CS2100: Computer Organisation

Tutorial #6: Boolean Algebra, Logic Gates and Simplification

(Week 8: 8 – 12 March 2021)

Answers to Selected Questsions

Tutorial Questions:

2. Using Boolean algebra, simplify each of the following expressions into simplified **sum-of-products (SOP) expressions**. Indicate the law/theorem used for each 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 simpler 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 law]
$= (x + y \cdot z') + x' \cdot (\mathbf{y \cdot z' + y})$	[identity law]
$= x + \mathbf{y \cdot z'} + \mathbf{x' \cdot y}$	[absorption theorem 1]
$= \mathbf{x} + \mathbf{x' \cdot y} + y \cdot z'$	[commutative law]
$= x + \mathbf{y} + \mathbf{y \cdot z'}$	[absorption theorem 2]
$= \mathbf{x + y}$	[absorption theorem 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 law]
$= (\mathbf{0} + (\mathbf{q' + r + s'})) \cdot (p' + q + r + s')$	[complement law]
$= (q' + \mathbf{r + s'}) \cdot (p' + q + \mathbf{r + s'})$	[identity law]
$= (\mathbf{q' \cdot (p' + q)}) + (r + s')$	[distributive law]
$= \mathbf{p' \cdot q' + r + s'}$	[absorption theorem 2]

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

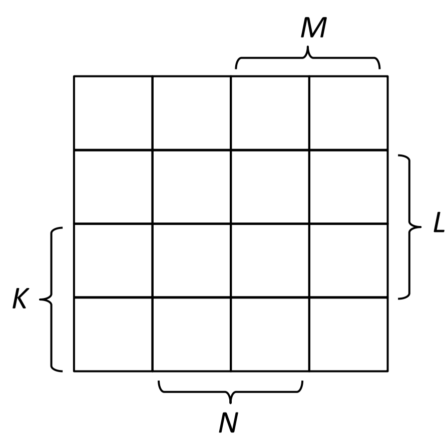
$X(K,L,M,N) = 1$ if $KL = MN$, or 0 otherwise,
where *KL* and *MN* are 2-bit unsigned integers.

$Y(K,L,M,N) = 1$ 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.

(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, with the assumption that the input 0000 will not occur.

(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

		N	*M*, *N*	*M*
	d	0	0	0
L	0	1	0	0
K, *L*	0	0	1	0
K	0	0	0	1

Y

		N	*M*, *N*	*M*
	d	1	1	1
L	0	1	1	1
K, *L*	0	0	1	0
K	0	0	1	1

Z

		N	*M*, *N*	*M*
	d	1	1	1
L	1	1	1	1
K, *L*	0	0	0	0
K	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 *X* = 0; *Y* = 0; *Z* = 1.