04 – SIMPLIFYING CIRCUITS

COMP256 – COMPUTING ABSTRACTIONS

DICKINSON COLLEGE

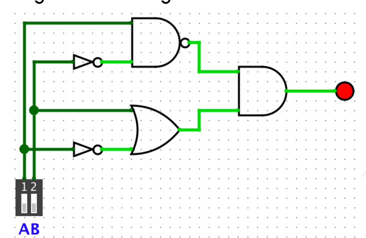
RECALL: GIVEN A LOGIC EXPRESSION …

$$Z = \overline{A \cdot \overline{B}} \cdot (\overline{A} + B)$$

- … generate a truth table.

A	B	$\overline{A}$	$\overline{B}$	$A \cdot \overline{B}$	$\overline{A \cdot \overline{B}}$	$\overline{A} + B$	Z
0	0	1	1	0	1	1	1
0	1	1	0	0	1	1	1
1	0	0	1	1	0	0	0
1	1	0	0	0	1	1	1

- … generate a logic circuit.



RECALL: GIVEN A LOGIC EXPRESSION …

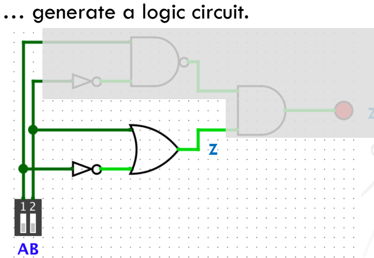
$$Z = \overline{A \cdot \overline{B}} \cdot (\overline{A} + B)$$

These gates
can be omitted a
We get the same

- … generate a truth table.

A	B	$\overline{A}$	$\overline{B}$	$A \cdot \overline{B}$	$\overline{A \cdot \overline{B}}$	$\overline{A} + B$	Z
0	0	1	1	0	1	1	1
0	1	1	0	0	1	1	1
1	0	0	1	1	0	0	0
1	1	0	0	0	1	1	1

- … generate a logic circuit.



LOGIC SIMPLIFICATION

- Boolean Algebra:
  - The Boolean Algebra consists of a collection of *identities* that can be used to rewrite logical expressions into *equivalent* logical expressions that may yield *better* circuits:
    - Fewer gates or inputs (cheaper)
    - Smaller *propagation delays* (faster)
    - Require less space on a chip (denser)
  - *NOTE:* Two logic expressions, Z_1 and Z_2, are *equivalent* if and only if all inputs that make Z_1 true also make Z_2 true and vice versa.

BOOLEAN IDENTITIES

Name	*And Form*	*Or Form*
Identity	$1 \cdot A = A$	$0 + A = A$
Null	$0 \cdot A = 0$	$1 + A = 1$
Idempotent	$A \cdot A = A$	$A + A = A$
Inverse	$A \cdot \overline{A} = 0$	$A + \overline{A} = 1$
Commutative	$A \cdot B = B \cdot A$	$A + B = B + A$
Associative	$(A \cdot B) \cdot C = A \cdot (B \cdot C)$	$(A + B) + C = A + (B + C)$
Distributive	$A + (B \cdot C) = (A + B) \cdot (A + C)$	$A \cdot (B + C) = A \cdot B + A \cdot C$
Absorption	$A \cdot (A + B) = A$	$A + A \cdot B = A$
DeMorgan's	$\overline{A \cdot B} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\,\overline{B}$

Double Negation $\overline{\overline{A}} = A$

ACTIVITY

- Use truth tables to show that the Absorption Identities are true.

SIMPLIFYING AN EXPRESSION

- Use the Boolean Identities to simplify the following expression:

$$Z = \overline{A \cdot \overline{B}} \cdot (\overline{A} + B)$$

SIMPLIFYING AN EXPRESSION

- Use the Boolean Identities to simplify the following expression:

$$Z = \overline{A \cdot \overline{B}} \cdot (\overline{A} + B)$$

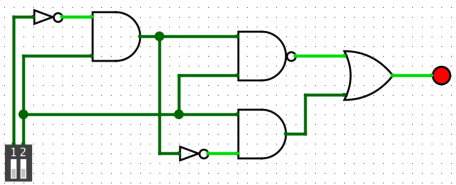
$Z = \overline{A \cdot \overline{B}} \cdot (\overline{A} + B)$	
$\overline{A \cdot \overline{B}}\,\overline{A} + \overline{A \cdot \overline{B}}\,B$	Distributive Or
$(\overline{A} + \overline{\overline{B}}) \cdot \overline{A} + (\overline{A} + \overline{\overline{B}}) \cdot B$	DeMorgan's And
$\overline{A} \cdot (\overline{A} + \overline{\overline{B}}) + B \cdot (\overline{A} + \overline{\overline{B}})$	Commutative And
$\overline{A}\,\overline{A} + \overline{A}\,\overline{\overline{B}} + B\overline{A} + B\overline{\overline{B}}$	Distributive Or
$\overline{A} + \overline{A}\,\overline{\overline{B}} + B\overline{A} + B\overline{\overline{B}}$	Idempotent And
$\overline{A} + \overline{A}B + B\overline{A} + BB$	Double negation
$\overline{A} + \overline{A}B + B\overline{A} + B$	Idempotent And
$\overline{A} + B\overline{A} + B$	Absorbtion Or
$\overline{A} + B + B\overline{A}$	Commutative Or
$Z = \overline{A} + B$	Absorbtion or

Truth table for a double check.

A	B	$\overline{A}$	Z $\overline{A} + B$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1

ACTIVITY

- Use Boolean Identities to reduce the number of gates in the following circuit:



$$Z = \overline{\bar{A} \cdot B \cdot B} + B \cdot \overline{\bar{A} B}$$

A	B	Z
0	0	1
0	1	0
1	0	1
1	1	1

ACTIVITY

$$Z = \overline{\bar{A} \cdot B \cdot B} + B \cdot \overline{\bar{A} B}$$

Truth table for a double check.

A	B	$\bar{B}$	Z $A + \bar{B}$
0	0	1	1
0	1	0	0
1	0	1	1
1	1	0	1

$Z = \overline{\bar{A} \cdot B \cdot B} + B \cdot \overline{\bar{A} B}$	
$= \overline{\bar{A} \cdot B} + B \cdot \overline{\bar{A} B}$	Idempotent And
$= \overline{\bar{A} \cdot B} + B \cdot (\bar{\bar{A}} + \bar{B})$	DeMorgan And
$= \overline{\bar{A} \cdot B} + B \cdot (A + \bar{B})$	Double Negation
$= \overline{\bar{A} \cdot B} + BA + B\bar{B}$	Distributive Or
$= \overline{\bar{A} \cdot B} + BA + 0$	Inverse And
$= \bar{\bar{A}} + \bar{B} + BA$	Demorgan And
$= A + \bar{B} + BA$	Double Negation
$= A + \bar{B} + AB$	Commutative And
$= A + AB + \bar{B}$	Commutative Or
$= A \cdot 1 + AB + \bar{B}$	Identity And, Commutative And
$= A(1 + B) + \bar{B}$	Distributive Or
$= A \cdot 1 + \bar{B}$	Null Or
$Z = A + \bar{B}$	Commutative And, Identity And

INTEGRATION SCALE

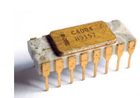
- Integrated circuits can be classified by the scale (magnitude) of Integration:
  - Small Scale Integration (1-10 transistors)
  - Medium Scale Integration (10-100 transistors)
  - Large Scale Integration (100-1000 transistors)
  - Very Large Scale Integration (*VLSI*) (1000+)
- "AMD 2nd Gen EPYC Rome Processors Feature **A Gargantuan 39.54 Billion Transistors**"
  - Oct 22, 2019
  - https://wccftech.com/amd-2nd-gen-epyc-rome-iod-ccd-chipshots-39-billion-transistors/

Image from: https://www.nteinc.com/integrated_circuits.php

MOORE'S LAW

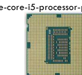
- Gordon Moore, co-founder of Intel, observed in 1965 that the number of transistors on a silicon chip had approximately doubled every year.
  - Later revised to every 18 to 24 months.

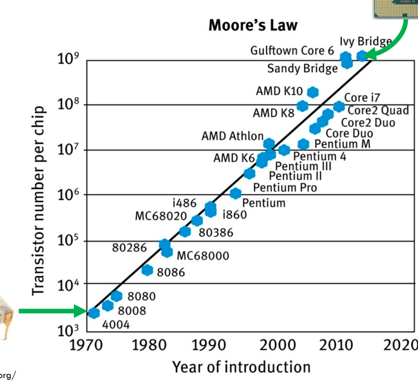
Image from: https://www.sfgate.com/business/article/Co-founder-of-Intel-honored-by-president-2798595.php



https://commons.wikimedia.org/wiki/File:Intel_C4004.jpg

https://wccftech.com/intels-3rd-generation-ivy-bridge-core-i5-processor-pitted/





Graph from: https://www.ncbi.nlm.nih.gov/books/NBK321721/figure/oin_tutorial.F3/