Hardware Model Checking: Status, Challenges and Opportunities

Pranav Ashar

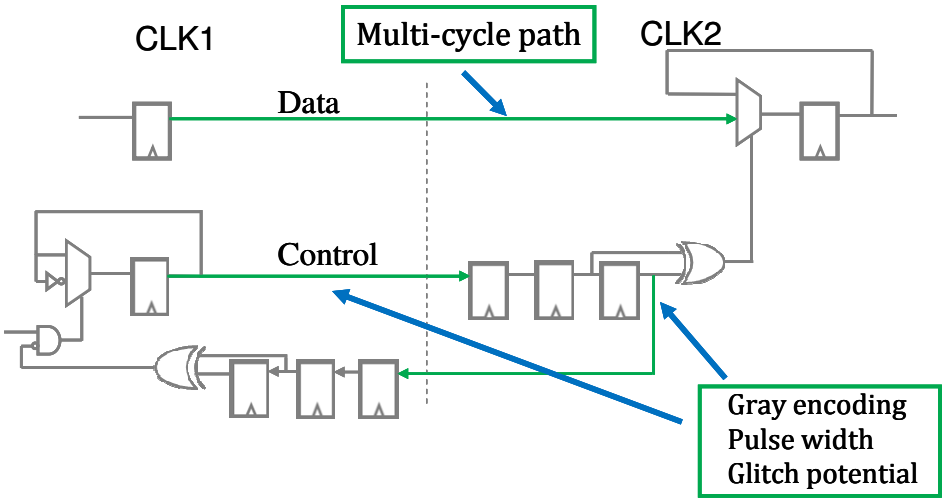
Real Intent Inc.

Model Checking is not ABV

- Model checking is a tool, not an application
  - Broader meaning than when the term was coined
- It's a bird, it's a plane
  - Constraint problem solver where the Kripke form gives structure, efficiency and closure criteria
  - Systematic simulation with closure criteria
- A tool to demystify the design and guide the design process
- Does more than prove assertions and generate traces
  - Why coverage target is not being hit
  - Why the proof was harder than expected
  - Find performance bugs and optimizations
  - ……

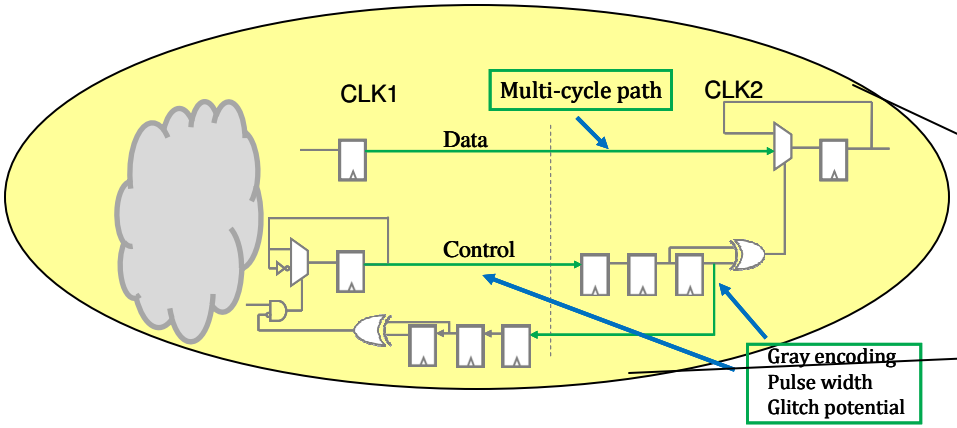
Much Specification is Implicit

- Common design idioms
- Simulation output
- The verification process
- Design constraints
  - set_false_path –from [get_ports IN1] –to [get_pins FF0/D]



Dead code
Uninitialized Memory
Constant RTL expressions
Constant nets
Constant state vector bits
Unreachable FSM states
Single FSM deadlock
Pairwise FSM deadlock
Bus contention
Floating buses
Full-case pragma violations
Parallel-case pragma violations
X value propagation
Array bounds violations

Much Analysis is Tractable



500 Million transistor SOC

- Analysis complexity does not necessarily grow with chip size

Making Model Checking Successful

- Use model checking to cut open the design rather than to beat the verification problem into submission
- ALWAYS return actionable information back to the user
  - If the result is bounded, show the user why it is bounded
  - Generate choices to make the next step easier for the user
- Bring all resources to bear to contain the formal analysis
- Solve the hard problems, you don't know what you will learn along the way
  - SAT/Model Checking is not fine-grain parallelizable

The Challenge

TILE64™ Processor Block Diagram