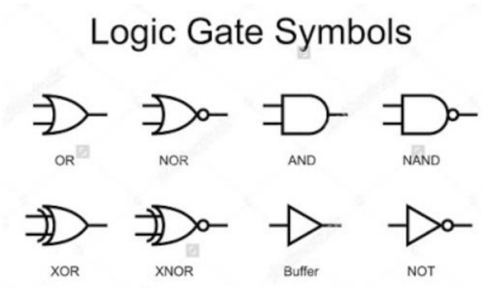
November 3, 2024 Math 8

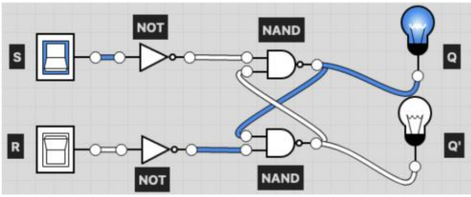
<u>Handout 8: More logic gates and circuits. Implications.</u>



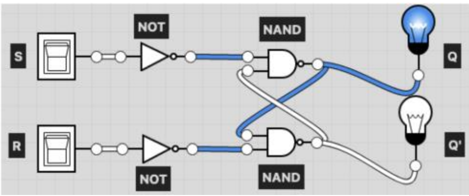
Combining such simple gates, one can create more complicated ones — and use that to create circuits which take as input a collection of binary digits and produce as output some function such as sum or product of inputs (interpreting n binary inputs as an n-digit binary number).

Consider the following circuit, which is called SP-Flip Flop. Interestingly, here the output of NAND-gates is also an input to other NAND-gates. Let us look at how it works. Let's imagine that we turn S on to 1. NOT-gate changes it to 0, and when it is fed to the NAND gate, the output of it would be 1, since 0 NAND X = 1 for any X (do you understand why it is so?). As a result, Q bulb will turn on.

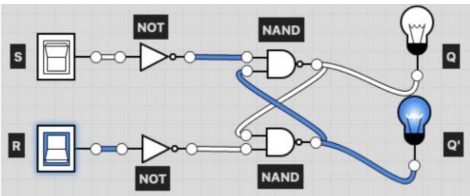
At the same time, R is off (is 0), and it is changed to 1 by the NOT-gate, and fed to NAND along with the output of the top NAND-gate, so the output of the bottom NAND-gate is 0 (since 1 NAND 1 = 0).



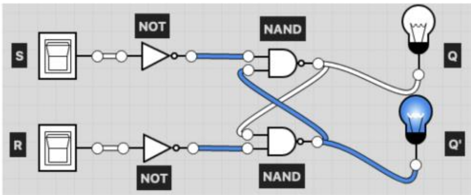
Interestingly, if we now flip S off, the lightbulbs will not change their state: lightbulb Q will stay on, and Q' will stay off: one of the inputs to the top NAND will always stay off, regardless of what S is.



Now, if we switch S to off, and turn R on, the lightbulbs will flip: Q' will be lit up, and Q will be off.



We will also observe a similar situation: now switching R to off will not change the state of lightbulbs:



The state when the top lightbulb is lit up is called S-state, that is the flip-flop is in SET position. When the lower lightbulb is on, the flip-flop is in RESET (R) position.

The interesting thing about this circuit is that it has memory: once it's in SET-state, the top lightbulb indicates it, and switching S-switch off won't change anything – the top lightbulb will still be on. Similarly, once we're in R-state, we can switch R-switch off, but the lightbulbs will still indicate that we are in the R-state (the 2nd light bulb is on).

There is a number of online simulators that allow you to create and test such circuits virtually; in particular, we can use https://logic.ly/demo/ (Demo version is enough for exercises).

Implication and equivalence

In addition to all previous logic operations, there are also operations expressing logical relationships, which we have not yet fully discussed. One is implication, also known as conditional and denoted by $A \Rightarrow B$ (reads A implies B, or "If A, then B"). It is defined by the following truth table:

A	B	$A \implies B$
T	T	T
T	F	F
F	T	T
F	F	T

Note that, in all situations where A is false $A \Rightarrow B$, is automatically true. E.g., a statement "if $2 \times 2 = 5$, then. . . " is automatically true, no matter what proposition one puts in place of dots. Every statement (be it true or false) "is implied by"/"follows from" any false statement! One may think of this in terms of a vacuous truth: Every statement is true in all cases when the antecedent false statement is true. "A implies B" indeed only means "whenever A, then B". Another logic operation, which we have already used, is called equivalence and defined as $A \Leftrightarrow B$, which is true if A, B always have the same value (both true or both false).

Exercise 1. Using truth tables, show that $A \Leftrightarrow B$ is equivalent to $A \Rightarrow B$ AND $B \Rightarrow A$.

Homework problems

1. Show that $A \Rightarrow B$ is not equivalent to $B \Rightarrow A$, i.e. they have different truth tables and one of them can be true while the other is false.
2. Prove the contrapositive law: $A \Rightarrow B$ is equivalent to $\neg B \Rightarrow \neg A$.
3. Show that $A \Rightarrow B$ t is equivalent to $(B \vee \neg A)$. Can you rewrite $\neg(A \Rightarrow B)$ without using implication operation?
4. Consider the following statement (from a parent to a son):

   "If you do not clean your room, you can't go to the movies"

   Using the laws of logic, establish whether it is the same as:

   (a) Clean your room, or you can't go to the movies
   (b) You must clean your room to go to the movies
   (c) If you clean your room, you can go to the movies

5. English language (and in particular, mathematical English) has a number of ways to say the same thing. Can you rewrite each of the verbal statements below using basic logic operations (including implications), and variables,
   - A: you get score of 90 or above on the final exam
   - B: you get an A grade for the class

   (As you will realize, many of these statements are in fact equivalent)

   (a) To get A for the class, it is required that you get 90 or higher on the midterm
   (b) To get A for the class, it is sufficient that you get 90 or higher on the midterm
   (c) You can't get A for the class unless you got 90 or above on the final exam
   (d) To get A for the class, it is necessary and sufficient that you get 90 or higher on the midterm

6. Show that in all situations where A is true and $A \Rightarrow B$ is true, B must also be true. [This simple rule has a name: it is called Modus Ponens.]

7. Show that if $A \Rightarrow B$ is true, and B is false, then A must be false. [This is called Modus Tollens.]

Recap: Logic operations

- NOT (for example, NOT A): true if A is false, and false if A is true. Commonly denoted by $\neg A$, $\sim A$, or (in computer science) $!\,A$.
- OR (for example A OR B): true if at least one of A, B is true, and false otherwise. Sometimes also called "inclusive or" to distinguish it from the "exclusive or" described in problem 4 below. Commonly denoted by $A \vee B$.
- AND (for example A AND B): true if both A, B are true, and false otherwise (i.e., if at least one of them is false). Commonly denoted by $A \wedge B$.
- NOR, NOT OR, for example NOT (A OR B), true if both A and B are false, and false if either A or B is true:
    - A NOR $B \Leftrightarrow \neg(A \vee B))$
- NAND, NOT AND, for example NOT (A AND B), true if at least one of A, B is false, and false otherwise (i.e., if both A, B are true):
    - A NAND $B \Leftrightarrow \neg(A \wedge B))$:
- XOR, exclusive OR gate: true if and only if exactly one of A, B is true and false otherwise:
    - A XOR $B \Leftrightarrow (\neg A \wedge B) \vee (A \wedge \neg B))$
- XNOR, equivalence gate, NOT exclusive OR: true if and only if both A, B are either true or false, and false otherwise, if A and B are different:
    - A XNOR $B \Leftrightarrow (\neg A \vee B) \wedge (A \vee \neg B))$

Recap: selected Logic Laws

- Double negation:

$$\neg(\neg A) \Leftrightarrow A$$

- Idempotency:

$$A \vee A \Leftrightarrow A$$

$$A \wedge A \Leftrightarrow A$$

- De Morgan (disjunction and conjunction negation):

$$\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$$

$$\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$$

- Distributive:

$$A \vee (B \wedge C) \Leftrightarrow (A \vee B) \wedge (A \vee C)$$

$$A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$$

Quiz

Exercise 1. Show that all basic logic operations, ¬, ∨, and ∧, can be expressed using NOR, which is defined as: A NOR $B =$ NOT (A OR B).

Exercise 2. Verify that NOR, NAND, XOR, and XNOR are commutative [you may use truth tables, or logic laws and the corresponding properties of basic operations, ¬, ∨, and ∧]:

- A NOR $B \Leftrightarrow B$ NOR A
- A NAND $B \Leftrightarrow B$ NAND A
- A XOR $B \Leftrightarrow B$ XOR A
- A XNOR $B \Leftrightarrow B$ XNOR A

Exercise 3. Using truth tables, logic laws, and corresponding properties of basic operations, ¬, ∨, and ∧, check which of the operations NAND, XOR, and XNOR are associative, and which are not:

- (A NOR B) NOR $C \neq A$ NOR (B NOR C) – this we showed last time in class
- (A NAND B) NAND C ? ? A NAND (B NAND C)
- (A XOR B) XOR C ? ? A XOR (B XOR C)
- (A XNOR B) XNOR C ? ? A XNOR (B XNOR C)

Exercise 4. Using logic laws, express in the simplest possible way using only basic operators, ¬, ∨, ∧,

- (A NAND B) NOR C
- A NAND (B NOR C)
- (A XOR B) XNOR C
- (A XNOR B) XOR C

Exercise 5. Using logic laws, express in the simplest possible way using only basic operators, ¬, ∨, ∧,

- (A XOR B) NAND C
- (A XNOR B) NOR C