Logical expressions

simplify a logic circuit/expression using Boolean algebra

Review

Last lesson we learned logical operators. Name which of them do you remember?

What is ALU?

Why do we need to know logical expressions?

Main logical operators and their truth tables

And

A	B	Ans
0	0	0
1	0	0
0	1	0
1	1	1

Or

A	B	Ans
0	0	0
1	0	1
0	1	1
1	1	1

Nand

A	B	Ans
0	0	1
1	0	1
0	1	1
1	1	0

Xor

A	B	Ans
0	0	0
1	0	1
0	1	1
1	1	0

Nor

A	B	Ans
0	0	1
1	0	0
0	1	0
1	1	0

Notice!!

When we write equations

+ is And

* is Or

- is Not

$\overline{A+B}$ is Nand

$\overline{A*B}$ is Nor

Simplifying boolean equations with Truth Tables

A common question is to give you a complex boolean equation, which you will then have to work out a simpler exact equivalent. This is useful when you are designing circuits and want to minimise the number of gates you are using or make circuits that only use particular types of gates. To simplify boolean equations you must be familiar with two methods. You can normally use either, but try to master both:

- Truth tables
- Boolean algebra - identities and De Morgan’s Law

Simplifying boolean equations with Truth Tables

____

A + notB

1. First of all we need to draw a truth table for **A and B**
2. Then we need to add there **notB**
3. After that we calculate **A + notB**
4. And finally we apply **NOT** for our equation **A + notB**

And we get this

A	B	notB	A+notB	$\overline{A+(notB)}$
0	0	1	0	1
0	1	0	0	1
1	0	1	1	0
1	1	0	0	1

Exercise 1

Simplify the following equation yourself using Truth table:

____

A * notB

Answer 1

A	B	notB	A*notB	$\overline{A*(notB)}$
0	0	1	1	0
0	1	0	0	1
1	0	1	1	0
1	1	0	1	0

De Morgan’s law

Not (**A** *and* **B**) is the same as *Not* **A** *or Not* **B**.

$$\overline{A + B} = \overline{A} * \overline{B}$$

Not (**A** *or* **B**) is the same as *Not* **A** *and Not* **B**.

$$\overline{A * B} = \overline{A} + \overline{B}$$

Prove

Let's prove that I'm not lying to you by creating a truth table to prove that

$$\overline{A * B} = \overline{A} + \overline{B}$$

A	B	A * B	$\overline{A * B}$	notA	notB	$\overline{A} + \overline{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Exercise 2

Prove the second De Morgan's Law with Truth table

$$\overline{A + B} = \overline{A} * \overline{B}$$

Answer 2

A	B	A + B	$\overline{A + B}$	notA	notB	$\overline{A} * \overline{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Now do some exercises on the paper sheets

Reflection

- Why do we need simplification?
- How can it help circuits?
- What way of simplification you like better Truth tables or De Morgan’s rule?

https://en.wikibooks.org/wiki/A-level_Computing/AQA/Computer_Components,_The_Stored_Program_Concept_and_the_Internet/Fundamental_Hardware_Elements_of_Computers/De_Morgan%27s_Laws