Random Access Scan

Virendra Singh
Associate Professor
Computer Architecture and Dependable Systems Lab
Dept. of Electrical Engineering
Indian Institute of Technology Bombay
viren@ee.iitb.ac.in

EE 709: Testing & Verification of VLSI Circuits

Lecture – 23 (Feb 27, 2012)

Should Serial Scan Continue?

- Three Problems with serial-scan
  - Test power
  - Test application time
  - Test data volume
- Efforts and limitations
  - ATPG for low test power consumption
    → Test power ↓ Test length ↑
  - Reducing scan clock frequency
    → Test power ↓ Test application time ↑
  - Scan-chain re-ordering (with additional logic insertion)
    → Test power/time ↓ Design time ↑
  - Test Compression
    →Test time/data size ↓ Has limited capability for Compacted test
- Orthogonal attack
  - Random access scan instead of Serial-scan
  - Hardware overhead? Silicon cost << Testing cost

Random Access Scan

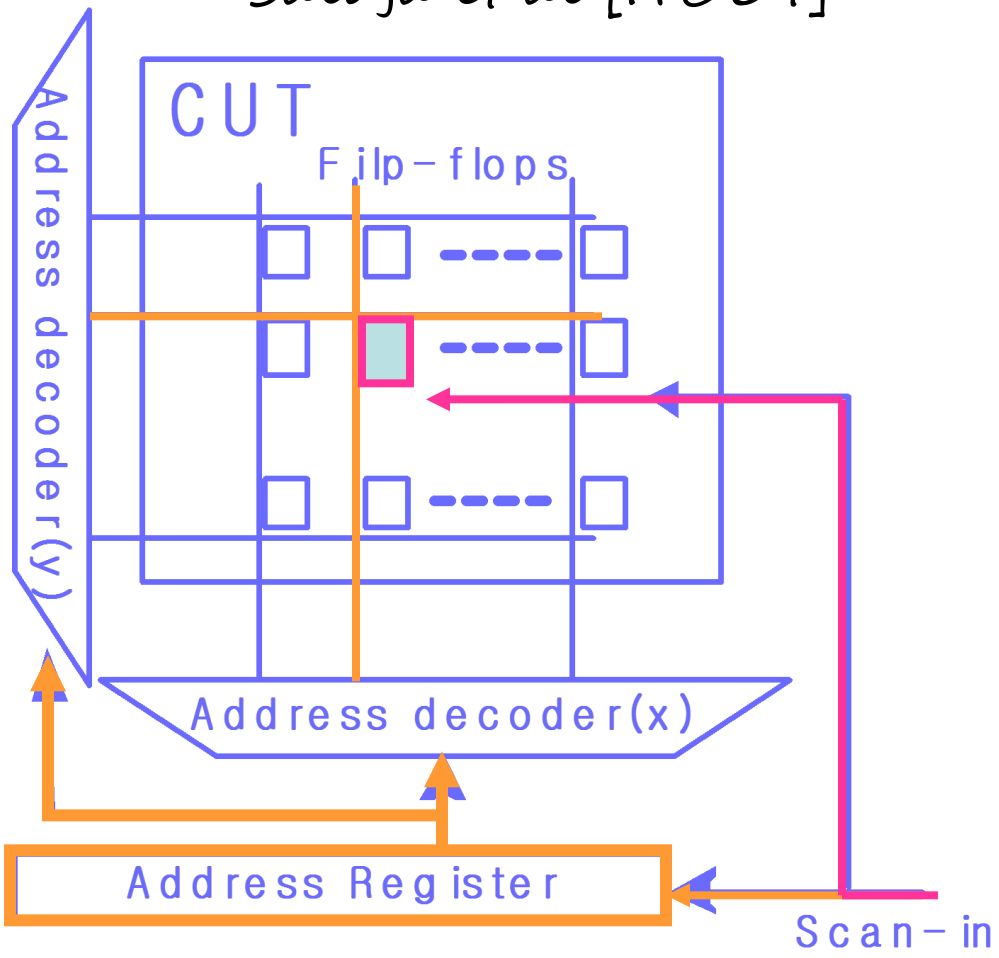
- First proposed by Ando in 1980
- It was considered impractical due to large area overhead.
- Baik at al. revisited it in 2004 [ITC'04]
- Proposed as a simultaneous solution to test power, test time, and test data volume
- Baik,2005, proposed PRAS which showed around 3x speed up, reduction in test data volume with only minor increase in area compared to Serial Scan, thus making RAS practical

Random Access Scan: Architecture

A solution to test power, test time and test data volume

- Architecture
- Each FF has unique address
- Address shift register
- X-Y Decoder
- Select FF to write/read

Saluja et al [ITC'04]

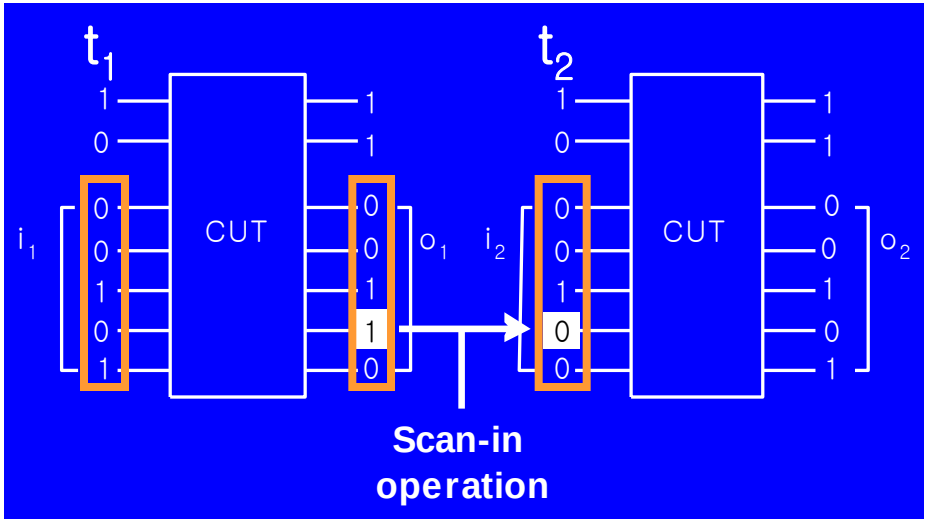


Scan Operation Example

- Test vector

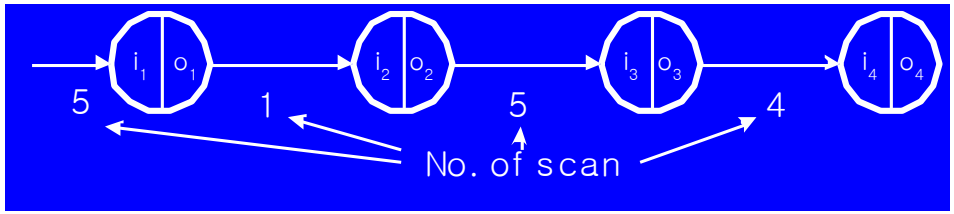
Test	PPI(i_i)	PPO(o_i)
t1	00101	00110
t2	00100	00101
t3	11010	11010
t4	00111	01011

- Scan operation for *t2*



- Complete test application

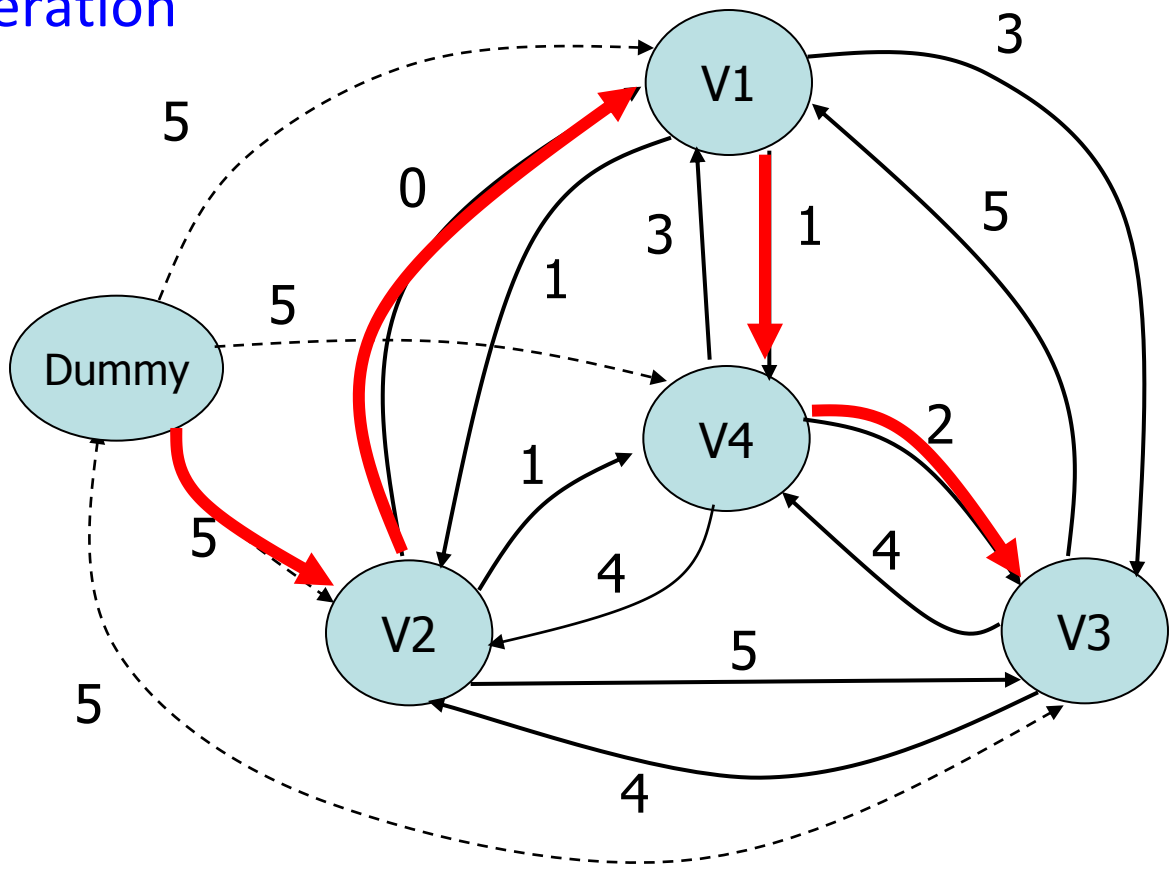
  Total number of scan operation = 15



Test Vector Ordering

- Test data volume and Test application time is proportional to the random access scan operation
- Goal: Reduce # scan operation

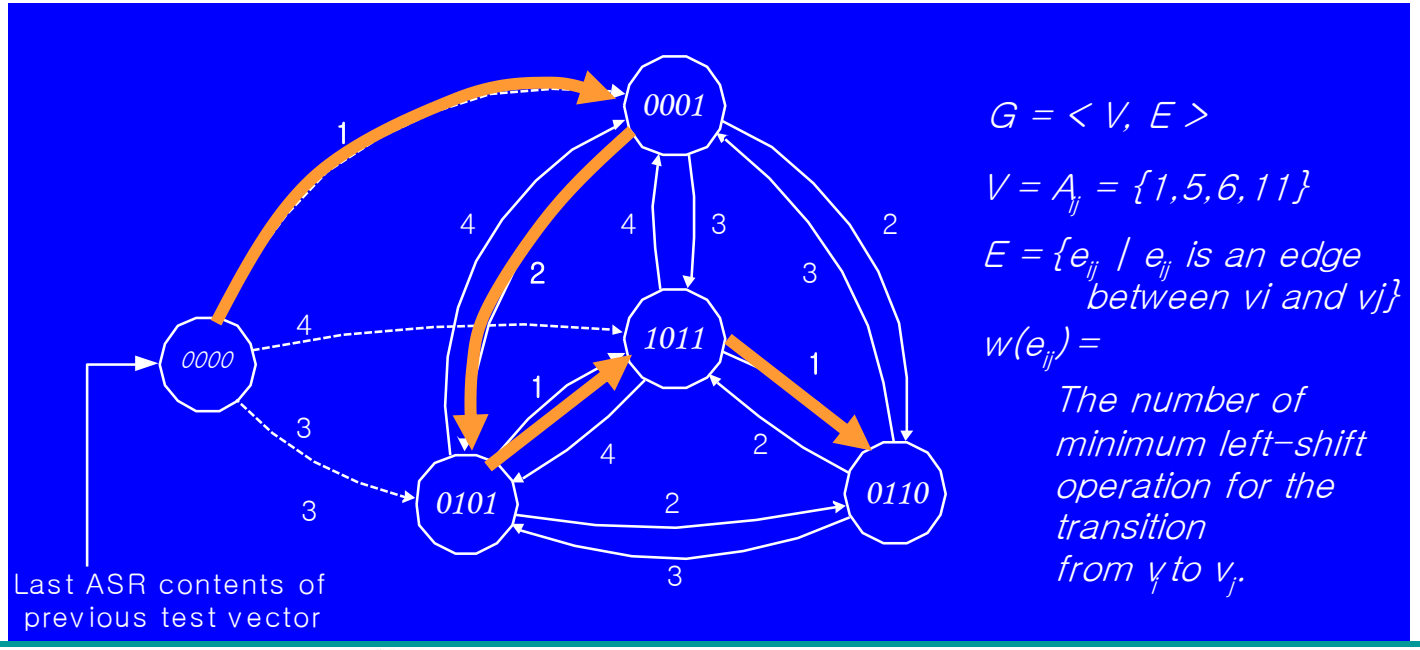
Test	PPI(i_i)	PPO(o_i)
t1	00101	00110
t2	00100	00101
t3	11010	11010
t4	00111	01011



Scan operation = 8

Optimizing Address Scan

- The cost of address shifting
  - # of scan operation x ASR width
  - Example address set = { 1, 5, 6, 11 } for 4-bit ASR
  - 4 X 4 = 16
- Proper ordering of address can minimize shifting cost
  - Apply 11(1011) after 5(0101) → needs only 1 left-shift
- Minimizing address shifting cost
  - Construct Address Shifting Distance Graph (ASD-graph)
  - Find min-cost Hamiltonian path using ATSP algorithm (Result : 5 shifts)



Thank You