Testing and Boundary Scan

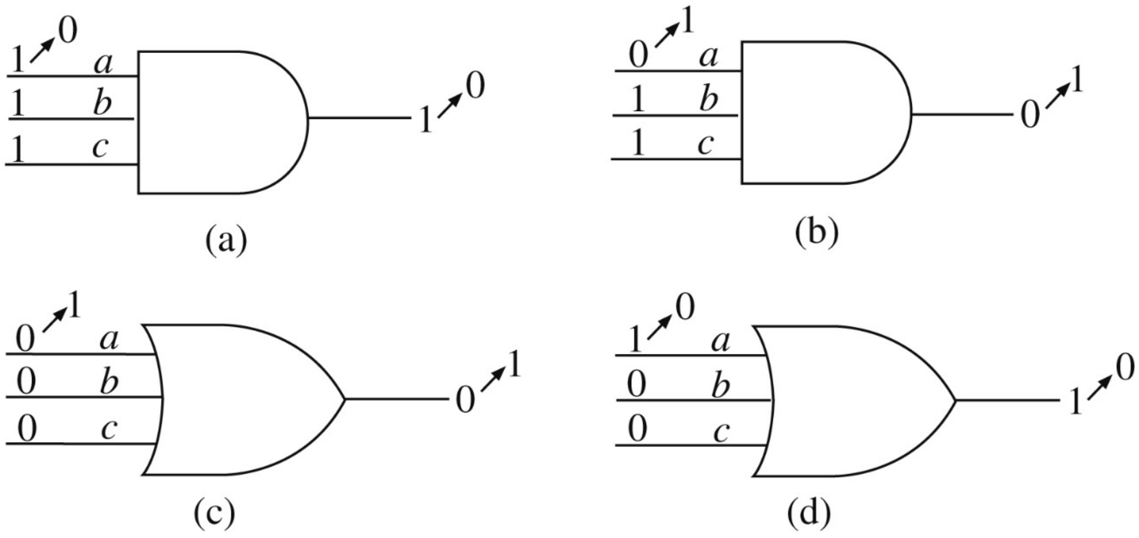
Roth text: Chapter 10.1 – 10.4

Digital circuit fault models

- Stuck-at fault: gate input/output appears to be always = logic 1 (stuck-at-1) or 0 (stuck at 0)
- Bridging fault: signal on one wire affects state of a second wire
- Delay fault: logic states correct, but switching time longer than expected
- Open circuit: open circuit on a wire
- Short circuit

Testing for stuck-at faults

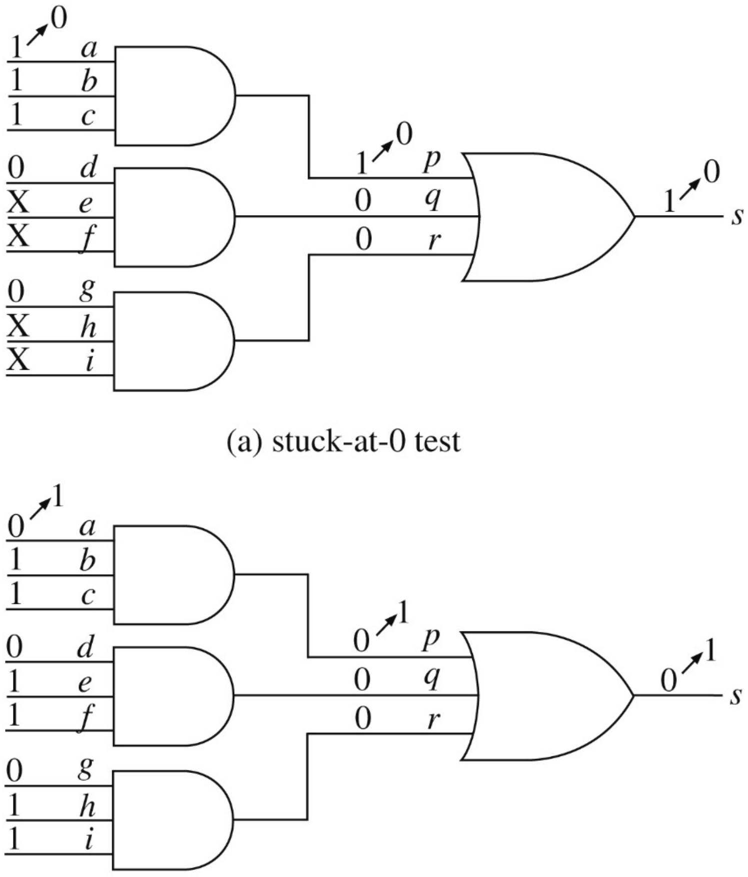
FIGURE 10-1: Testing AND and OR Gates for Stuck-At Faults



"Test" – apply an input pattern (vector) that produces different values in fault-free and faulty circuits.

a) Test a s-a-0 with vector 111 (expected output 1, erroneous output 0)
b) Test a s-a-1 with vector 011 (expected output 0, erroneous output 1)
c) Test a s-a-1 with vector 000 (expected output 0, erroneous output 1)
d) Test a s-a-0 with vector 100 (expected output 1, erroneous output 0)

FIGURE 10-2: Testing an AND-OR Circuit



(a) stuck-at-0 test

(b) stuck-at-1 test

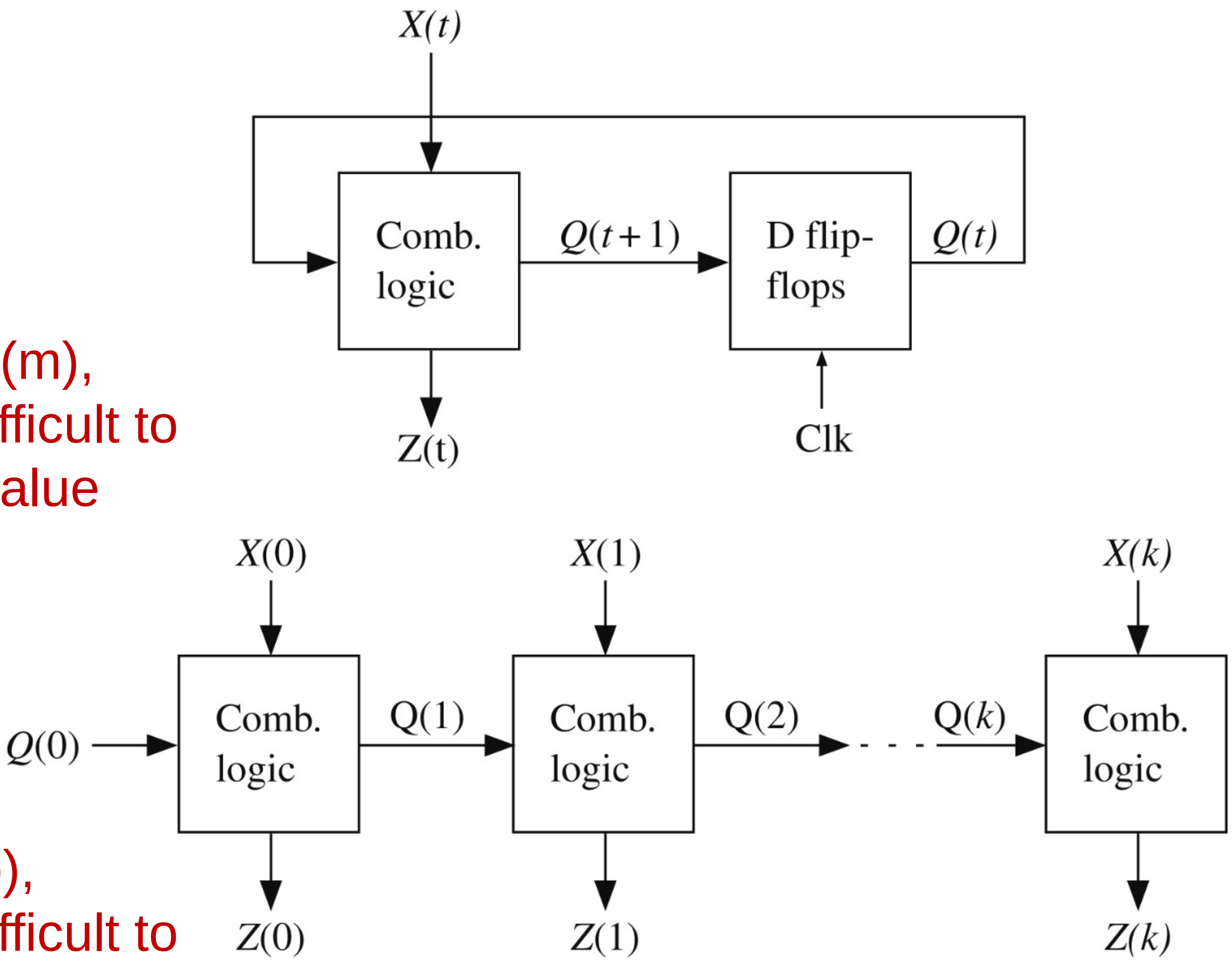
Test set to detect all stuck-at faults.

TABLE 10-1: Test Vectors for Figure 10-2

a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	Faults Tested
1	1	1	0	X	X	0	X	X	*a*0, *b*0, *c*0, *p*0
0	X	X	1	1	1	0	X	X	*d*0, *e*0, *f*0, *q*0
0	X	X	0	X	X	1	1	1	*g*0, *h*0, *i*0, *r*0
0	1	1	0	1	1	0	1	1	*a*1, *d*1, *g*1, *p*1, *q*1, *r*1
1	0	1	1	0	1	1	0	1	*b*1, *e*1, *h*1, *p*1, *q*1, *r*1
1	1	0	1	1	0	1	1	0	*c*1, *f*1, *i*1, *p*1, *q*1, *r*1

Sequential circuit testing

FIGURE 10-5:
Sequential and Iterative Circuits

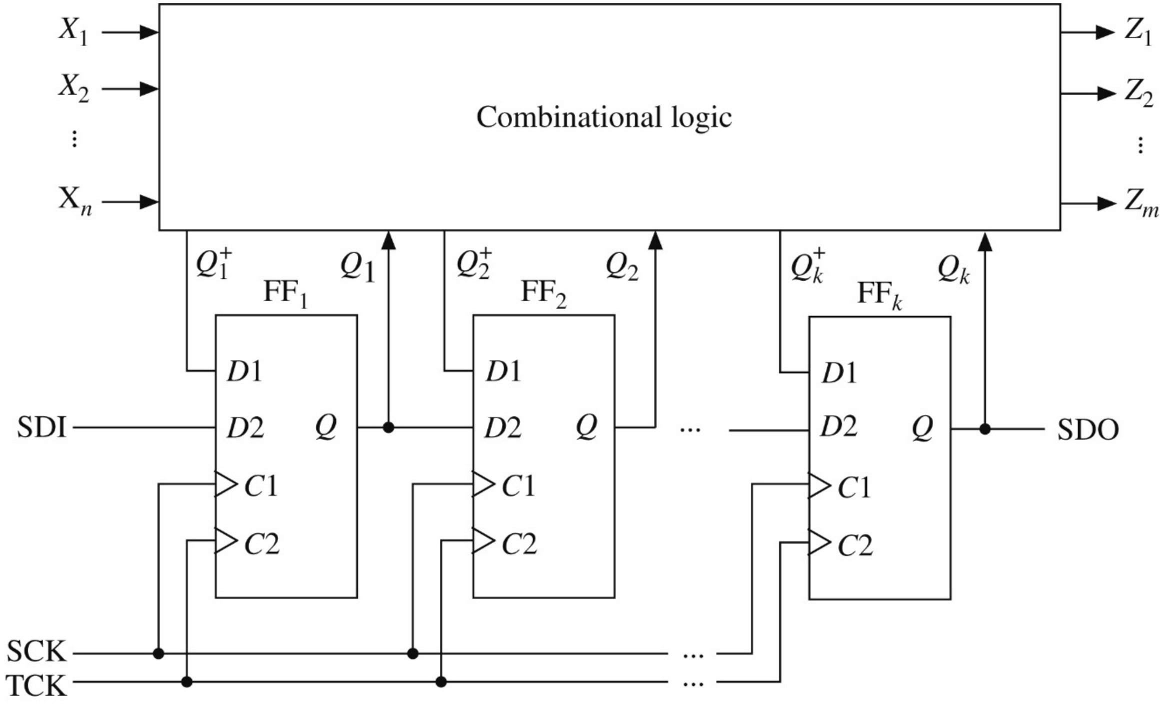


From external inputs X(m), internal signals Q(n) difficult to control/set to desired value

At external outputs Z(p), internal signals Q(n) difficult to observe/determine state

Scan path testing – improves controllability/observability

FIGURE 10-8: Scan Path Test Circuit Using Two-Port Flip-Flops



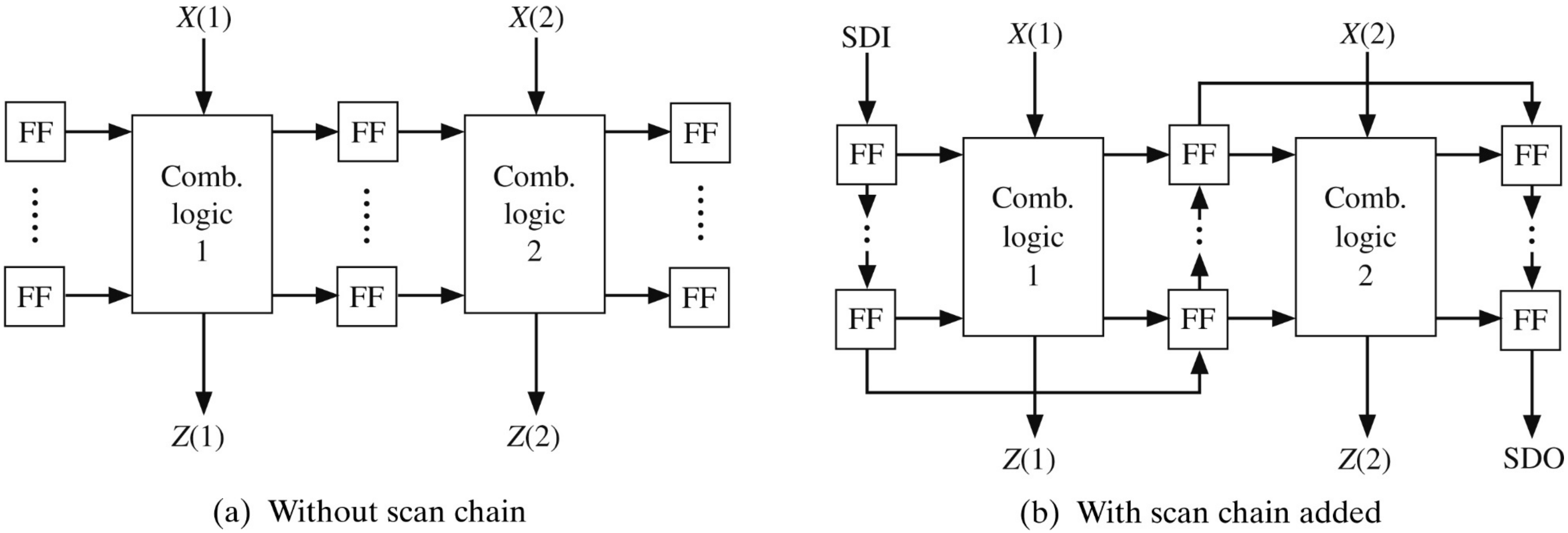
SDI is scan data input
SCK is system clock

SDO is scan data output
TCK is test clock

During test (TCK clock): configure flip flops as a shift register (scan path) to load test patterns via input from SDI and observe flip flop states at output SDO.
Normal operation (SCK clock): flip flop inputs/outputs connect from/to combinational ckt.

Circuit with and without scan chain

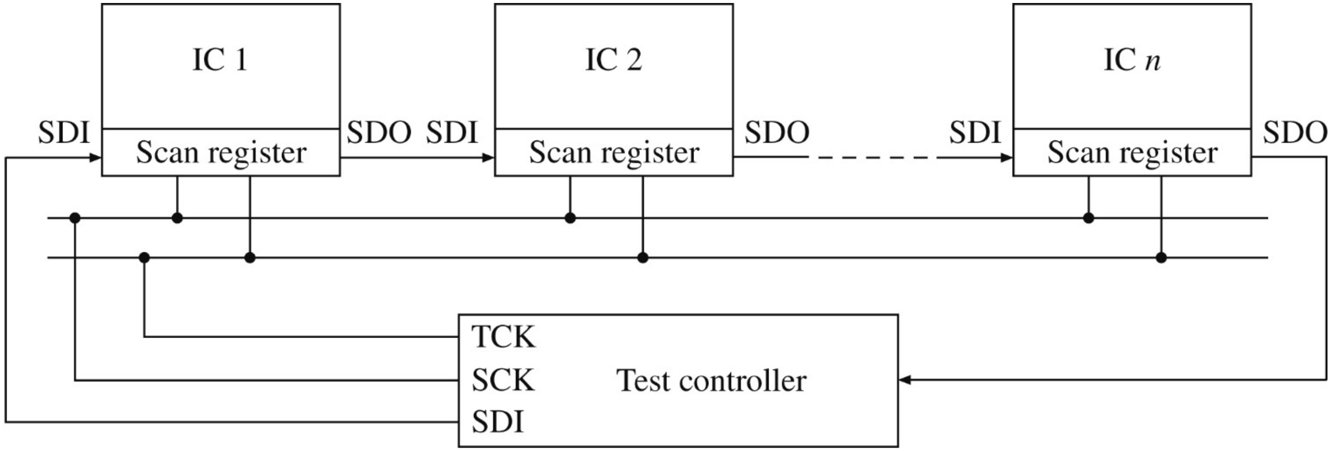
FIGURE 10-10: System with Flip-Flop Registers and Combinational Logic Blocks



(a) Without scan chain

(b) With scan chain added

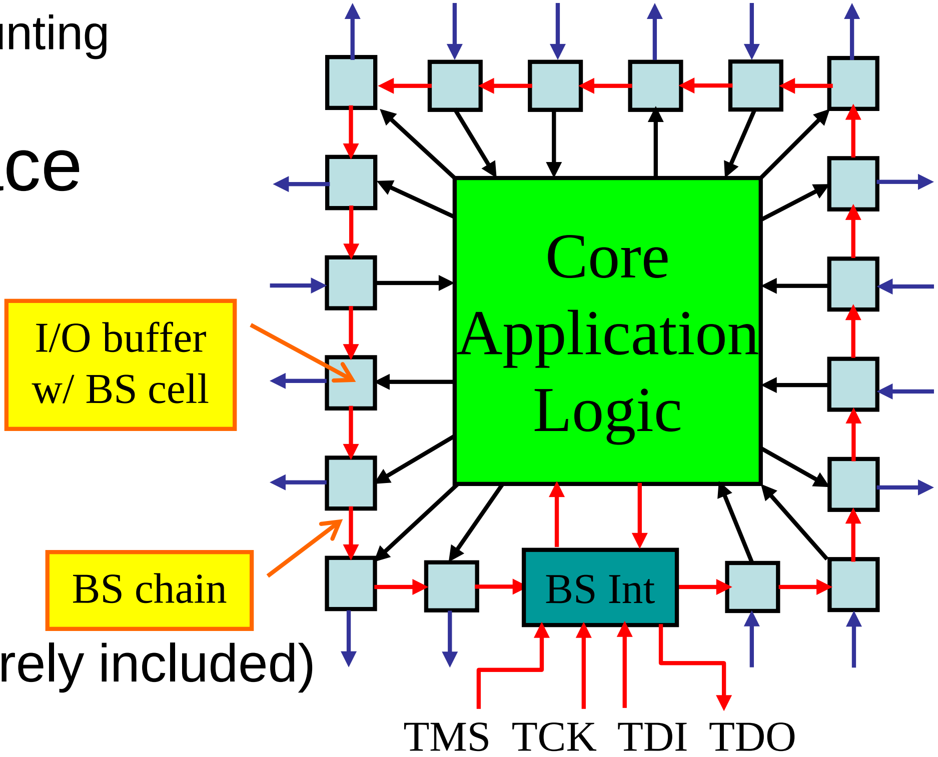
FIGURE 10-11: Scan Test Configuration with Multiple ICs



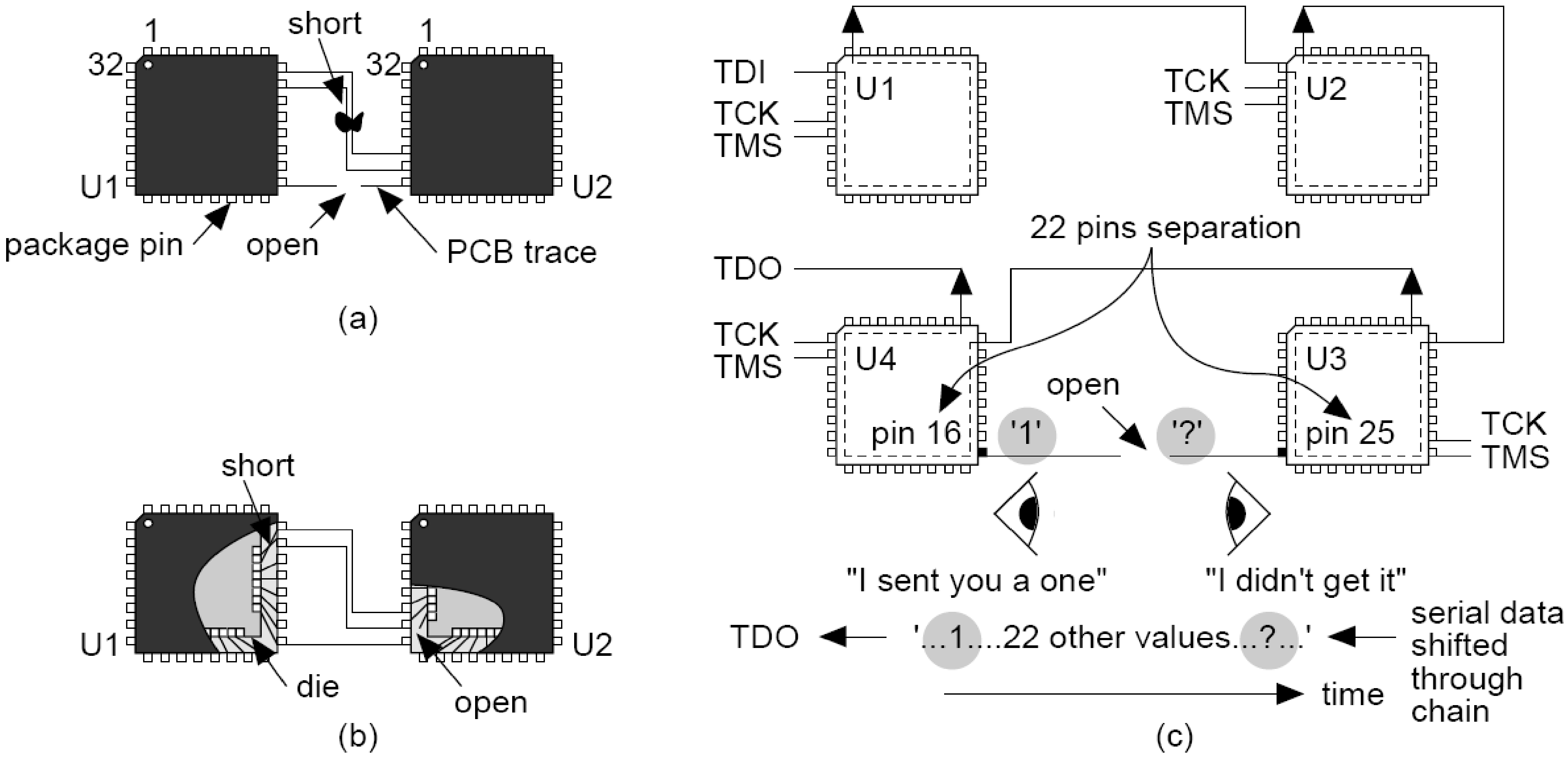
One long scan path

Boundary Scan (Text: Chap. 10.4)

- Developed to test interconnect between chips on PCB
  - Originally referred to as JTAG (Joint Test Action Group)
  - Uses scan design approach to test external interconnect
  - No-contact probe overcomes problem of “in-circuit” test:
    - surface mount components with less than 100 mil pin spacing
    - double-sided component mounting
    - micro- and floating vias
- Standardized test interface
  - IEEE standard 1149.1
  - Four wire interface
    - TMS - Test Mode Select
    - TCK - Test Clock
    - TDI - Test Data In
    - TDO - Test Data Out
    - TRST - reset (optional & rarely included)



Use of boundary scan to detect shorts/opens between ICs



(a)

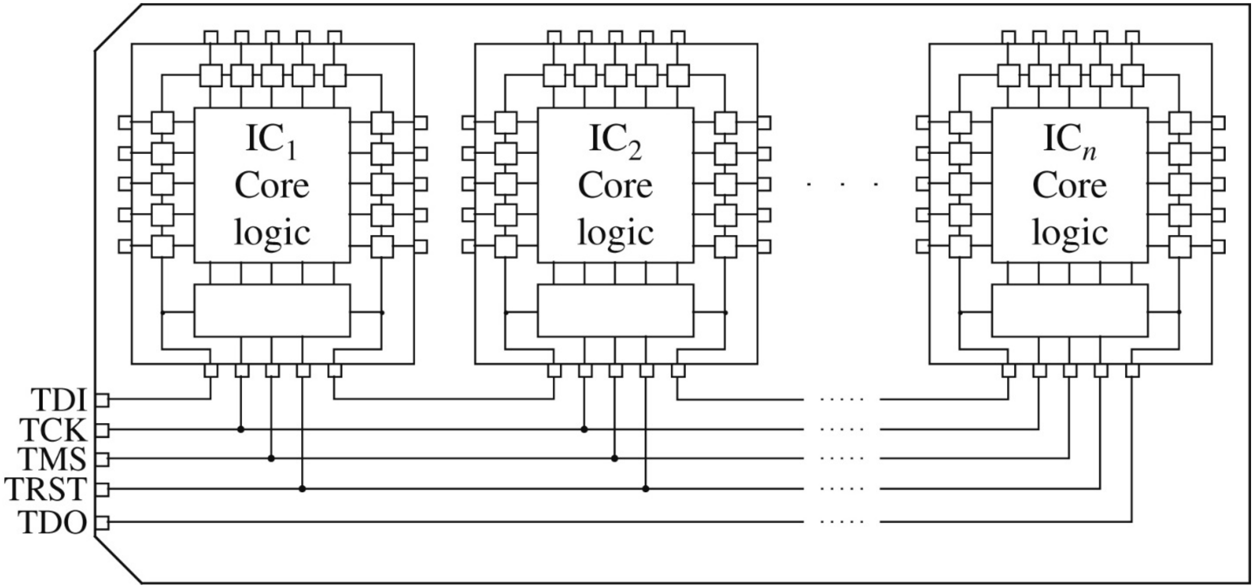
(b)

(c)

Source: M. Smith *Application-Specific IC's,* Figure 14.1

PC board test with boundary scan

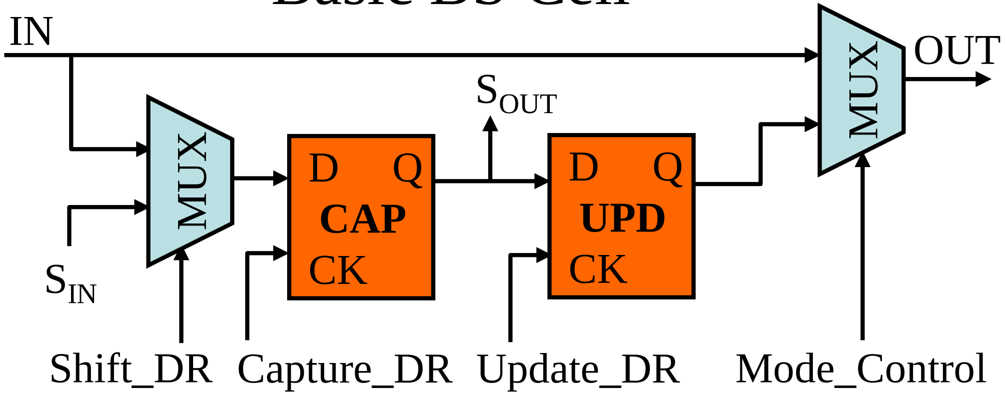
FIGURE 10-13:
PC Board with Boundary Scan ICs



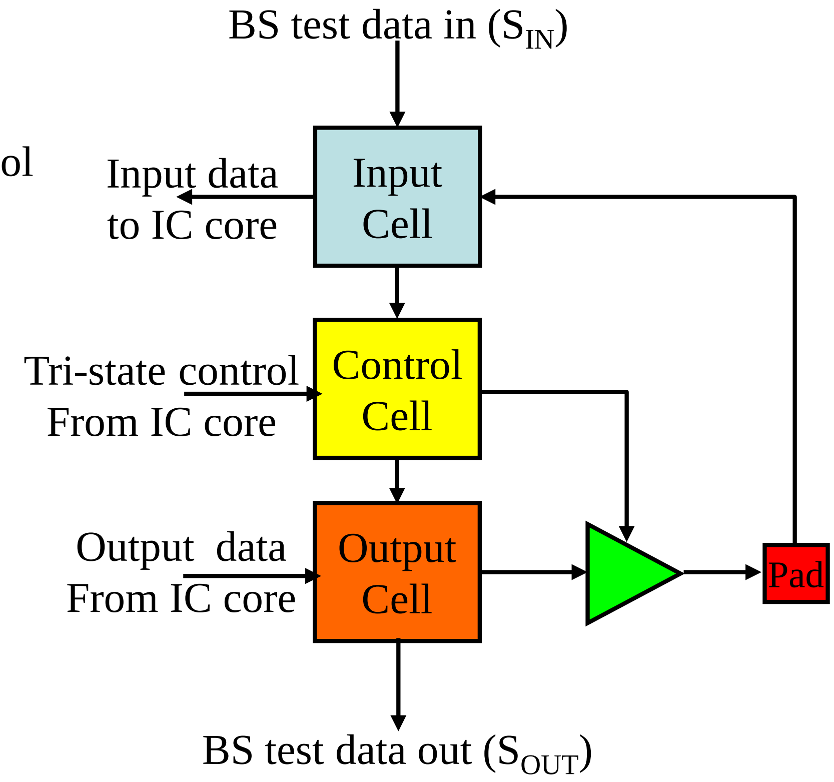
Link I/O cells of all ICs into one long scan chain

Boundary Scan Cell Architecture

Basic BS Cell



Bi-directional buffers require multiple BS cells



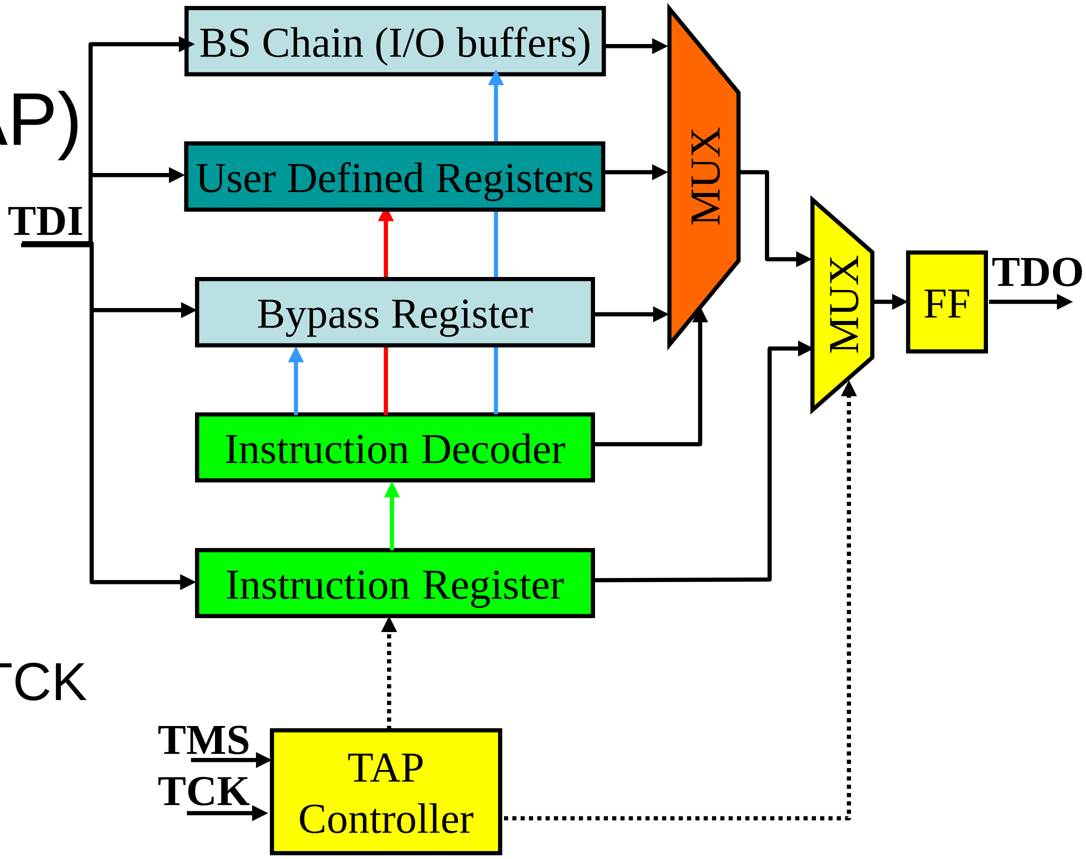
BS Cell Operation

Operational Mode	Data Transfer
Normal	IN → OUT
Scan	S_{IN} → CAP
Capture	IN → CAP
Update	CAP → UPD

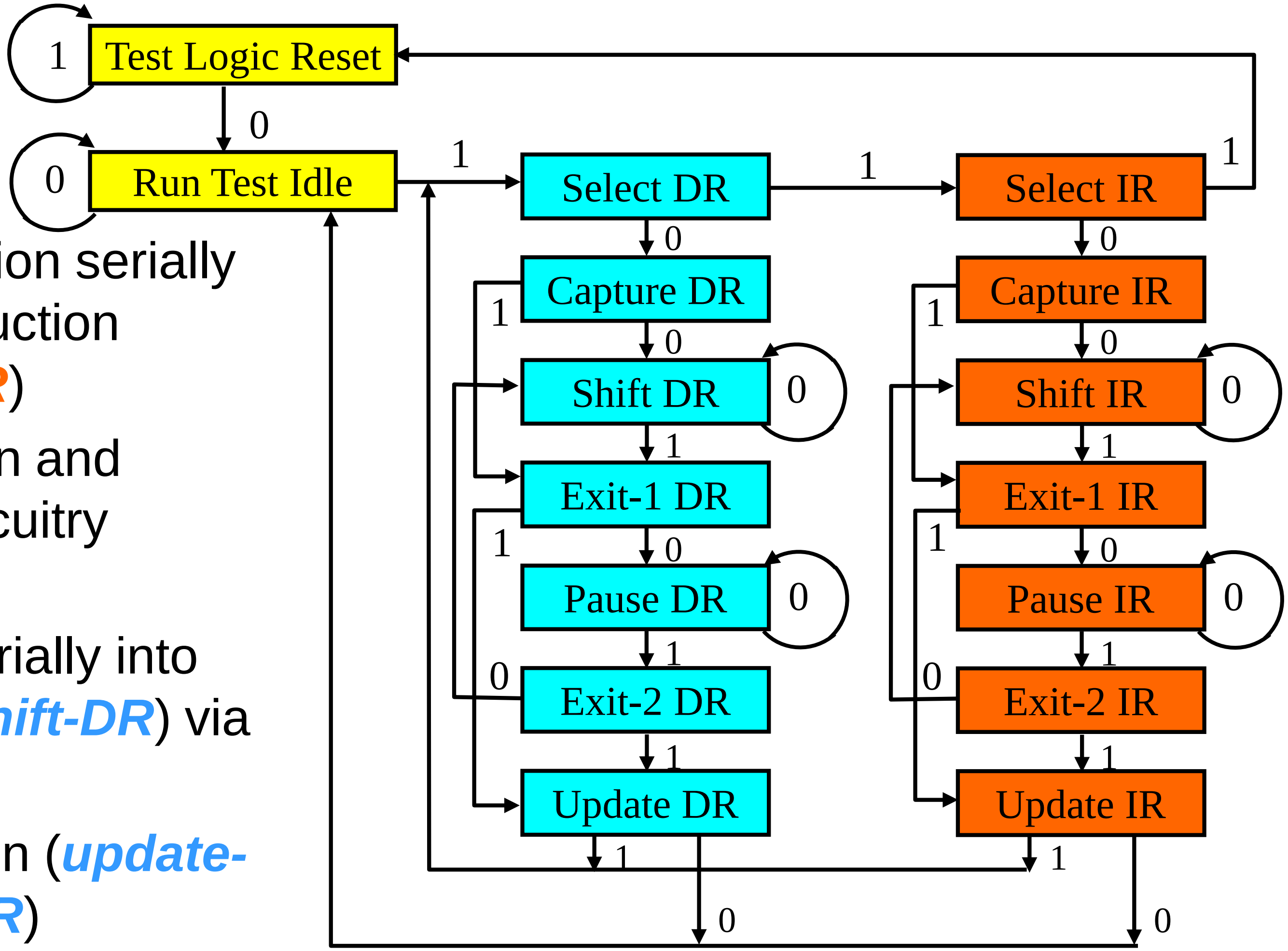
Boundary Scan Architecture

Additional logic :

- 1 Boundary Scan cell per I/O pin
- Test Access Port (TAP)
  - 4-wire interface
    - TMS
    - TCK
    - TDI
    - TDO
  - TAP controller
    - 16-state FSM
    - controlled by TMS & TCK
  - various registers for
    - instructions
    - operations



Boundary Scan TAP Controller Operation



Note: transitions on rising edge of TCK based on TMS value

1. Send test instruction serially via *TDI* into Instruction Register (***shift-IR***)
2. Decode instruction and configure test circuitry (***update-IR***)
3. Send test data serially into Data Register (***shift-DR***) via *TDI*
4. Execute instruction (***update-DR*** & ***capture-DR***)
5. Retrieve test results captured in Data Register (***shift-DR***) serially via *TDO*

Boundary Scan Instructions

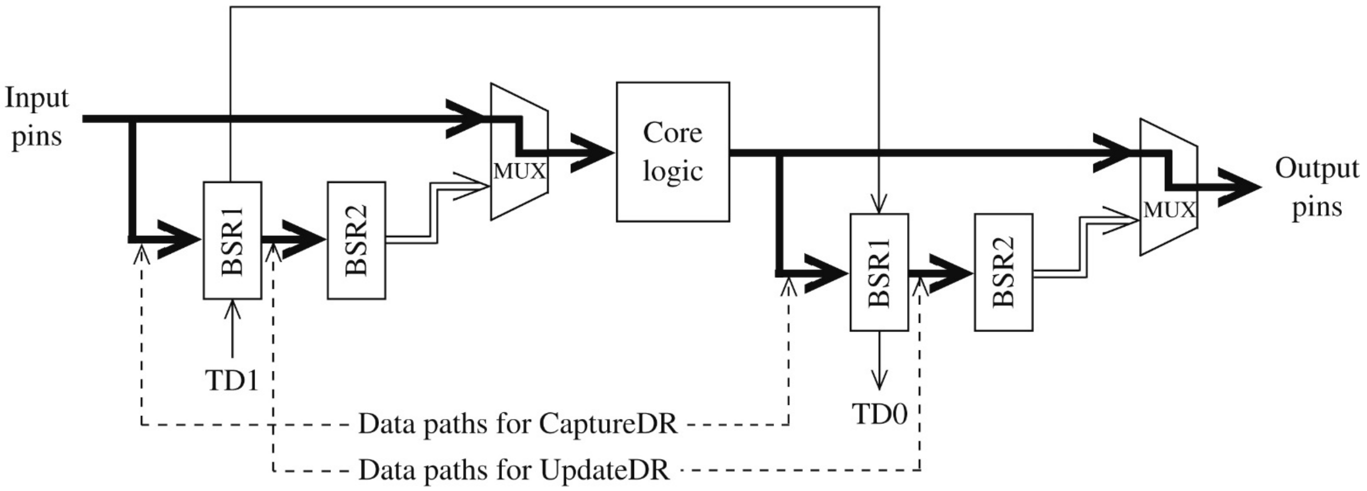
Defined by IEEE 1149.1 standard:

- Mandatory Instructions
  - Extest – to test external interconnect between ICs
  - Bypass – to bypass BS chain in IC
  - Sample/Preload – BS chain samples external I/O; can shift patterns from TDI-BS-TDO while ckt operates
  - IDCode – 32-bit device ID
- Optional Instructions
  - Intest – to test internal logic within the IC
  - RunBIST – to execute internal Built-In Self-Test
    - if applicable (this is rare)
  - UserCode – 32-bit programming data code
    - for programmable logic circuits
  - User Defined Instructions

Sample/Preload Instruction

- Capture external inputs in BSR1/2
- Capture core outputs in BSR1/2

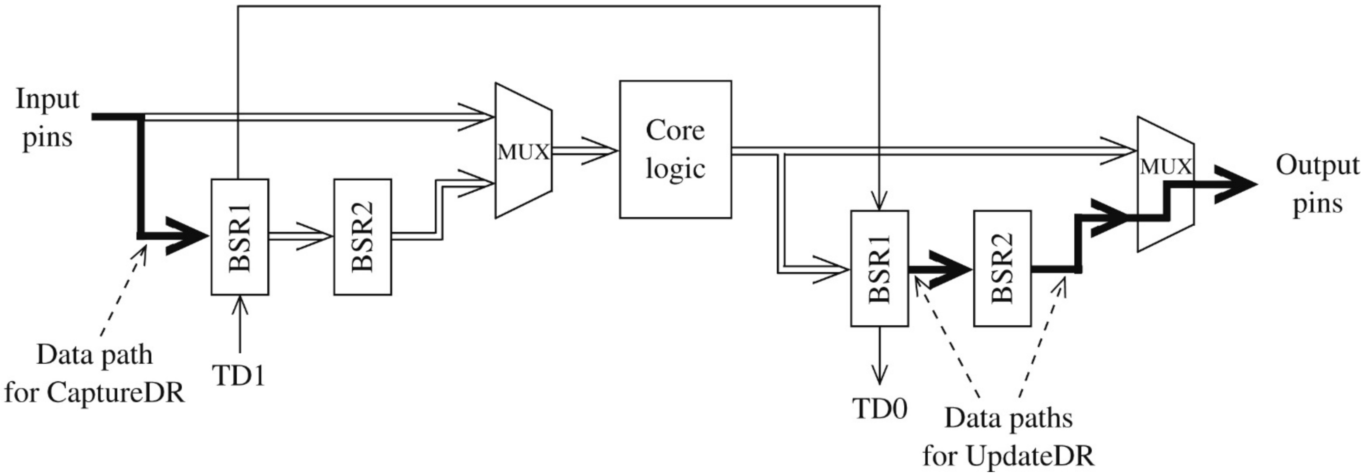
FIGURE 10-17: Signal Paths for Sample/Preload Instruction (highlighted)



Extest instruction

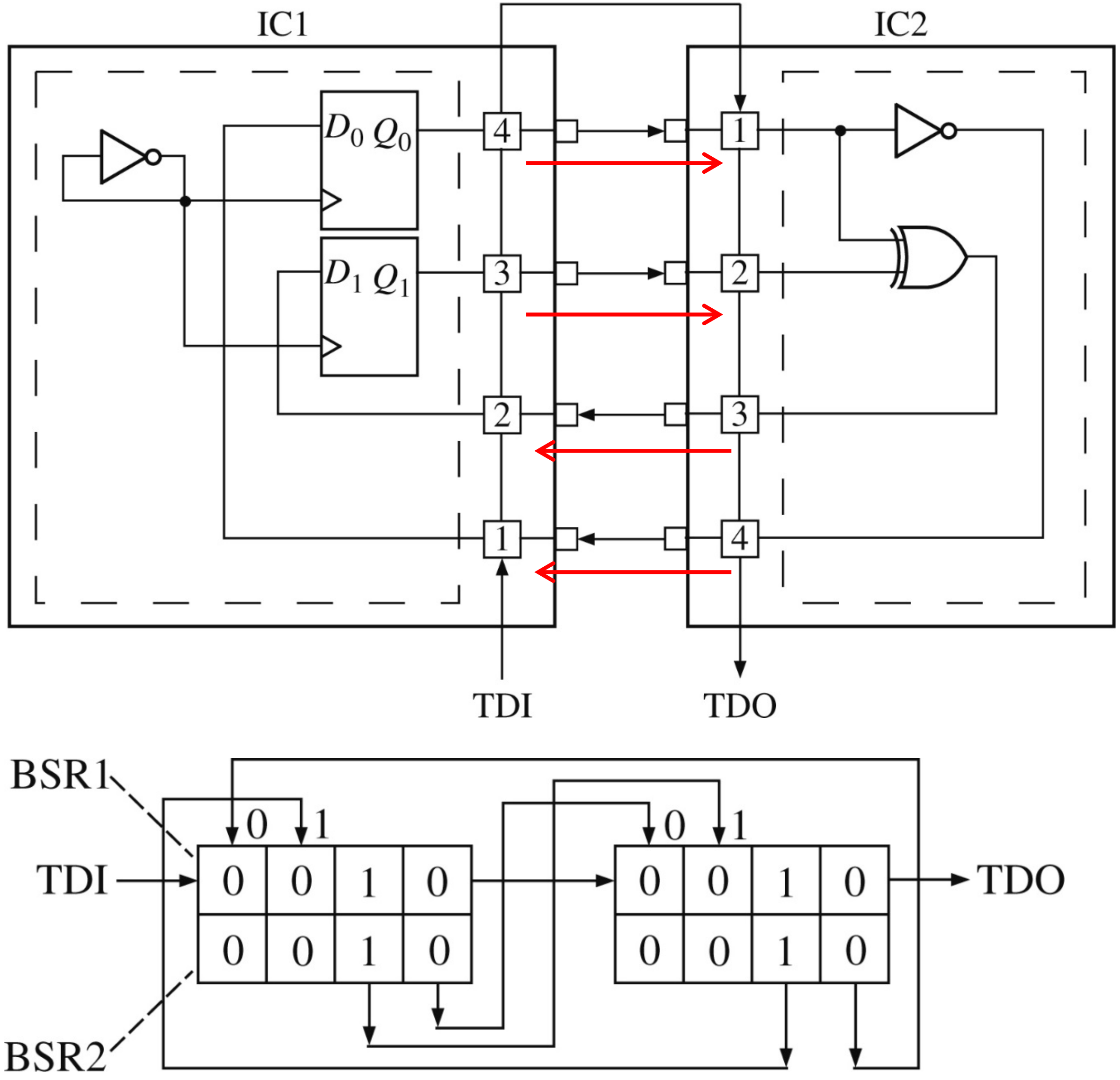
- Tests connections between IC pins
  - Previously: shift test pattern into BSR1/2 cells
  - Drive output pins with BSR2 (to external connections) and capture input pins in BSR1 (from external sources)
  - Later: shift out BSR1/2 to check for correct results

FIGURE 10-18: Signal Paths for Extest Instruction (highlighted)



PCB interconnect test

FIGURE 10-20: Interconnection Testing Using Boundary Scan



1. Shift pattern into the 8 boundary scan cells via TDI

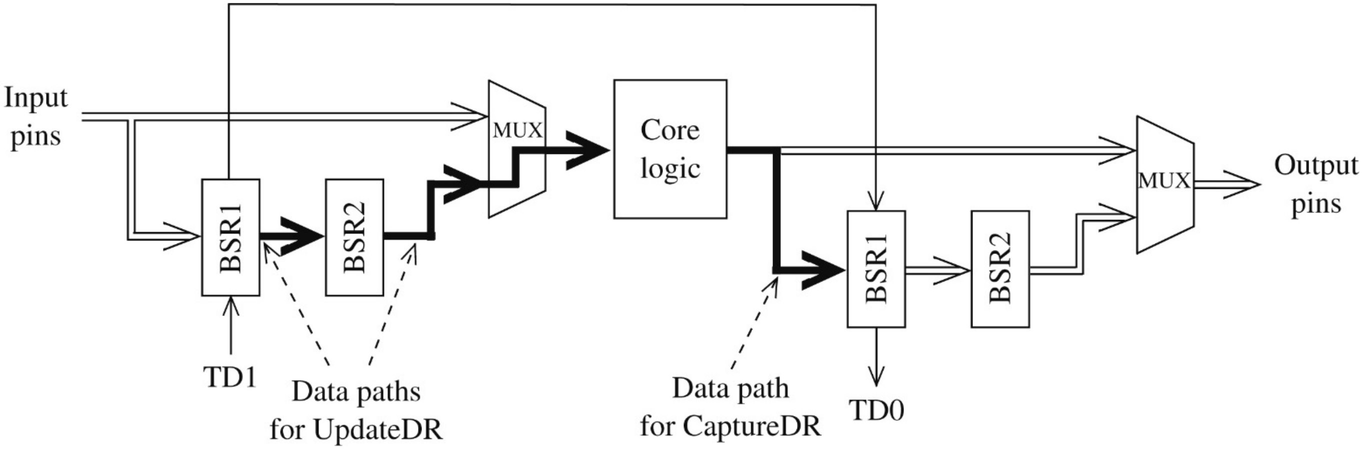
- Test pattern bits in cells 3,4 of C1 and IC2
- Capture results in cells 1,2 of IC1 and IC2

2. Shift results out via TDO.

Intest instruction

- Tests core logic
  - Previously: shift test patterns to BSR1/2
  - Apply patterns to core logic inputs from BSR2 and capture core logic outputs in BSR1
  - Later: shift out BSR1/2 to verify correct core outputs

FIGURE 10-19: **Signal Paths for Intest Instruction (highlighted)**



Boundary Scan: User-Defined Instructions

- User-defined instructions facilitate:
  - public instructions (available for customer use)
  - private instructions (for the manufacturer use only)
  - extending the standard to a universal interface
    - for any system operation feature or function
    - a communication protocol to access new IC test functions
- In FPGAs
  - Access to configuration memory to program device
  - Access to FPGA core programmable logic & routing resources
    - Xilinx is one of few to offer this

Boundary Scan: Advantages

- It's a standard! (IEEE 1149.1)
  - allows mixing components from different vendors
  - provides excellent interface to internal circuitry
- Supported by CAD tool vendors, IC & FPGA manufacturers
- Allows testing of board & system interconnect
  - back-plane interconnect test w/o using PCB functionality
  - very high fault coverage for interconnect
- Useful in diagnosis & FMA
  - provides component-level fault isolation
  - allows real-time sampling of devices on board
  - useful at wafer test (fewer probes needed)
- BS path reconfigured to bypass ICs for faster access

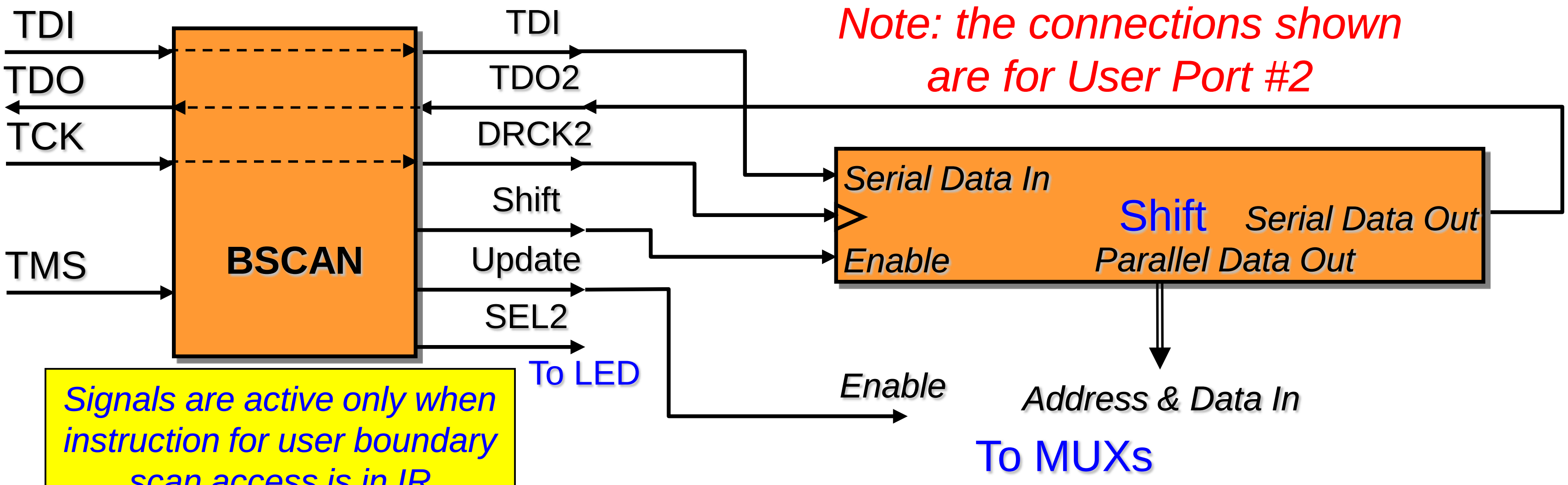
- IEEE P1500 uses BS circuitry around cores inside SoCs
  - TRST pin is not optional in order to initialize all cores

Boundary Scan: Disadvantages

- Overhead:
  - Logic: about 300 gates/chip for TAP + about 15 gates/pin
    - overall overhead typically small (1-3%)
    - but significant for only testing external interconnect
      - especially tri-state (2 cells) & bi-directional buffers (3 cells)
  - I/O Pins: 4
    - 5 if optional TRST (Test Reset) pin is included
      - Must be included in SoC cores to meet P1500 standards
  - I/O delay penalty
    - 1 MUX delay on all input & output pins
      - this can be reduced by design
- Cannot test at system clock speed
  - But internal BIST can run at system clock speed

Labs 8/10 BScan Communications

- As you serially shift in address and data on TDI via the USER2 port of the BSCAN module, the previous values you sent to the register file will shift out on TDO
  - As a sanity check, you can continue to shift in data and see it come out 6 clock cycles later in TDO in the Impact GUI
- As you shift in the last bit of your address and data bring TMS high to Exit-DR and the next clock with TMS high will activate Update and load your data into the appropriate register in your register file
  - You can go back to Shift-DR to shift in a new set of address and data values for another register and repeat the process as long as you like without changing IR

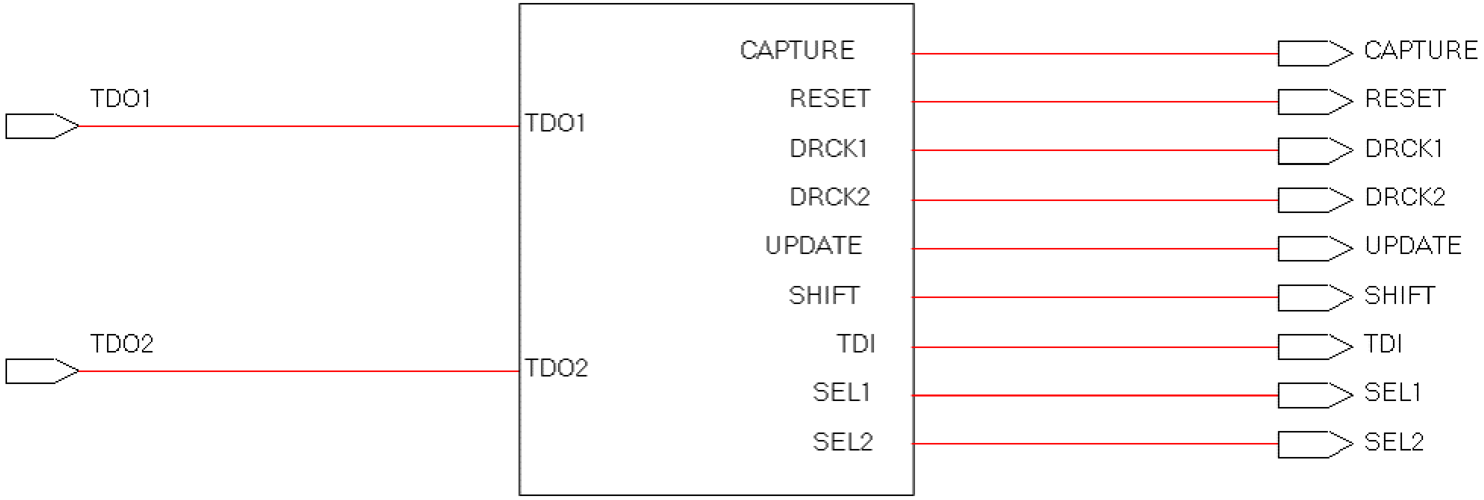


Note: the connections shown are for User Port #2

Signals are active only when instruction for user boundary scan access is in IR

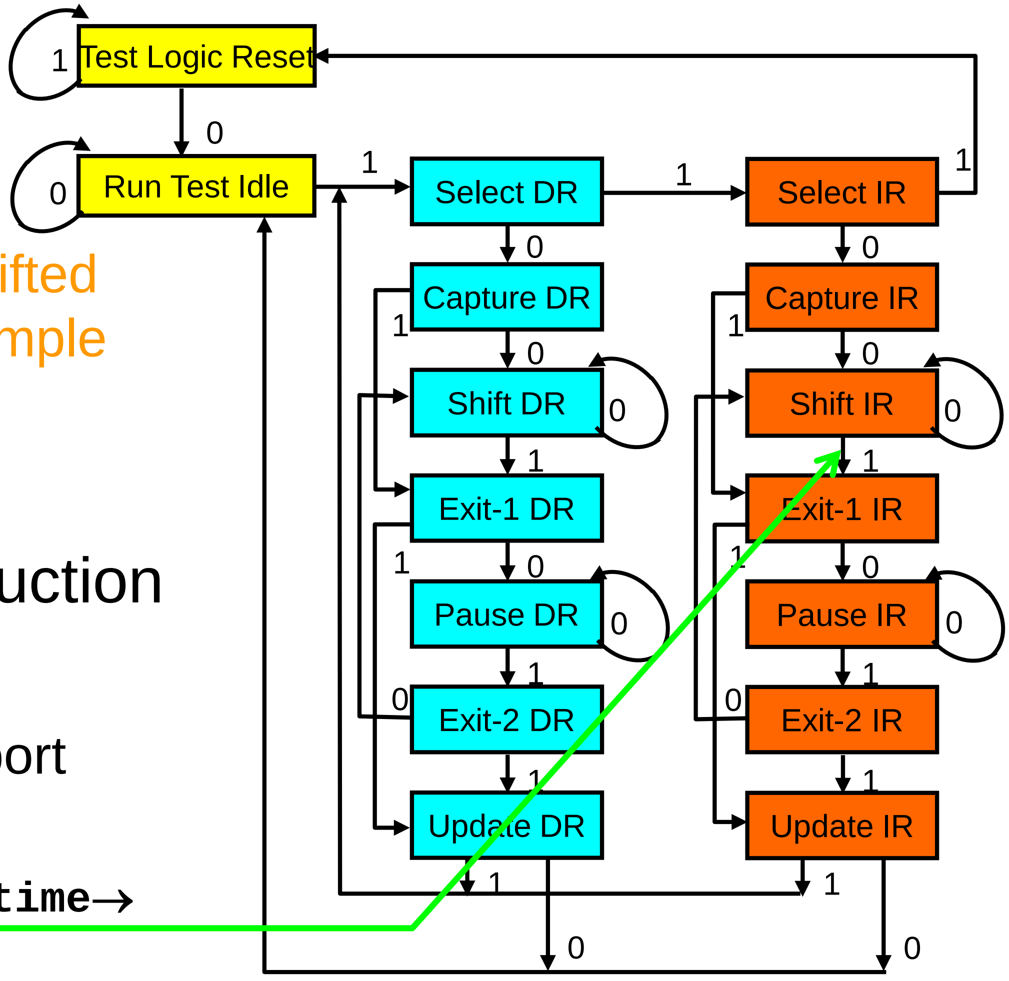
Spartan-3 BScan Module

```
library UNISIM;
use UNISIM.vcomponents.all;
BSCAN_SPARTAN3_inst : BSCAN_SPARTAN3
   port map (
      CAPTURE => CAPTURE, -- CAPTURE output from TAP controller (not used in your design)
      DRCK1 => DRCK1,     -- Data register output for USER1 functions (clock for shift register – bring to LED)
      DRCK2 => DRCK2,     -- Data register output for USER2 functions (clock for shift register – bring to LED)
      RESET => RESET,     -- Reset output from TAP controller (not used in your design)
      SEL1 => SEL1,       -- USER1 active output (not used in your design but bring it out to an LED)
      SEL2 => SEL2,       -- USER2 active output (not used in your design but bring it out to an LED)
      SHIFT => SHIFT,     -- SHIFT output from TAP controller (enable for shift register – bring it out to an LED)
      TDI => TDI,         -- TDI output from TAP controller (input data to shift register – bring it out to an LED)
      UPDATE => UPDATE,   -- UPDATE output from TAP controller (write enable to Lab 6 circuit)
      TDO1 => TDO1,       -- Data input for USER1 function (output data from shift register – bring it to an LED)
      TDO2 => TDO2        -- Data input for USER2 function (output data from shift register – bring it to an LED)
   );
```



Instructions for BSCAN Access

- 6-bit instruction
  - User port #1: 0x02
  - User port #2: 0x03
    - note that this is what is shifted into the IR in the time example below
  - Shifted into IR LSB first
  - Exit IR on last bit of instruction
  - You are now in Shift-DR
    - ready to access BSCAN port



```
TMS=01100 000001 1100 time→
TDI=xxxxx 110000 xxxx
```

moving from Test-Logic-Reset to Shift-IR

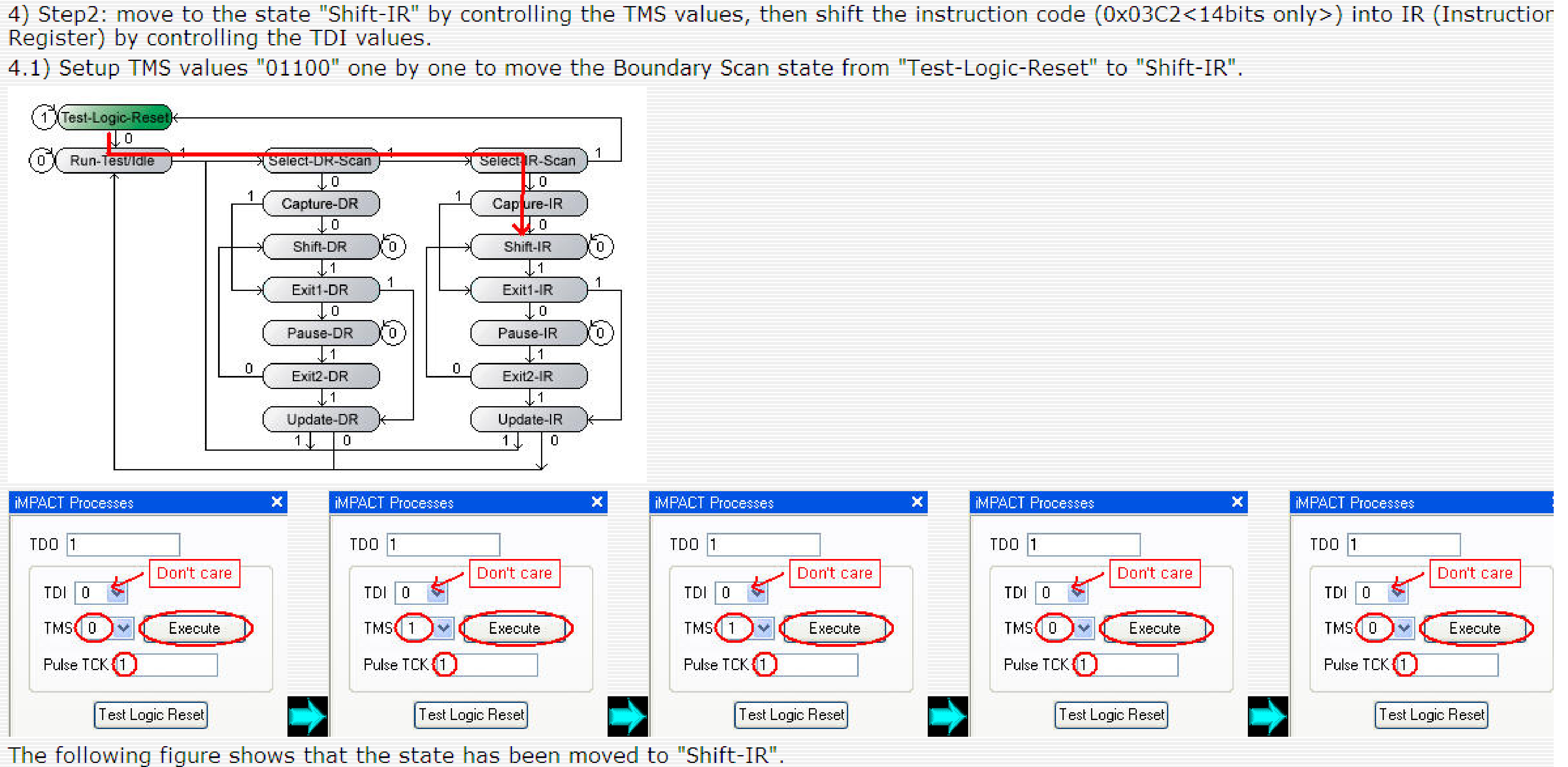
shifting in instrcution 0x03 LSB first

moving from Exit-IR to Shift-DR

Note: transitions on rising edge of TCK based on TMS value

Communicating With BS

- See tutorial by Gefu Xu (1st GTA for 4200 Fall ’04)
- Use Impact BS GUI to access Bscan module
  - Communicate with your circuit
  - We use this interface to control & execute BIST in FPGAs



4) Step2: move to the state "Shift-IR" by controlling the TMS values, then shift the instruction code (0x03C2<14bits only>) into IR (Instructior Register) by controlling the TDI values.

4.1) Setup TMS values "01100" one by one to move the Boundary Scan state from "Test-Logic-Reset" to "Shift-IR".

iMPACT Processes

TDO 1

Don't care

TDI 0

TMS 0

Pulse TCK 1

iMPACT Processes

TDO 1

Don't care

TDI 0

TMS 1

Pulse TCK 1

iMPACT Processes

TDO 1

Don't care

TDI 0

TMS 1

Pulse TCK 1

iMPACT Processes

TDO 1

Don't care

TDI 0

TMS 0

Pulse TCK 1

iMPACT Processes

TDO 1

Don't care

TDI 0

TMS 0

Pulse TCK 1

The following figure shows that the state has been moved to "Shift-IR".