

CSE140 Exercises before Midterm 1 Solution

(I) (Problem Formulation) Convert the following statement to Boolean equations. A car should slow down and stop one foot before an object. List needed sensors for inputs and output control signals. Express the outputs as Boolean equations of the inputs.

In order to stop one foot before an object, the sensors need to detect the object from faraway; meanwhile they should detect the current speed of the car and determine whether they need to send output control signal to slow down the car. Let A_k denotes the input boolean variable whether the sensor detects an object from k feet away; e.g. $A_{100} = 1$ means the sensor detects an object from 100 feet away. Let B_v denotes the input boolean variable whether the current car speed is over v mph; e.g. $B_{20} = 1$ means the current speed is over 20 mph. Assume we have n checkpoints with n variables A_{k_1} to A_{k_n} , and n variables B_{k_1} to B_{k_n} , the boolean function should be $\sum_{i=1}^n A_{k_i} * B_{k_i}$. That is, in each checkpoint, if the object is detected and the speed is over the preset limit, output control signal should be sent to slowdown the car.

Note: This is an open question. Thus any reasonable solution and explanation will be acceptable.

(II.1) (Laws and Theorems of Boolean Algebra) Prove using Boolean algebra that $ab + a'c + bc = ab + a'c$. Write the particular law you are using in each step.

$$\begin{aligned}
 & ab + a'c + bc \\
 = & ab + a'c + 1 * bc && \text{(Identity)} \\
 = & ab + a'c + (a + a') * bc && \text{(Complement)} \\
 = & ab + a'c + abc + a'bc && \text{(Distributive)} \\
 = & ab + abc + a'c + a'bc && \text{(Commutative)} \\
 = & (ab + abc) + (a'c + a'bc) && \text{(Associative)} \\
 = & ab * (1 + c) + a'c * (1 + b) && \text{(Distributive)} \\
 = & ab * 1 + a'c * 1 && \text{(Null Elements)} \\
 = & ab + a'c && \text{(Identity)}
 \end{aligned}$$

(II.2) (Laws and Theorems of Boolean Algebra) Prove using Boolean algebra that $(a + b)(a' + c)(b + c) = (a + b)(a' + c)$. Write the particular law you are using in each step.

$$\begin{aligned}
 & (a + b)(a' + c)(b + c) \\
 = & (a + b)(a' + c)(0 + b + c) && \text{(Identity)} \\
 = & (a + b)(a' + c)(aa' + b + c) && \text{(Complement)} \\
 = & (a + b)(a' + c)(a + b + c)(a' + b + c) && \text{(Distributive)} \\
 = & (a + b)(a + b + c)(a' + c)(a' + b + c) && \text{(Commutative)} \\
 = & ((a + b)(a + b + c))((a' + c)(a' + b + c)) && \text{(Associative)} \\
 = & (a + b)(1 + c)(a' + c)(1 + b) && \text{(Distributive)} \\
 = & (a + b) * 1 * (a' + c) * 1 && \text{(Null Elements)} \\
 = & (a + b)(a' + c) && \text{(Identity)}
 \end{aligned}$$

(III.1) (Karnaugh Map) Use Karnaugh map to simplify function $f(a, b, c, d) = \sum m(6, 7, 8, 9, 14) + \sum d(0, 10, 13)$. List **all possible** minimal two-level **sum of products** expressions. Show the switching functions.

Possible Solution 1:

Prime Implicants: $\sum m(6, 7), \sum m(6, 14), \sum m(8, 9)$

Essential Prime Implicant: $\sum m(6, 7)$

Function 1: $f(a, b, c, d) = a'bc + bcd' + ab'c'$

Possible Solution 2:

Prime Implicants: $\sum m(6, 7), \sum m(10, 14), \sum m(8, 9)$

Essential Prime Implicant: $\sum m(6, 7)$

Function 2: $f(a, b, c, d) = a'bc + acd' + ab'c'$

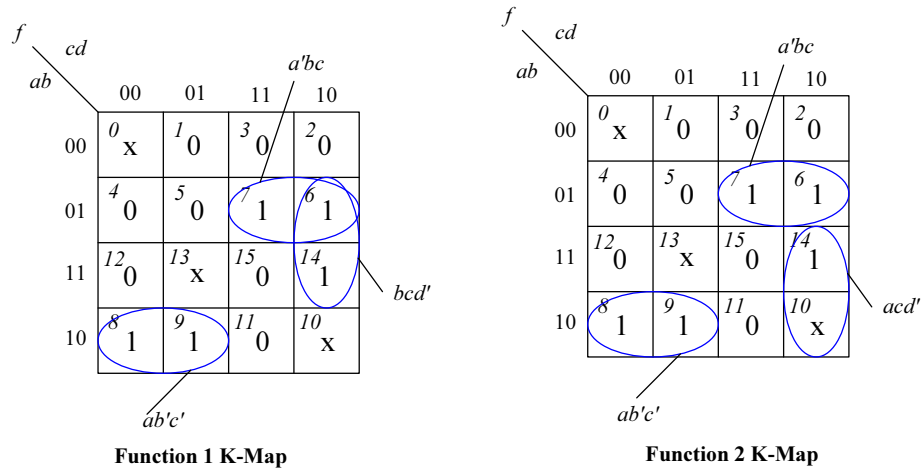


Figure 1: Problem III.1 Karnaugh Maps

(III.2) (Karnaugh Map) Use the same function in problem III, i.e. $f(a, b, c, d) = \sum m(6, 7, 8, 9, 14) + \sum d(0, 10, 13)$. List **all possible** minimal two-level **product of sums** expressions. Show the switching functions.

Possible Solution 1:

Prime Implicates: $\prod M(0, 1, 2, 3), \prod M(4, 5, 12, 13), \prod M(11, 15)$

Essential Prime Implicates: $\prod M(4, 5, 12, 13)$

Function 1: $f(a, b, c, d) = (a + b)(b' + c)(a' + c' + d')$

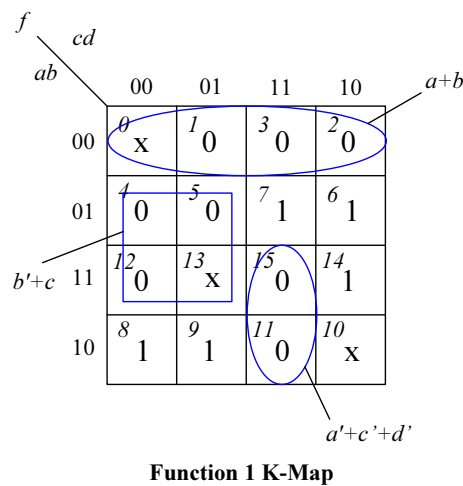


Figure 2: Problem III.2 Karnaugh Map

(IV) (Quine-McCluskey Method) Use the Quine-McCluskey method to find the minimum sum-of-products expression of function $f(a, b, c, d, e, f) = \sum m(1, 5, 7, 33) + \sum d(0, 34)$. Show the implication chart, and give the result in Boolean expressions.

id	a	b	c	d	e	f	F
0	0	0	0	0	0	0	-
1	0	0	0	0	0	1	1
5	0	0	0	1	0	1	1
33	1	0	0	0	0	1	1
34	1	0	0	0	1	0	-
7	0	0	0	1	1	1	1

id	a	b	c	d	e	f	F
(0,1)	0	0	0	0	0	-	1
(1,5)	0	0	0	-	0	1	1
(1,33)	-	0	0	0	0	1	1
(5,7)	0	0	0	1	-	1	1

Step 1: Find prime implicants: $\sum m(0,1), \sum m(1,5), \sum(1,33), \sum(5,7)$

Step 2: Find essential prime implicants: $\sum m(1,33), \sum m(5,7)$

Step 3: Select remaining prime implicants to cover all minterms: None

The result expression is

$$f(a,b,c,d,e,f) = \sum m(1,33) + \sum m(5,7) = b'c'd'e'f + a'b'c'df$$

(V) (Universal Set of Gates) Check if the set in the following list is universal and explain your decision. Assuming constants 0 and 1 are available as inputs.

i. {AND, NOT}

Yes. AND and NOT operations are directly available. OR operation can be obtained using De Morgan's law: $F = (x'y')' = x'' + y'' = x + y$.

ii. {NAND}

Yes. NOT operation can be implemented by connecting one input of the NAND gate to 1. AND operation is therefore obtained by the combination of NAND and NOT operations. OR operation is implemented in the same way as (i).

iii. {XOR, NOT}

No. AND cannot be implemented by XOR.

iv. $\{f(x,y)\}$, where $f(x,y) = x'y$

Yes. NOT is implemented as $f(x,1) = x' * 1 = x'$. AND operation can be implemented as $f(f(x,1),y) = f(x',y) = x''y = xy$. According to (i), OR can also be implemented.

v. $\{g(x,y,z)\}$, where $g(x,y,z) = (x+y)z'$

Yes. Because $g(x,0,z) = (x+0)z' = xz'$. Then it is the same function as the one in (iv).

vi. $\{f(x,y), g(x,y)\}$, where $f(x,y) = x'y + xy'$ and $g(x,y) = x'y'$

Yes. In fact g itself is universal. NOT is implemented as $g(x,0) = x' * 1 = x'$. OR is implemented as $g(g(x,y),0) = (x'y')' = x + y$. AND is implemented as $g(g(x,0),g(y,0)) = g(x',y') = x''y'' = xy$.