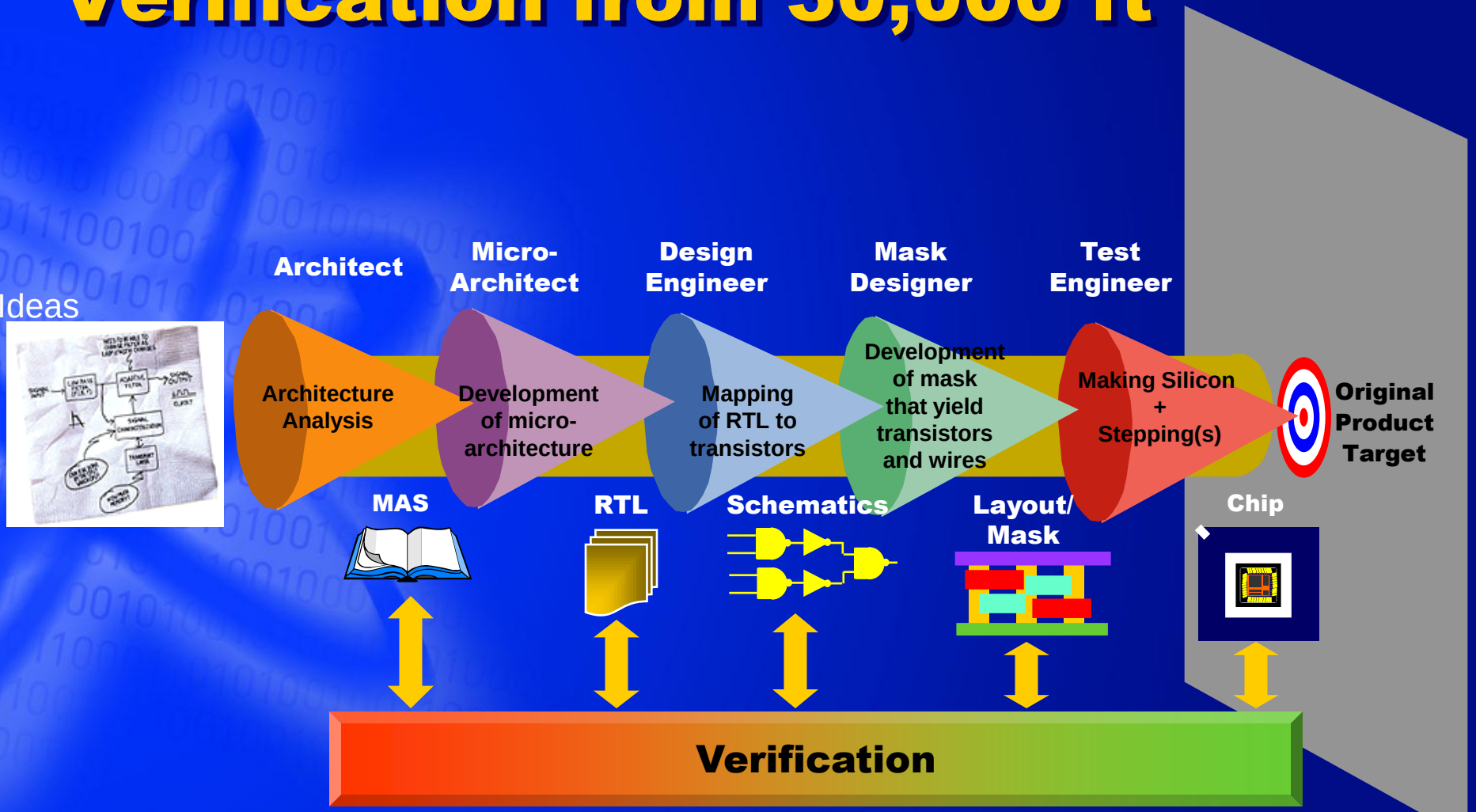
Top-to-Bottom Integrated Design and Verification

Dr. Carl Seger

Intel

January 15, 2013

Verification from 30,000 ft



- How to: 1) check we captured what we wanted
  2) check that we did not make a mistake along the way
  3) check that it does nothing else

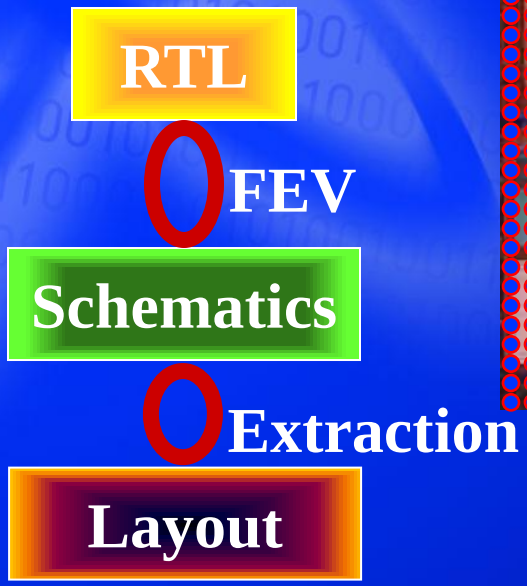
Approaches to Functional Verification

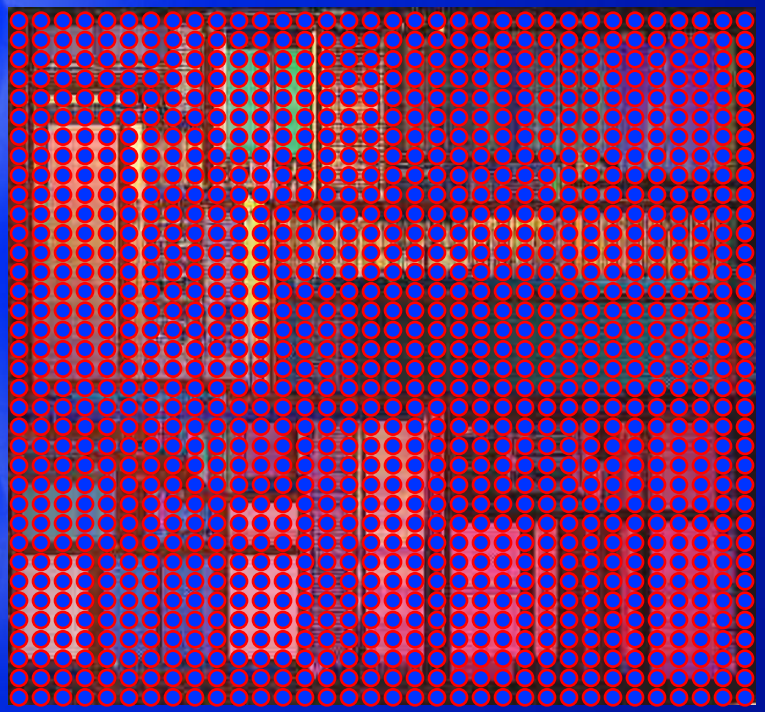
- Two broad categories:
  - Dynamic approaches
    - Simulation
    - Emulation
    - Prototyping
  - Static approaches
    - Code review
    - “Lint” like tools
    - Formal Equivalence Verification
    - Formal Property Verification

Practical and Deployed Formal Verification

Formal Equivalence Verification

- Use of symbolic/algebraic methods to completely verify that a circuit implements a specification





Today: 100% of the design is FEV clean before tape-out

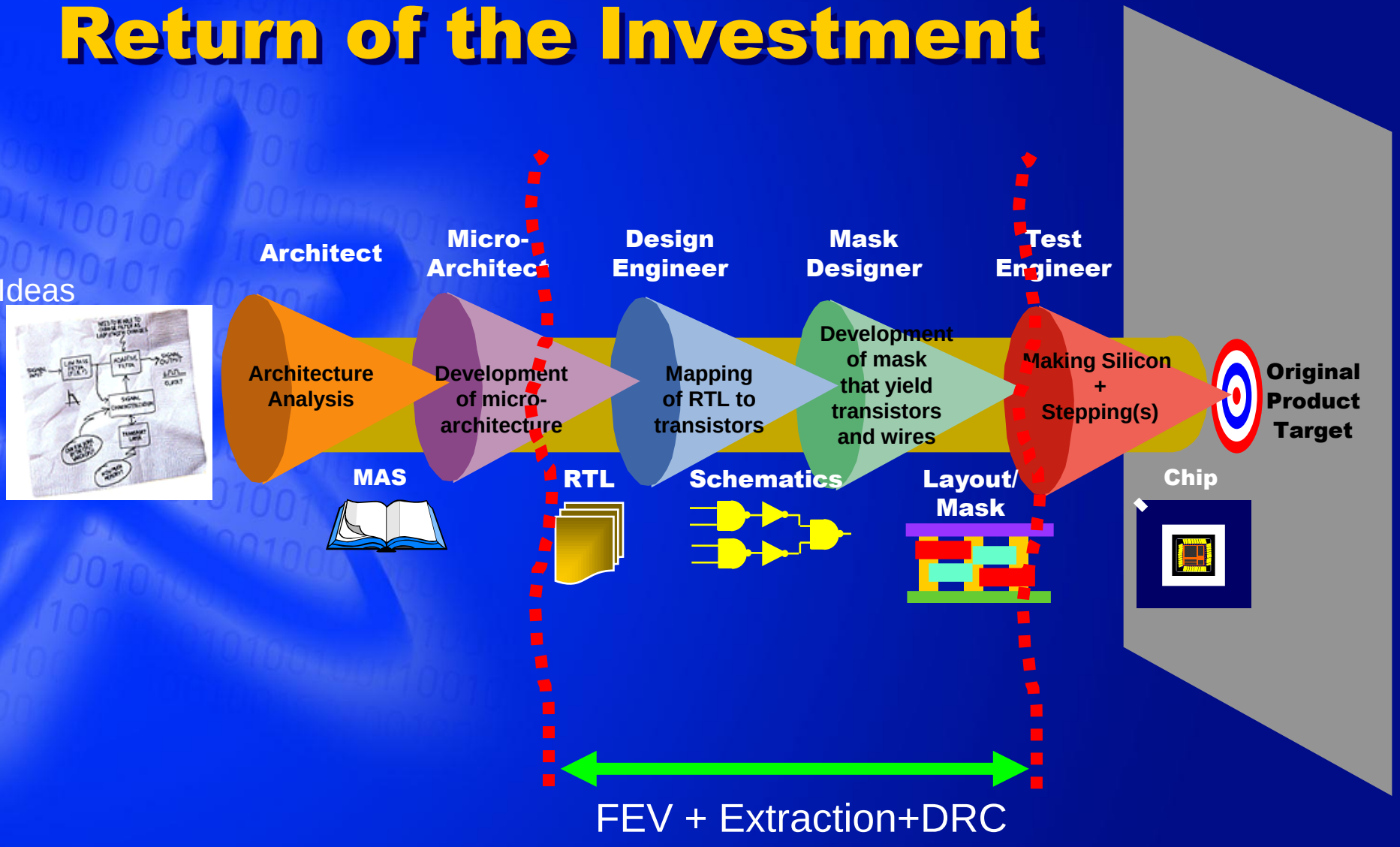
Extremely successful application of math, logic and computer science in practical engineering!

Usability high enough that every design engineer is able to run the verification.

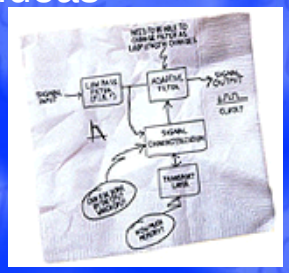
Formal Property Verification of Computation Units

- Symbolic Trajectory Evaluation (STE), a form of symbolic simulation, are today used to formally verify computation units/blocks
  - Complete formal property verification of all (>3,000) uops in the execution cluster is now routinely done
    - Includes all control, clock gating logic, test features etc. as well as the actual data-path computations
    - FPV is primary pre-Si verification for this unit
- Combining STE with theorem proving increases the quality of specification
  - Floating point spec is mathematical statement of IEEE standard

Solid Formal Link with Good Return of the Investment



Mind the Gap(s)...



Ideas

Architect

Micro-Architect

Design Engineer

Mask Designer

Test Engineer

Architecture Analysis

Development of micro-architecture

Mapping of RTL to transistors

Development of mask that yield transistors and wires

Making Silicon + Stepping(s)

Original Product Target

MAS

RTL

Schematics

Layout/ Mask

Chip

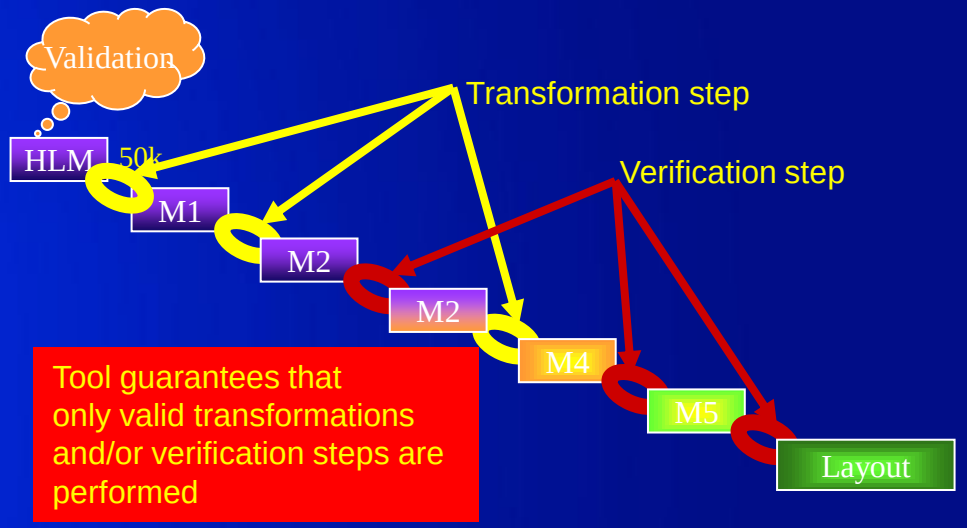
?

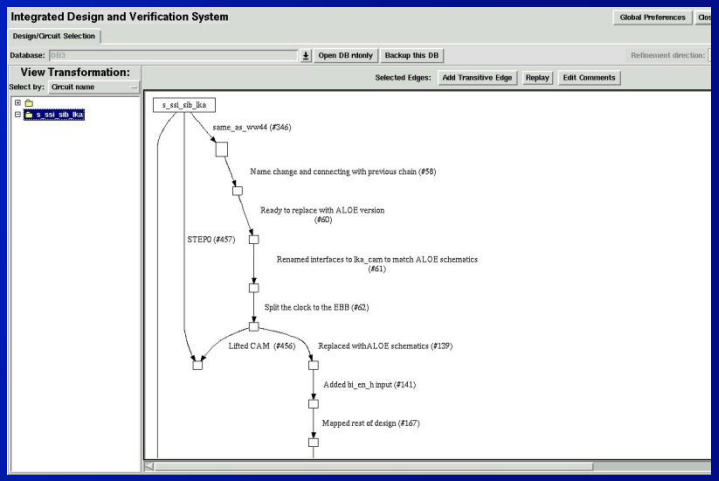
?

Potential New Uses of FV to Increase Confidence in Designs

Integrate Verification and Design

- Rather than trying to do post-design verification, verify each step along the way.
  - Can mix “correct-by-construction” and “trust-but-verify” parts.
  - Can use different verification engines at different levels of abstraction
  - Imposes a relatively modest overhead on the design process for a big payoff.
  - A system can be built to track the “quality” of a design from correctness point of view.





IDV prototype system for abstract RTL to layout with complete verification