing a logic expression for each one.

$$A\bar{B}, \bar{A}\bar{C}, \bar{B}\bar{C}, \bar{B}\bar{D}, \bar{A}\bar{B}, \bar{A}\bar{D}, B\bar{C}, A\bar{C}D$$

- b. (4 points) Identify all of the Essential Prime Implicants (EPIs) for $L(A, B, C, D)$, writing a logic expression for each one.

None is essential: each 1 included in at least 2 PIs.

- c. (9 points) Find a minimum-cost (as we have defined cost in this class) SoP synthesis/implementation for $L(A, B, C, D)$. Your answer should be a logic expression for L , not a logic network/circuit. You do not need to provide the cost of your solution.

- 1) Choose $A\bar{B} \rightarrow$ contributes 3 $L=1$ minterms to cover.
 2) Choose $\bar{A}\bar{C} \rightarrow$ 2-literal term that contributes 1 more.
 3) Choose $BCD \rightarrow$ contributes the final 2 $L=1$ minterms.

$$L = A\bar{B} + \bar{A}\bar{C} + BCD$$

Prof. Petr

Copyright 2020 David W. Petr

Spring 2020

5. (12 points) Prove the following Boolean equality in variables A , B , and C using Boolean algebraic manipulation. You may *not* use a Truth Table, K-map, or Venn diagram to prove the equality. You *must* justify each step of your manipulation with the Boolean Algebra property number or numbers used to obtain that step.

$$A \cdot \bar{B} \cdot \bar{C} + A \cdot B + A \cdot C + \bar{A} \cdot B \cdot C = A + B \cdot C$$

$$LHS: A \cdot \bar{B} \cdot \bar{C} + A \cdot B + A \cdot C + \bar{A} \cdot B \cdot C$$

$$12a: A \cdot (\bar{B} \cdot \bar{C} + B + C) + \bar{A} \cdot B \cdot C$$

$$15b, 11b: A \cdot (\overline{B+C} + (B+C)) + \bar{A} \cdot B \cdot C$$

$$8b: A \cdot (1) + \bar{A} \cdot B \cdot C$$

$$6a: A + \bar{A} \cdot B \cdot C$$

$$16a: A + BC \quad X = A \quad Y = BC$$

A RHS Done.

Other proofs possible.

Prof. Petr

Copyright 2020 David W. Petr

Spring 2020

6. You are to begin the design of a digital logic circuit with the following characteristics. The circuit has two binary data inputs A and B , plus as many binary control inputs as needed. The binary output G is determined by the control inputs as follows:

For the first combination of control input values: G should be A AND B . You are free to specify the specific combination of control input values (you will do this below).

For the second combination of control input values: G should be A OR B .

For the third combination of control input values: G should be A XOR B , where XOR stands for exclusive OR.

- a. (4 points) Determine the minimum number of control inputs that are required, then *define* each one (that is, give it a name). Finally, *assign* values to the control inputs for the first, second, and third combinations as indicated above.

1 binary control input can only provide

2 conditions/combinations, so need

2 control inputs. Call them C_1, C_0

1st combination: $C_1 = 0$; $C_0 = 0$; $G = A \cdot B$

2nd " ; $C_1 = 0$; $C_0 = 1$; $G = A + B$

3rd " ; $C_1 = 1$; $C_0 = 0$; $G = A \oplus B$

- b. (8 points) Draw a complete Truth Table for G , showing all possible combinations of all inputs. List the control inputs first in the top row of the Truth Table, followed by data inputs A and B , followed by the output G . If needed, use d for the output for input combinations where the output does not matter.

C_1	C_0	A	B	G
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
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	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Not Used.