Logic Circuits

Combinatorial Circuits I

Boolean Algebra: Background

Named after George Boole - who introduced the idea in:

> *An Investigation of the Laws of Thought*(1852)

John Atanasoff used the idea -

> did not recognise that the idea was Boolean Logic.

In 1940, Claude Shannon, in MS thesis at MIT

>*A Symbolic Analysis of Relay and Switching Circuits*

>proved that it was possible to simplify circuits using B. Algebra and that it was possible to solve B.Alg. problems using circuits

Formed the basis of modern digital circuit design.

BASICS

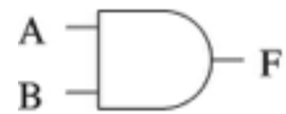
- Functions of n binary i/p vars. to ONE o/p var. are called BOOLEAN FUNCTIONS
- Boolean vars. (A,B,..X…) can only take on two discrete values (call them whatever)….
- Arb. Boolean functions can always be defined by a truth table of 2^n rows
- Since the output of B. func. is binary, only a finite number of n-var B. func. exist -

Q. How many?

Basic Logic Operations/Gates

- Simple operations:
  - AND
  - OR
  - NOT
- Functionality can be expressed by a truth table
  - A truth table lists output for each possible input combination
- Precedence
  - NOT > AND > OR
  - F = A notB + A B

    = (A (not B)) + ((A) B)

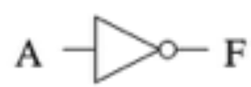


AND gate

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

OR gate

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1



NOT gate

A	F
0	1
1	0

Logic symbol

Truth table

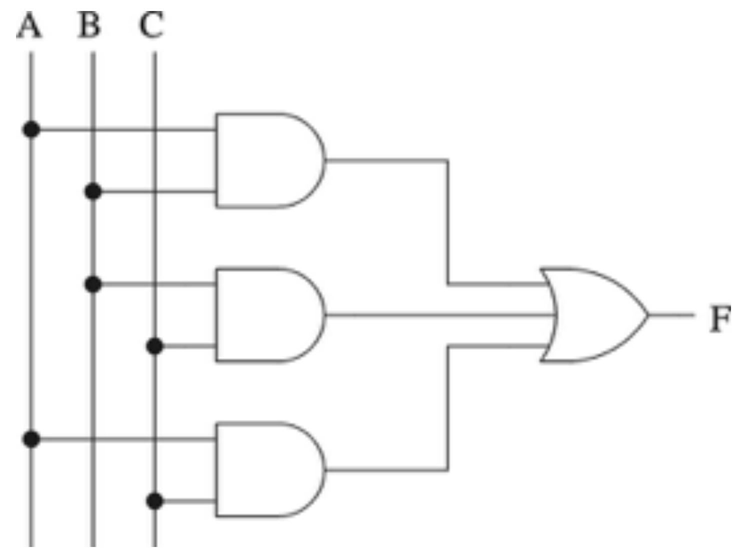
Boolean Algebra

- Logical functions can be expressed in several ways:
  - Truth table
  - Logical expressions - Using AND, OR , NOT
  - Graphical form - as a CIRCUIT
- Example:
  - Majority function
    - Output is 1 whenever majority of inputs is 1
    - We use 3-input majority function

Three Representations of A Boolean Function

3-input majority function

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



- Logical expression form

$$F = A B + B C + A C$$

- Proving logical equivalence of two circuits
  - Derive the logical expression for the output of each circuit
  - Show that these two expressions are equivalent
    - Two ways:
      - You can use the truth table method
        - For every combination of inputs, if both expressions yield the same output, they are equivalent
        - Good for logical expressions with small number of variables
      - You can also use algebraic manipulation
        - Need Boolean identities (see handout for identities and examples).

Logic Circuits

- Complete sets
  - A set of logic gates (operations) is complete
    - If we can implement any logical function using only the type of gates in the set
  - Some example of complete sets
    - {AND, OR, NOT} (Not a minimal complete set)
    - {AND, NOT}
    - {OR, NOT}
    - {NAND} (Proof of universality in class)
    - {NOR} (Proof of completeness: Assignment1)
  - Minimal complete set
    - A complete set with no redundant elements.