VLSI SYSTEMS LAB

LAYOUTDRIVEN SYNTHESIS FOR SUBMICRON TECHNOLOGY MAPPING EXPANSIONS TO REGULAR LATTICES

20023671
Woong Hwangbo

KAIST

Abstract

- A basic concept in VLSI layout
- Applications to submicron design, quantum devices,and designing new fine-grain FPGAs
- In a regular arrangement of cells, every cell is connected to 4, 6, 8 neighbors and to vertical, horizontal and diagonal buses.

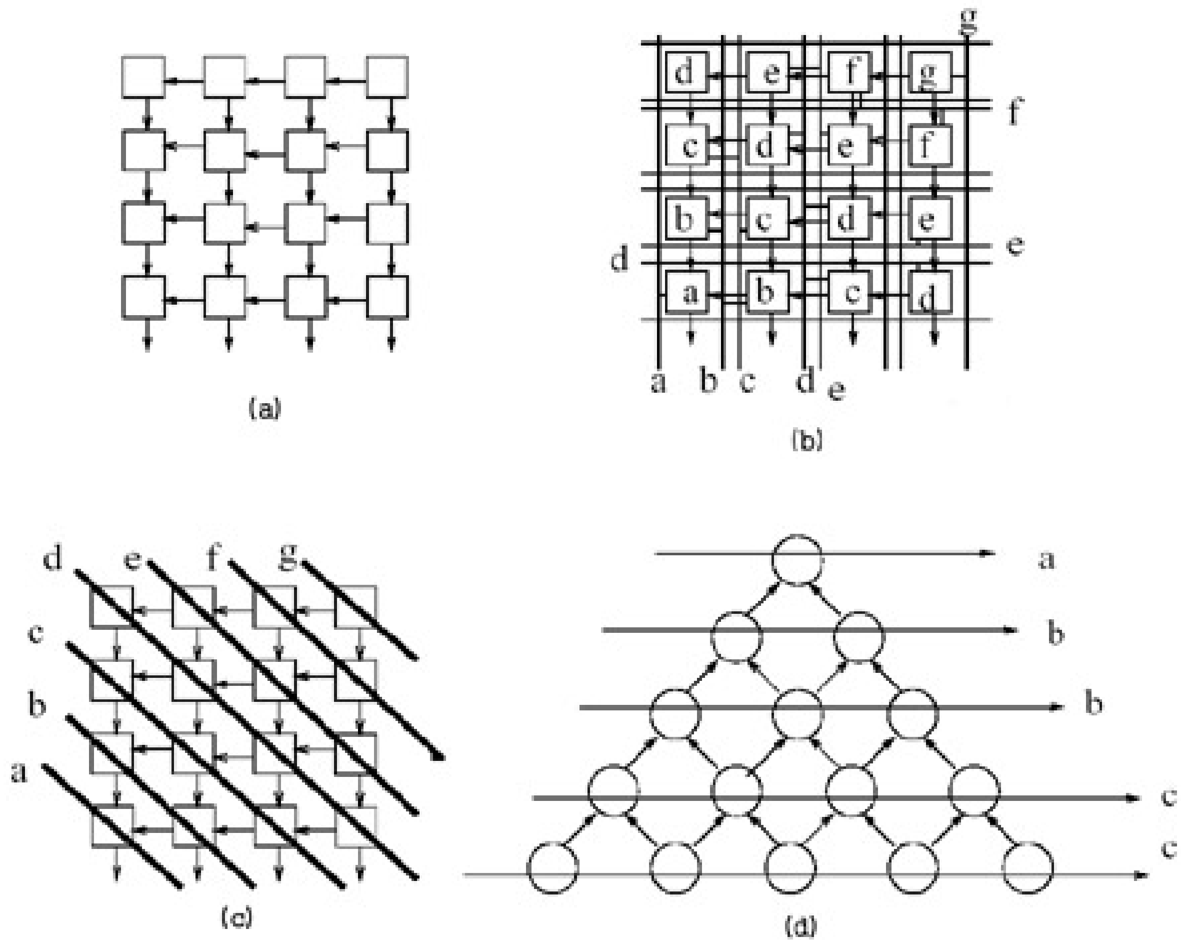
Expansions of functions

- Operators that transform a function to a few simpler functions
- Two types
  - 1st category
    - canonical (such as Shannon)
    - noncanonical(such as Sum-of-Product)
  - 2nd category
    - Maximum-type
    - Linearly-Independent type

Three components of lattice diagrams

- Expansion of a node function, which creates several successor nodes of this node.
- Joining operation joins several nodes of a bottom of the lattice.
- Regular geometry to which the nodes are mapped, this geometry guides which nodes of the level are to be joined.

Layout Geometries–4 neighbors



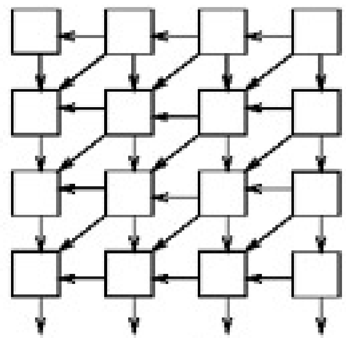
(a) (b) (c) (d)

Layout Geometries–4 neighbors

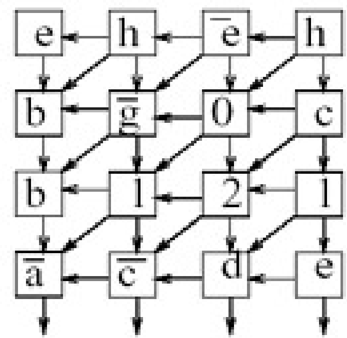
- planar & based on a rectangular grid
- Each cell has two inputs and two outputs.
- For a 4-neighbor lattice geometry , any canonical form of Reed-Muller logic and its Linearly Independent generalizations can be realized.
- Any MV logic can be also realized in the 4-neighbor lattice.

VLSI SYSTEMS LAB

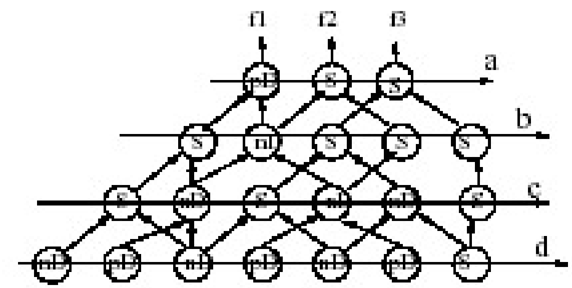
Layout Geometries–6 neighbors



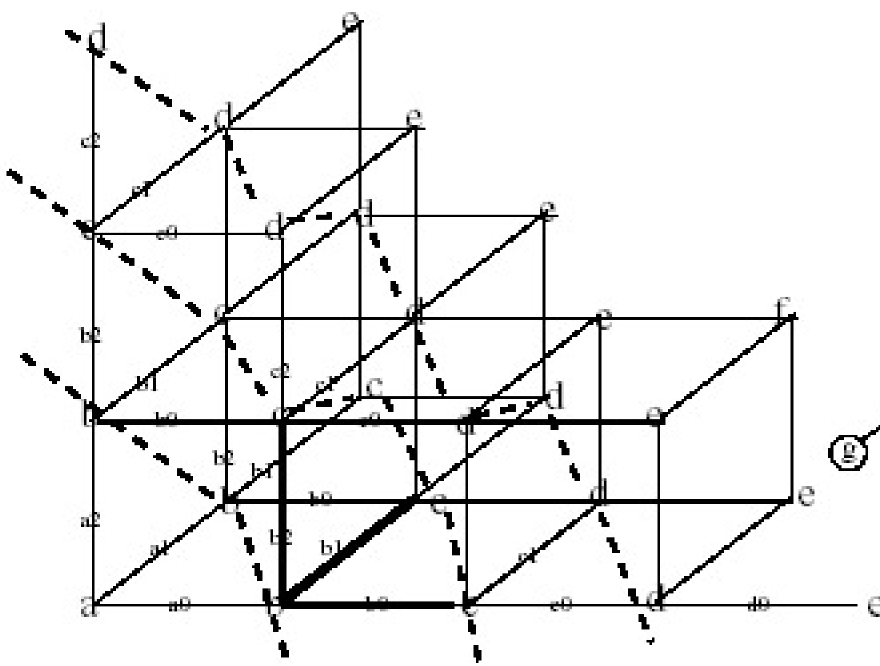
(a)



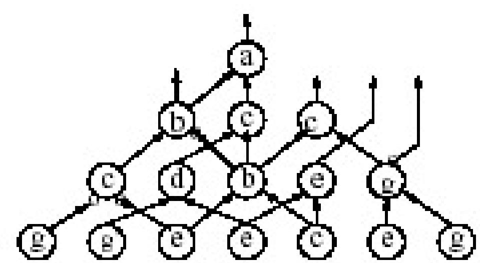
(b)



(c)



(d)



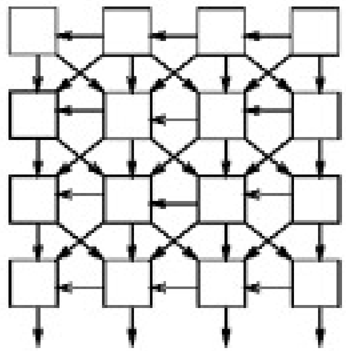
(e)

KAIST

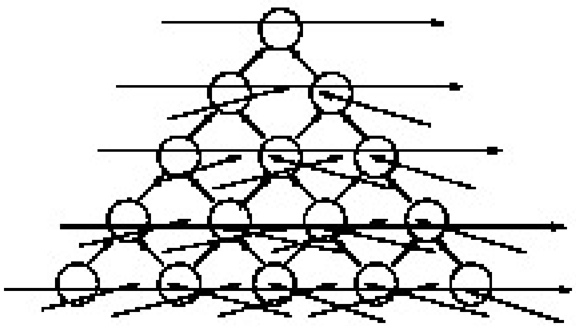
Layout Geometries–6 neighbors

- The rectangular grid is enhanced with one diagonal connection, thus every cell has three inputs and three outputs.
- This can realize the generalized ternary diagrams.

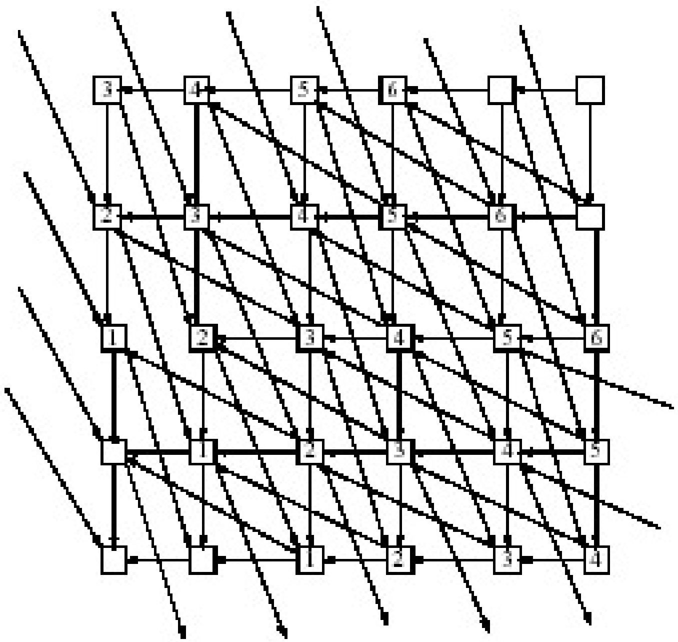
Layout Geometries–8 neighbors



(a)



(b)



(c)

Layout Geometries–8 neighbors

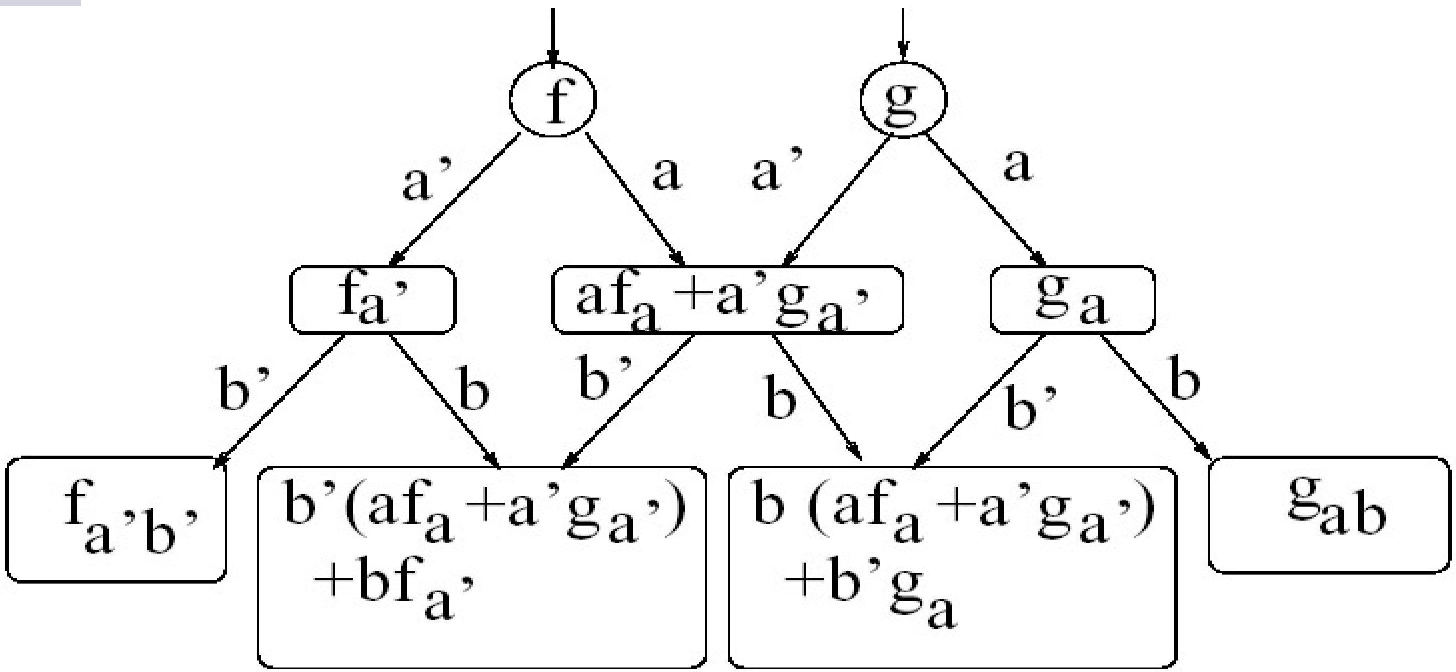
- The rectangular grid is enhanced with two diagonal connection, thus every cell has four inputs and four outputs.
- This can realize the generalized quaternary diagrams.

Linearly Independent type

- Generalizations of Positive Davio and Negative Davio expansions.
- For arbitrary algebras they should have at least one linear group operation.
- Usually based on the algebraic structure of an arbitrary field.
- They include
  - S(Shannon expansion)
  - Positive and Negative Davio pD and nD respectively
  - general Linearly Independent(binary and MV)
  - EXOR Ternary expansions

Maximum-type

- Assumption : each binary function(f)
  - f is represented by a pair [ON(f),OFF(f)]
  - fa = [ON(f),OFF(f)]
  - Shannon



Maximum-type operations

- Prior figure explains the principle of creating a Shannon Lattice based on ordered Shannon expansions for f , g.
- A standard cofactor fx where x is a variable does not depend on this variable.
- This cofactor is vacuous cofactor(denoted by v-cofactor) though fx is still a function of all variables including x, but as a result of cofactoring the variable x becomes vacuous.

Maximum-type operations

- For any two disjoint products a_1 and a_2, the v-cofactors f_{a1} and g_{a2} are disjoint.
- Therefore functions f_{a1} and g_{a2} are in an incomplete tautology relation, and functions f and g are not changed when f_{a1} and g_{a2} are joined (OR-ed) to create a new function $f_{a1} + g_{a2}$, as in prior figure(where $a_1=a_2=a$)
- This way the entire lattice is created level-by-level.

Design Methodology

- One level of function is expanded to an assumed type of a decision tree(Shannon, SOP, etc).
- The level of the tree is mapped to the assumed type of Lattice.
- This means joining together some nodes of the treelike lower part of the lattice.
- The procedure requires repeating some variables in the lattice.