CS2100: Computer Organisation

Tutorial #6: Boolean Algebra, Logic Gates and Simplification

(Week 8: 11 – 15 March 2024)

Answers

Discussion Questions:

D1. (a) One common mistake that students make is the following: $A \cdot B + A' \cdot B' = 1$ … (equation 1)

This seems to be erroneously "derived" from the following rule: $X + X' = 1$.
Explain why the rule is wrongly applied here.

(b) Is the following equation correct? Why? $A \cdot B + (A \cdot B)' = 1$ … (equation 2)

Answers:

(a) The rule is wrongly applied because if $X = A \cdot B$, then $X' = (A \cdot B)' = A'+B'$ and not $A' \cdot B'$.

(b) Equation 2 is correct since it is of the form $X + X' = 1$.

D2. Given the following two 3-variable Boolean functions:

$$F(A,B,C) = \Sigma\, m(0, 2, 4, 6, 7)$$
$$G(A,B,C) = \Sigma\, m(1, 2, 3, 6)$$

(a) Write the product-of-maxterms expressions in ΠM notation for F and G.

(b) If $X = F + G$, write the sum-of-minterms expressions in Σm notation for X.

(c) If $Y = F \cdot G$, write the sum-of-minterms expressions in Σm notation for Y.

(d) If $Z = F \oplus G$, write the sum-of-minterms expressions in Σm notation for Z.

Do you know how to generalise the above for any arbitrary Boolean functions F and G?

[If it is not convenient to type symbols like Σ and Π in the forums, you may use Sum-m to mean Σm and Prod-M to mean ΠM. Example: Sum-m(0, 2, 4, 6, 7), Prod-M(2, 3, 5).]

Answers:

(a) $F = \Pi\, M(1, 3, 5)$, $G = \Pi\, M(0, 4, 5, 7)$

(b) $X = \Sigma\, m(0, 1, 2, 3, 4, 6, 7)$ [$F+G$ contains the minterms in either F or G]

(c) $Y = \Sigma\, m(2, 6)$ [$F \cdot G$ contains the minterms common to both F and G]

(d) $Z = \Sigma\, m(0, 1, 3, 4, 7)$ [$F \oplus G$ contains the minterms in either F or G, but <u>not both</u>]

$F+G$ is the union of the minterms of F and G, while $F \cdot G$ is the intersection of minterms of F and G. These can be illustrated using truth tables.

D3. For each part below, how many prime implicants (PIs) and essential prime implicants (EPIs) are there in the K-map? What is/are the simplified **SOP expressions**? List out all alternative answers. [*d*(…) and *D*(…) denote don't-cares.]

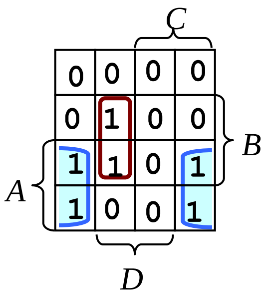
(a) $F1(A,B,C,D) = \Sigma\, m(5, 8, 10, 12, 13, 14)$

(b) $F2(W,X,Y,Z) = \Pi\, M(0, 1, 2, 8, 9, 10)$

(c) $F3(K,L,M,N) = \Sigma\, m(1, 7, 10, 13, 14) + d(0, 5, 8, 15)$

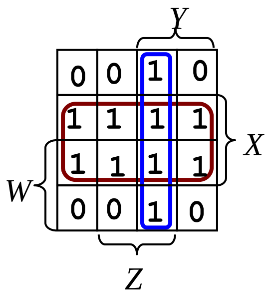
(d) $F4(A,B,C,D) = \Pi\, M(4, 8, 9, 11, 12) \cdot D(2, 3, 6, 7, 10, 14)$

(a) Answer: $F1 = \boldsymbol{A \cdot D' + B \cdot C' \cdot D}$



PIs: 3
EPIs: 2

(b) Answer: $F2 = \boldsymbol{X + Y \cdot Z}$

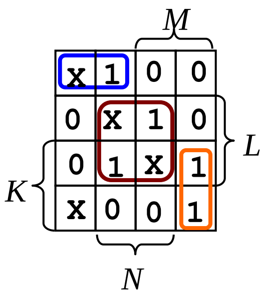


PIs: 2
EPIs: 2

(c) Answer:

$F3 = \boldsymbol{L \cdot N + K \cdot M \cdot N' + K' \cdot L' \cdot M'}$

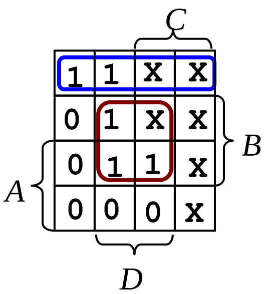
or $F3 = \boldsymbol{L \cdot N + K \cdot M \cdot N' + K' \cdot M' \cdot N}$



PIs: 6
EPIs: 1 ($L \cdot N$)

(d) Answer:

$F4 = \boldsymbol{A' \cdot B' + B \cdot D}$

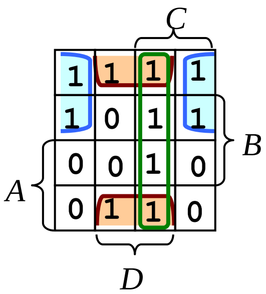


PIs: 4
EPIs: 2

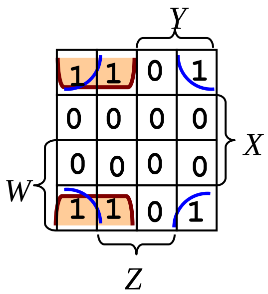
D4. For each of the functions in D3 above, find the simplified **POS expression**. List out all alternative answers, if any.

K-maps of complement functions are shown below.

(a) Answer: $F1' = A'\cdot D' + B'\cdot D + C\cdot D$
$F1 = \mathbf{(A+D)\cdot(B+D')\cdot(C'+D')}$



(b) Answer: $F2' = X'\cdot Y' + X'\cdot Z'$
$F2 = \mathbf{(X+Y)\cdot(X+Z)}$



(c) Answer:

$F3' = M'\cdot N' + K'\cdot N' + K'\cdot L'\cdot M + K\cdot L'\cdot N$

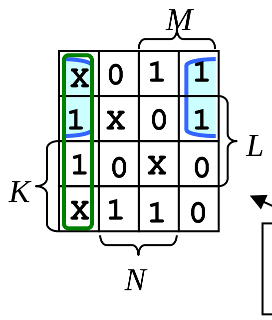
or $F3' = M'\cdot N' + K'\cdot N' + L'\cdot M\cdot N + K\cdot L'\cdot N$

or $F3' = M'\cdot N' + K'\cdot N' + L'\cdot M\cdot N + K\cdot L'\cdot M'$

$F3 = \mathbf{(M+N)\cdot(K+N)\cdot(K+L+M')\cdot(K'+L+N')}$

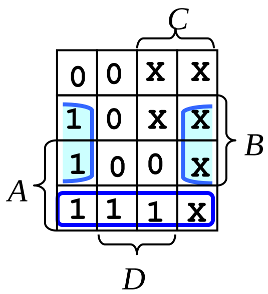
or $F3 = \mathbf{(M+N)\cdot(K+N)\cdot(L+M'+N')\cdot(K'+L+N')}$

or $F3 = \mathbf{(M+N)\cdot(K+N)\cdot(L+M'+N')\cdot(K'+L+M)}$



Only EPIs are shown here.

(d) Answer: $F4' = A\cdot B' + B\cdot D'$
$F4 = \mathbf{(A'+B)\cdot(B'+D)}$



You are encouraged to do the above discussion questions and discuss them on Canvas or QnA. These are fundamental concepts that you must know, before you attempt the tutorial questions below.

Reminder:
Do not omit the · operator when writing product terms such as $A\cdot B$. Omitting the · operator, like *AB*, means that *AB* is a 2-bit value. Marks will be deducted.

Tutorial Questions:

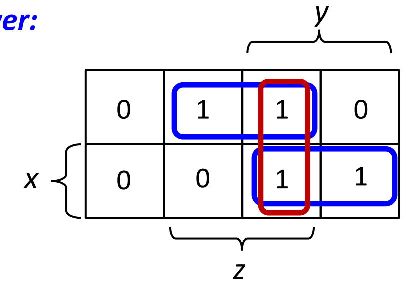
Note: By default, we assume that complemented literals are <u>NOT</u> available, unless otherwise stated.

1. The **consensus theorem** is given as

$$x{\cdot}y + x'{\cdot}z + y{\cdot}z = x{\cdot}y + x'{\cdot}z$$

Can you prove this using the laws and theorems of Boolean algebra given in class?
[Hint: Draw the K-map which will give you an idea which laws/theorems should be used.]

Answer:



You can see that the red implicant ($y{\cdot}z$) is redundant, so we can split it into $x'{\cdot}y{\cdot}z + x{\cdot}y{\cdot}z$, then let each product term be absorbed by $x'{\cdot}z$ and $x{\cdot}y$ respectively.

Hence, we have

$x{\cdot}y + x'{\cdot}z + y{\cdot}z$	$= x{\cdot}y + x'{\cdot}z + 1{\cdot}y{\cdot}z$	[identity law] ← ask students what law before showing it
	$= x{\cdot}y + x'{\cdot}z + (x+x'){\cdot}y{\cdot}z$	[complement law]
	$= x{\cdot}y + x'{\cdot}z + x{\cdot}y{\cdot}z + x'{\cdot}y{\cdot}z$	[distributive law]
	$= x{\cdot}y + x{\cdot}y{\cdot}z + x'{\cdot}z + x'{\cdot}y{\cdot}z$	[commutative law]
	$= (x{\cdot}y + x{\cdot}y{\cdot}z) + (x'{\cdot}z + x'{\cdot}y{\cdot}z)$	[associative law]
	$= x{\cdot}y + x'{\cdot}z$	[absorption theorem 1]

Tutors:
Don't spend too much time on this.
Unlike CS1231S, we are not that strict here that students must show the complete sequence of commutative and associative laws.

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 about 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]
	$= \mathbf{x + x'{\cdot}y} + y{\cdot}z'$	[commutative]
	$= x + \mathbf{y + y{\cdot}z'}$	[absorption 2]
	$= \mathbf{x + y}$	[absorption 1]

Tutors:
Spend 10 minutes on this.
This question may take the students a bit of time, especially if they made careless mistakes in their derivation.
To save time, you may get 2 students to write the answers on the board concurrently.

(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]

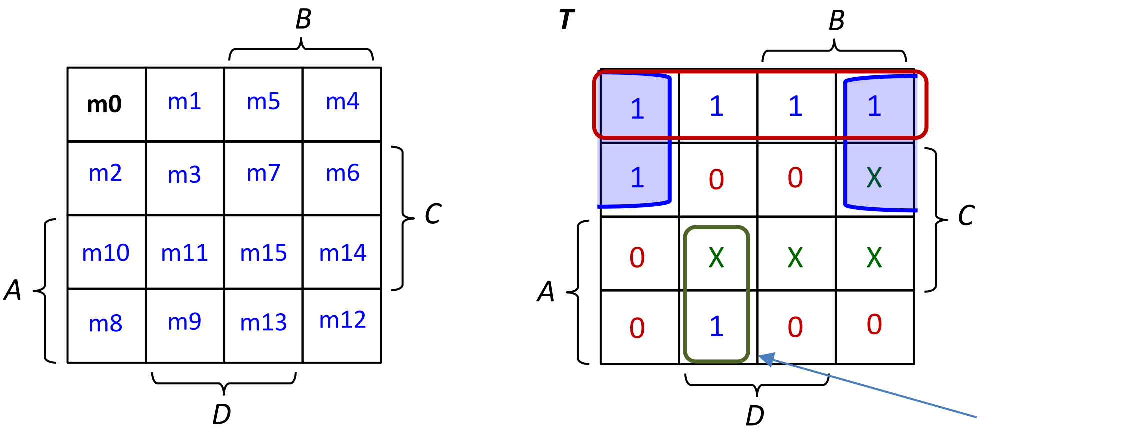
$= (\mathbf{q' \cdot (p' + q)}) + (r + s')$ [distributive]

$= \mathbf{p' \cdot q' + r + s'}$ [absorption 2]

Check: Do students remember the definition of **maxterm**?

Tutors:
Remind student to use · for AND, and not to leave it out. For example, $x \cdot y$ and not xy.
Marks will be deducted if this is not followed.

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-cares, 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 and a 2-level NAND only circuit.

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 $B'{\cdot}C'{\cdot}D$.

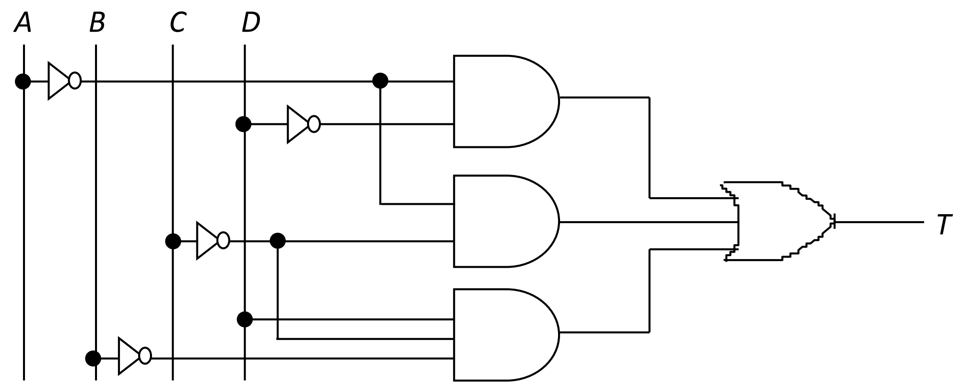
(e) 2 EPIs: $A'{\cdot}D'$ and $A'{\cdot}C'$.

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

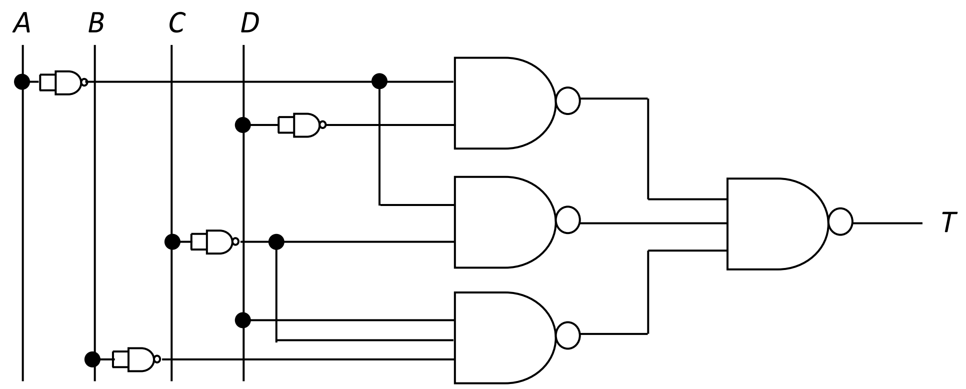
(g) POS expression: $T(A,B,C,D) = (A' + D){\cdot}(C' + D'){\cdot}(A' + B')$.
[Working: $T'(A,B,C,D) = A{\cdot}D' + C{\cdot}D + A{\cdot}B$.]

(h) Take the expression $A'{\cdot}D' + A'{\cdot}C' + B'{\cdot}C'{\cdot}D$

2-level AND-OR circuit:



2-level NAND circuit:



Students: Draw logic diagrams neatly with <u>straight lines</u>. Draw thick dots to represent forks.

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	✓0,1: 000-	0,1,4,5: 0-0-
	✓0,2: 00-0	0,2,4,6: 0--0
	✓0,4: 0-00	
✓1: 0001	✓1,5: 0-01	
✓2: 0010	1,9: -001	
✓4: 0100	✓2,6: 0-10	
	✓4,5: 010-	
	✓4,6: 01-0	
✓5: 0101	6,14: -110	
✓6: 0110	9,11: 10-1	
✓9: 1001		
✓11: 1011	11,15: 1-11	
✓14: 1110	14,15: 111-	
✓15: 1111		

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 $A \cdot B' \cdot D$.

4. A circuit takes in four inputs *K,L,M,N* and generates 3 outputs *X,Y,Z* as follow:

$X(K,L,M,N) = 1$ if $KL = MN$, or 0 otherwise,
where *KL* and *MN* are 2-bit unsigned integers.

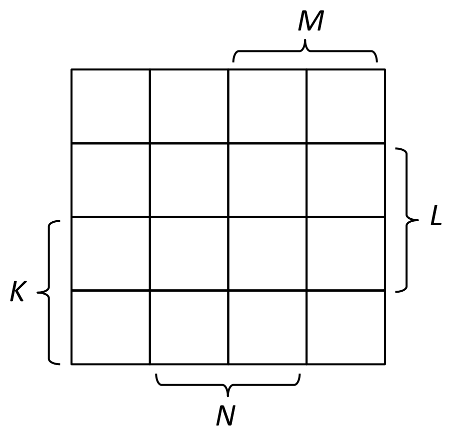
$Y(K,L,M,N) = 1$ if $KL \leq MN$, or 0 otherwise,
where *KL* and *MN* are 2-bit unsigned integers.

$Z(K,L,M,N) = 1$ if $KLM < LMN$, or 0 otherwise,
where *KLM* and *LMN* are 3-bit unsigned integers.

For parts (a) – (c) below, you may assume that the input 0000 will not occur.

(a) Fill in the truth table for the circuit. Write 'd' for don't cares.

(b) Fill in the K-maps of *X*, *Y* and *Z* using the layout given below.



(c) Write out the simplified SOP expressions of *X*, *Y* and Z.

(d) After designing the circuit according to the simplified SOP expressions in (c), if you feed the input 0000 into it, what will be the outputs?

Answers:

(a)

K	*L*	*M*	*N*	*X*	*Y*	*Z*
0	0	0	0	d	d	d
0	0	0	1	0	1	1
0	0	1	0	0	1	1
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	1	1	1
0	1	1	0	0	1	1
0	1	1	1	0	1	1

	K	*L*	*M*	*N*	*X*	*Y*	*Z*
	1	0	0	0	0	0	0
	1	0	0	1	0	0	0
	1	0	1	0	1	1	0
	1	0	1	1	0	1	0
	1	1	0	0	0	0	0
	1	1	0	1	0	0	0
	1	1	1	0	0	0	0
	1	1	1	1	1	1	0

(b)

X

d	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Y

d	1	1	1
0	1	1	1
0	0	1	0
0	0	1	1

Z

d	1	1	1
1	1	1	1
0	0	0	0
0	0	0	0

(c) ***X = K'·L·M'·N + K·L'·M·N' + K·L·M·N***

Y = M·N + K'·N + K'·M + L'·M

Z = K'

(d) Input *KLMN* = 0000; output *XYZ* = 001.