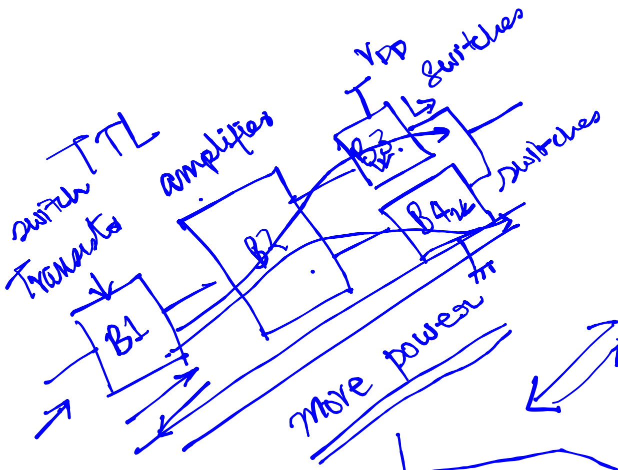
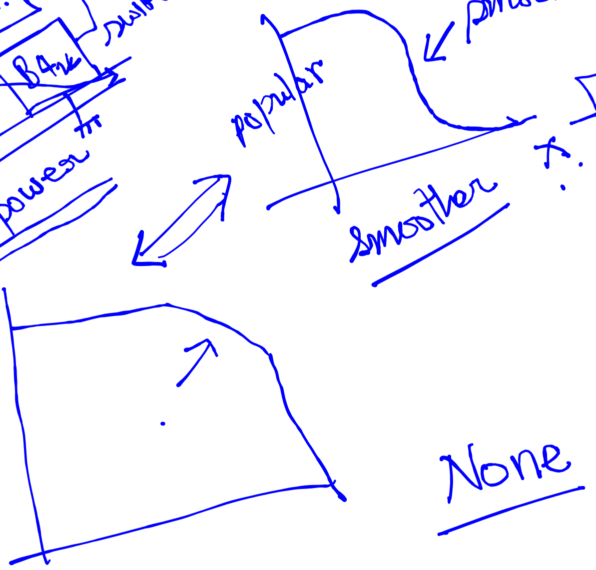
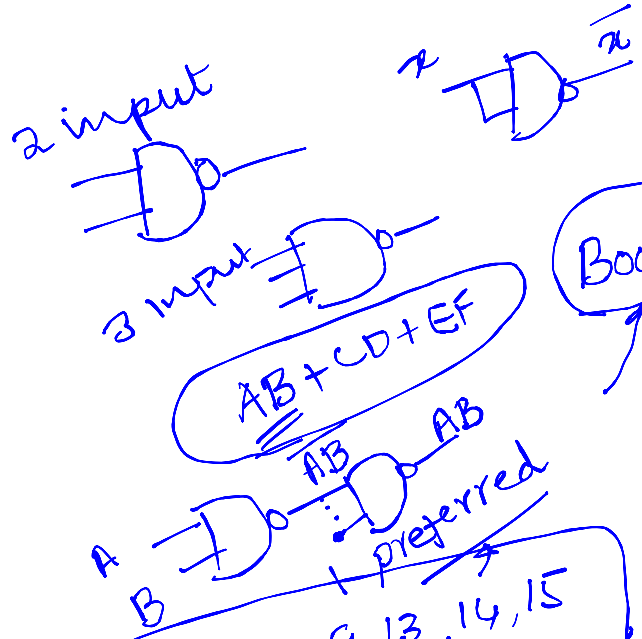




None



AB + CD + EF

Boole

~~AB~~

{3, 4, 5, 7, 9, 13, 14, 15}

Boolean Algebra

Any mathematics begins with an invention of some kind. Boole invented the basis for his algebra by forming a series of postulates or axioms upon which he built the rest of the algebra. One form of these contains five postulates that relate very nicely to what we'll need for developing digital logic circuits.

Basic Operations: NOT ($^{-}$), AND($\cdot$) and OR($+$) $B\{0,1\}$

Postulates (No Proof Required)

axioms or rules

P1: The operations are closed.

$x, y \in B\{0,1\}$ $\bar{x} \in B\{0,1\}$; $x \cdot y \in B\{0,1\}$; and $x + y \in B\{0,1\}$

P2: For every operation there exists and identity element

$$x \cdot 1 = x \qquad x + 0 = x$$

Postulates

P3: The operations are commutative.

$$x + y = y + x \qquad and \qquad x \cdot y = y \cdot x$$

algebra, arithmetic

P4: (a) The operator $(\cdot)$ is distributive over $(+)$; i.e., $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

(b) The operator $(\cdot)$ is distributive over $(\cdot)$; i.e., $x + (y \cdot z) = (x + y) \cdot (x + z)$

$x \cdot x + x \cdot z + y \cdot x + y \cdot z$

P5: For every element $x \in B$, there exists an element $\bar{x} \in B$ (called complement of x) such that $(a) x + \bar{x} = 1$ and $(b)\ x \cdot \bar{x} = 0$

P6: There exist at least two elements $x, y \in B$ such that $x \neq y$.

Boolean Algebra Duality:

A+B

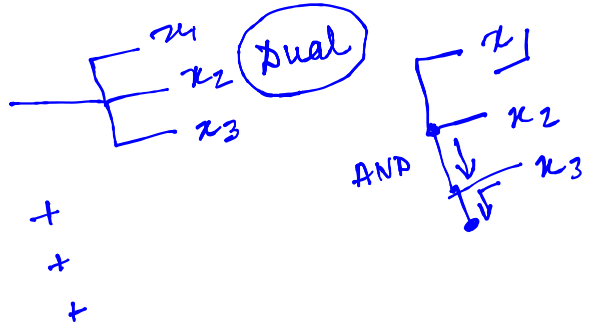
Dual → operation
AND → OR
OR → AND
but no dual for NOT

- Given a logical expression, its dual is obtained by replacing $+$ $(\cdot)$operators with operators $\cdot$ $(+)$ respectively and by replacing 0 (1) with 1 (0) respectively.

$x \neq \bar{x}$

$0 \xrightarrow{\text{dual}} 1$

- The dual of any true statement in Boolean algebra is also a true statement



$$(x + y) + z \xrightarrow{\text{Dual}} (x \cdot y) \cdot z$$

$$\bar{x} \cdot \bar{y} \xrightarrow{\text{Dual}} \bar{x} + \bar{y}$$

Boolean Algebra ----- Some Theorems/Properties

Theorems (Proof exists)	Expression	Dual Expression
T1: Idempotency	$x + x = x$	$x \cdot x = x$
T2: Complete Cover	$x + 1 = 1$	$x \cdot 0 = 0$
T3: Involution	$\overline{(\bar{x})} = x$	$\overline{(\bar{x})} = x$
T4: Associativity	$(x + y) + z = x + (y + z)$	$(x \cdot y) \cdot z = x \cdot (y \cdot z)$
T5: DeMorgan's Law	$\overline{(x + y)} = \bar{x} \cdot \bar{y}$	$\overline{(x \cdot y)} = \bar{x} + \bar{y}$
T6: Absorption/Combining	$x + x \cdot y = x$	$x \cdot (x + y) = x$
T6a: Second Absorption	$x + \bar{x} \cdot y = x + y$	$x \cdot (\bar{x} + y) = x \cdot y$
T7: Consensus	$x \cdot y + \bar{x} \cdot z + y \cdot z$ $= x \cdot y + \bar{x} \cdot z$	$(x + y) \cdot (\bar{x} + z) \cdot (y + z)$ $= (x + y) \cdot (\bar{x} + z)$

Property	Expression	Dual Expression
Py1: Combination	$x \cdot y + x \cdot z = x \cdot (y + z)$	$(x + y) \cdot (x + z) = x + y \cdot z$
PY2: Combination	$x \cdot y + x \cdot \bar{y} = x$	$(x + y) \cdot (x + \bar{y}) = x$

Theorems

Application of of DeMorgan's Theorem that is easier to say in words than in mathematical form.

To complement a complete expression,

(a) put in all parentheses to group terms correctly—don't rely on the assumed hierarchy of • over +,

(b) change all the • to + and all the + to •, and

(c) change all complemented variables to uncomplemented ones and vice versa.

(d) Now clean up the parentheses if you need too.

- Complement of a function is obtained by taking the dual of the function and complement each literal.

Dual literal

$$F_1(x, y) = x + (\bar{x} \cdot y) \dashrightarrow F_{1D}(x, y) = x \cdot (\bar{x} + y)$$

$$\overline{F_1(x, y)} = x + \bar{x} \cdot y \dashrightarrow \overline{F_1(x, y)} = \bar{x} \cdot (x + \bar{y})$$

$(x + \bar{x} + y)$

$x \cdot (x + y)$

De Morgan's Law

Applying DeMorgan Theorem

DeMorgan's Theorem is quite useful because complementing an expression arises from time to time.

To illustrate the use of the generalized form of the theorem, we will do an example.

Consider the logic expression that I want to complement:

$$Z = A + B \bullet \bar{C} + \overline{(\bar{D} + E \bullet \bar{F} \bullet G)}$$

We want $\bar{Z} = \overline{A + B \bullet \bar{C} + \overline{(\bar{D} + (E \bullet \bar{F} \bullet G))}}$

First, we will install the parentheses to group terms so that we are not relying on the assumed hierarchy of • over +:

$$\bar{Z} = \overline{A + (B \bullet \bar{C}) + \overline{(\bar{D} + (E \bullet \bar{F} \bullet G))}}$$

$D \cdot (\bar{E} + F + \bar{G})$

$\bar{A} \cdot (\bar{B} + C) \cdot \underbrace{(D \cdot (\bar{E} + F + \bar{G})}$

Applying DeMorgan Theorem

Then we will unprime the primed terms and prime the unprimed ones and also exchange · and + operators.

Note that there is a "big" term in the parentheses at the right that has a prime on it (just after double parentheses).

I remove this prime and leave all the terms and operators inside alone.

$$\bar{Z} = \bar{A} \bullet (\bar{B} + C) \bullet (\bar{D} + (E \bullet \bar{F} \bullet G)).$$

Finally, I clean up any unnecessary parentheses:

$$Z' = \bar{A} \bullet (\bar{B} + C) \bullet (\bar{D} + E \bullet \bar{F} \bullet G).$$

The result is now the complement of the original expression.

This can be carried to any depth of the parentheses or complication of expression, but "real-world" logic functions rarely get that messy.

USING BOOLEAN ALGEBRA

- Let's use Boolean algebra to prove one of the theorems.
- We want to demonstrate a way of showing that something is true (or false, for that matter).
- The simplest approach, which is almost a no-brainer, uses perfect induction to demonstrate that every possible combination of 0s and 1s yields the correct result.
- Perfect induction is most easily carried out by writing a truth table that tells the whole truth and nothing but the truth.
- We will prove using perfect induction the second half of property PY1, which says $\underbrace{(A + B) \cdot (A + C)}_{\text{Truth Table}} = \underbrace{A + B \cdot C}_{\text{brute force or induction}}$. Theorems.

USING BOOLEAN ALGEBRA

Figure shows the truth table for the given function

$(A + B) \cdot (A + C) = A + B \cdot C$

same

A	B	C	X= A+B	Y= A+C	X•Y	A	Z= B•C	A+Z
0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0
0	1	0	1	0	0	0	0	0
0	1	1	1	1	1	0	1	1
1	0	0	1	1	1	1	0	1
1	0	1	1	1	1	1	0	1
1	1	0	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1

=

Proved by induction

Figure Truth Table

USING BOOLEAN ALGEBRA

- Here we developed the table by creating the "input" columns *A*, B, and *C* and listing all eight of the possible arrangements of 0s and 1s. (There are three variables, each of which can have one of two values, so there are $2^3 (= 8)$ rows in this truth table.) We list them in ascending binary number order
- Compute two columns of the parenthesized combinations (*A* + B) and (*A* + *C*) using postulates P3, P4, and P5 that tell how the + operation works.
- Now combine these two columns using the • operation and postulates P3, P4, and P5.

That's the result for the left-hand side of the = sign of the theorem we are proving.

USING BOOLEAN ALGEBRA

Now do the same for the right-hand side:

- Repeat the A column for convenience.
- Compute the value of B•C using postulates P3, P4, and P5.
- Now combine these two columns using the + operation and postulates P3, P4, and P5.

That's the result for the right-hand side of the = sign.

Are the two columns equal all

- the way down? If so, we have proven the theorem by showing that it works for every possible combination of values of A, B, and C. That's proof by perfect induction

De Morgan's Law

Proving (A + B) • (A + C) = A + B•C, using postulates and Theorems

Let $A + B = X$ and write $(A+B)\cdot(A+C) = X\cdot(A+C) = \boxed{X\cdot A + X\cdot C}$ (by postulate 4)

$X\cdot A + X\cdot C = (A+B)\cdot A + (A+B)\cdot C = A + A\cdot C + B\cdot C = \boxed{A + B\cdot C}$ (using T6 and postulate 4)

$A\cdot A + A\cdot B \Rightarrow A(A+B)$ $A\cdot A = A$

$A\cdot(1+B) = A$

$A\cdot(1+C)$

$\downarrow$ 1

$A\cdot A + A\cdot B$

$A + A\cdot B \rightarrow A\cdot(1+B) \rightarrow A\cdot 1$

Logic Simplification

A logic circuit has been designed to to implement the function

$$Z = \boxed{A \cdot B \cdot C' + A \cdot C + A' \cdot C.}$$

$A \cdot C + \bar{A} \cdot C = C(A + \bar{A}) = C \cdot 1 = C$

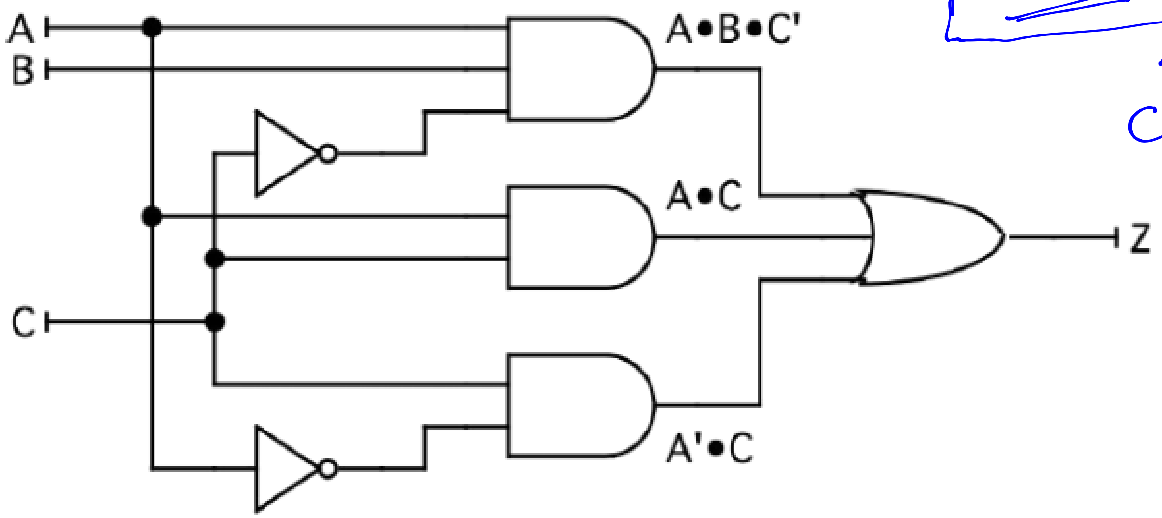
$\boxed{A \cdot B \cdot \bar{C} + C}$

$C + \bar{C} \cdot (A \cdot B)$

$x + \bar{x} \cdot y$

$= x + y$

The logic circuit is shown in Fig.



This circuit requires four gates and two inverters.

Logic Simplification

We will use Boolean algebra to reduce the number of gates if possible.

The process is by no means algorithmic, which means we can't write any definitive rules to tell how to proceed.

This depends on combining and simplifying terms.

Our goal is to reduce the number of gates, which means combining terms to get rid of as many + and • operations as we can.

Logic Simplification

canonical form

Here's the original expression: $A \cdot B \cdot C + A \cdot \bar{B} C$

$\bar{A} \cdot B \cdot C + \bar{A} \cdot \bar{B} \cdot C$

$$Z = A \bullet B \bullet \bar{C} + A \bullet C + \bar{A} \bullet C.$$

reduced / minimized

- Apply PY2 to the last two terms, reducing them to simply C:

$$Z = A \bullet B \bullet \bar{C} + C.$$

- Use T4(associativity) and P3 (commutativity) to regroup the remaining terms:

$$Z = C + \bar{C} \bullet (A \bullet B).$$

- Now expand this expression somewhat by using T6a $(A + \bar{A}B = A + B)$ to get the simplified form:

$Z = C + (A \bullet B)$ ------ A much simpler expression

introduce definitions.

proving

The real test, though, is whether the gate circuit is simpler. Look at Figure
Have we succeeded in reducing the circuit from six logic gates to only two logic gates.

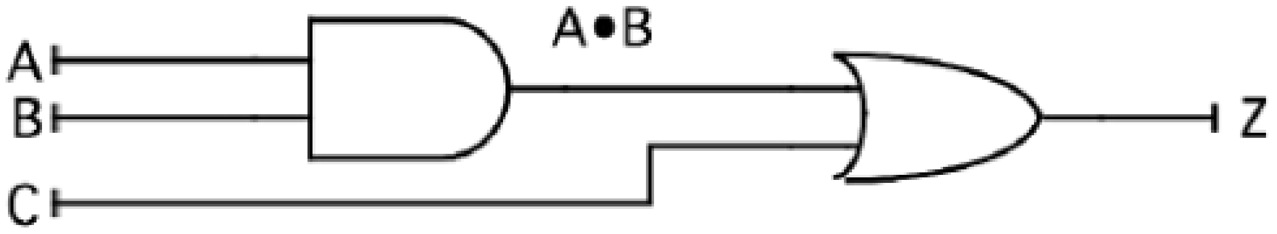


FIGURE A logic circuit simplified

CANONICAL FORMS

- The starting point for most of the algorithms for simplification is an expression in canonical form.
- While that sounds like something awful, it isn't. It's just a neat starting point that gets the algorithmic process off to a proper start.
- To get through this, we have some new terms to learn.
- In every field there are terms that are specific to that field, that help define things sharply.
- Digital logic is no different.

a, b, c, d

Definitions

a+b+c+d
sum

canonical

$f = a \cdot b \cdot c \cdot d + a \cdot b \cdot c \cdot \bar{d}$

product $a \cdot b \cdot c$ minimised form

Here are the important definitions that we'll need:

- Product term is a term where the variables are connected by and operators, terms such as $A \bullet B$, $\bar{C} \bullet D \bullet E$, and even the simple term F.
- Sum term is a term where the variables are connected by or operators, terms such as $A + B$, $\bar{C} + D + E$, and even the simple term F.
- Literal is any appearance of a variable in a term, whether primed or not. In the terms above, the literals are $A, B, \bar{C}, D, E$, and F. If the variable appears both primed and unprimed, as in G and $\bar{G}$, each is a literal.
- Sum-of-products is an expression consisting of product terms linked by or operators, as in $A \bullet B + \bar{C} \bullet D \bullet E + F$. product term SOP expression $\boxed{F = F \cdot 1}$
- Product-of-sums is an expression consisting of sum terms linked by and operators, as in $(A + B) \bullet (\bar{C} + D + E) \bullet F$. POS expression sum term $F+0$

Definitions

$f = A \cdot B \cdot \bar{C} + \bar{A} \cdot BC + A \cdot \bar{B}\bar{C}$

SOP

- Normal term is a term, either a product term or a sum term, in which no variable appears more than once.
  - For example, the term $\boxed{A \bullet B \bullet C \bullet C}$ is not normal because we can use theorem T3 to remove one appearance of C to yield $A \bullet B \bullet C$.

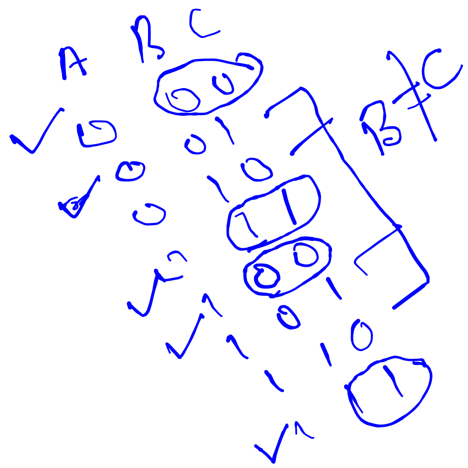
    never in an exp.
  - Also, the term $D \bullet E \bullet \bar{E}$ is not normal because we can use theorem T5 to reduce the term to 0.
  - Non-normal terms are pretty obvious and represent unnecessary complications in our logic circuits.

$A \cdot B = A \cdot C$

$B = C$

8

ABC



Definitions

Condition	A	B	C	D	Z
	0	0	0	0	1

$Z = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D}$ minterm
canonical

- Minterm is a normal product term that contains all the variables of the problem.
  - For example, if the problem contains the variables A, B, and C, then all minterms must contain these three variables, with or without primes.
  - So $\overline{A} \bullet B \bullet C$ and $A \bullet \overline{B} \bullet \bar{C}$ are two possible minterms in this case.
  - The term A•B is not a minterm in this case because it does not contain C.
  - To make life a little more orderly, we usually write the variable names in alphabetical order.

$Z = \bar{A} + \bar{B} + \bar{C} + \bar{D}$

$\overline{A \cdot B \cdot C \cdot D}$

~~$\bar{A} + \bar{B} + \bar{C} + \bar{D}$~~

A	B	C	D	Z
0	0	0	0	1
0	0	0	1	1

$Z = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot D$

realization

$\bar{A} \cdot \bar{B} \cdot \bar{C} \cdot (D + \bar{D})$

$= \bar{A} \cdot \bar{B} \cdot \bar{C}$ not a minterm