CS2100: Computer Organisation

Tutorial #6: Boolean Algebra, Logic Gates and Simplification

(Week 8: 4 – 8 October 2021)

Answers to Selected Questions

---

1. Name the essential theorems A, B, C and D used in the following derivation:

$F(j, k, m, p) = k' \cdot (j' \cdot p \cdot (j' + m'))' + (p + k' + j)'$

$= k' \cdot (j + p' + j \cdot m) + p' \cdot k \cdot j'$ … [A; involution]

$= j \cdot k' + k' \cdot p' + j \cdot k' \cdot m + p' \cdot k \cdot j'$ … [distributive; commutative]

$= j \cdot k' + p' \cdot (k' + k \cdot j') + j \cdot k' \cdot m$ … [associative; commutative; distributive]

$= j \cdot k' + k' \cdot p' + j' \cdot p' + j \cdot k' \cdot m$ … [B; distributive; commutative]

$= j \cdot k' + k' \cdot p' + j' \cdot p'$ … [associative; C]

$= j \cdot k' + j' \cdot p'$ … [D]

Note: In writing out terms, you should write the literals in the order of significance, especially in your final answer. For instance, for the above Boolean function $F(j, k, m, p)$, you should write the final answer as $j \cdot k' + j' \cdot p'$ and not $k' \cdot j + j' \cdot p'$ or $j \cdot k' + p' \cdot j'$.

Answers:

A = DeMorgan's Theorem

B = Absorption Theorem 2: $a + a' \cdot b = a + b$ ➔ $k' + k \cdot j' = k' + j'$

C = Absorption Theorem 1: $a + a \cdot b = a$ ➔ $j \cdot k' + j \cdot k' \cdot m = j \cdot k'$

D = Consensus Theorem: $a \cdot b + a' \cdot c + b \cdot c = a \cdot b + a' \cdot c$

➔ $j \cdot k' + k' \cdot p' + j' \cdot p' = j \cdot k' + j' \cdot p'$

2. Using Boolean algebra, simplify each of the following expressions into simplified **sum-of-products (SOP) expression**. Indicate the law/theorem used at every step.

   (a) $F(x,y,z) = (x + y \cdot z') \cdot (y' + y) + x' \cdot (y \cdot z' + y)$

   (b) $G(p,q,r,s) = \Pi M(5, 9, 13)$

   Tip: For (b), it is easier to start with the given expression and get done in 5 steps, rather than to expand it into sum-of-products/sum-of-minterms expression first.

Answers:

Note: There are more than one way of derivation.

(a) $(x + y \cdot z') \cdot \mathbf{(y' + y)} + x' \cdot (y \cdot z' + y)$

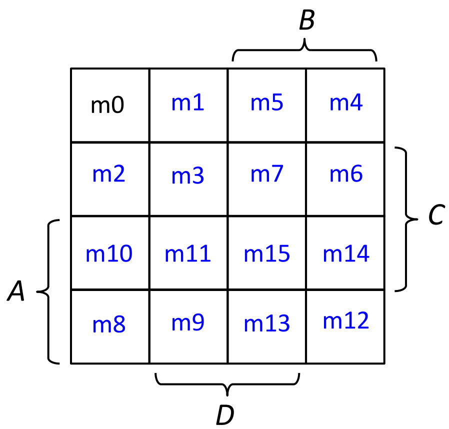
$= (x + y \cdot z') \cdot \mathbf{1} + x' \cdot (y \cdot z' + y)$	[complement]
$= (x + y \cdot z') + x' \cdot \mathbf{(y \cdot z' + y)}$	[identity]
$= x + \mathbf{y \cdot z'} + \mathbf{x' \cdot y}$	[absorption 1]
$= \mathbf{x} + \mathbf{x' \cdot y} + y \cdot z'$	[commutative]
$= x + \mathbf{y} + \mathbf{y \cdot z'}$	[absorption 2]
$= \mathbf{x + y}$	[absorption 1]

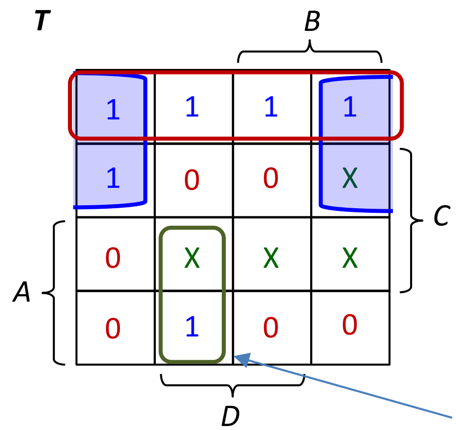
(b) $G(p,q,r,s) = \Pi M(5, 9, 13)$

$= \mathbf{(p + q' + r + s')} \cdot (p' + q + r + s') \cdot \mathbf{(p' + q' + r + s')}$	
$= ((\mathbf{p \cdot p'}) + (q' + r + s')) \cdot (p' + q + r + s')$	[distributive]
$= (\mathbf{0} + \mathbf{(q' + r + s')}) \cdot (p' + q + r + s')$	[complement]
$= (q' + \mathbf{r + s'}) \cdot (p' + q + \mathbf{r + s'})$	[identity]
$= (\mathbf{q' \cdot (p' + q)}) + (r + s')$	[distributive]
$= \mathbf{p' \cdot q' + r + s'}$	[absorption 2]

> Students: Remember to use $\cdot$ for AND, and not to leave it out.
> For example, for "*x* AND *y*", write $x \cdot y$ and not xy.

3. (a) The following K-map layout is used for a 4-variable Boolean function $T(A,B,C,D)$. Fill in the minterm positions m1 to m15 into the respective cells. m0 has been filled for you.





Alternative PI: $B'\cdot C'\cdot D$

(b) Given the following 4-variable Boolean function:

$$T(A,B,C,D) = \Pi M(3,7,8,10,12,13) \cdot X(6,11,14,15)$$

where X's are the don't-care conditions, write out the Σm notation for $T(A,B,C,D)$.

(c) Draw the K-map for T using the layout above.

(d) How many PIs (prime implicants) are there in the K-map? List out all the PIs.

(e) How many EPIs (essential prime implicants) are there? List out all the EPIs.

(f) What is the simplified SOP expression for T? List out all alternative solutions.

(g) What is the simplified POS expression for T? List out all alternative solutions.

(h) Implement the simplified SOP expression for T using a 2-level AND-OR circuit or a 2-level NAND only circuit, assuming that primed literals are not available.

Note: Always assume that prime literals are not available unless otherwise stated.

Answers:

(a) See above.

(b) $T(A,B,C,D) = \Sigma m(0,1,2,4,5,9) + X(6,11,14,15)$.

(c) See K-map above.

(d) 4 PIs: $A'\cdot D'$, $A'\cdot C'$, $A\cdot B'\cdot D$ and $B'\cdot C'\cdot D$.

(e) 2 EPIs: $A'\cdot D'$ and $A'\cdot C'$.

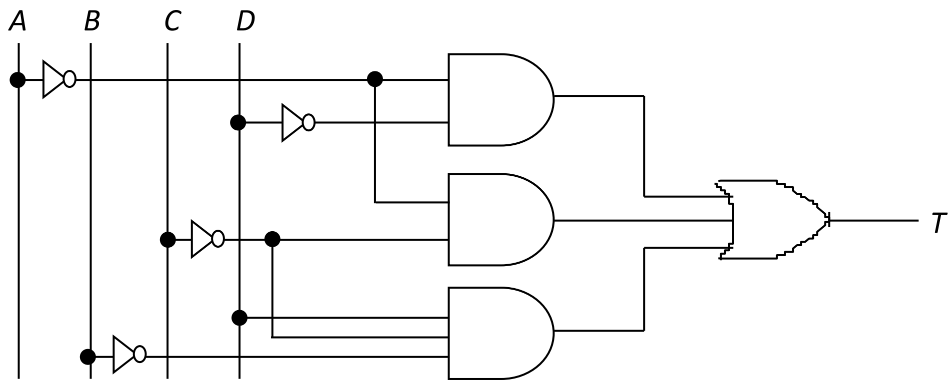
(f) SOP expression: $T(A,B,C,D) = A'\cdot D' + A'\cdot C' + B'\cdot C'\cdot D$ or $A'\cdot D' + A'\cdot C' + A\cdot B'\cdot D$.

(g) POS expression: $T(A,B,C,D) = (A' + D)\cdot(C' + D')\cdot(A' + B')$.

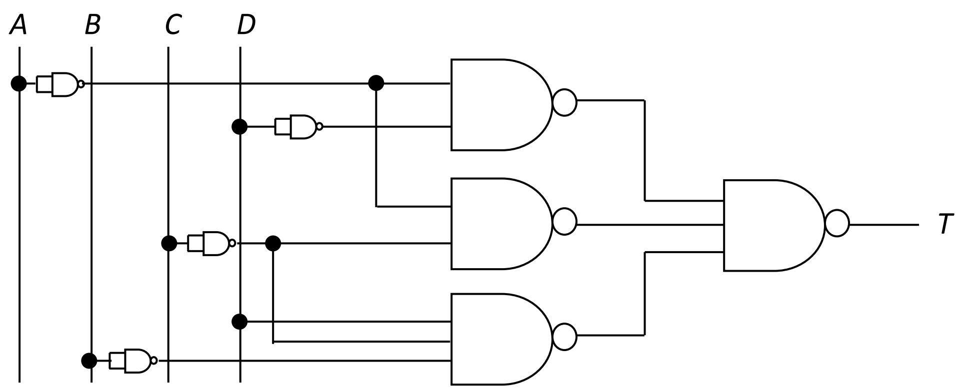
[Working: $T'(A,B,C,D) = A\cdot D' + C\cdot D + A\cdot B$.]

(h) Take $A' \cdot D' + A' \cdot C' + B' \cdot C' \cdot D$

2-level AND-OR circuit:



2-level NAND circuit:



Students: Draw logic diagrams neatly with straight lines.

Using Quine McCluskey to find the simplified SOP expression for T.

(Just for illustration. Quine McCluskey is not in the scope of CS2100, but knowing it will strengthen your understanding of K-map, and appreciate why K-map is faster and easier.)

$T(A,B,C,D) = \Sigma m(0,1,2,4,5,9) + X(6,11,14,15)$.

PI chart:

✓0: 0000

✓1: 0001
✓2: 0010
✓4: 0100

✓5: 0101
✓6: 0110
✓9: 1001

✓11: 1011
✓14: 1110

✓15: 1111

✓0,1: 000-
✓0,2: 00-0
✓0,4: 0-00

✓1,5: 0-01
1,9: -001
✓2,6: 0-10
✓4,5: 010-
✓4,6: 01-0

6,14: -110
9,11: 10-1

11,15: 1-11
14,15: 111-

0,1,4,5: 0-0-
0,2,4,6: 0--0

These are PIs.

These are not PIs because they contain solely don't-cares.

Reduced PI Chart:

Collecting the 4 PIs, we draw this reduced PI chart:

PI	Minterms						Don't-cares	
	0	1	2	4	5	9	6	11
0,1,4,5: 0 - 0 - ($A' \cdot C'$)	▪	▪		▪	▪			
0,2,4,6: 0 - - 0 ($A' \cdot D'$)	▪		▪	▪			▪	
1,9: -0 0 1 ($B' \cdot C' \cdot D$)		▪				▪		
9,11: 1 0 - 1 ($A \cdot B' \cdot D$)						▪		▪

Look under the minterms columns to find any column containing just one dot.
Since minterm m2 is covered only by $A' \cdot D'$, so $A' \cdot D'$ must be an EPI.
Likewise, minterm m5 is covered only by $A' \cdot C'$, so $A' \cdot C'$ must be an EPI.
Minterms m0, m1, m2, m4, m5 are covered by these 2 EPIs, leaving only minterm m9, which can be covered either by $B' \cdot C' \cdot D$ or $A \cdot B' \cdot D$.