Computer and VLSI design

Structured circuit testing

P.J. Drongowski
SandSoftwareSound.net

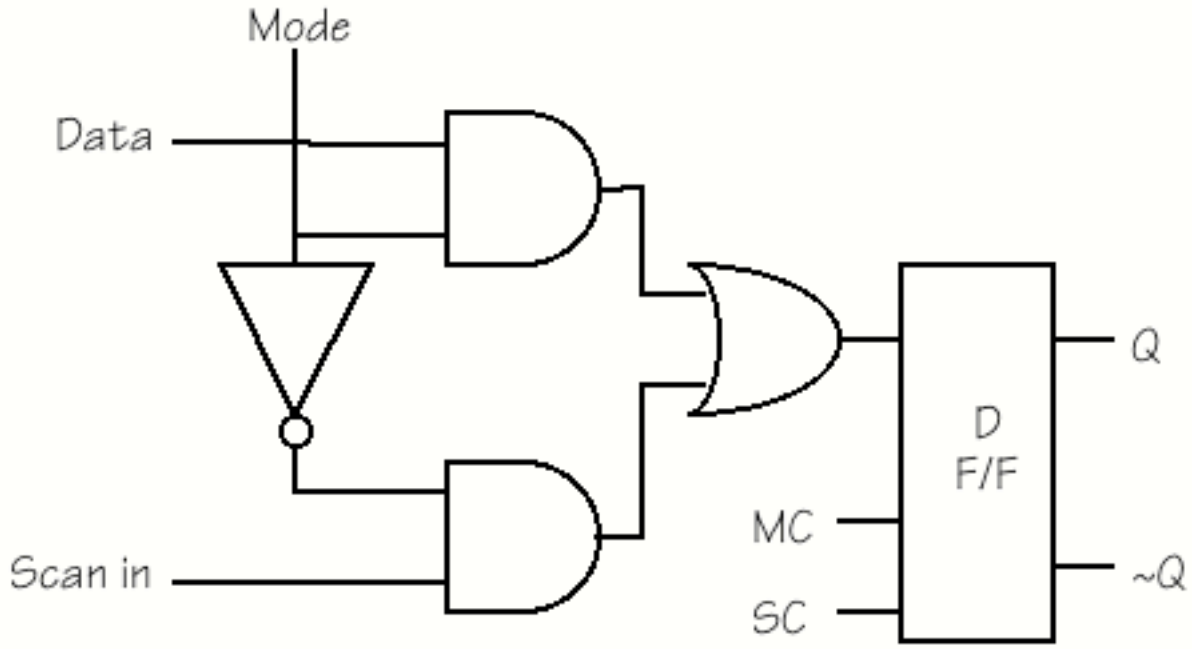
Functional testing

- Design for testability (DFT)
  - Increase fault coverage
  - Improved controllability and observability
  - Extra hardware to load / read-back test data
  - Pattern generation
    - Use internal system structure
    - Consistent with on-chip test hardware
  - Automatic test equipment
    - Access to primary test inputs / outputs
    - Compatible with DFT scheme
- Ad hoc approach
  - Set-up and observe registers via busses
  - Each register has a different read / write procedure
  - Partition logic into testable subdivisions
  - Uses manually generated test patterns (typical)
  - Controllability and observability may be limited
- Structured approach
  - Separate system into sequential and comb'l logic
  - Control / observe memory in a race-free manner
  - Test procedure reduces to combinational logic test
  - Patterns more easily generated automatically
  - Employ some form of scan design
- Built-in test (BIT)
  - Generate test patterns on-chip
  - Compress test results
  - Possibly report go or no-go

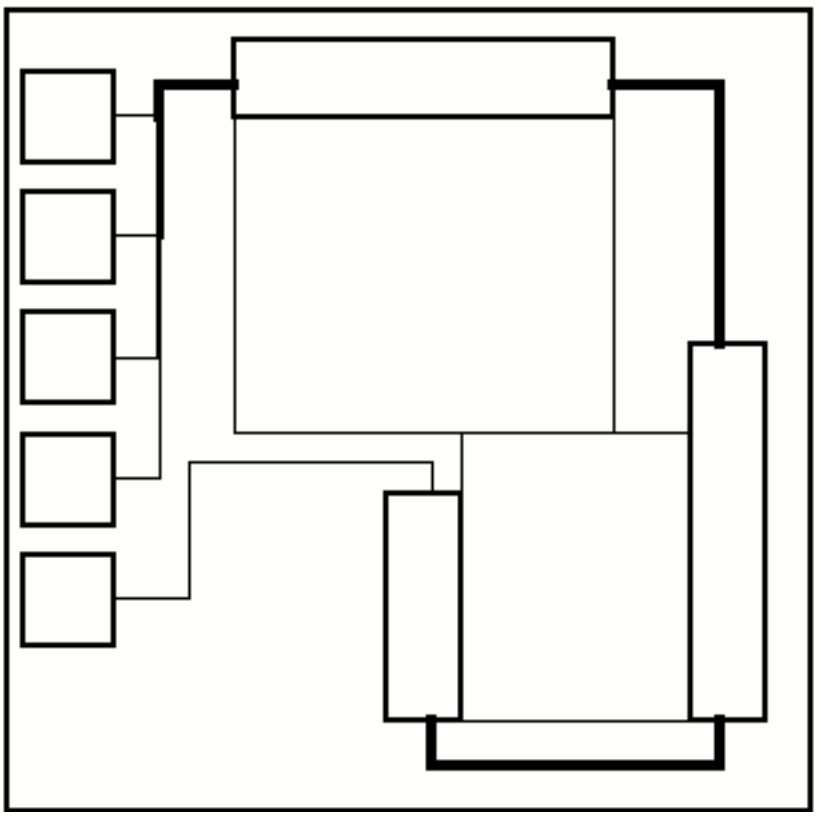
Scan design

- Scan-in, scan-out (SISO)
- Partition design into blocks
- Reconfigurable sequential elements
  - Normal operating mode: parallel read / write
  - Test mode: long shift chain to load / read vectors
- Test procedure
  - Put system into test mode
  - Scan (shift) in test pattern
  - Return system to operational mode
  - Perform one clock step
  - Put system into test mode
  - Scan (shift) out test results
  - Compare results, iterate
- Advantages
  - Good controllability and observability
  - Higher coverage
  - Pattern generation for combinational logic only
  - Reduced number of patterns WRT ad hoc approach
  - Fewer patterns just to load and read test data
- Disadvantages
  - Long time to shift in patterns and shift out results
  - Hint: shift in while shifting out
  - Additional signals (test clock, mode, data) to route
  - Requires additional space
    - Larger latches (i.e., input MUX, dual clock)
    - Routing area
  - Input MUX may increase operational delay

Basic scan design latch



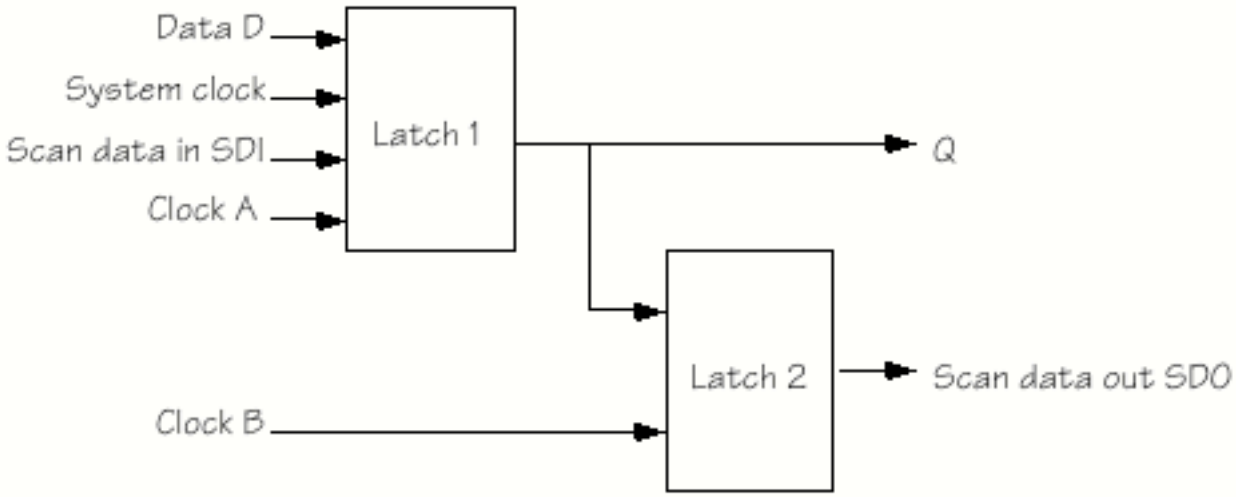
Scan design layout

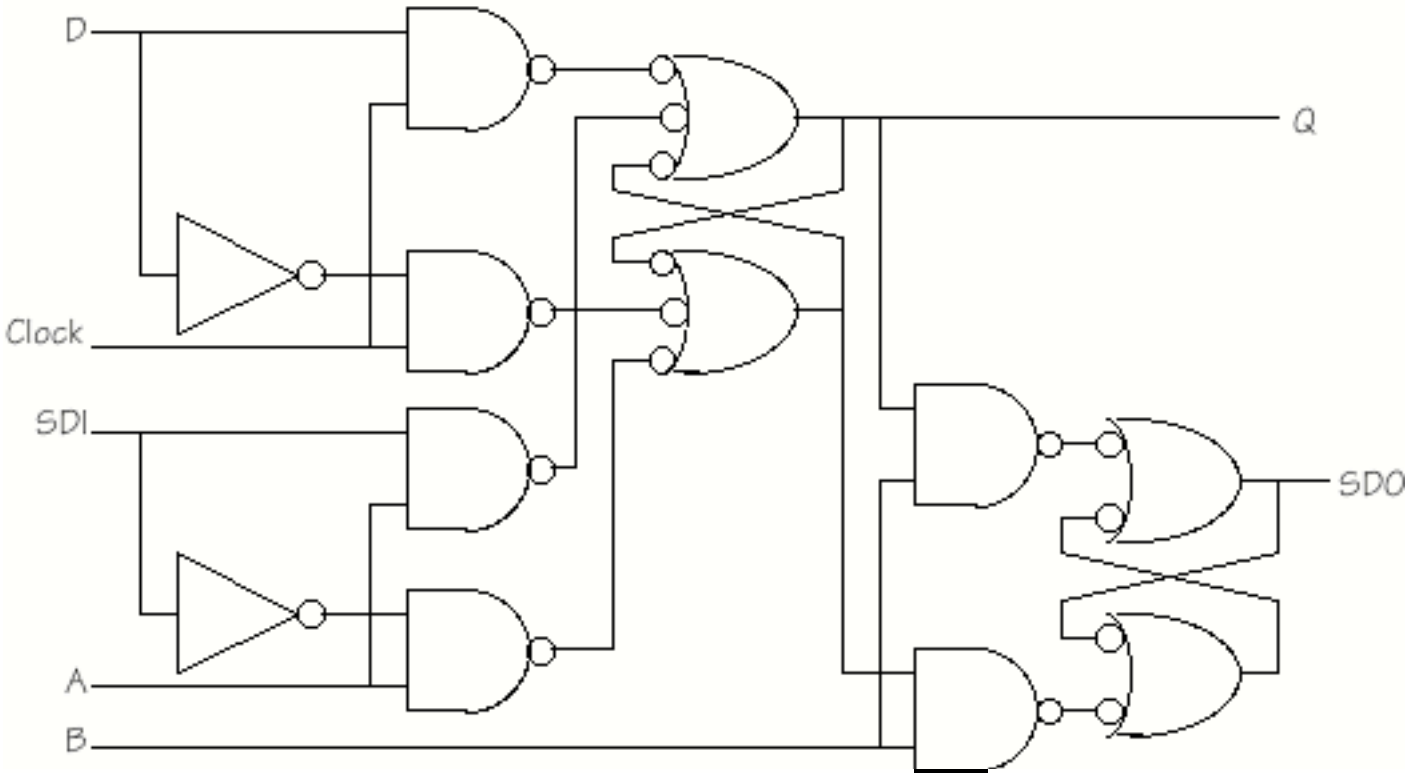


Level-sensitive scan design

- Level-sensitive scan design (LSSD)
- IBM corporation
- Basic features
  - Storage elements are latches (not flip / flops)
  - Latch responds whenever control input is active
  - Level sensitivity eases timing constraints
  - Input can settle any time during clock pulse
  - Latch is insensitive to clock edge degradation
  - Connected chain of shift register latches (SRL)
- SRL operation
  - Normal operation
    - Shift clocks A and B are inactive
    - Active system clock transfers input D to Q
    - Value is held by latch L1
  - Test mode
    - System clock is inactive
    - Input D is disconnected from latch L1
    - Phases A and B are non-overlapping shift clock
    - Assert B first transferring Q to input of latch L2
    - L2 holds this data and drives SDO
    - Phases A and B are non-overlapping shift clock
    - Assert B first transferring Q to input of latch L2
    - Assert A next transferring SDI into latch L1
    - Next B pulse will transfer new Q to SDO
- Notes
  - Each storage bit uses two latches
  - Clever design can use L2 during normal operation
  - 4% to 20% complexity increase in practice

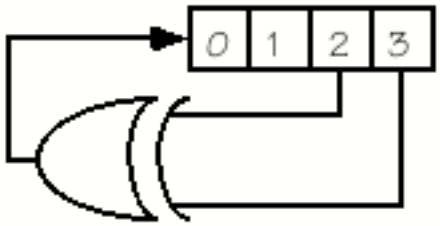
Level-sensitive scan design

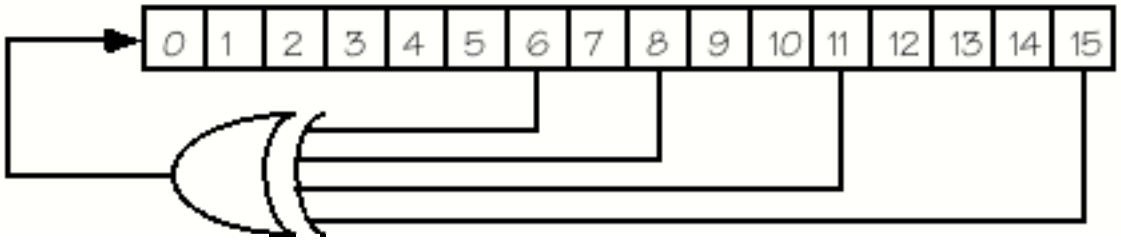




Built-in pattern generation

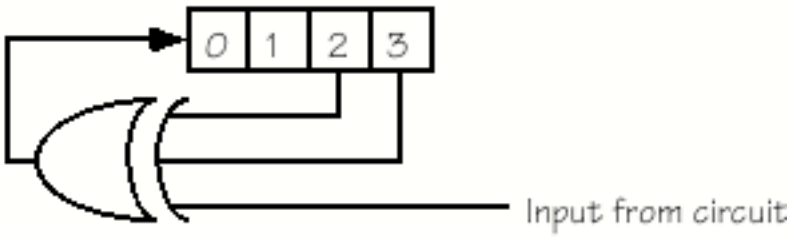
- SISO gives good coverage, controllability, observability
- Shift in, shift out of data is time consuming
- Why not generate test patterns internally?
  - Systems can be self-testing
  - Stored memory patterns are not practical
  - Exhaustive testing (via a counter) takes too long
  - Try (pseudo) random patterns
- Pseudo-random number (PRN) generation
  - Linear feedback shift register (LFSR)
  - Feed selected stages into input through XOR gate
  - Tap stages to obtain maximal sequence of values
  - Sequence composition depends on choice of taps
  - LFSR cannot become zero - it remains at zero!
- Problems
  - Sequence length to obtain good coverage
  - Random sequence may "hang" sequential logic

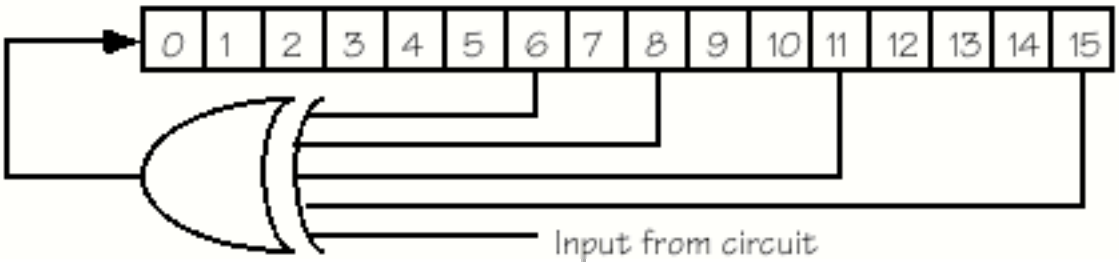




Signature analysis

- Technique devised by Hewlett-Packard (~1975)
- Use LFSR to compact test output results
- Similar to cyclic redundancy checking (CRC)
- Test procedure
  - Initialize LFSR to known value
  - Perform operational step
  - Apply test output data to input of LFSR
  - Clock LFSR to new state, iterate
- Final value of LFSR depends upon:
  - Initial value
  - Sequence of output values
  - Number of test steps / LFSR configuration
- Compute expected "signature" using KGD
- Change in any output value yields different signature
- Possible that a fault will still yield correct signature
- "Aliasing" can be eliminated by lengthening LFSR





BILBO

- Built-in logic block observer (BILBO
- Standard LFSR-based test structure
- Multiple modes
  - Parallel load register
  - Pattern generator
  - Signature generator