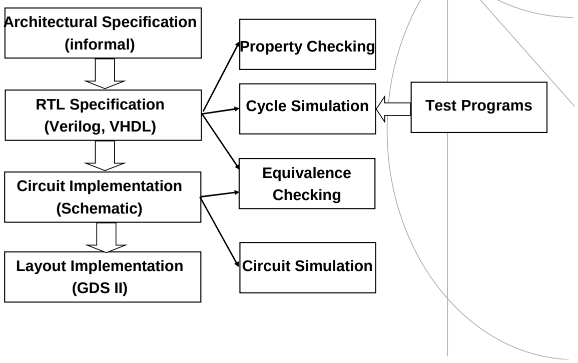
Symbolic Verification: Tassonomy

Gianpiero Cabodi ***Stefano Quer***

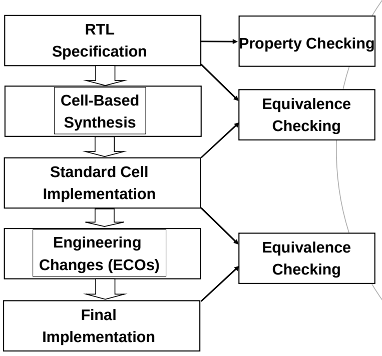
Politecnico di Torino
Torino, Italy

{gianpiero.cabodi,stefano.quer}@polito.it
http://staff.polito.it/{gianpiero.cabodi,stefano.quer}/

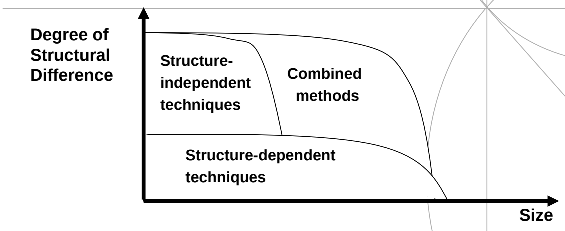
Application of Verification in μP Designs


Architectural Specification (informal)
RTL Specification (Verilog, VHDL)
Circuit Implementation (Schematic)
Layout Implementation (GDS II)
Property Checking
Cycle Simulation
Equivalence Checking
Circuit Simulation
Test Programs

Application of EC in ASIC Designs

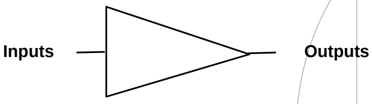


Methods



- ❖ **Structure-independent techniques**
  - ◆ **Exhaustive simulation**
  - ◆ **Decision diagrams (*DD*)**
- ❖ **Structure-dependent techniques**
  - ◆ **Graph hashing**
  - ◆ **SAT solvers including learning techniques**

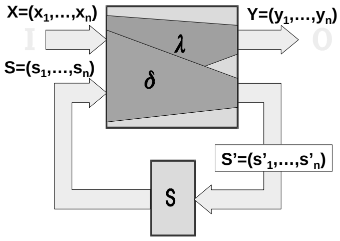
Combinational Circuits



- ❖ Basic methods:
  - ◆ Random simulation, good for finding miscompares
  - ◆ BDD based and modifications
  - ◆ Structural SAT based with modifications

Sequential Circuits

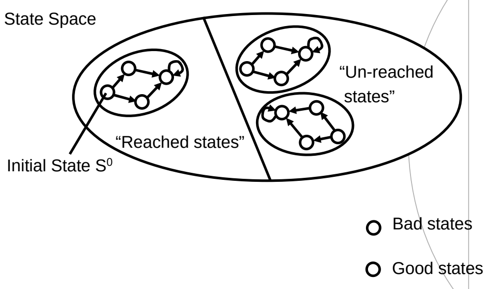
- ❖ **The Finite State Machine (FSM) Model**
  - ◆ ***M = (I, O, S, S_0, δ, λ)***



X: Inputs
Y: Outputs
S: Current State
S_0: Initial State(s)
δ: X × S → S
(next state function)
λ: X × S → Y
(output function)

- ❖ Deterministic machine
- ❖ Completely specified
- ❖ Delay element
  - ◆ Clocked: Synchronous
  - ◆ Single-phase clock vs Multiple-phase clocks
  - ◆ Unclocked: asynchronous
- ❖ Moore Machine
  - ◆ output = f (state)
- ❖ Mealy
  - ◆ output = f (state, input)

Problem: Reachable State Set



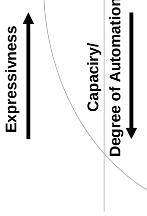
- ❖ Adapt combinationa verification to sequential circuits
- ❖ If combinational verification paradigm fails (e.g. we have no name matching) there are two options
  - ◆ Run full sequential verification based on state traversal
    - ✧Very expensive but most general
  - ◆ Try to match registers automatically
    - ✧Functional register correspondence
    - ✧Structural register correspondence
    - ✧Consider retiming
  - ◆ In essence, use all internal nets as candidates for possible matches
- ❖ Worst case: full sequential verification
  - ◆ Prove that the output of the product machine is not satisfiable (sequentially)
  - ◆ Special case of general property checking

How Do We Obtain R?

- ❖ Reachability analysis
  - ◆ State traversal until no more states can be explored
    - ✧Forward
    - ✧Backward
    - ✧Explicit
    - ✧Symbolic
- ❖ Relying on the design methodology to provide R
  - ◆ Equivalent state encoding in both machines
  - ◆ Synthesis tool provides hint for R from sequential optimization
  - ◆ Manual register correspondence
  - ◆ Automatic register correspondence
- ❖ Combination of them

Property Checking

- ❖ Assertion-based verification
  - ◆ Properties are expressed as RTL annotations in terms or assertions ("This statement must hold true")
  - ◆ E.g. AG(x=y) "For all paths from the initial state and all successor states x=y"
- ❖ Formal verification methods
  - ◆ Exhaustive, do not require simulation vectors
- ❖ Main methods
  - ◆ Theorem proving
  - ◆ Model Checking
    - ✧Liveness property checking
    - ✧Safety property checking
  - ◆ Refinement checking


Expressivness
Capaciry/
Degree of Automation