CS1110
Systems Organisation

19/10/2015

Proving T6 by Algebraic Manipulation

$$A+(\bar{A}.B) \xrightarrow{P_3} (A+\bar{A}).(A+B) \xrightarrow{P_5} 1.(A+B) \xrightarrow{P_4} A+B \quad Q.E.D.$$

Proof by Perfect Induction (exhaustive)

A	B	$\bar{A}$	$\bar{A}.B$	$A+(\bar{A}.B)$	$A+B$
0	0	1	0	0	0
0	1	1	1	1	1
1	0	0	0	1	1
1	1	0	0	1	1

As these columns are the same, the expressions are equal.

You can always use either method to prove any of the theorems.

De Morgan's Theorems

- These provide a way of replacing the . operator with the + operator, and vice-versa
- Do this using the inverse
- This is useful for changing logic expressions to make them easier to implement.
  (E.g. NAND gates are v.easy to construct, so we could make everything out of those).
  (Other e.g. changing something to AND can reduce the amount that needs to be evaluated if one condition is false.)

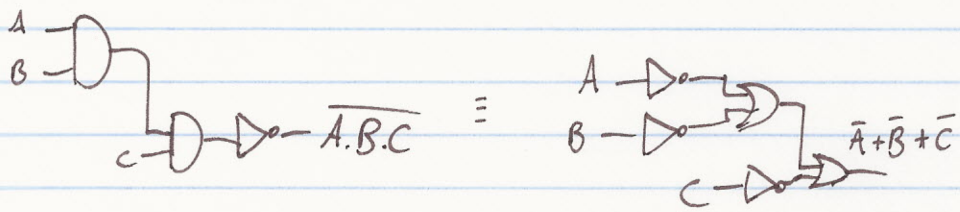
De Morgan's Theorems (cont.)

Method:

1. Change the operator
2. Invert each input variable
3. Invert the whole expression

Example: $\overline{A.B.C}$

1. $\Rightarrow \overline{A+B+C}$
2. $\Rightarrow \overline{\bar{A}+\bar{B}+\bar{C}}$
3. $\Rightarrow \bar{A}+\bar{B}+\bar{C}$



Note: You can change multiple . operators simultaneously, as we have done above.
You can do the same with multiple + operators.
⇒ Do all (or some) of the +s separate from the .s.

Where are we going?

We're looking for a mechanistic way of deriving equations from a truth table.

- Start by laying out the inputs and the desired outputs for each input combination (truth table).
- From this we get an equation, which we can then optimise with DeMorgan's Laws.

CS1110
Systems
Organisation

19/10/2015

Some More Terminology (in pursuit of a mechanism)

Minterm: An expression formed by ANDing all input variables in either their original or complemented form

Example: Given A, B, C, there are minterms:

$A.B.C\,;\ \bar{A}.\bar{B}.\bar{C}\,;\ A.B.\bar{C}\,;\ A.\bar{B}.C \ldots$

There will be 2^3 $(2^{\text{number of inputs}}) = 8$ minterms in this case.

Think of this as all possible value combinations in a Truth Table.

i.e. $A.B.C \equiv$

A	B	C
1	1	1

$\bar{A}.B.\bar{C} \equiv 0\ 1\ 0$

We will use this in our mechanistic process.

The following are not minterms:

$A.B\,;\ \overline{A.B.C}\,;\ A.A.B.C\,;\ A+B.C$

have to use all variables

can't invert after .

can't use any variable more than 1 time

can't use +

A	B	C	Minterms	
0	0	0	$\bar{A}.\bar{B}.\bar{C}$	m_0
0	0	1	$\bar{A}.\bar{B}.C$	m_1
0	1	0	$\bar{A}.B.\bar{C}$	m_2
0	1	1	$\bar{A}.B.C$	m_3
1	0	0	$A.\bar{B}.\bar{C}$	m_4
1	0	1	$A.\bar{B}.C$	m_5
1	1	0	$A.B.\bar{C}$	m_6
1	1	1	$A.B.C$	m_7