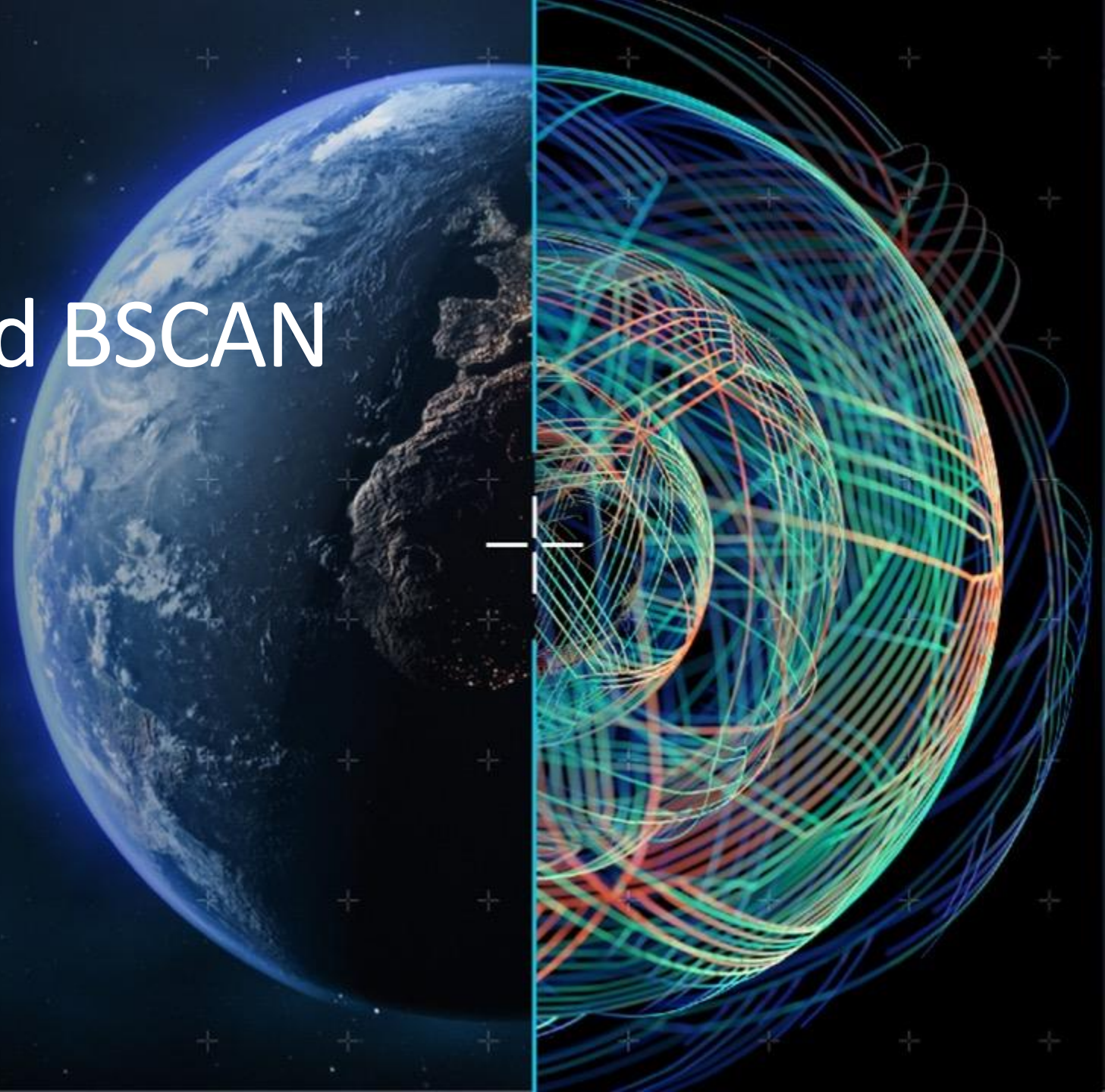
arm

Basics of MBIST and BSCAN



VLSI to System design: Silicon-to-end application approach

Vamsikrishna Dabbara

July 2023

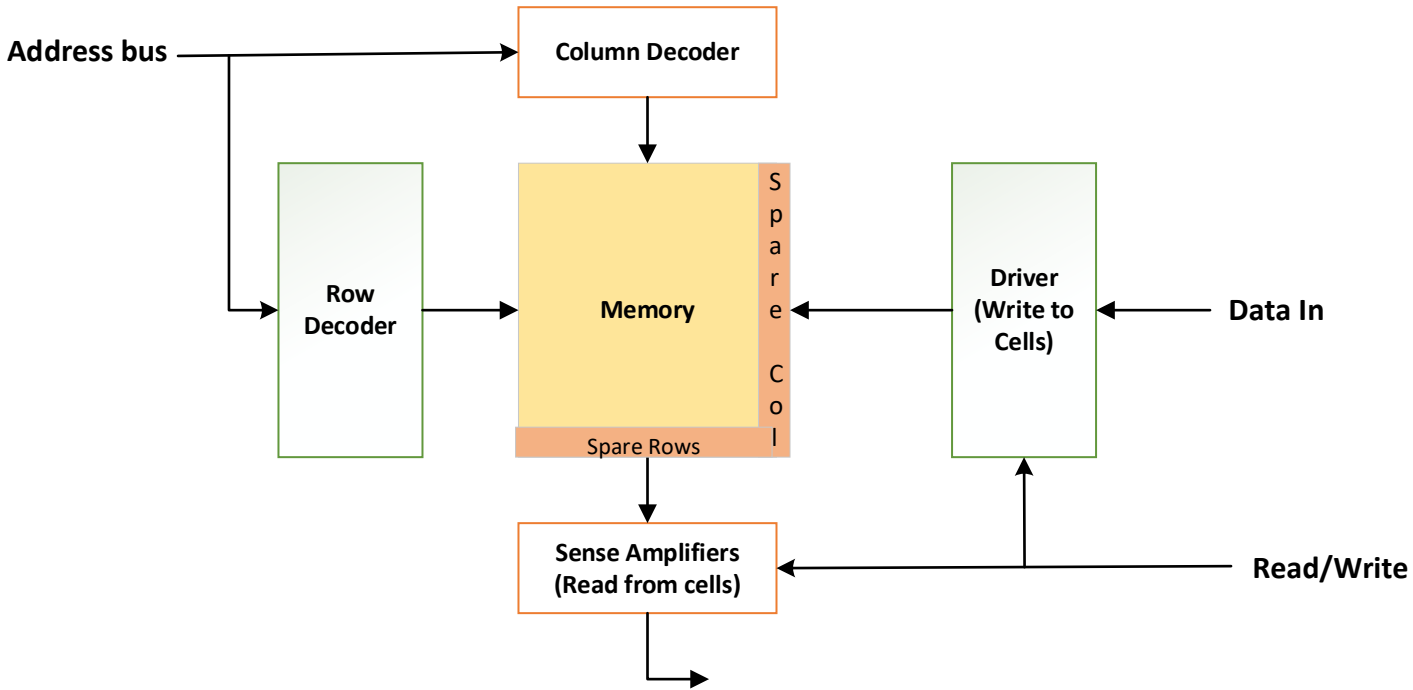
Agenda

- Memory architecture
- Why memory test
- MBIST algorithms
- Memory Test Methodologies
- Memory Redundancy and Repair
- BSCAN
- In-Circuit test
- Why BSCAN
- Principle of BSCAN
- Boundary scan instructions
- TAP controller state machine
- Boundary scan cell
- Boundary operation for EXTEST instruction

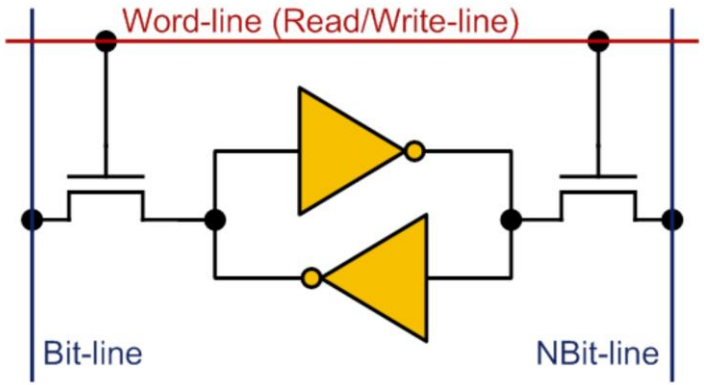
Memory architecture

- Memory helps us to store the large amounts of data getting by our surroundings.
- Typically, memories are occupying >40% of the die area in certain applications(Base station, storage devices etc.)
- Everyday memory technologies are getting improved to have better PPA(Performance Power & Area)
- Memories are not built with simple logic gates or flops.
- Typical memory model is as shown below,
  - It is constructed using array(Row/Column) structure.

Typical memory model

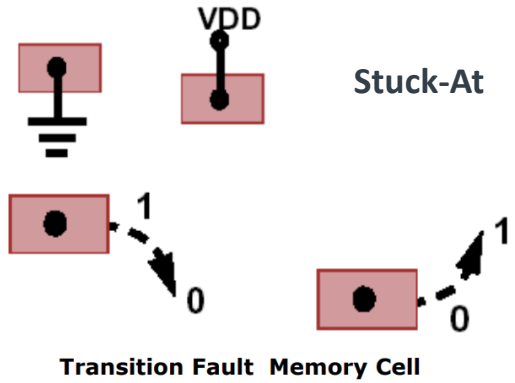


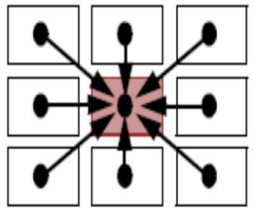
Typical 6T SRAM cell



Why Memory test

- Since memories consume a significant amount of area in design/silicon, they are dense in nature and hence they are susceptible to manufacture defects.
- The nature of memory defects are different from regular logic defects and cannot be covered using regular SCAN design techniques.
- Typical memory fault models are,
  - Stuck-At fault(SA)
  - Transient fault(TF)
  - Coupling fault(CF)
  - Neighbourhood pattern sensitive fault (NPSF)
  - Address decoder faults(ADF)



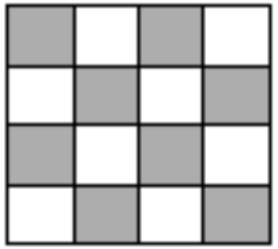


MBIST algorithms

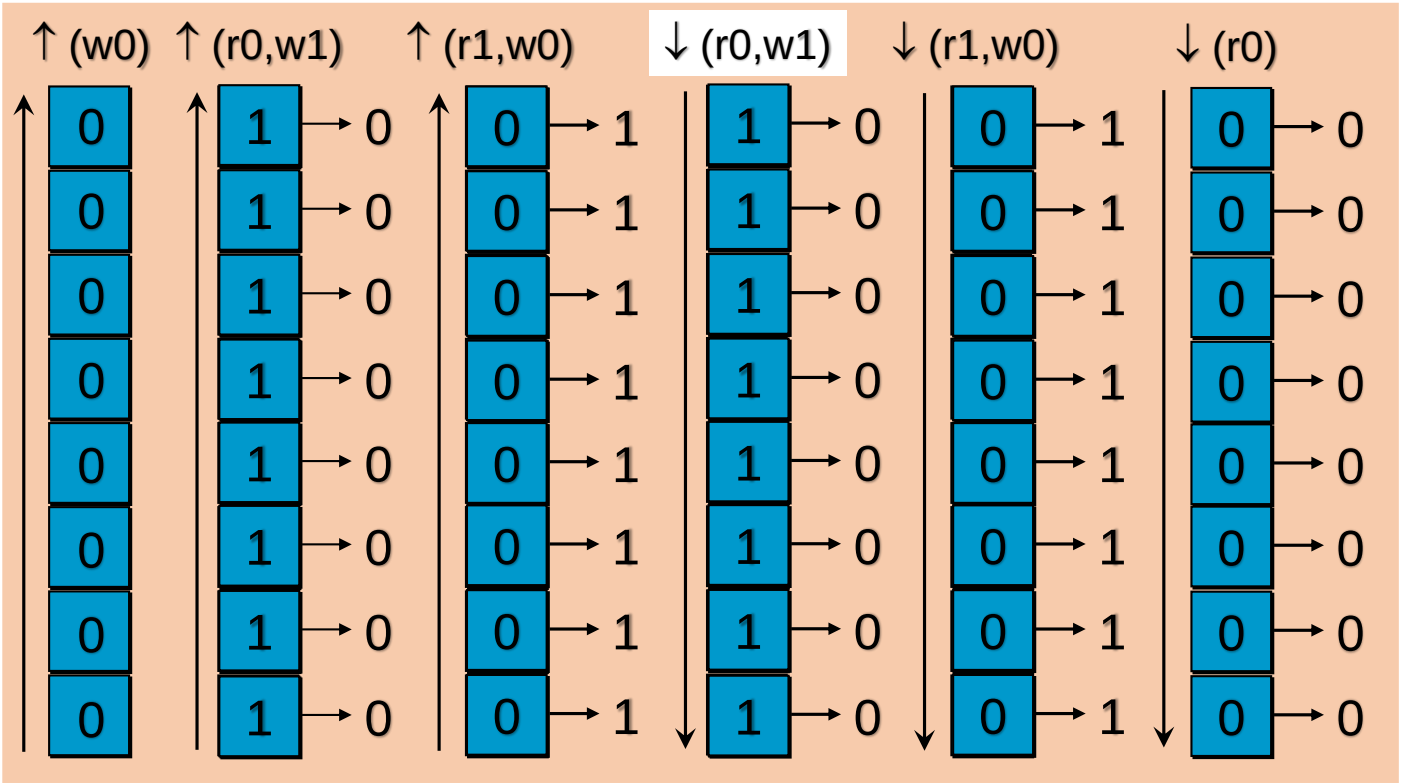
- Memories are tested with special algorithms which detect the faults occurring in memories. Several different algorithms can be used to test RAMs and ROMs.
  - Classical algorithms
    - MSCAN
    - Checkerboard
  - March algorithms
    - March X
    - March C

MBIST algorithms(Cont.)

Checkerboard algorithm



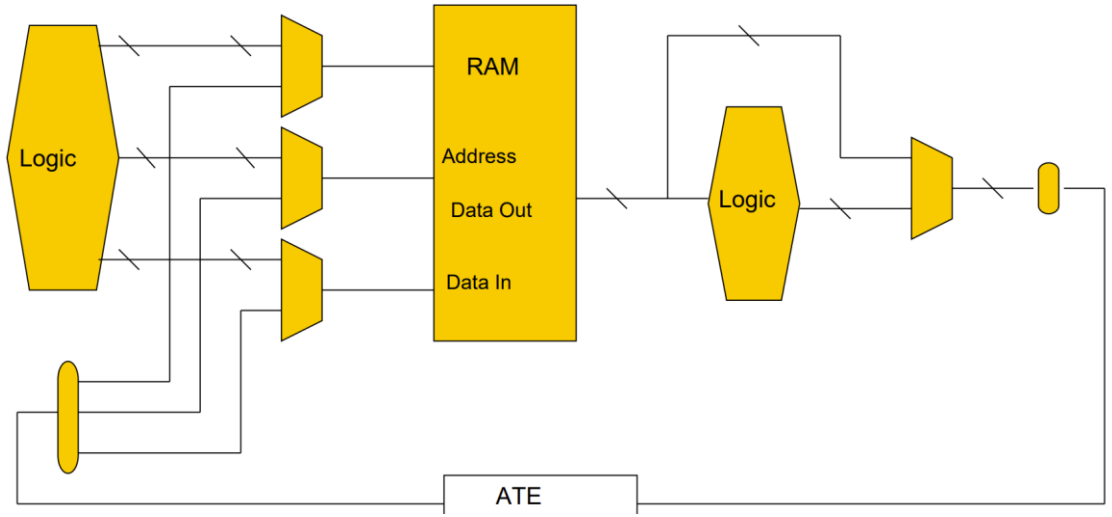
Write		Read	
0	1	0	1
1	0	1	0
0	1	0	1
1	0	1	0



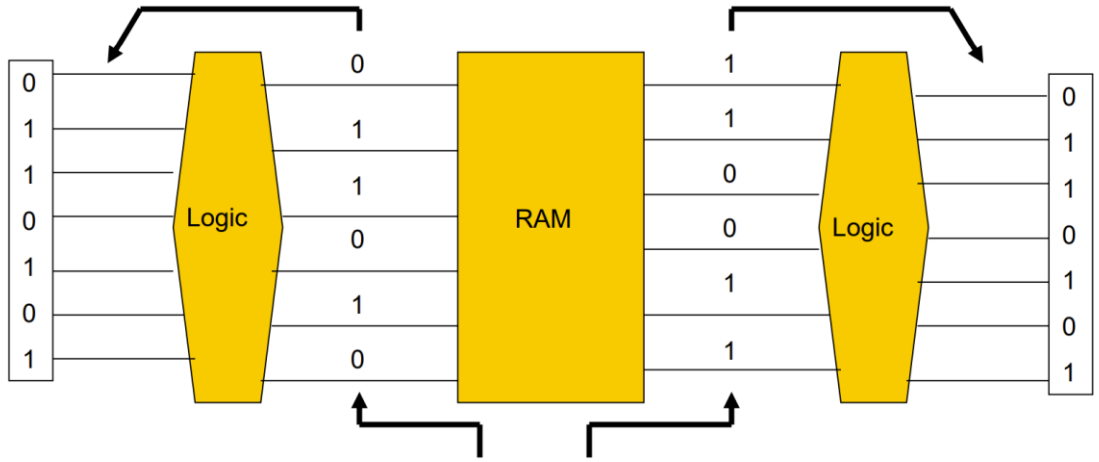
March algorithm

Memory Test Methodologies

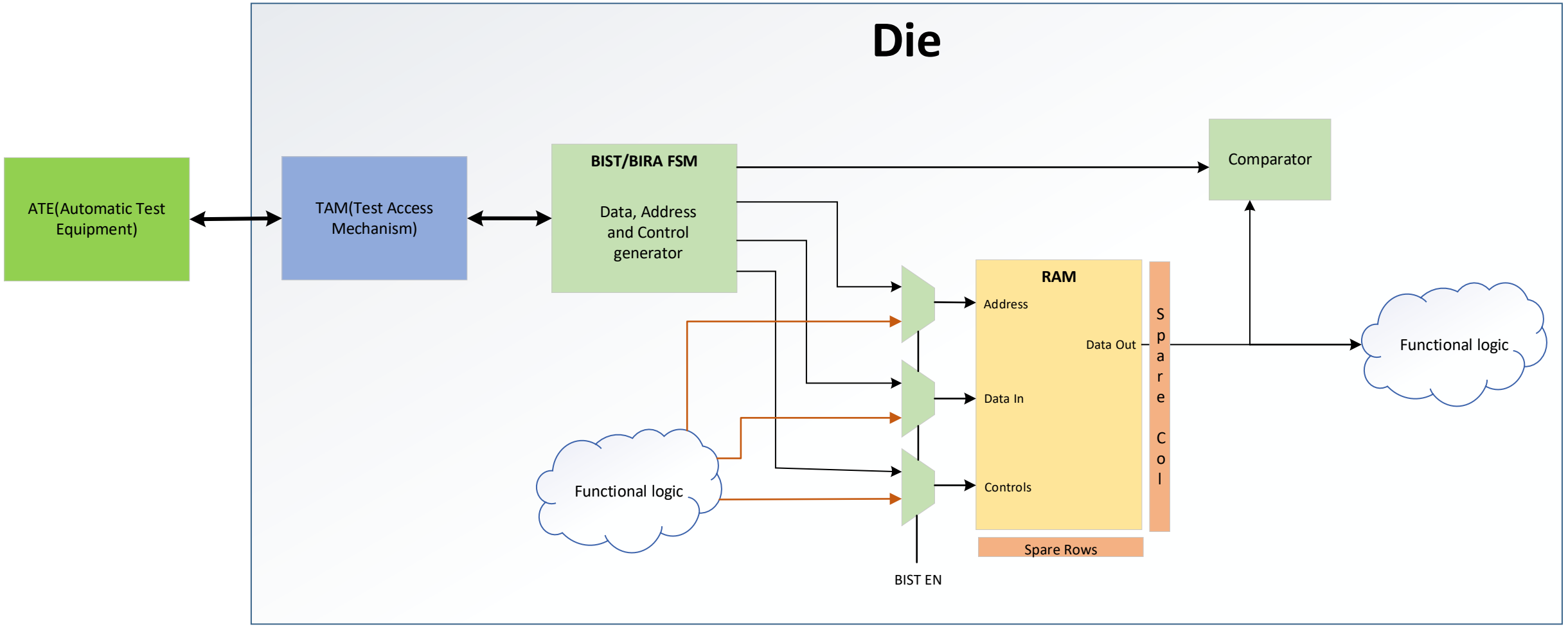
- **Direct Access**



- **Macro Testing**



Memory BIST(Built In Self-Test), BIRA(Built In Redundancy Analysis)



arm

Challenges with MBIST

- Area overhead
- Performance impact
- Design complexity and TAT

Memory Redundancy and Repair

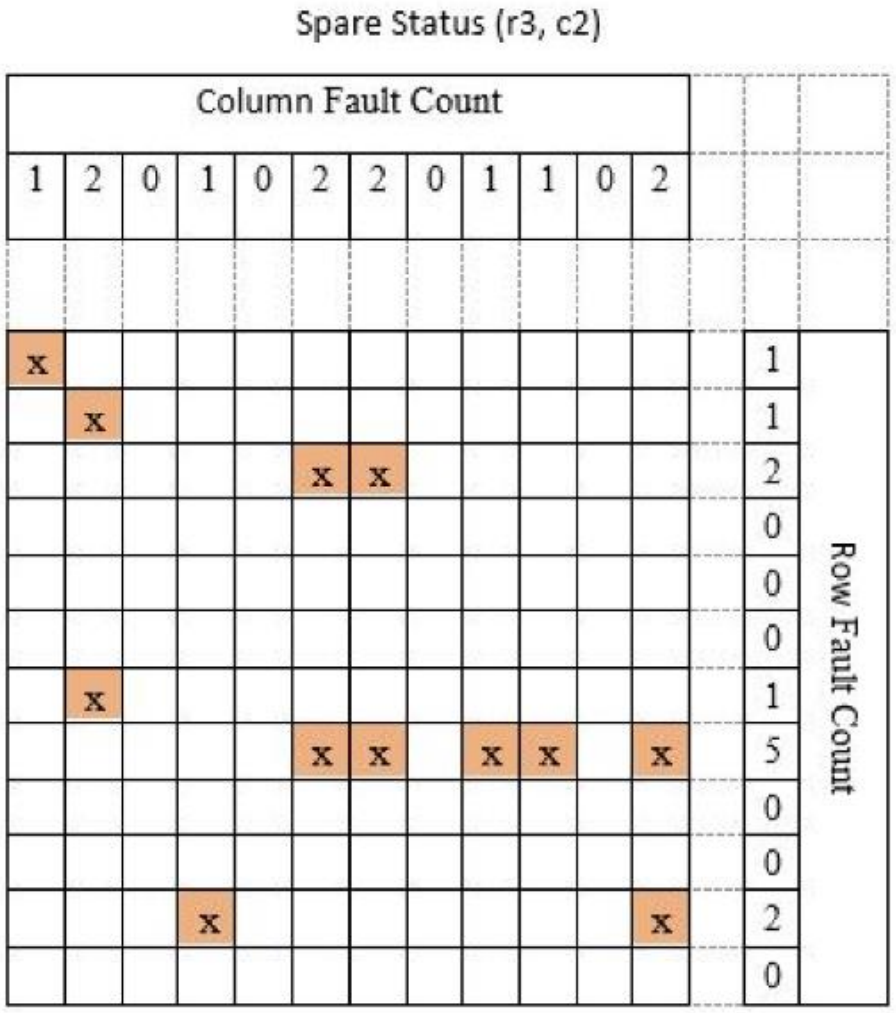
- Memories have a significant impact on yield.

- To avoid yield loss, redundant spare rows and/or columns of storage cells are added to memories.
  - Faulty cells can be redirected to redundant cells, rows and columns.
  - Repairable memories have repair registers which hold the repair signature.

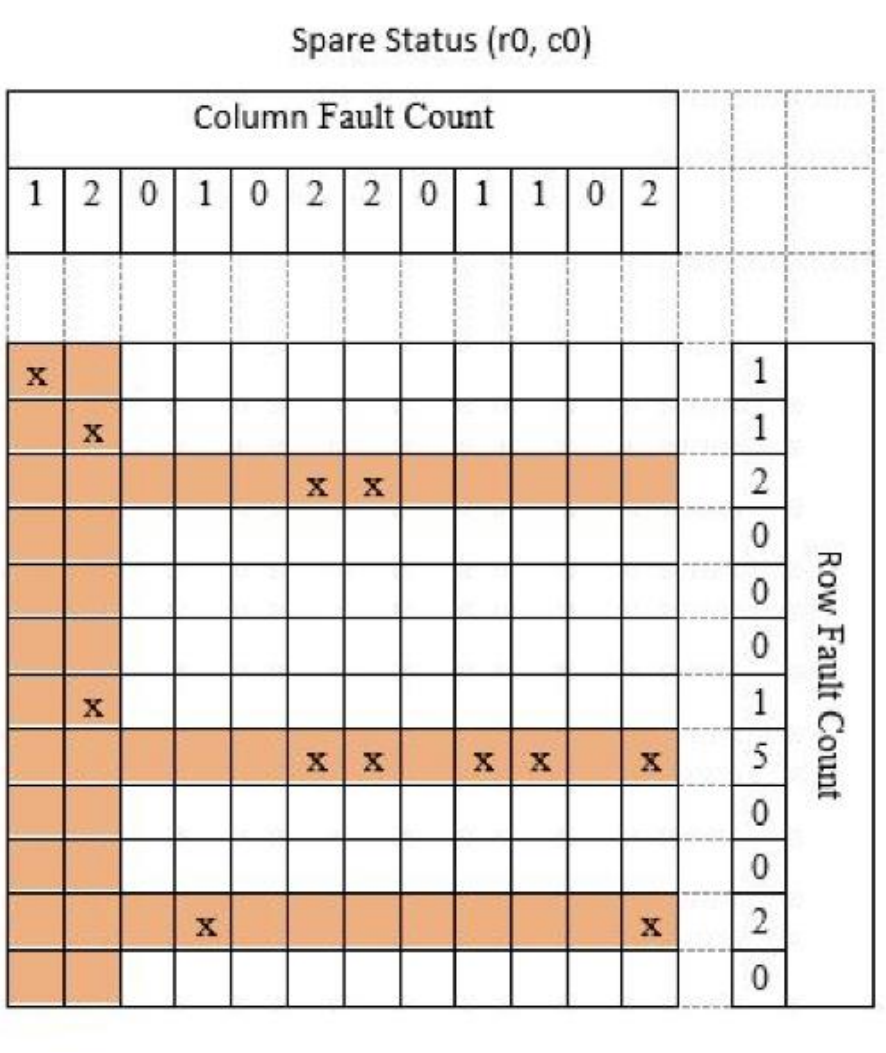
- Two types of repairs
  - Soft repair → Repair data is calculated every time during chip bootup
  - Hard Repair → Repair data is stored in efuse/OTP and loaded during chip bootup

Memory Redundancy and Repair

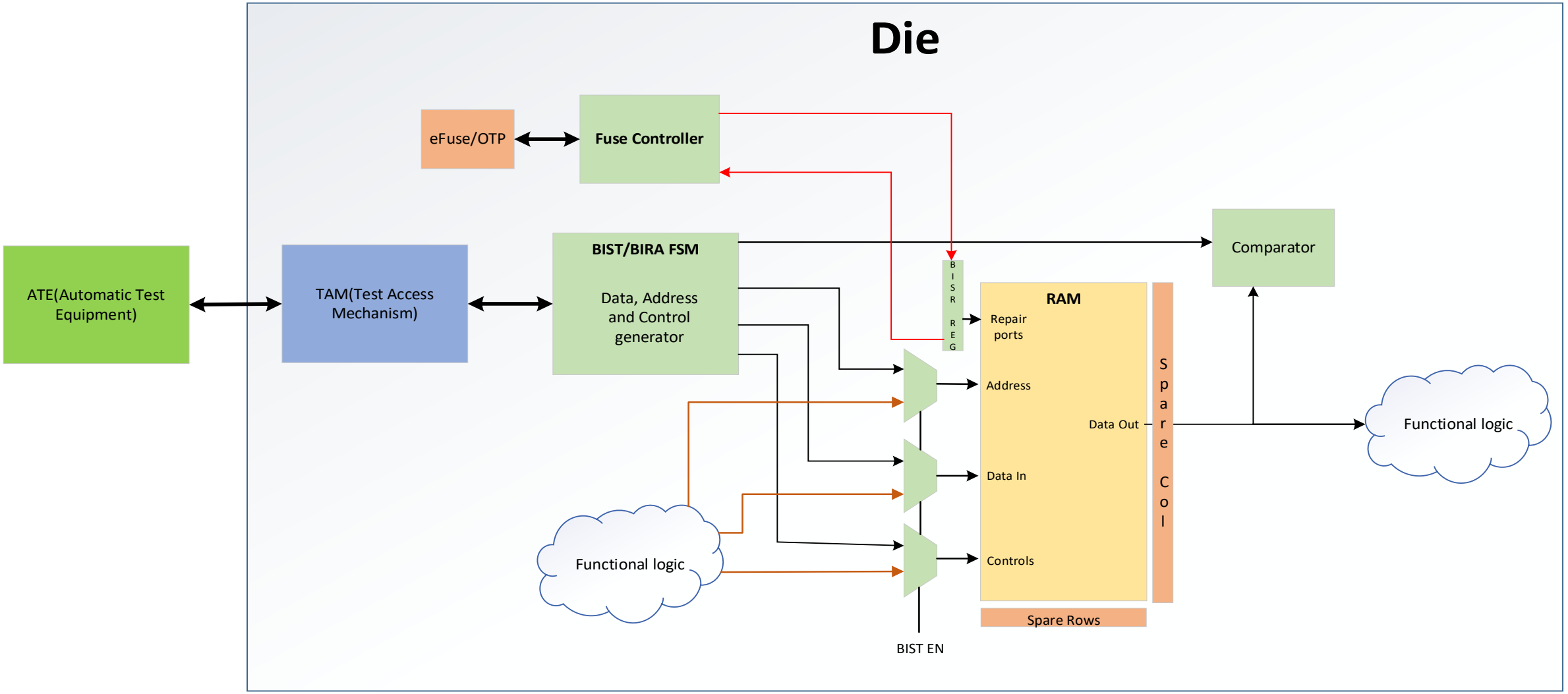
Pre-Repair



Post-Repair



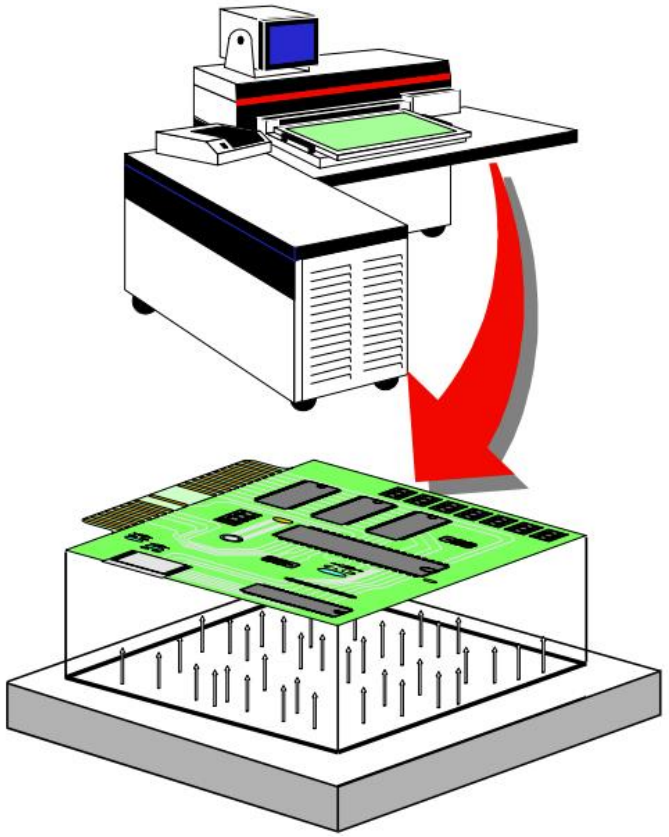
Memory Redundancy and Repair



arm

Boundary SCAN(BSCAN)

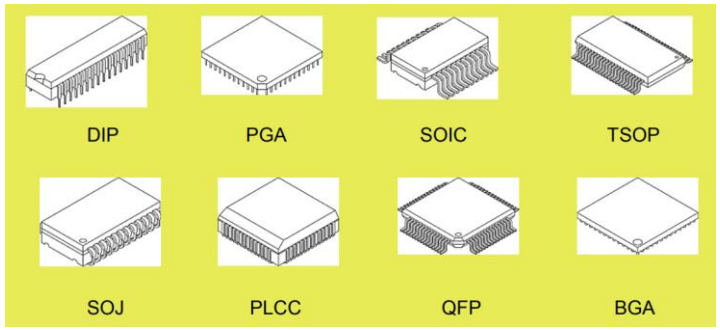
ICT(In Circuit test)



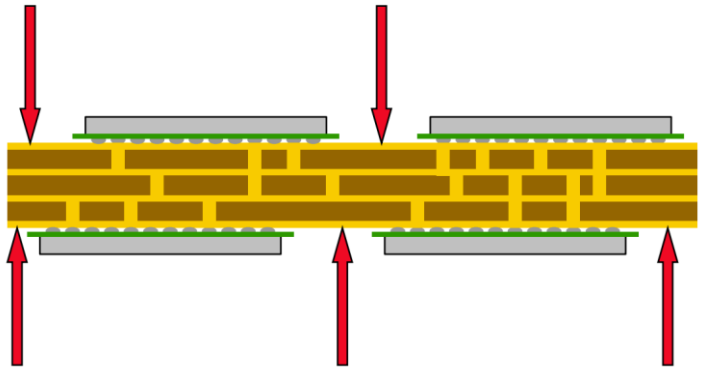
Bed-Of-Nails (MDA, ICT)

- The structural testing of loaded printed circuit boards has relied very heavily on the use of the so-called in-circuit bed-of-nails technique
- This method of testing makes use of a fixture containing a bed-of-nails to access individual devices on the board through test lands laid into the copper interconnect, or into other convenient physical contact points.
- Due to change in package styles and multi-layer boards, ICT is not efficient.

Package styles



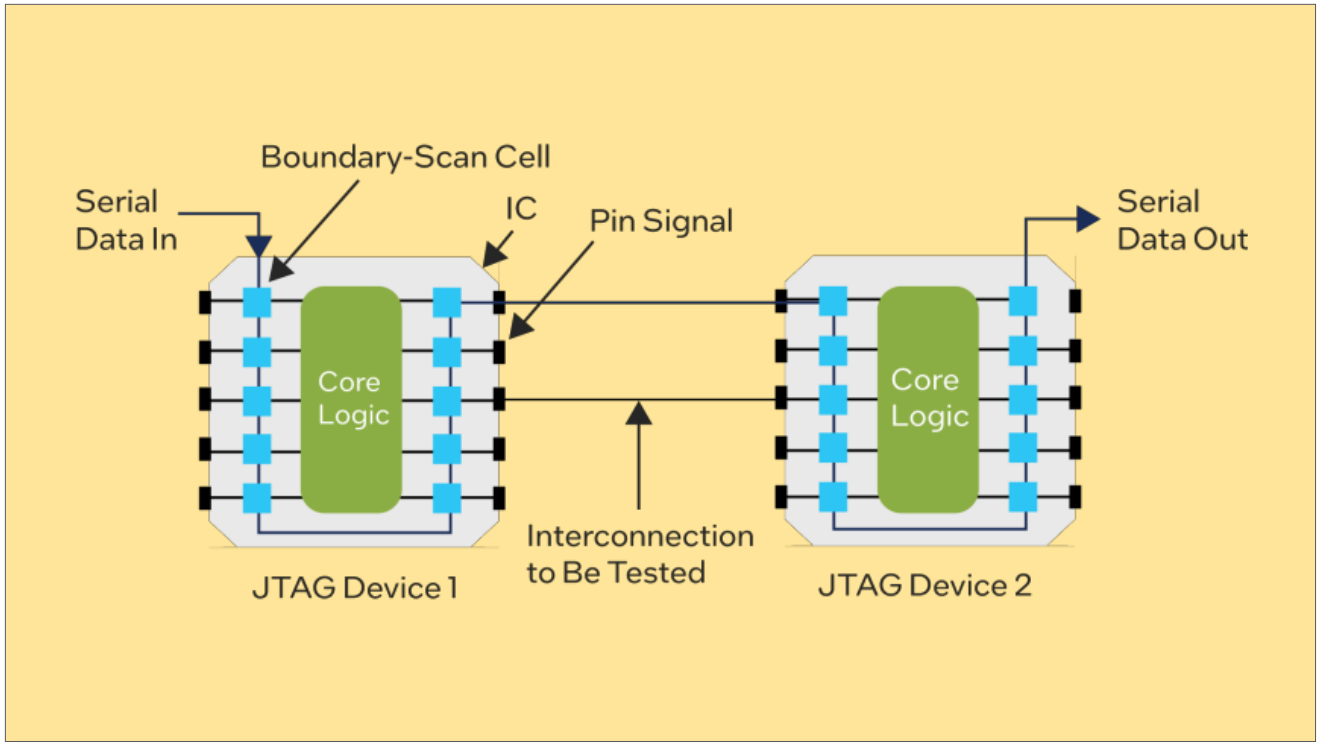
Multi-layer boards



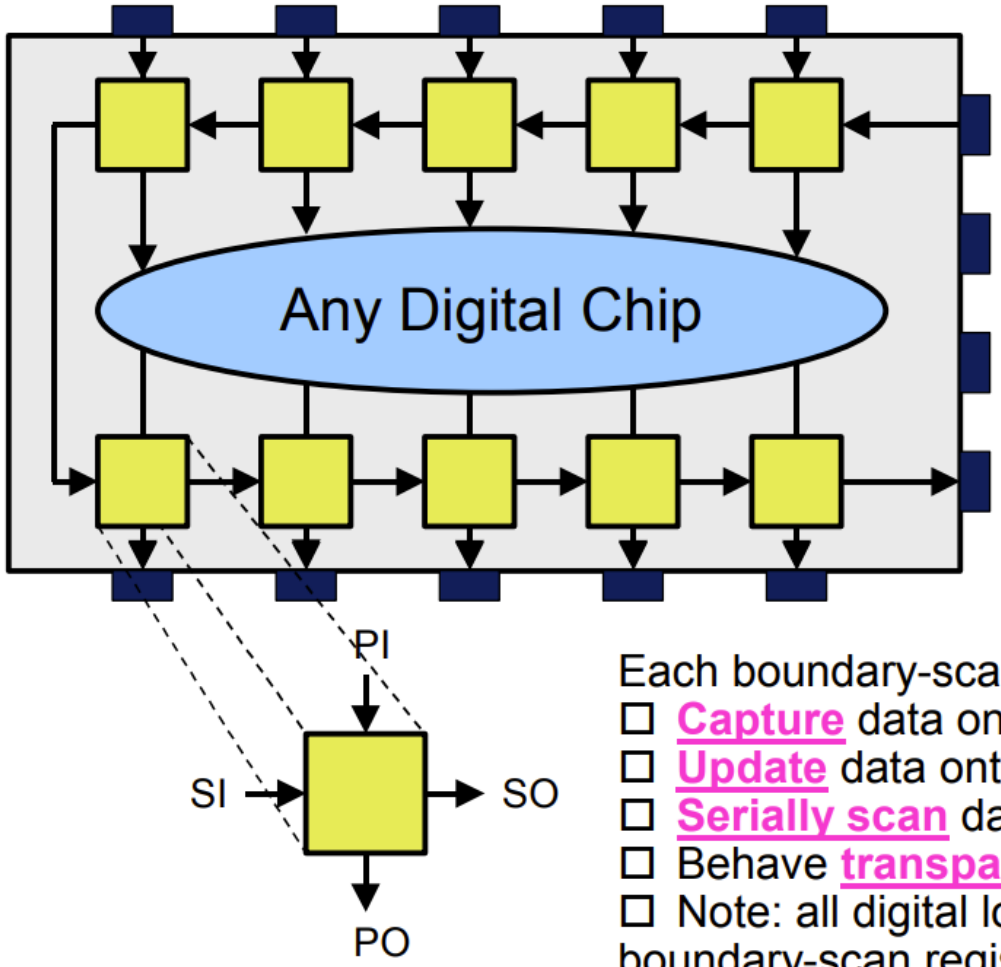
Boundary SCAN(BSCAN)

Boundary-scan is an integrated method for testing interconnects on printed circuit boards (PCBs) that are implemented at the integrated circuit (IC) level.

Boundary Scan is commonly referred to as JTAG and defined by the Institute of Electrical and Electronic Engineers (IEEE) 1149.1, which originally began as an integrated method for testing interconnects on printed circuit boards (PCBs) implemented at the integrated circuit (IC) level.



Principle of BSCAN



Each boundary-scan cell can:

- ☐ Capture data on its parallel input PI
- ☐ Update data onto its parallel output PO
- ☐ Serially scan data from SO to its neighbour's SI
- ☐ Behave transparently: PI passes to PO
- ☐ Note: all digital logic is contained inside the boundary-scan register

Boundary scan instructions

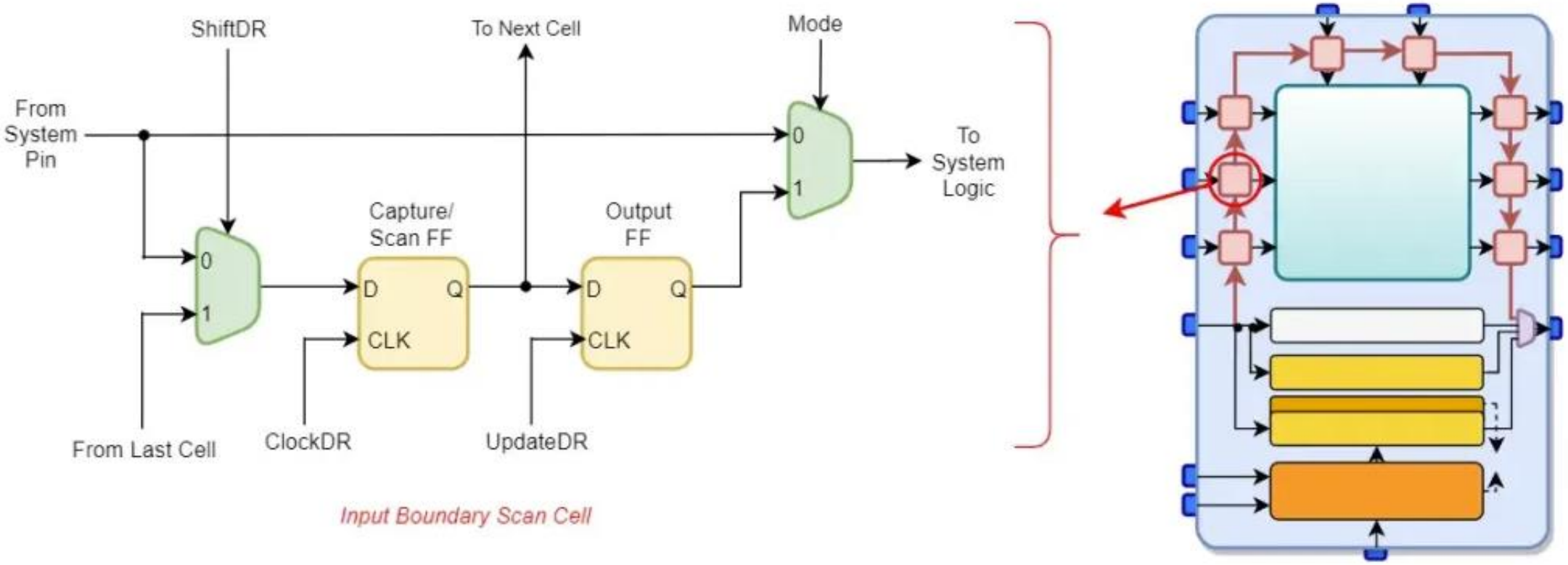
Mandatory BSCAN instructions

- **EXTEST**
  - During EXTEST instruction, the boundary scan cells associated with outputs are preloaded with test patterns to test downstream devices. The input boundary cells are set up to capture the input data for later analysis

- **SAMPLE/PRELOAD**
  - During SAMPLE/PRELOAD instruction, the boundary scan register can be accessed through a data scan operation, to take a sample of the functional data input/output of the device. Test data can also be preloaded into the boundary-scan register prior to loading an EXTEST instruction

- **BYPASS**
  - Using the BYPASS instruction, a device's boundary scan chain can be skipped, allowing the data to pass through the bypass register

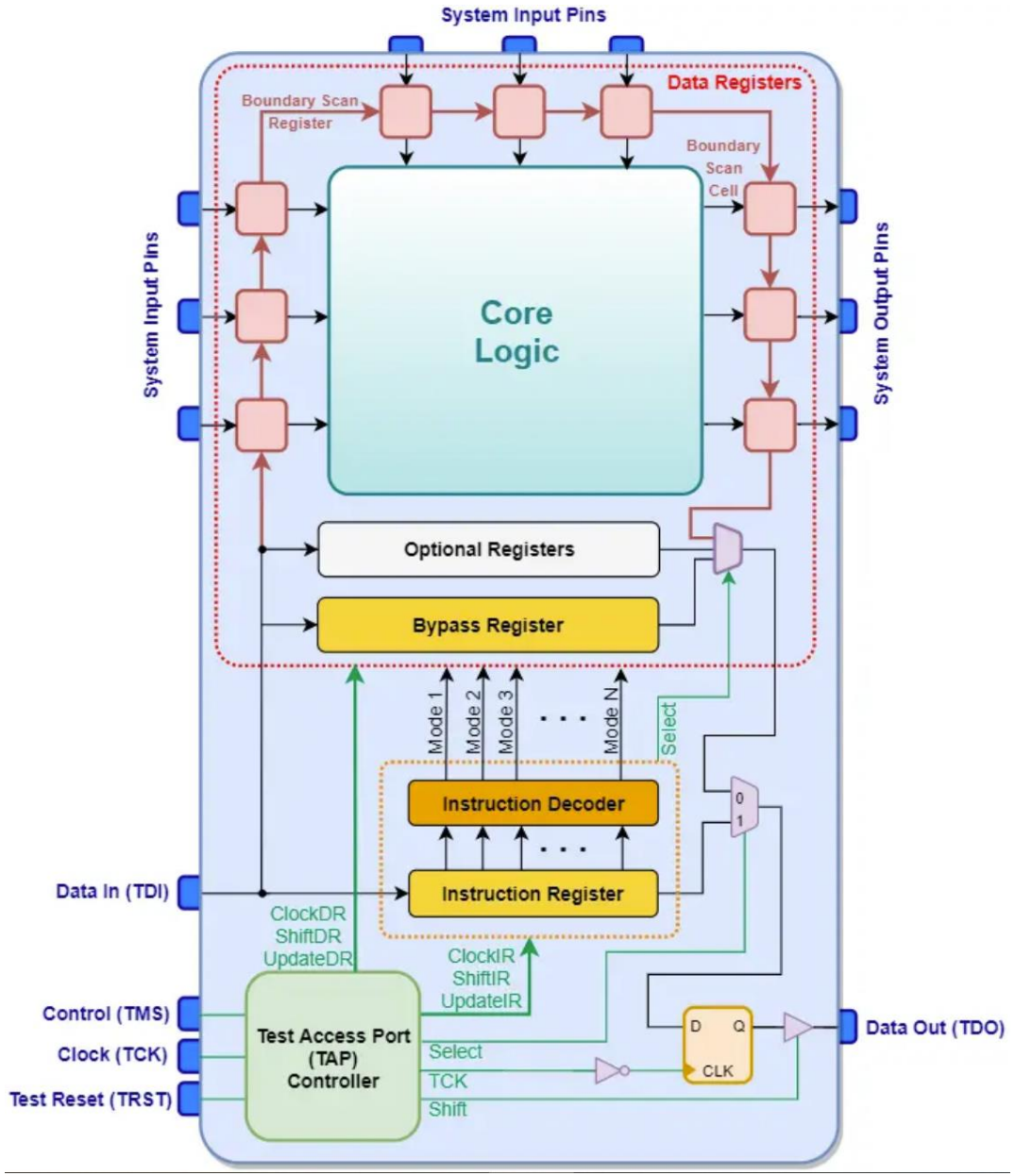
Boundary scan cell

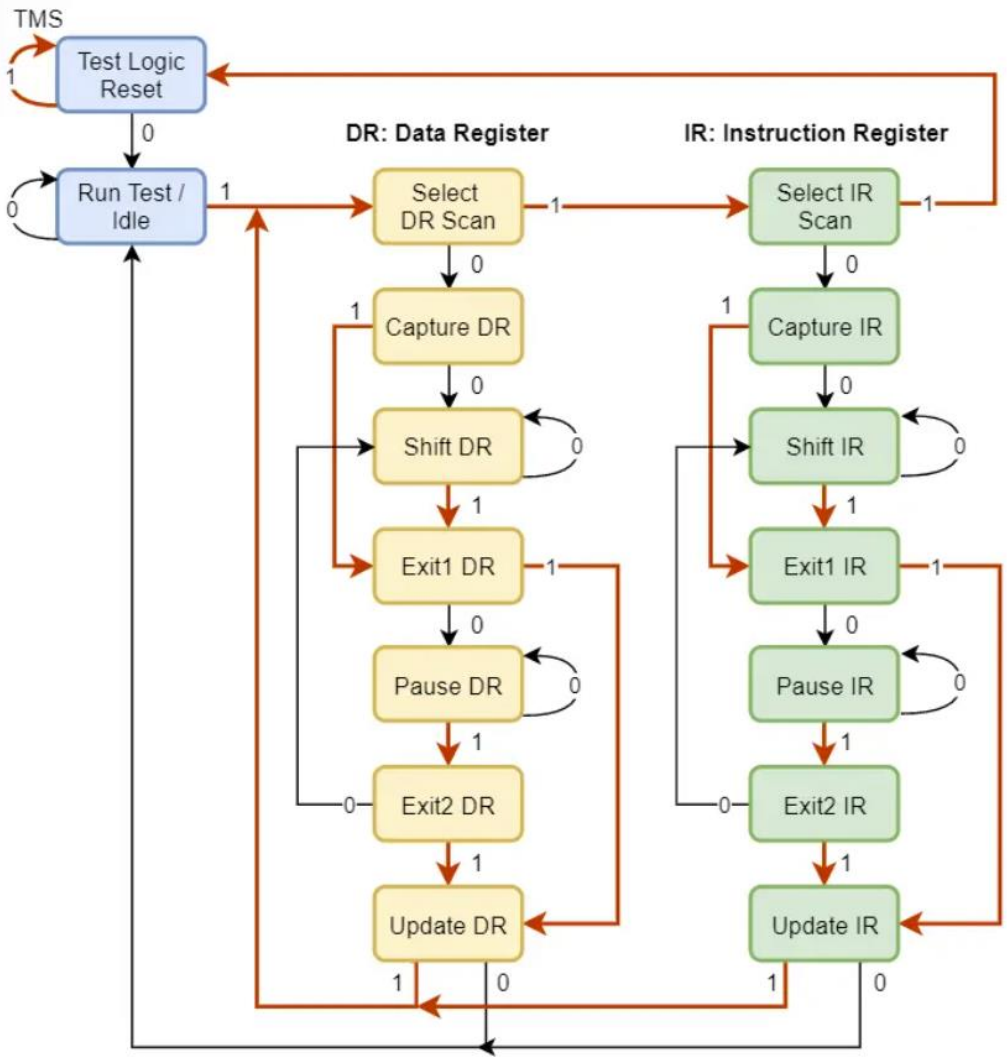


Input Boundary Scan Cell

arm

TAP controller state machine

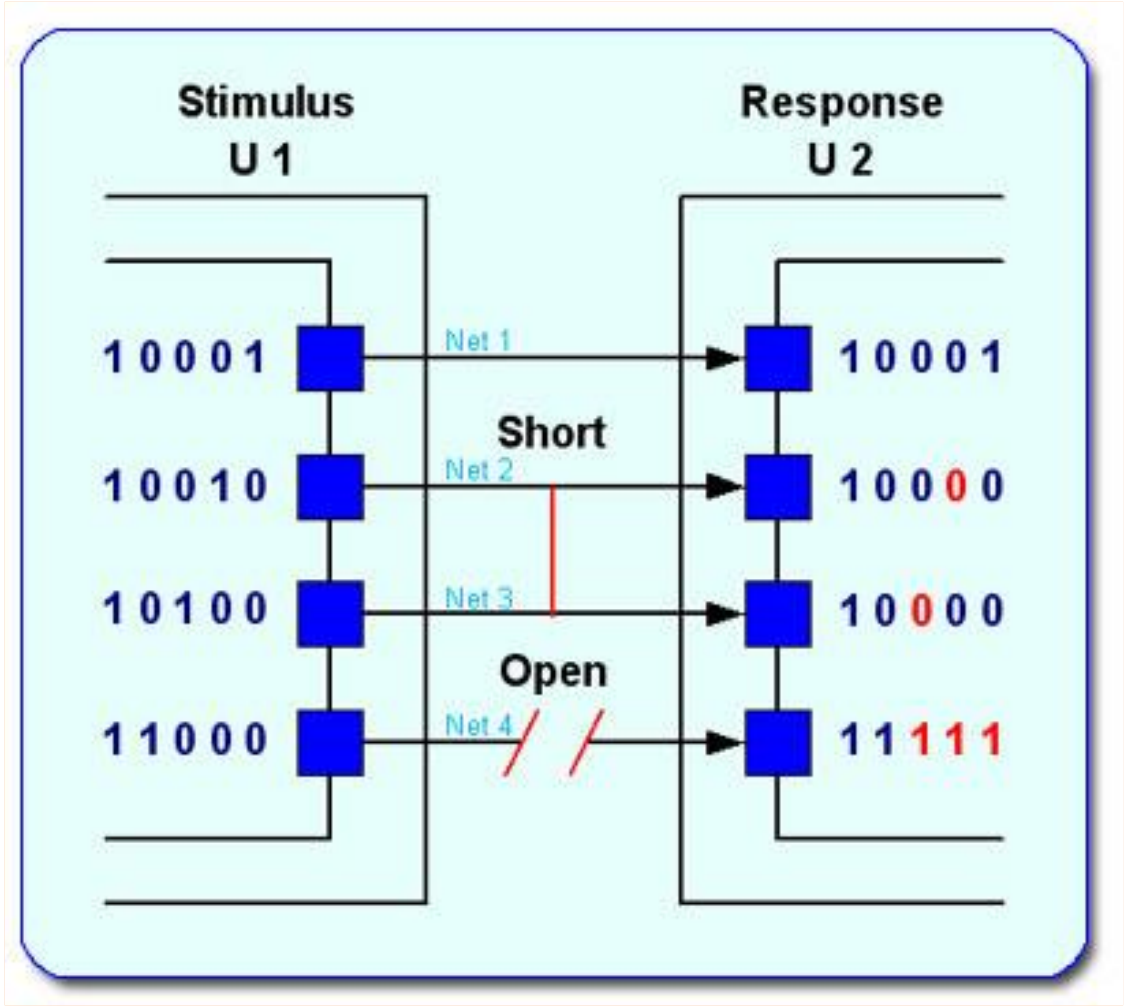




IR ≥ 2
Reset:
TMS = 1, 5 x TCK

arm

Boundary operation for EXTEST instruction



arm