# High-Level and Hierarchical Test Sequence Generation

**Gert Jervan, Zebo Peng**
**Embedded Systems Laboratory**
**Linköping University, Sweden**

**Olga Goloubeva, Matteo Sonza Reorda, Massimo Violante**
**Dipartimento di Automatica e Informatica**
**Politecnico di Torino, Italy**

This work has been supported by the EC project IST-2000-29212 COTEST

## Overview

- Analysis of available high-level fault models in order to select the most suitable one for estimating the testability by reasoning only on circuits behavioral descriptions.
- Assessment of the effectiveness of high-level test generation for manufacturing test based on the adopted high-level fault model.
- A novel high-level hierarchical test generation approach for improving the high-level test generation effectiveness by exploiting structural information.

## High-Level Fault Models

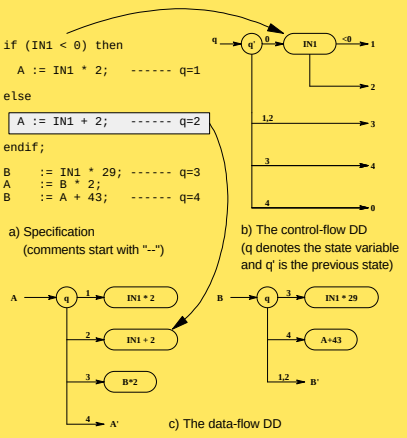
- We analyze statement coverage, bit coverage and condition coverage in terms of the correlation they provide between high-level fault coverage and gate-level stuck at coverage.
- We fault simulate the *same* input sequence with two *different* models of the same circuit (high-level and gate-level ones) and compare the attained gate-level and high-level fault coverage figures.
- The proposed high-level fault models can be fruitfully exploited to estimate the quality of different test sets and to predict the gate-level fault coverage before synthesis.

## High-Level Test Generation

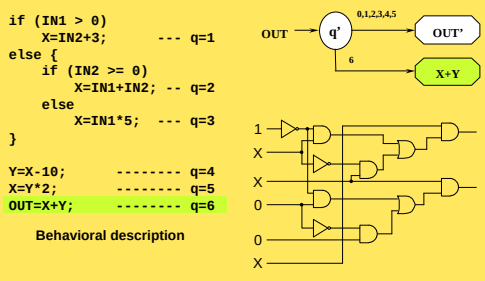
- Based on purely behavioral models: no details about the circuit structure are used.
- Exploits a Simulated Annealing algorithm
  - solution ⇒ a sequence of vectors;
  - evaluation function ⇒ Bit Coverage + Condition Coverage + Statement Coverage;
  - neighborhood exploration ⇒ three operators: add one vector, delete one vector, complement one bit in one vector;
- Experiments gathered on a prototype:
  - SystemC descriptions
  - 1,000 lines of C code

## Hierarchical Test Generation

- Improvement of pure high-level ATPG by using a hierarchical fault model targeting errors in the system behavior and in its final implementation.
- Two types of tests: Conformity test and functional unit test



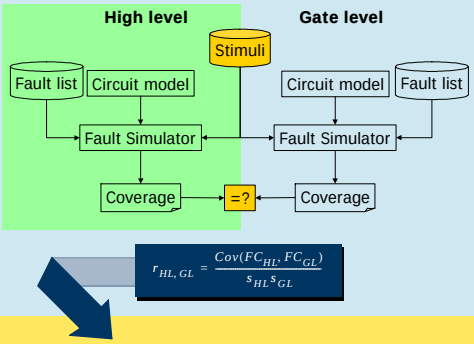

**A Decision Diagram example**




**Hierarchical Design Representation**

- The fault coverage attained by the hierarchical ATPG is higher than that of the pure high-level ATPG
- The generated test sequences can be efficiently used for testing stuck-at faults.

## Fault Model Evaluation




## Fault Models Comparison

| Benchmark | Statement Coverage | Bit Coverage | Condition Coverage | Bit + Condition |
|---|---|---|---|---|
| BIQUAD 1 | 0.63 | 0.97 | 0.60 | 0.97 |
| BIQUAD 2 | 0.64 | 0.97 | 0.61 | 0.98 |
| DIFFEQ 1 | 0.62 | 0.97 | 0.65 | 0.98 |
| DIFFEQ 1 | 0.63 | 0.97 | 0.64 | 0.98 |
| TLC | 0.83 | 0.45 | 0.72 | 0.80 |

## Comparison of Test Generators

| High-level ATPG | Gate-level FC [%] | Test Len [#] | CPU [s] |
|---|---|---|---|
| DIFFEQ 1 | 97.25 | 553 | 954 |
| DIFFEQ 2 | 94.57 | 553 | 954 |
| High-level Hierarchical ATPG | | | |
| DIFFEQ 1 | 98.05 | 199 | 468 |
| DIFFEQ 2 | 96.46 | 199 | 468 |
| Gate-level ATPG (`testgen`) | | | |
| DIFFEQ 1 | 99.62 | 1,177 | 4,792 |
| DIFFEQ 2 | 96.75 | 923 | 4,475 |

## Hierarchical Test Generation Algorithm