

LOGIC EXPRESION MINIMIZATION

- Goal is to find an equivalent of an original logic expression that:
 - a) has fewer variables per term
 - b) has fewer terms
 - c) needs less logic to implement
- There are three main manual methods
 - Algebraic minimization
 - Karnaugh Map minimization
 - Quine-McCluskey (tabular) minimization

ALGEBRAIC MINIMIZATION

- Process is to apply the switching algebra postulates, laws, and theorems to transform the original expression
 - Hard to recognize when a particular law can be applied
 - Difficult to know if resulting expression is truly minimal
 - Very easy to make a mistake
 - Incorrect complementation
 - Dropped variables

CONSENSUS THEOREM (4.1 p. 66)

Consensus theorem states:

$$XY + X'Z + YZ = XY + X'Z$$

The **YZ** term is called the *consensus term* and is redundant. The consensus term is formed from a PAIR OF TERMS in which a variable (X) and its complement (X') are present; the consensus term is formed by multiplying the two terms and leaving out the selected variable and its complement.

The consensus of XY, X'Z is YZ .

PROVE THE CONSENSUS THEOREM

Consensus Theorem Proof:

$$\begin{aligned} XY + X'Z + YZ &= XY + X'Z + (X + X')YZ \\ &= XY + X'Z + XYZ + X'YZ \\ &= (XY + XYZ) + (X'Z + X'YZ) \\ &= XY(1 + Z) + X'Z(1 + Y) \\ &= XY + X'Z \end{aligned}$$

You could also use a truth table to prove this.

DUAL OF THE CONSENSUS THEOREM

$$(X + Y) (X' + Z) (Y + Z) = (X + Y) (X' + Z)$$

The consensus of $(X + Y)(X' + Z)$ is $(Y + Z)$.

How do you use the consensus theorem? Simply be suspicious anytime you have two terms that have a variable and its complement. Form the consensus term and see if it is present; if consensus term is present, just get rid of it.


THE CONSENSUS THEOREM

EXAMPLE:

$A'C'D + \textcolor{brown}{A'BD} + \underline{BCD} + \textcolor{brown}{ABC} + ACD'$ underlined term can be eliminated by consensus thm.



$A'C'D + \underline{A'BD} + BCD + \underline{ABC} + ACD'$ start over -- this time eliminate two other terms



Now Consider:

$F = ABCD + B'CDE + A'B' + BCE'$ cannot reduce by consensus thm.

$F = \underline{ABCD} + \underline{B'CDE} + A'B' + BCE' + \textcolor{green}{ACDE}$ add the consensus term $\textcolor{green}{ACDE}$ first

$F = BCE' + \underline{ABCD} + ACDE + A'B' + \underline{B'CDE} + ACDE$



Then the two underlined terms become redundant by consensus thm.

ADJACENCY

$$XY + XY' = X \quad (\#9)$$

$$(X+Y)(X+Y') = X \quad (\#9D)$$

Look for two terms that are identical except for compl. in one variable.
-Application removes one term and one variable from the remaining term.

Combining Terms by Adjacency:

Example:

$$abc'd' + abcd' = abd'$$

Example: (duplicating abc first then eliminating by adjacency)

$$\begin{aligned} ab'c + abc + a'bc &= ab'c + abc + abc + a'bc \\ &= ac + bc \end{aligned}$$

ABSORPTION

$$X + XY = X \quad (\#10)$$

$$X(X+Y) = X \quad (\#10D)$$

Look for two terms that are the same except for an extra variable

Eliminating Terms by Absorption:

Example:

$$a'b + a'bc = a'b$$

—

SIMPLIFICATION

$$X + X'Y = X + Y$$

$$X(X' + Y) = XY$$

Eliminating Literals by Simplification:

Example:

$$\begin{aligned} A'B + A'B'C'D' + ABCD' &= A'(B + \underline{B'}C'D') + ABCD' \quad (\text{factored}) \\ &= \underline{A'}(B + \underline{C'D'}) + ABCD' \quad (\text{simplification}) \\ &= A'\underline{B} + A'C'D' + A\underline{B}CD' \quad (\text{distributed } A') \\ &= B(A' + \underline{A}CD') + A'C'D' \quad (\text{factored out } \underline{B}) \\ &= B(A' + CD') + A'C'D' \quad (\text{simplification}) \end{aligned}$$

PROVING VALIDITY OF AN EQUATION

1. Construct Truth Table and evaluate both sides of eqn.
2. Make L.S. equal R.S. or R.S. equal L.S. by algebraic manipulation.
3. Reduce both L.S. and R.S. independently to the same expression.

To show an equation is NOT Valid

Give one combination of values ('0', '1') of the variables for which $L.S. \neq R.S.$

(This is equivalent to finding one line in a truth table for which L.S. and R.S. have different values.)

VALID OPERATIONS

When attempting to prove that an equation. is valid:

It is permissible to perform the same operation on on both sides of the eqn. as long as the operation is reversible (has an inverse) within Boolean Algebra..

Complement both sides -- Allowed

Mult. both sides by same expression -- Not Allowed

Add same term to both sides --- Not Allowed

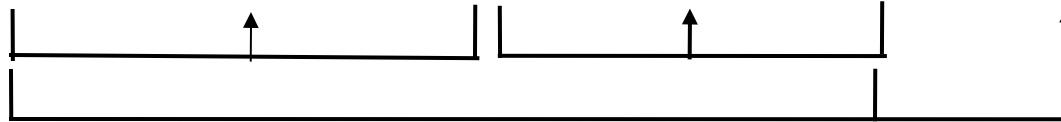
PROVING VALIDITY OF AN EQUATION

Example 1 p. 71 -- Show that:

$$A'BD' + BCD + ABC' + AB'D = BC'D' + AD + A'BC$$

L.S. $= A'BD' + BCD + ABC' + AB'D$

$$= A'BD' + \textcolor{violet}{BC'D'} + ABC' + \textcolor{violet}{ABD} + BCD + \textcolor{violet}{A'BC} + AB'D$$



(added consensus terms)

$$= AD + \underline{A'BD'} + \textcolor{violet}{BC'D'} + \underline{ABC'} + \underline{BCD} + \textcolor{violet}{A'BC}$$

($AB'D + \textcolor{violet}{ABD} = AD$ -- adjacency)

$$= AD + \textcolor{violet}{BC'D'} + \textcolor{violet}{A'BC} = \text{R.S.}$$

(underlined terms removed as consensus terms)

($BC'D' + AD \rightarrow \underline{ABC'}$, $AD + A'BC \rightarrow \underline{BCD}$, $BC'D' + A'BC \rightarrow \underline{A'BD'}$)

PROVING VALIDITY OF AN EQUATION

Example 2 p. 71 -- Show that: $A'BC'D + (A' + BC)(A + C'D') + BC'D + A'BC'$

$$= ABCD + A'C'D' + ABD + ABCD' + BC'D$$

$$\text{L.S.} = \underline{A'BC'D} + (A' + BC)(A + C'D') + BC'D + A'BC'$$

$$= (A' + BC)(A + C'D') + BC'D + A'BC'$$

(eliminated $A'BC'D$ by absorption (#10) with $A'BC'$)

$$= \underline{AA'} + ABC + A'C'D' + \underline{BCC'D'} + BC'D + \underline{A'BC'}$$

(by distributive law)

$$= ABC + A'C'D' + BC'D + \underline{A'BC'}$$

$$= ABC + A'C'D' + BC'D$$

(eliminated $A'BC'$ by consensus, $A'C'D' + BC'D \rightarrow A'BC'$)

$$\text{R.S.} = ABCD + A'C'D' + ABD + ABCD' + BC'D$$

$$= ABC + A'C'D' + \underline{ABD} + BC'D \text{ (ABCD and ABCD' adjacent)}$$

$$= ABC + A'C'D' + BC'D$$

(eliminated ABD by consensus, $ABC + BC'D \rightarrow ABD$)

Since **L.S. = R.S.** the original eqn. is valid

What do you need to know?

- Basic Boolean Theorems
- Algebraic Simplification
- Consensus Theorem
- Adjacency, Absorption, Simplification
- Proving Validity of Equation