

Logic Design

Sandhya Dwarkadas

Logic Minimization

Essence of simplification -

- repeatedly find two-element sub-sets of true values in which only one variable changes its values while the other variables do not
- Apply the Unifying Theorem to eliminate the single varying variable -
 - $F = A.\bar{B} + A.B$
 - $F = A.(\bar{B} + B)$ - applying the Distributive law of Boolean Algebra
 - $F = A$ - applying the Inverse law of Boolean Algebra

Karnaugh Maps

A graphical representation of the truth table - boolean cube in n-dimensional space where n is the number of input variables

Entry for each combination of input variables specifying the value of the output function

Uses Gray Code encoding - advancing from 1 index to the next changes the value of only a single input variable/bit

Multi-dimensional table with logical adjacency along a dimension - any two adjacent elements (horizontal or vertical) are distance one apart

Adjacencies provide clues about whether uniting theorem can be

applied

Goal: find a minimum cover of the 1's using rectangles or squares containing a power of two number of 1's

In essence, a mechanical method to find the *don't cares* in the truth table

Finite State Machines

Sequential system that contains state

Described with a state table

Set of states and two functions -

- Next state function
- Output function

Moore Machine - Output = F(Current State)

Mealy Machine - Output = F(Current State + Current Input)

Clocks

Free-running signal with a fixed cycle time

Edge-triggered clocking - state changes occur on clock edge

Constraint in synchronous or clocked system: Input must be valid immediately before and after active edge - *setup* and *hold* times

To avoid races -

$$T_{period} > t_{prop} + t_{combinational} + t_{setup} + t_{skew}$$

$$t_{prop} + t_{combinational} - t_{skew} > t_{hold}$$