ENGR 101
Engineering Technology

Dr. Kerese Manueli

School of Engineering and Computer Science
Victoria University of Wellington

Victoria
UNIVERSITY OF WELLINGTON
Te Whare Wānanga
o te Ūpoko o te Ika a Māui

CAPITAL CITY UNIVERSITY

Week 7 Lecture 11a

- Combinational circuit

- Assignment 2 – submit before midnight Monday
- Test 1 – Thursday 22 April (10:15 – 11:55am)

- Course web page:

  https://ecs.wgtn.ac.nz/Courses/XMUT101_2021T1/

- kerese@ecs.vuw.ac.nz

Combinational Logic

- Basic logic gate functions will be combined in *combinational* logic circuits.

Combinational Logic

- Basic logic gate functions will be combined in *combinatorial (also called combinational)* logic circuits.

- At any time the output depends only on the combination of logic levels at the input to the circuit.

Combinational Logic

- Basic logic gate functions will be combined in *combinatorial (also called combinational)* logic circuits.

- At any time the output depends only on the combination of logic levels at the input to the circuit.

- Simplification of logic circuits will be done using Boolean algebra and K-Map techniques.

Simplifying logic circuits

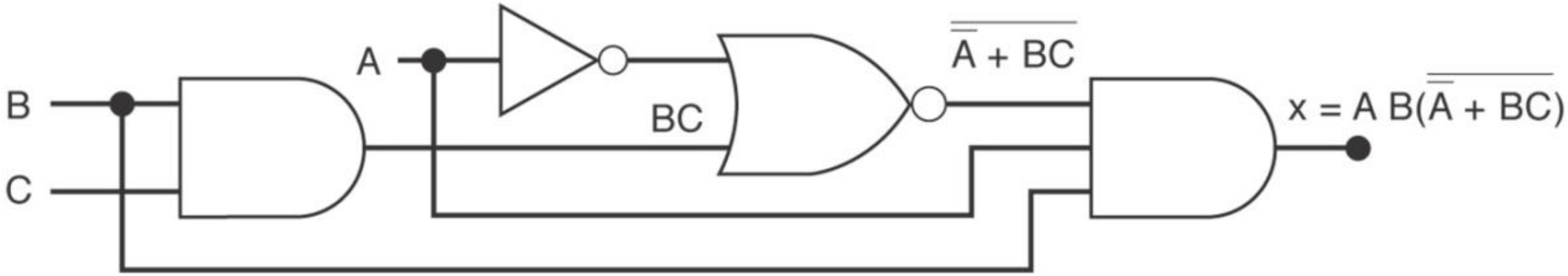
- Logic circuit simplification and design requires the logic expression to be in a sum-of-products (SOP) form.

Simplifying logic circuits

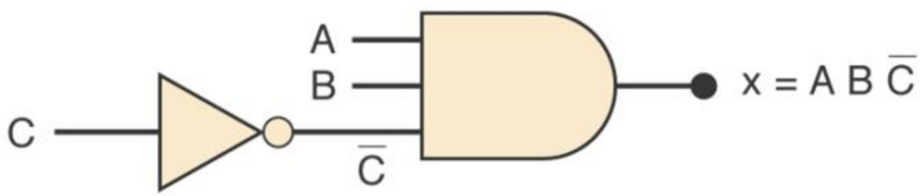
- Logic circuit simplification and design requires the logic expression to be in a sum-of-products (SOP) form.
- This expression will appear as two or more AND terms ORed together.
- Example 1: ABC + A'BC' → $ABC + \overline{A}B\overline{C}$
- Example 2: AB+A'BC'+C'D'+D → $AB + \overline{A}B\overline{C} + \overline{C}\,\overline{D} + D$

Simplifying Logic Circuits

The circuits below both provide the same output, but the lower one is clearly less complex.



(a)



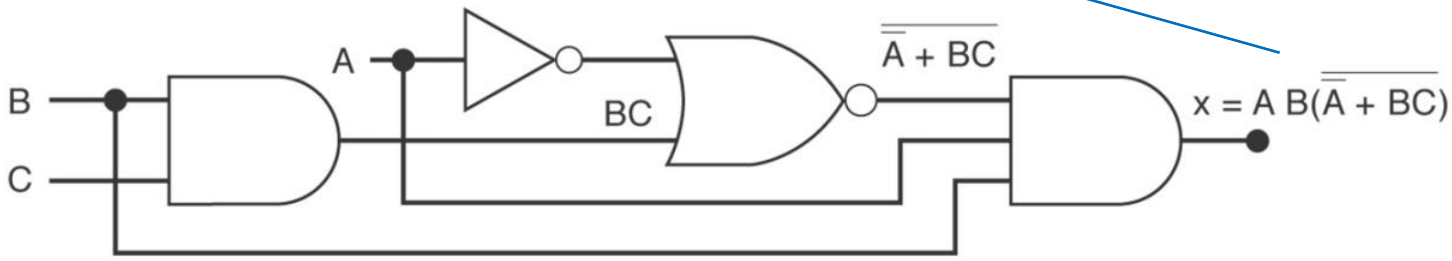
(b)

Boolean Algebra Rules

	BOOLEAN LAW or RULE	
1.	Identity Law	A+0=A; A+1=1; A.0=0; A.1=A
2.	Commutative Law	A.B = B.A; A+B = B+A
3.	Associative Law	A.(B.C) = A.B.C; A+(B+C) = A+B+C
4.	Idempotent Law	A.A = A; A+A = A
5.	Double Negative Law	A'' = A
6.	Complement Law	A.A' = 0; A+A' = 1
7.	Law of Intersection	A.1 = A; A.0 =0
8.	Law of Union	A+1 = A; A+0 = 0
9.	DeMorgan's Theorem	(AB)' = A'+B'; (A+B)' = A'.B'
10.	Distributive Law	A.(B+C) = (A.B) + (A.C); A+(BC) = (A+B).(A+C)
11.	Absorption Law	A.(A+B) = A; A+(A.B) = A
12.	Common Identities Law	A.(A'+B) = AB; A+(A'B) = A+B

Example 1

Simplify the expression: $x = AB(A' + BC)'$



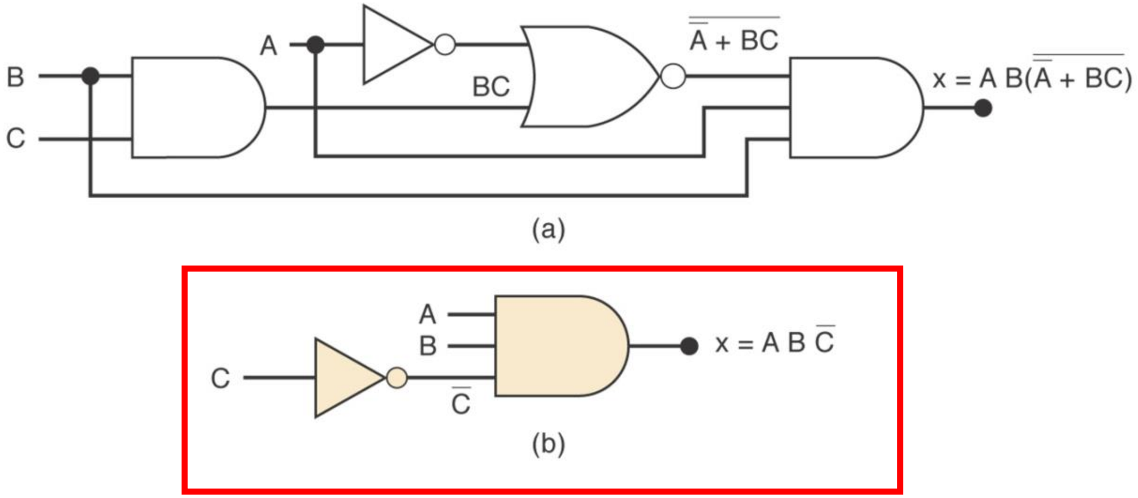
(a)

Simplify the expression: x = AB(A' + BC)'

Expression	Boolean Law used
x = A B(A' + BC)'	Step 1 - DeMorgan's Law (A+B)' = A'B'
= A B (A''.B'C')	Step 2 - Double Negative Law A'' = A
= A B(AB'C')	Step 3 - Associative Law A(B.C) = A.B.C
= A (B(B'C'))	Step 4 - DeMorgan's Law (AB)' = A' + B'
= A B(B' + C')	Step 5 - Distributive Law A(B+C) = AB + AC
= ABB' + ABC'	Step 6 - Complement Law A.A' = 0
= A(0) + ABC'	
= ABC'	Simplified expression

Simplifying Logic Circuits

The circuits below both provide the same output, but the lower one is clearly less complex.

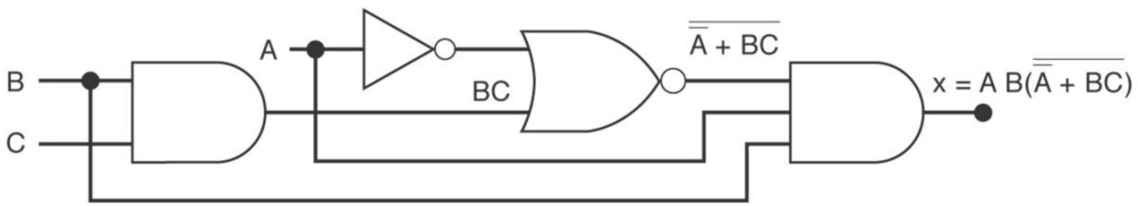


(a)

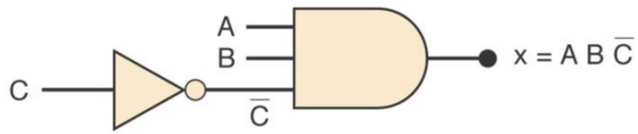
(b)

Simplifying Logic Circuits

The circuits below both provide the same output, but the lower one is clearly less complex.

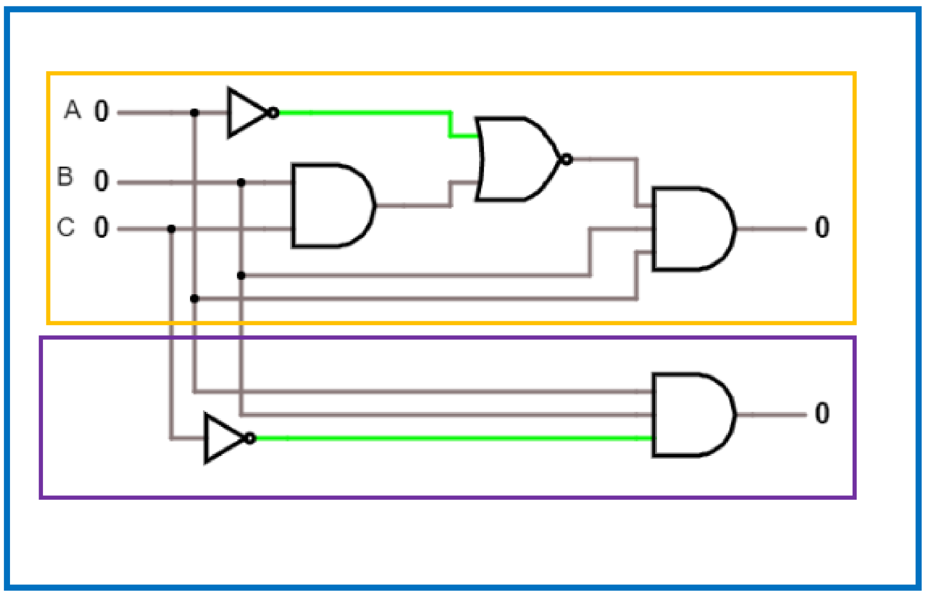


(a)



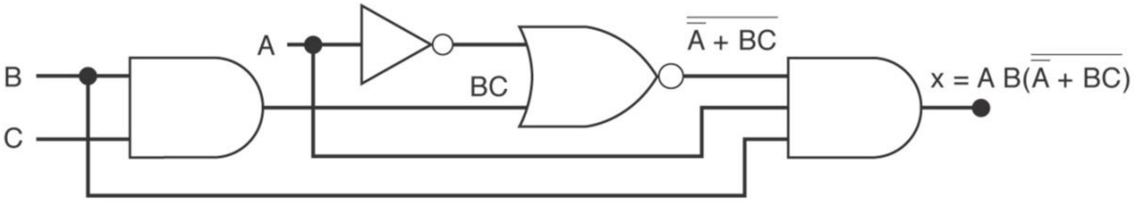
(b)

Using the online logic simulator: https://www.falstad.com/circuit/

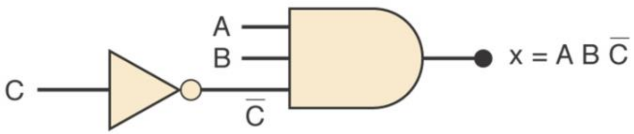


Simplifying Logic Circuits

The circuits below both provide the same output, but the lower one is clearly less complex.

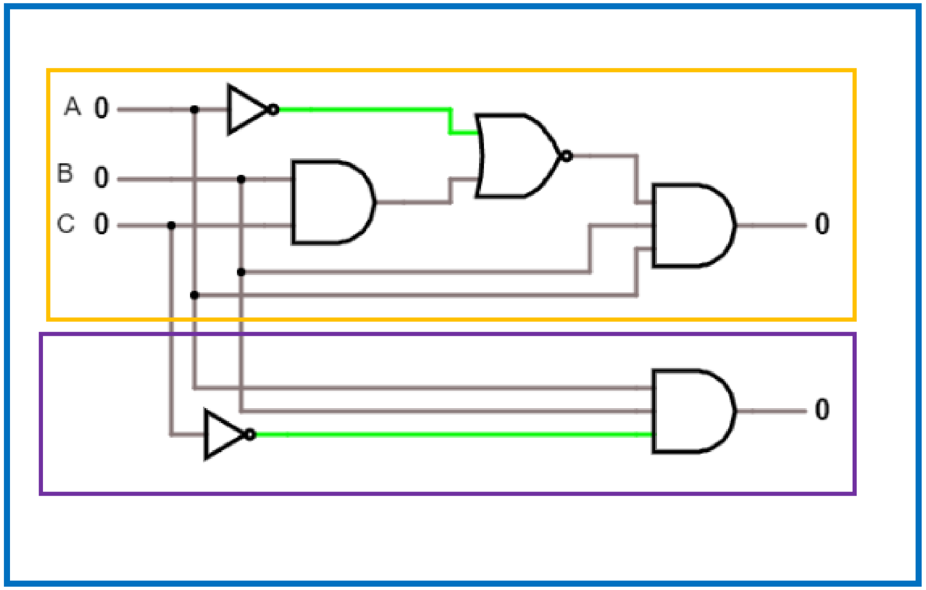


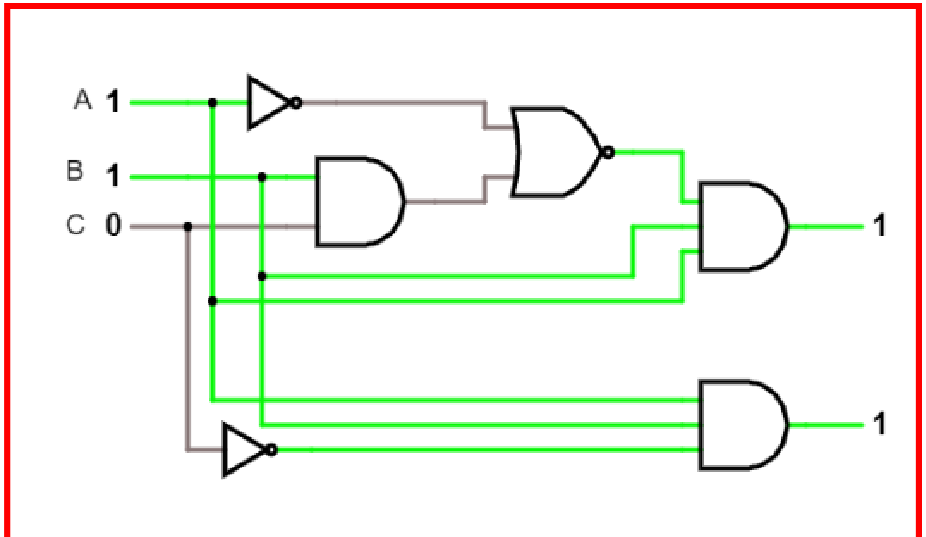
(a)



(b)

Using the online logic simulator:
https://www.falstad.com/circuit/





Week 7 Lecture 11a

- Combinational circuit

- Assignment 2 – submit before midnight Monday

- Course web page:

  https://ecs.wgtn.ac.nz/Courses/XMUT101_2021T1/

- kerese@ecs.vuw.ac.nz