The 6th COE Technical Presentation

Acceleration of Test Generation for Sequential Circuits Using Knowledge Obtained from Synthesis for Testability

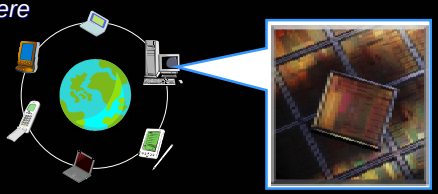
Masato NAKAZATO
Computer Design and Test Lab.

Outline

- Background
- Our proposed methods
  - Test generation method for sequential circuits
  - Synthesis for testability method
- Experimental results
- Conclusions

Importance of VLSI testing

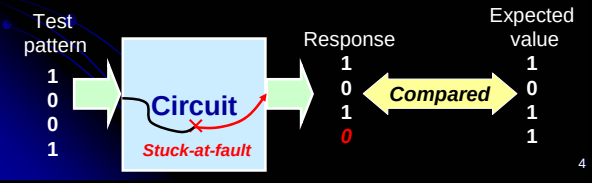
- ***Ubiquitous computing***
  - Various tasks are performed *anytime* and *anywhere*



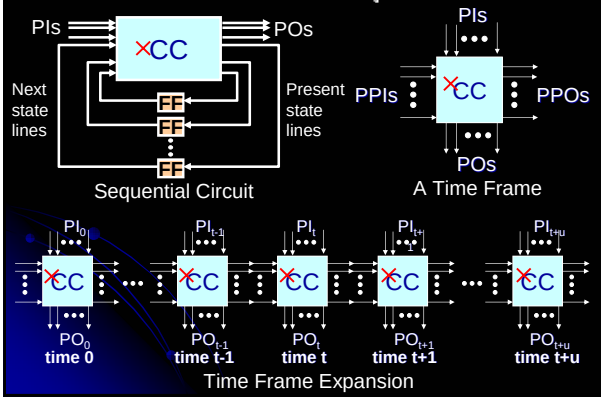
- ***Testing of VLSI circuits***
  - Essential technology to realize ***dependable ubiquitous systems***

VLSI Testing

- VLSI Testing
  - Check whether faults exist or not
- Fault efficiency
  - A rate scale which presents the quality of test patterns

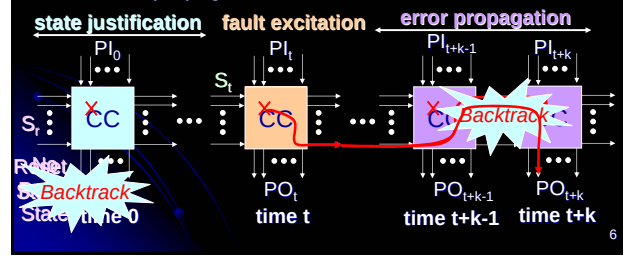


Time Frame Expansion



Sequential Test Generation

- A sequential TG consists of three phases
  - Fault excitation
  - State justification
  - Error propagation



Related Works

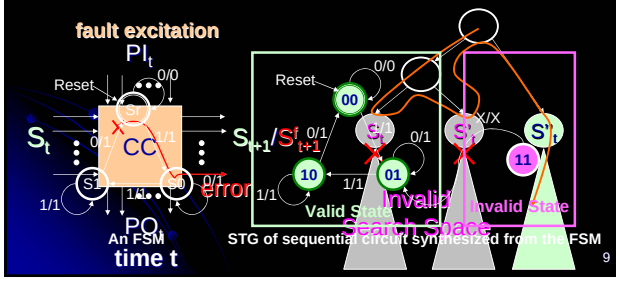
- Previous works of sequential TG
  - Gate level
    - HITEC [Patel '91], TestGen [Synopsys], FASTEST [Kelsey '89], etc.
  - Utilizing knowledge extracted from FSMs
    - VERITAS [Cho '93]
      - State justification was accelerated
      - Error propagation was ***not*** accelerated

Our Proposed Methods

- Synthesis for Testability (SfT) method to synthesize a sequential circuit with three specific characteristics from an FSM
- TG method for the sequential circuit to utilize higher level knowledge of its features
  - Acceleration of fault excitation
  - Acceleration of state justification
  - Acceleration of error propagation

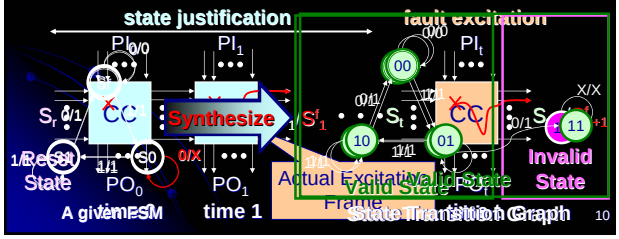
I. Acceleration of fault excitation

- Searching for directly valid search space by using *information of the valid states*
- Pruning the search space of a test generation



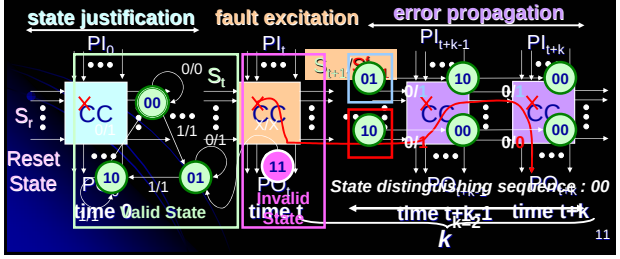
II. Acceleration of state justification

- Justifying the excitation state by using *state justification sequences* extracted from a given FSM
- If an invalidation occurs, we try to propagate errors from the actual excitation frame



III. Acceleration of error propagation

- For fault-free circuit, a *state distinguishing sequence* for the pair of states *of length* $\leq k$ is guaranteed
- For faulty circuit, if the sequence is *valid* for the pair of states (S_{t+1}/S^f_{t+1}), fast error propagation will be done



Experiment Environment

- We applied three methods to MCNC benchmark
  - TestGen + Original circuit
  - TestGen + SfT circuit
  - Proposed TG + SfT circuit
    - Objective for comparison :
      Test Generation Time (TGT)
      Fault Efficiency (FE)

Method	equivalent fault analysis	fault simulation	Test generation algorithm for combinational circuits
TestGen	Yes	Yes	Heuristic
Proposed TG			Non-heuristic

Experimental Results (TGT and FE)

Circuits	TGT (sec)			FE (%)		
	TestGen + Original circuit	TestGen + SfT circuit	Proposed + SfT circuit	TestGen + Original circuit	TestGen + SfT circuit	Proposed + SfT circuit
bbara	10h	249.84	0.27	98.89	100.00	100.00
ex1	391.80	23841.81	1.20	100.00	100.00	100.00
keyb	1333.31	10h	1.54	100.00	100.00	100.00
planet	69.66	125.62	3.12	100.00	100.00	100.00
pma	458.09	76.81	23.14	100.00	100.00	100.00
s1	29918.96	7.76	1.85	100.00	100.00	100.00
s298	10h	10h	58.50	97.64	99.98	100.00
tma	682.22	223.86	1.42	100.00	100.00	100.00
tbk	10h	5.53	4.03	99.17	100.00	100.00

Conclusions

- We have proposed an efficient test generation method with an SfT method
  - Acceleration of each process by utilizing knowledge extracted from FSM during the SfT
    - Information of the valid states
    - State justification sequences
    - Length of state distinguishing sequences
- Our proposed method can achieve 100% fault efficiency in shorter test generation time
- This research was presented by the international conference WRTLT'05

Future Work

- Study on Design for Testability of Instruction-Based Self-Test for pipelined microprocessors
  - It is difficult for Instruction-Based Self-Test to achive 100% fault efficiency
  - We will be studying the design for testability method for instruction-based self-test which can achieve 100% fault efficiency