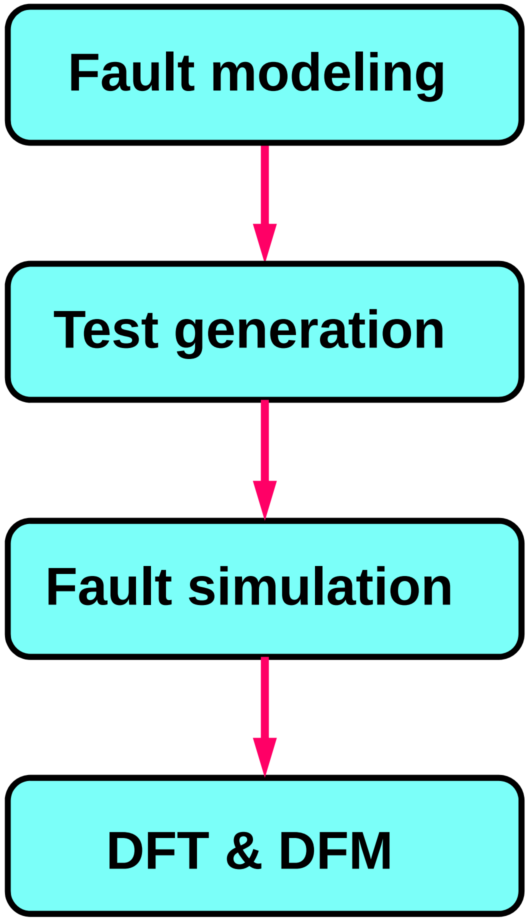
Chapter 2
RAM Functional Faults

Jin-Fu Li
Advanced Reliable Systems (ARES) Lab.
Dept. of Electrical Engineering
National Central University
Jhongli, Taiwan

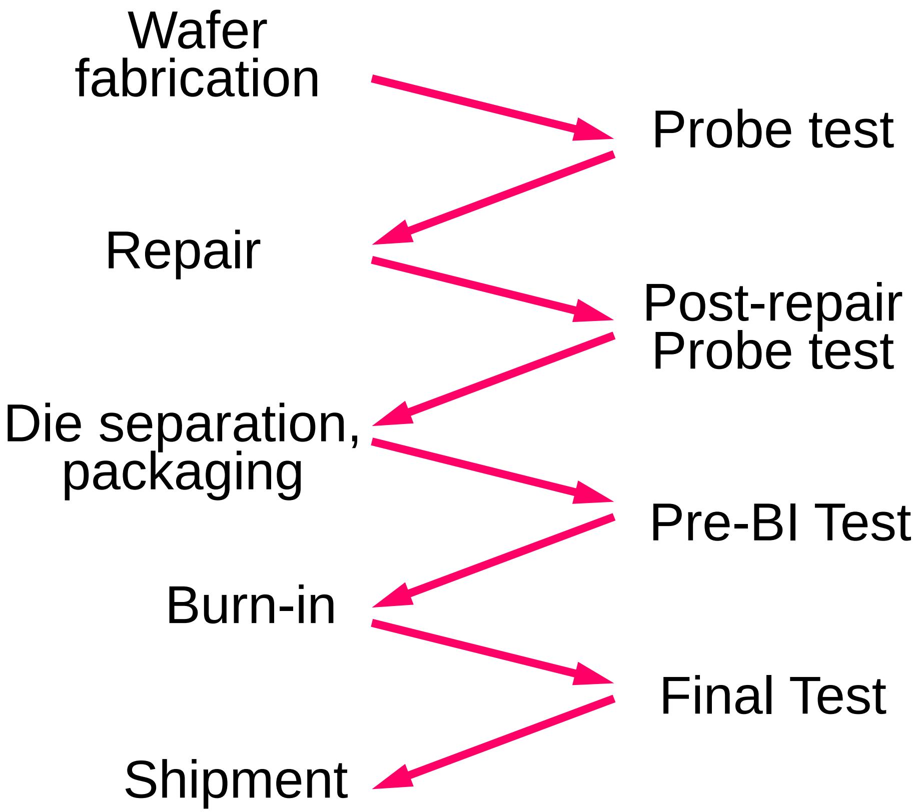
Outline

- Functional RAM Model
- Reducing the Functional Model
- Definition of Reduced Functional Faults
- Fault Combinations
- Fault Mapping

Memory Testing



Test-Preparation Phase



Test-Execution Phase

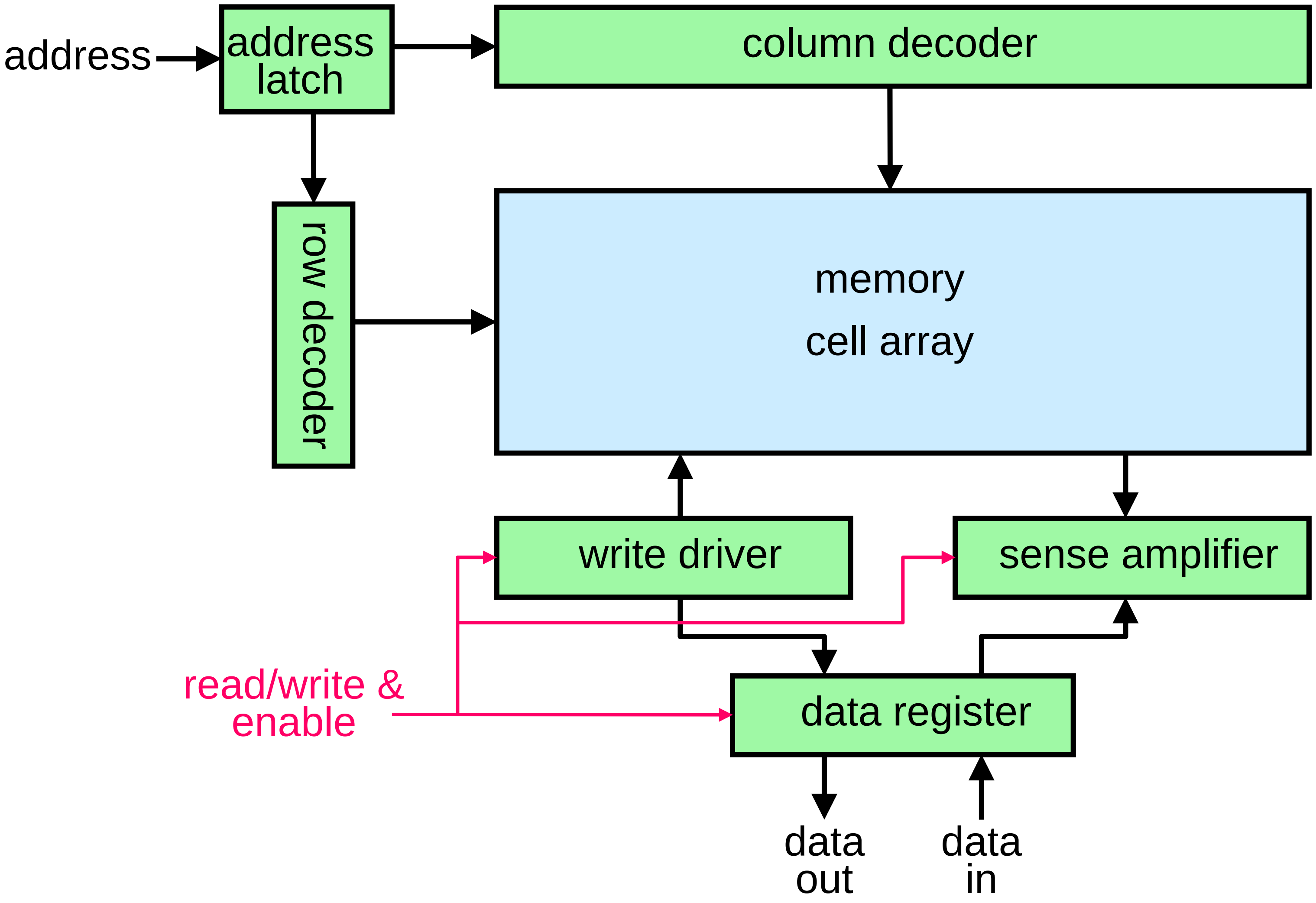
Memory Test Design

- A memory test consists of a series of measurements that applied by programmable automatic test equipment (ATE) or built-in test circuitry to unseparated dice or packaged ICs
- Tests can be divided into the following four categories
  - *DC parametric tests*
    - Measure static analog characteristics of the device’ s input/output interface
    - E.g., measure the leakage current, static and dynamic power supply current, output voltage level, and so on

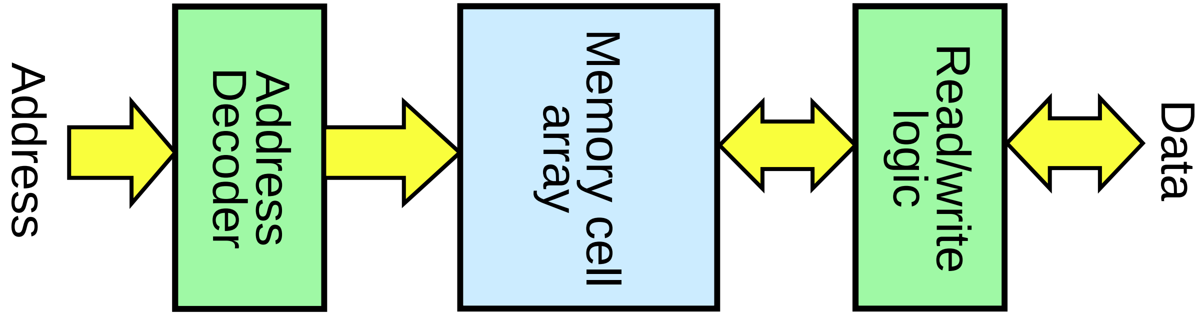
Memory Test Design

- *AC parametric tests*
  - ∗ Measure dynamic parameters of the device’ s input/output interface
  - ∗ E.g., measure setup time, hold time, rise time, fall, time, etc.
- *Functional tests*
  - ∗ Also called Boolean tests, verify whether or not the memory under test performs the correct logic functions
- *Dynamic tests*
  - ∗ Detect timing faults affecting internal device-under- test circuitry
  - ∗ E.g., detect slow sense amplifier operation, slow address decoder operation, etc.

Functional RAM Model



Reduced Functional Model



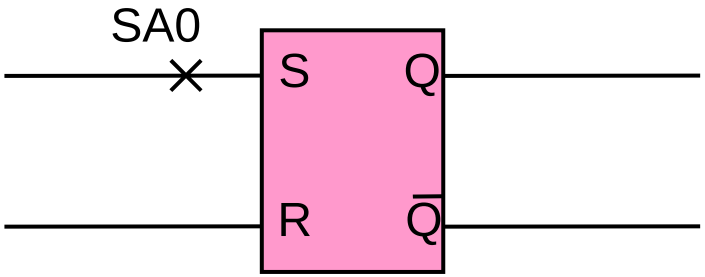
- The address latch, the row, and the column-decoder are combined to the address decoder
  - They all concern addressing the right cell or word
- The write driver, the sense amplifier, and the data register are combined to the read/write logic
  - They all concern the transport of data from and to the memory cell array

Reduced Functional Faults

- Stuck-at fault (SAF)
  - Definition:
    - The logic value of a stuck-at (SA) cell or line is always 0 or 1.
    - It is always in state 0 or in state 1 and cannot be changed to the opposite state
  - Detection requirement:
    - From each cell or line, a 0 or 1 must be read
- Transition fault (TF)
  - Definition:
    - A cell that fails to undergo a *0* to *1* transition when it is written is said to contain an *up transition fault*
    - A *down transition fault* indicates that a cell fails to undergo a *1* to *0* transition

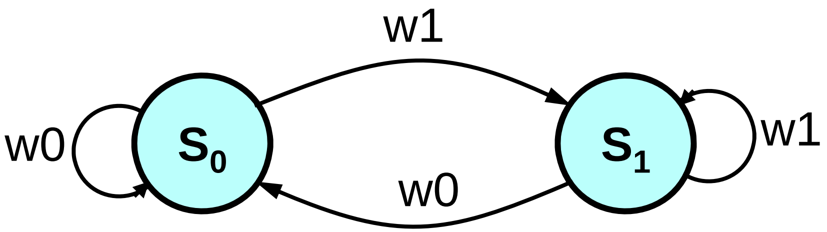
Reduced Functional Faults

- A TF can be thought of as a set/reset (S/R)-type flip-flop with a SAF on the set or reset input

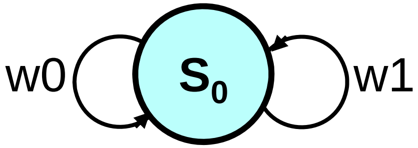


- Detection requirement:
  - ∗ Each cell should undergo up and down transitions and be read after each transition before undergoing further transitions

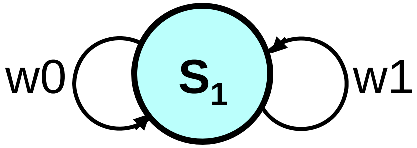
State Diagram for SAFs & TFs



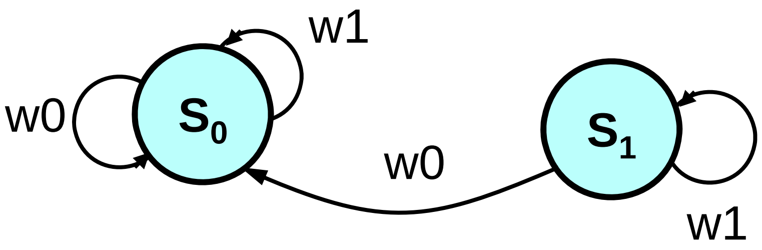
State diagram of a good cell



SA0 fault



SA1 fault

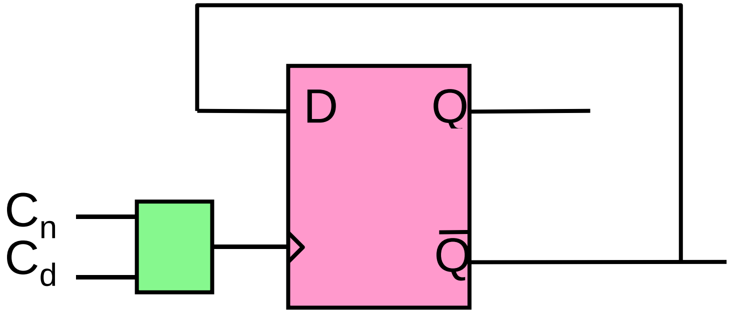


TFu

Reduced Functional Faults

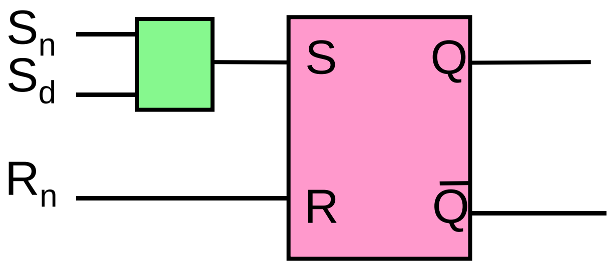
- Coupling faults (CFs)
  - 2-coupling faults
  - Different types of CFs
    - ∗ Inversion CF (CFin)
    - ∗ Idempotent CF (CFid)
    - ∗ State CF (CFst)
    - ∗ Dynamic CF (CFd)
- CFin
  - A 0 to 1 (or 1 to 0) transition in one cell inverts the content of a second cell
  - An CFin can be thought of as a D-type flip-flop with an extra clock input C_d and the Q' output tied to the D input, as depicted in the following figure.

Reduced Functional Faults



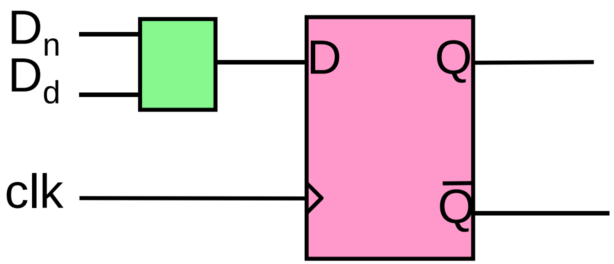
- CFid
  - A 0 to 1 (or 1 to 0) transition in one cell forces the content of a second cell to a certain value, 0 or 1.
  - An idempotent coupling fault can be thought of as an S/R-type flip-flop with an OR-gate in the Set or Reset line, as depicted in the following figure. S_n is the normal set input whereas S_d is the undesired set input due to coupling with one or more other flip flops.

Reduced Functional Faults



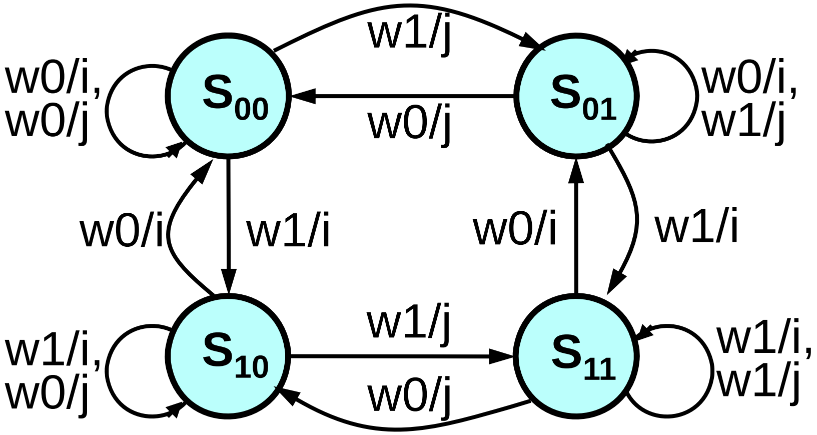
- CFst
  - A 0 or 1 state in one cell forces the content of a second cell to a certain value, 0 or 1.
  - It can be thought of as a D-type flip-flop with an OR/AND-gate in the data line (D), as depicted in the following figure. Dn is the normal set input whereas Dd is the undesired set input due to coupling with one or more other flip flops.

Reduced Functional Faults

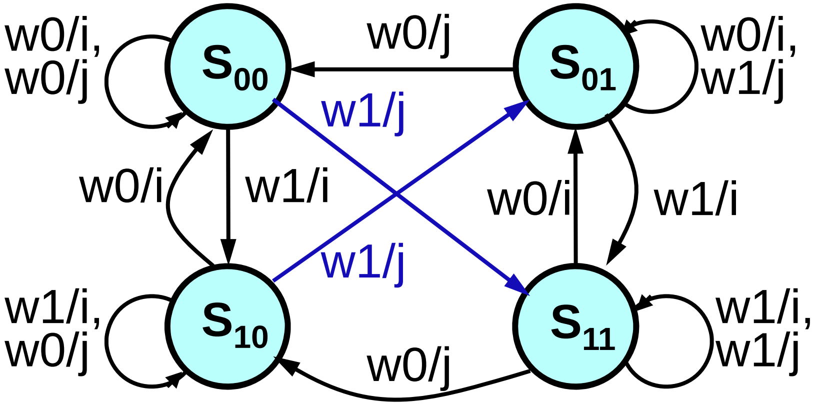


- Dynamic Coupling Fault (CF_d)
  - Occur between cells in different words. A *read* or *write* operation on one cell forces the content of the second cell either 0 or 1.
  - Denoted as <rx|wy; z> where | denotes the *or* of the read and write operations
  - Four possible CF_d faults
    - <r0|w0;0>, <r0|w0;1>,<r1|w1;0>, and <r1|w1;1>

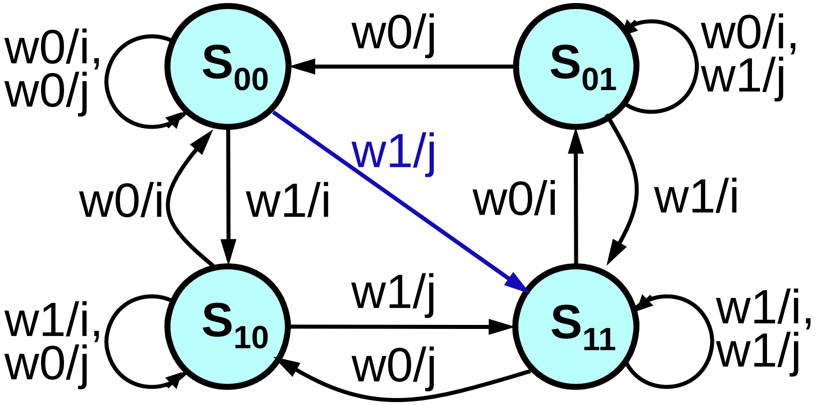
State Diagrams for CFin & CFid



State diagram of two good cells

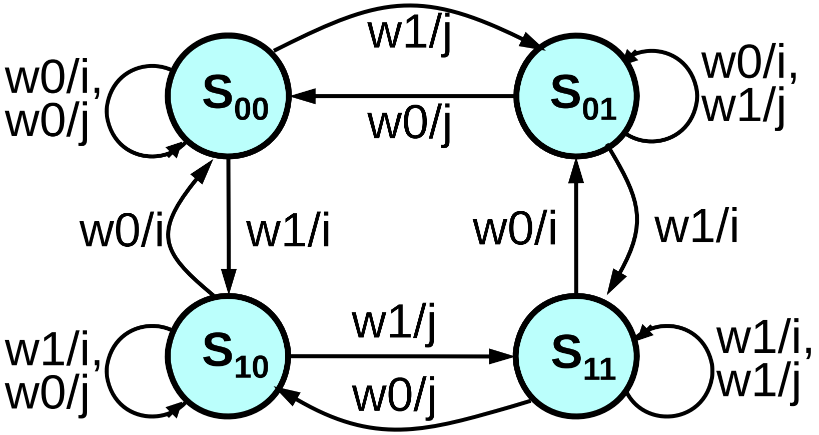


State diagram of an CFin<u;i>

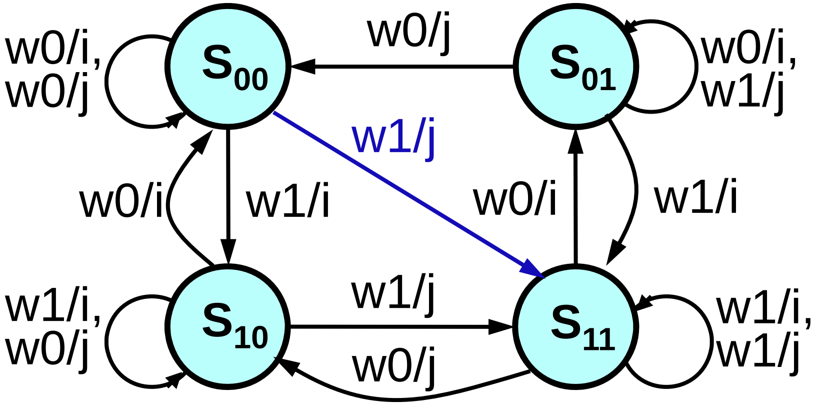


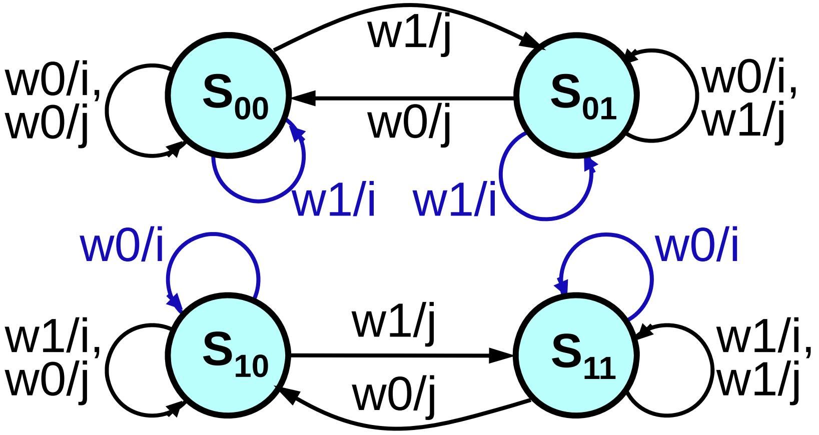
State diagram of an CFid<u;1>

State Diagrams for CFst & TF



State diagram of two good cells





State diagram of a CFst<1;1>

State diagram of TFu & TFd

Summary of CFs

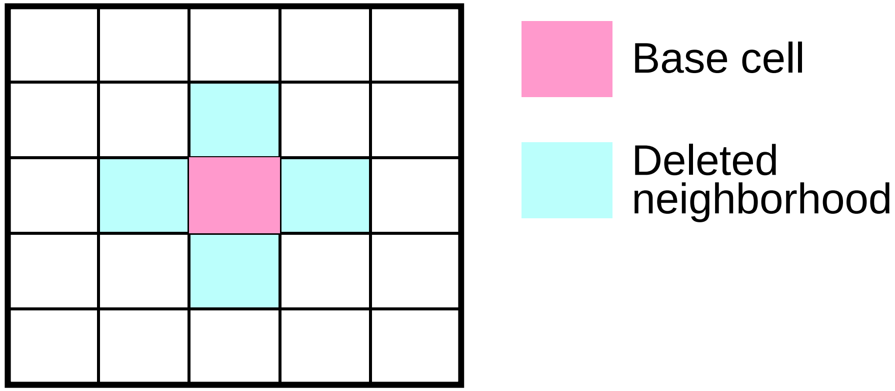
- Note that all definitions talk about single-way faults, that is, the presence of a CF from *cell i* to *cell j* does not imply the presence of a CF from *cell j* to *cell i*.
- Suppose that a transition or state in cell j can induce a coupling fault in cell i. Cell i is then said to be *coupled* cell (or *victim*); cell j is called the *coupling* cell (or *aggressor*).
- A test that has to detect and locate all coupling faults should satisfy
  - For all coupled cells, each cell should be read after a series of possible coupling faults may have occurred

NPSFs

- Neighborhood Pattern Sensitive Fault (NPSF)
  - A subset of the Pattern Sensitive Fault (PSF)
- PSF is defined as follows:
  - The contents of a cell is affected by the contents of a group of cells
- The PSF is the most general *k-coupling fault*, where *k=n* (all of the memory)
- In the PSF, the neighborhood could be anywhere in the memory array, whereas in a NPSF, the neighborhood must be in a single position surrounding the base cell

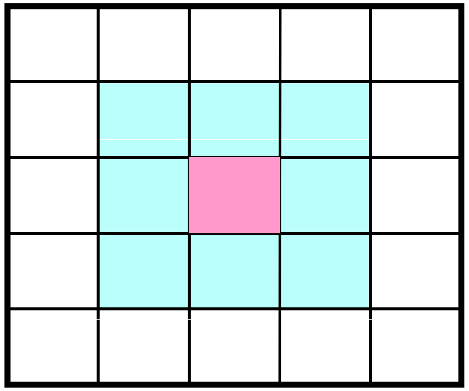
NPSFs

- Neighborhood Pattern Sensitive Fault (NPSF)
  - Type-1 neighborhood



		0		
	1	2	3	
		4		

  - Type-2 neighborhood



	0	1	2	
	3	4	5	
	6	7	8	

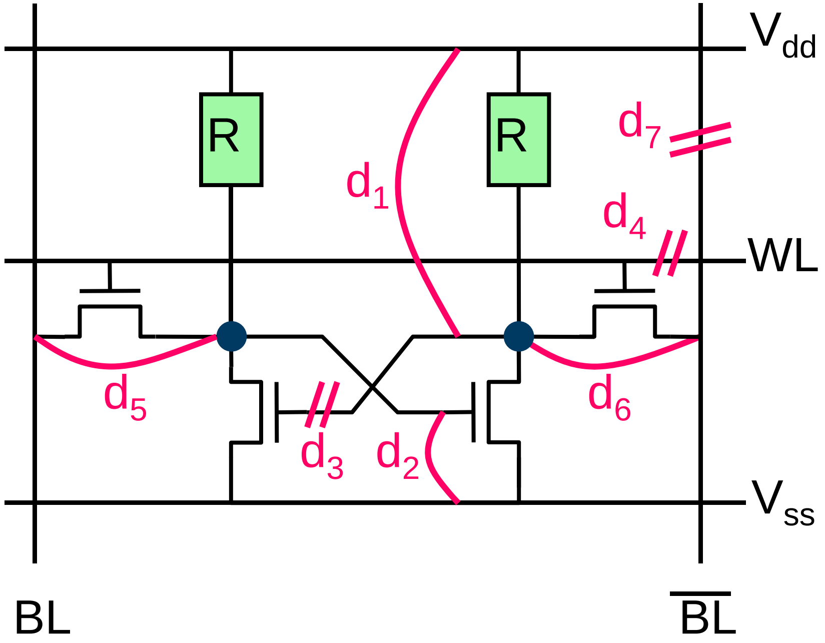
Types of NPSFs

- Three types of NPSFs
  - Active NPSF (ANPSF)
  - Passive NPSF (PNPSF)
  - Static NPSF (SNPSF)
- ANPSF
  - The base cell changes due to a change in the pattern of the deleted neighborhood
  - An ANPSF test has this necessary condition
    - Each base cell must be read in state 0 and state 1, for all possible deleted neighborhood pattern changes

Types of NPSFs

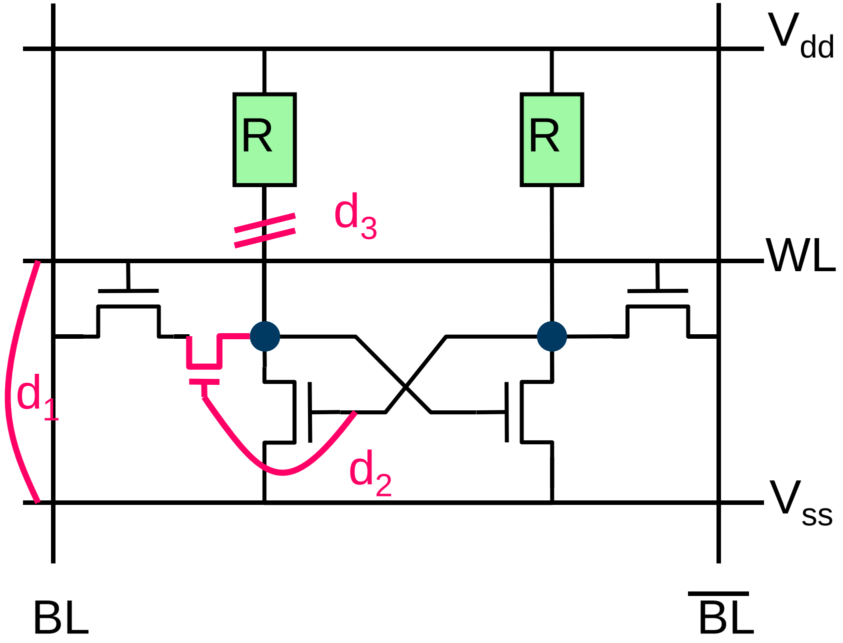
- PNPSF
  - A specific neighborhood pattern prevents the base cell from changing
  - The necessary condition to detect and locate a PNPSF
    - Each base cell must be written and read in state 0 and in state 1, for all deleted neighborhood pattern permutations
- SNPSF
  - The base cell is forced into a particular state when the deleted neighborhood contains a particular pattern
  - The necessary condition of test is
    - Each base cell must be read in state 0 and in state 1, for all deleted neighborhood pattern permutations

Relation Between Faults & Defects



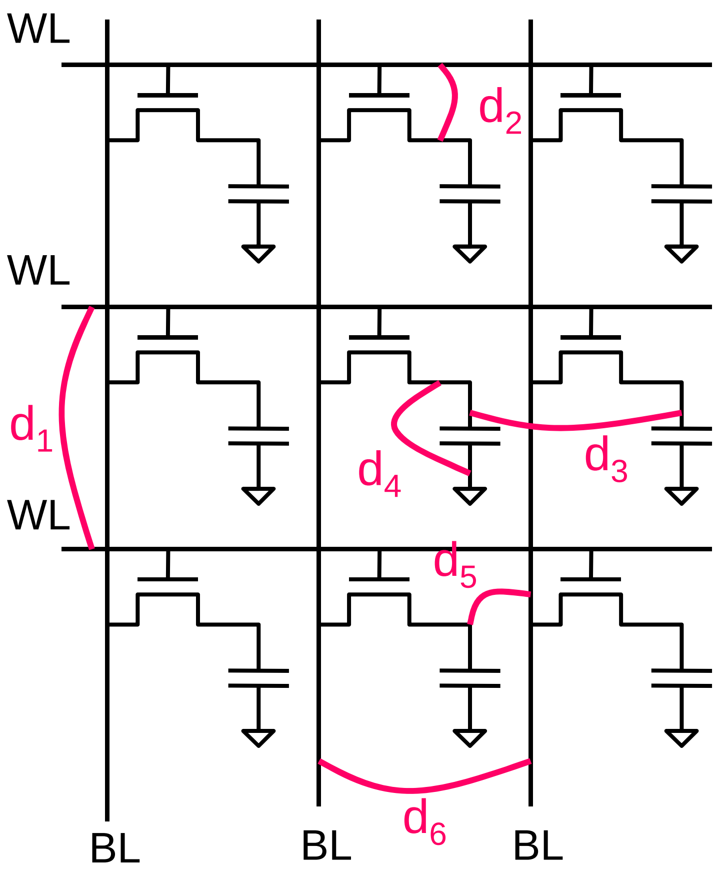
- Defect d_1 (inverse node shorted to V_{dd}) causes a SA0 fault
- Defect d_2 (true node shorted to V_{ss}) causes a SA0 fault
- Defect d_3 (open true node gate) cause a SA1 fault
- Defect d_4 (an open word line) causes all cells after the WL fault to be inaccessible (AF)
- Defect d_5 (a short between the true node and BL) will pull BL down if the cell contains a 0, but will not affect BL if the cell contains 1. This is the state coupling fault <0;0>
- Defect d_6 (short between inverse node and BL’) is similar, as is the state coupling fault <1;1>
- Defect d_7 (open BL’) prevents cells after the open defect from passing a logic 1 value on BL’

Relation Between Faults & Defects



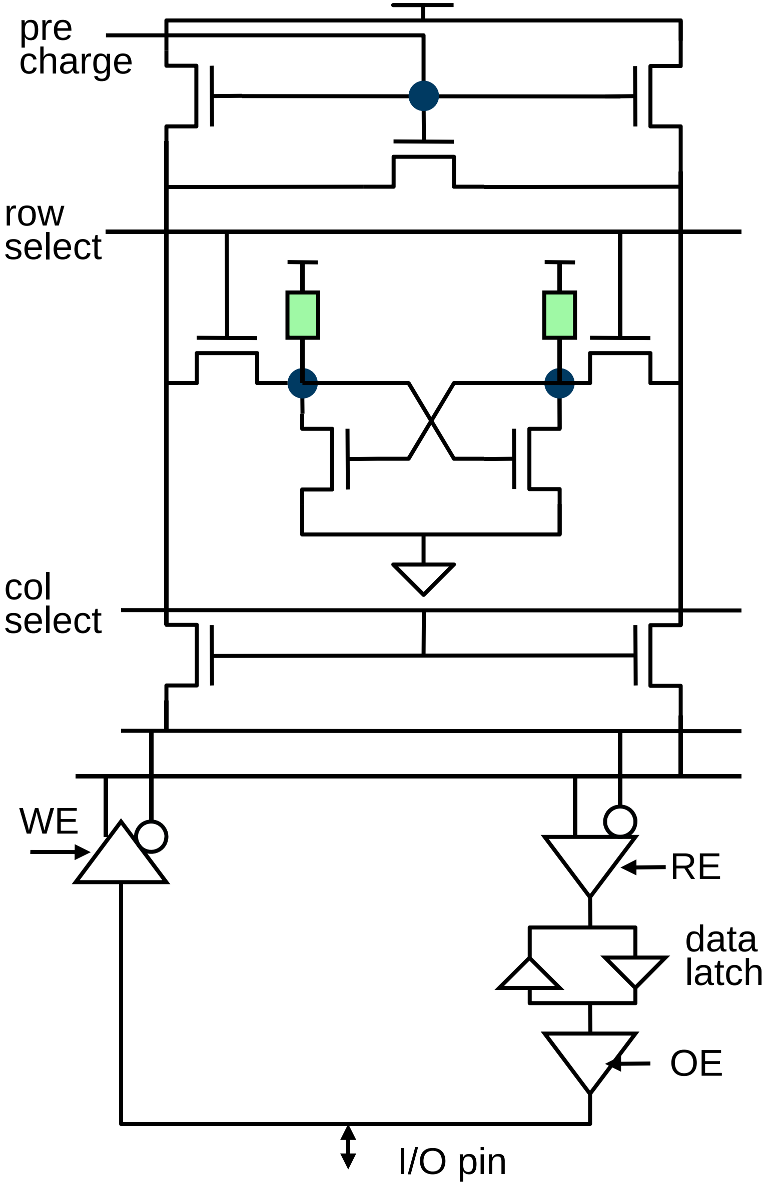
- Defect d_1 (a word line connected to V_{ss}) causes all cells in the word line to be *stuck open*.
- A defect in the poly silicon layer covering a diffusion region may result in the creation of an extra pass transistor. This defect (d_2) causes a *transition fault*.
- Defect d_3 (a broken pull up resistor) introduces a *data retention fault*. If the cell is not accessed, the cell node with the broken pull up resistor can be floating high or active low. If the node is floating high, the leakage current from the cell node to the substrate will decline the voltage at the node. If the node voltage passes the threshold voltage V_{th} the data in the cell will invert. If the node is active low, the cell will function correctly.

Relation Between Faults & Defects



- Defect d_1 (a short between two WLs) results in an AND bridging fault between pairs of cells located in the same column for the two shorted WLs
- Defect d_2 (a capacitor-WL short) causes a SA1 fault
- Defect d_3 (a short between two capacitors) causes a state coupling fault
- Defect d_4 (a shorted capacitor) is a SA0 fault
- Defect d_5 (a short between capacitor and BL) is an AND bridging fault with all cells in the same column
- Defect d_6 (a short between two neighboring BLs) causes an AND bridging fault between pairs of cells on the same word line and on the shorted bit lines

Design Dependent Conditions



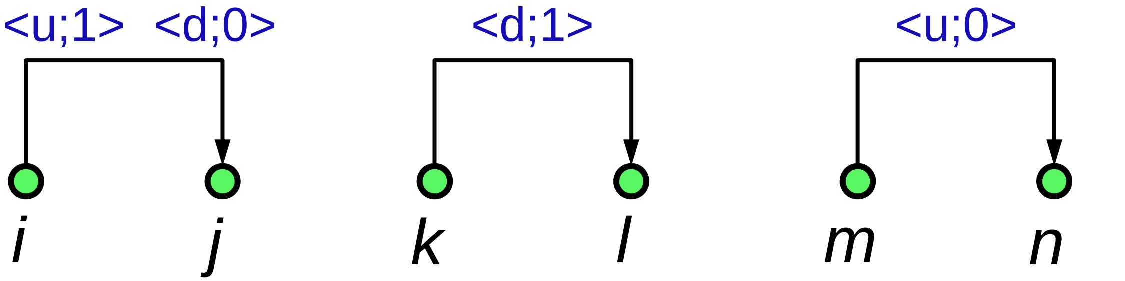
- Detection of all faults is only assured if the R/W logic is *transparent* for all faults in the cell array.
- Suppose a cell is SA0. When the cell is read, the sense amplifier in the R/W logic will interpret this as a logic 0. The fault is thus detected as SA0 at the output pin and the R/W logic is transparent for SA0 faults.
- Suppose a cell is stuck open, i.e., the cell is not accessible. When the cell is read, the sense amplifier in the R/W logic will detect that both bit lines are high (due to the preceded bit line precharge). It is not clear what value the sense amplifier will pass to the output pin. The transparency of the R/W logic for stuck-open faults depends on the design of the sense amplifier.

Fault Combinations

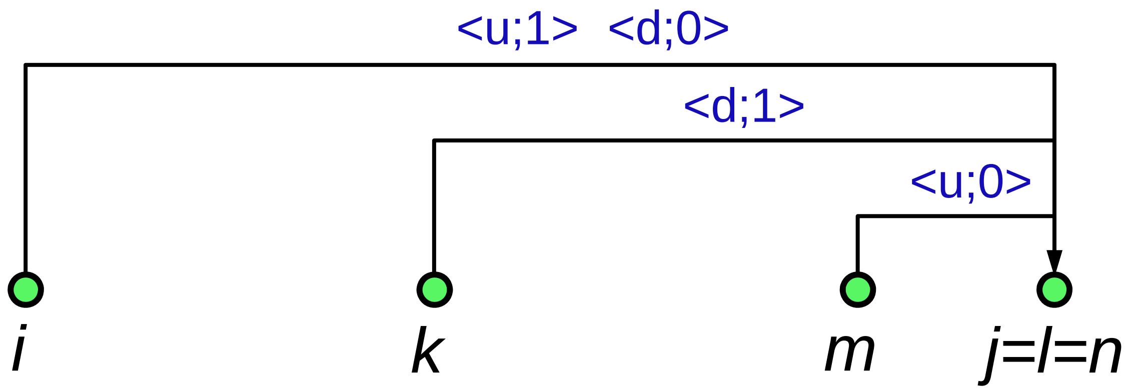
- The test engineer has assumed that a certain complicated functional fault (such as CF) is likely to occur
  - It is not only assumed that any number of faults of that particular type can occur but also that any number of faults of a more simple functional fault type (such as SAFs) can occur
- When multiple faults occur, these faults can be *linked* or *unlinked*
- Linked faults
  - A fault is linked when that fault influences the behavior of other faults
    - May cause the *fault masking* effect
- Unlinked faults
  - A fault is unlinked when that fault does not influence the behavior of other faults

Fault Combinations

- For example, suppose there are four CFs in a memory as in the following figure

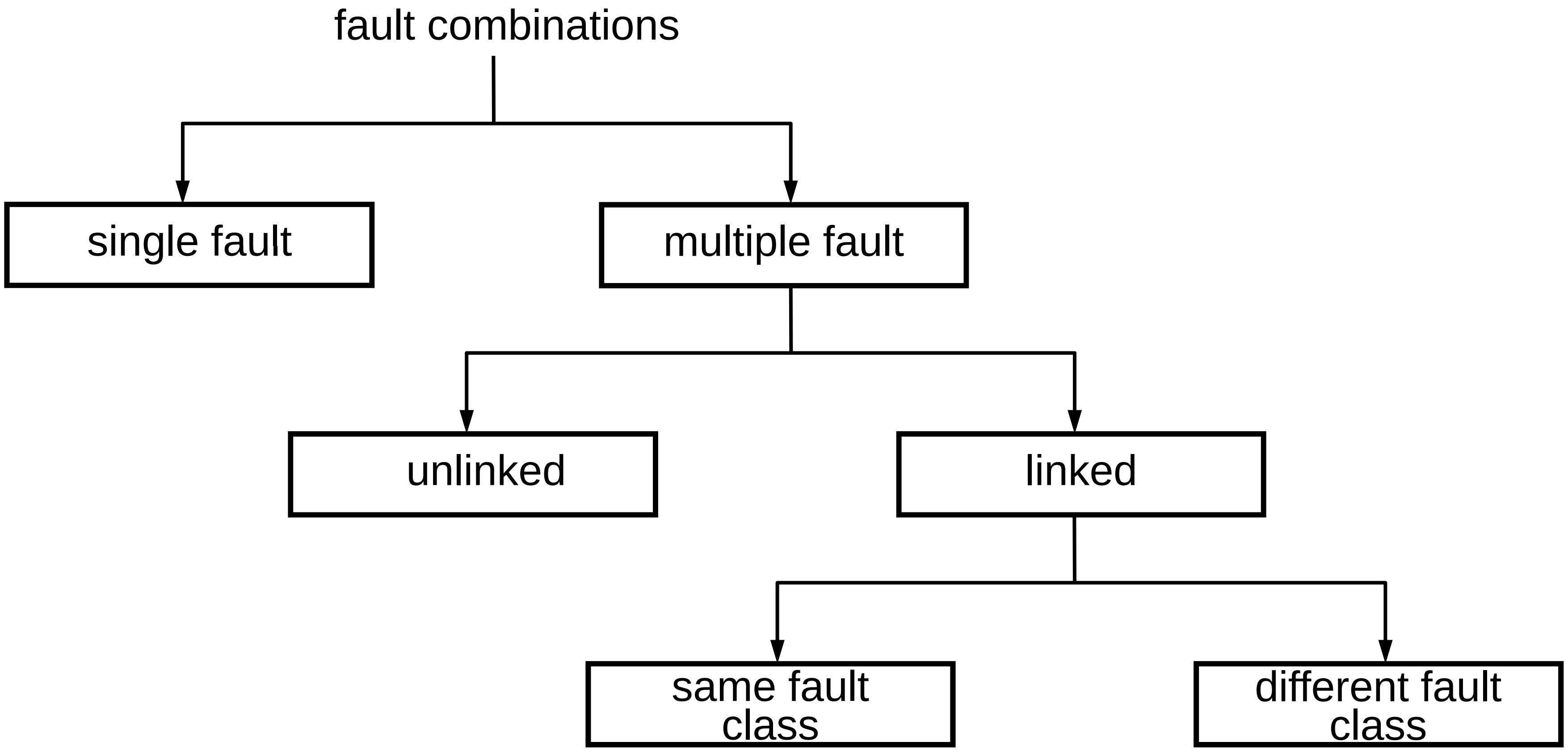


Unlinked faults



Linked faults

Fault Combinations

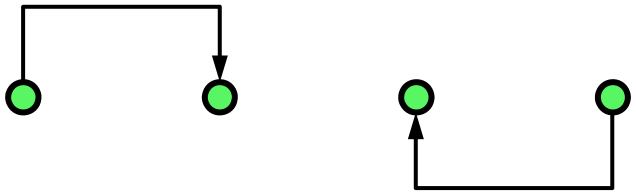


Linked Faults of the Same Class

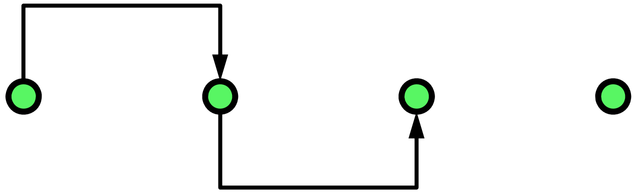
- SAF
  - A SAF involves only one cell, and only one SAF can occur in any one cell. Because only one cell is involved, a SAF is inherently unlinked.
- TF
  - A TF involves only one cell, and at most two different transition faults can occur in any one cell (TFu and TFd). Because only one cell, a TF cannot be linked to another TF.
- CF
  - With a CF (of the 2-coupling type) two cells per fault are involved. The case of multiple faults will be analyzed by considering two faults

Linked Faults of the Same Class

- These two faults can have the following relationships:
  - All four cells are different:

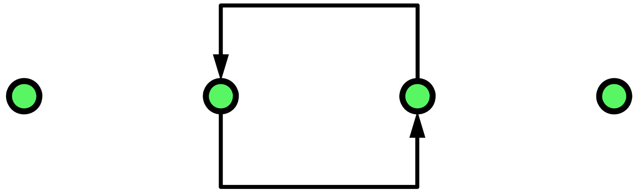


  - The coupling cell of one of the faults is the coupled cell of the other fault, whereas the other two cells are different

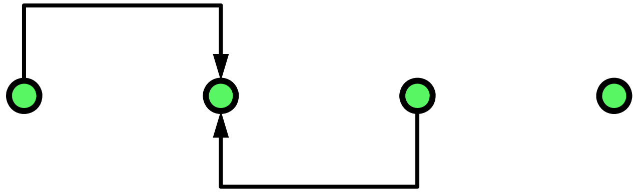


Linked Faults of the Same Class

- The coupling cell of the first fault is the coupled cell of the other fault, and, vice versa, the coupling cell of the second fault is the coupled cell of the first fault

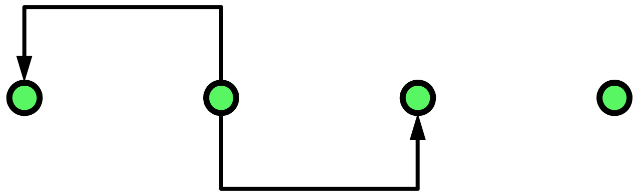


- The coupled cells are the same cell, whereas the coupling cells are different

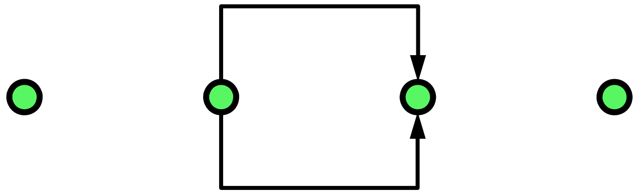


Linked Faults of the Same Class

- The coupling cells are the same cell, whereas the coupled cells are different



- The coupled cells are the same cell, and the coupling cells are the same cell



Fault Mapping

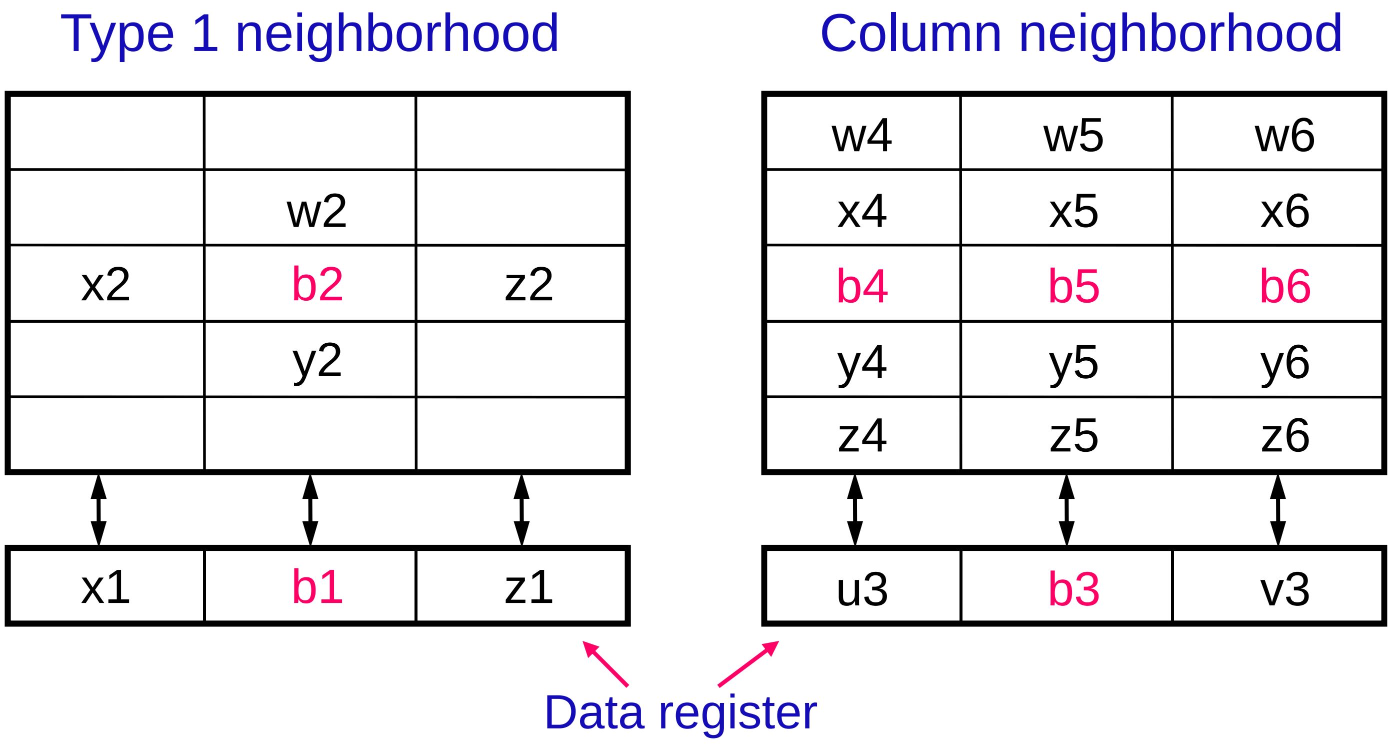
- A reduced functional model has been defined consisting of three blocks:
  - The address decoder, the memory cell array, and the read/write logic
- When one wants to test a memory for SAFs, three tests would be necessary
  - One test that detects SAFs in the address decoder
  - One test that detects SAFs in the memory cell array
  - One test that detects SAFs in the read/write logic
- *Most faults occurring in the read/write logic and the address decoder can be mapped to faults in the memory cell array*

Fault Mapping

- Mapping read/write logic faults into memory array faults
  - A test that detects SAFs in the memory array will also detect SAFs in the read/write logic
    - A SAF in the read/write logic will appear as a large group of cells stuck-at faults
  - The same arguments are valid for TFs and CFs
- For NPSFs, this cannot be proven generally
  - A test that detects NPSFs in the memory cell array will also detect NPSFs in the read/write logic under the condition
    - Each neighborhood in the *data register* is a *subset* of a neighborhood in the cells of the memory cell array

Fault Mapping

- For example
  - A test for defecting Type 1 NPSFs can also detect the neighborhood faults in data register
  - A test for detecting column neighborhood faults can not detect the neighborhood faults in the data register



Type 1 neighborhood

	w2	
x2	b2	z2
	y2	

x1	b1	z1

Column neighborhood

w4	w5	w6
x4	x5	x6
b4	b5	b6
y4	y5	y6
z4	z5	z6

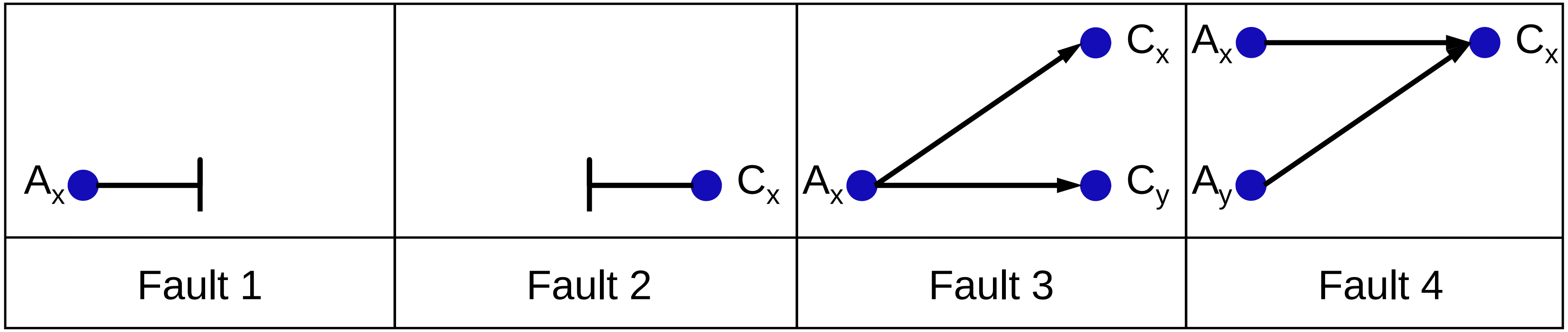
u3	b3	v3

Data register

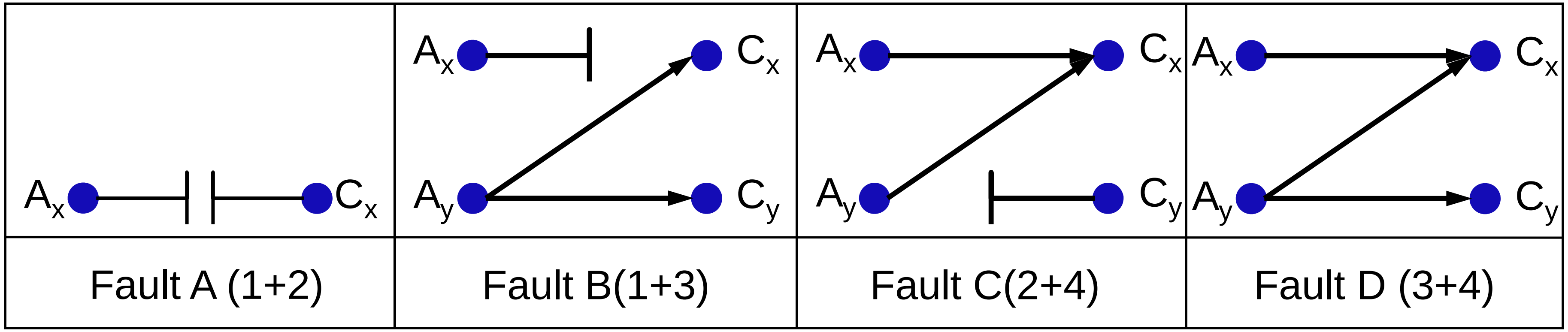
Address Decoder Faults

- An address decoder fault (AF) represents an address decoding error
- Assumptions
  - The decoder logic does not become sequential
  - The fault is the same during both read and write operations
- AFs can be classified into four cases
  - Fault 1: no cell is accessed for a certain address
  - Fault 2: no address can access a certain cell
  - Fault 3: with a particular address, multiple cells are simultaneously accessed
  - Fault 4: a particular cell can be accessed with multiple addresses

Address Decoder Faults



Address decoder faults

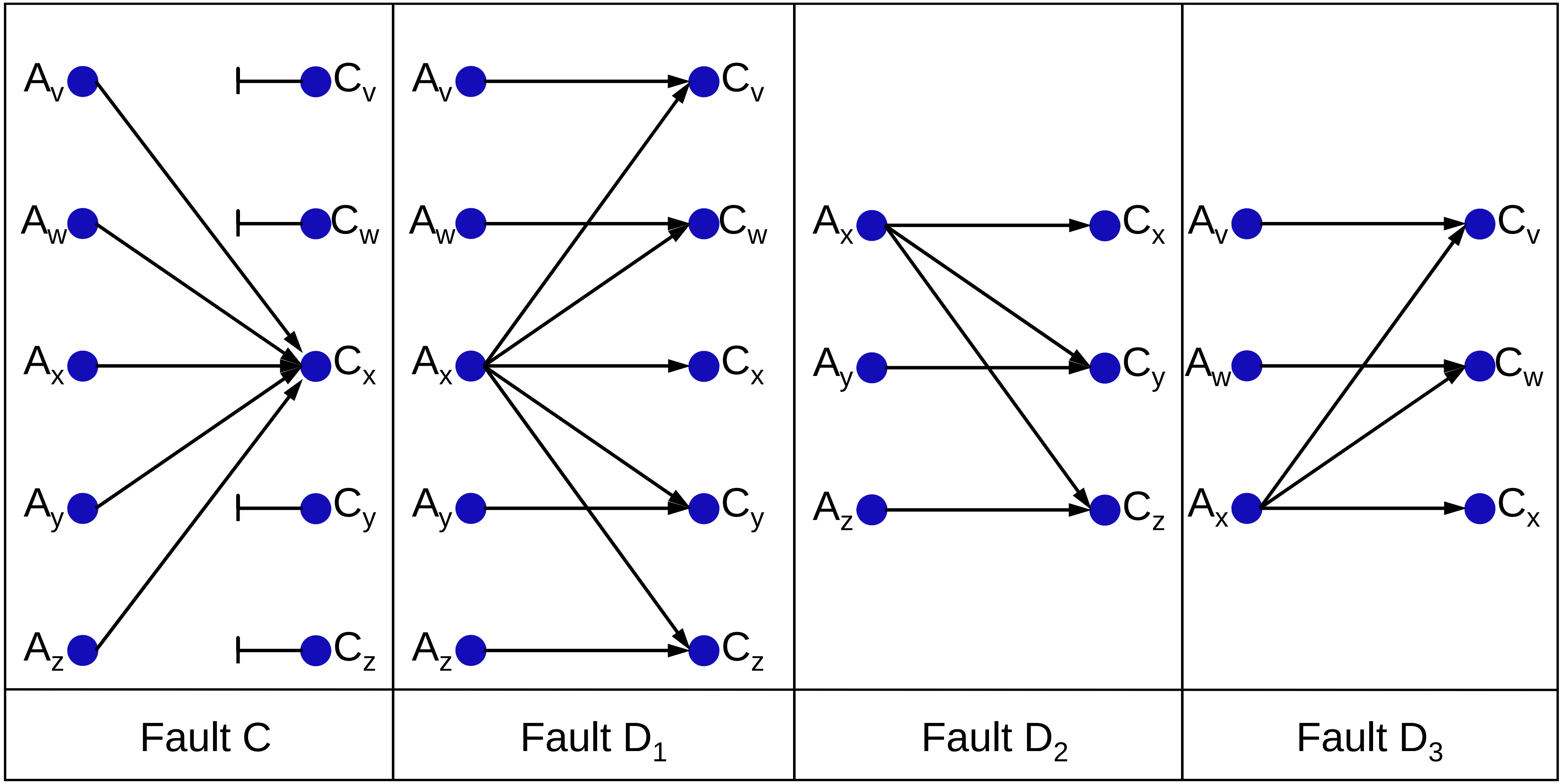


Combinations of AFs that must be tested

Address Decoder Faults

- Fault A & fault B are inherently unlinked
  - It is impossible to mask the fault occurring when one reads A_x
- Fault C & fault D may be linked
  - It is possible that writing in A_x masks the fault that occurred when C_x was erroneously written through A_y
- Therefore, fault C & fault D are extended to the general case, with more than two addresses
  - See the figure shown in the next page

Address Decoder Faults



Note that any number of cells and addresses with the same fault can be inserted between v and w and between y and z.

Address Decoder Faults

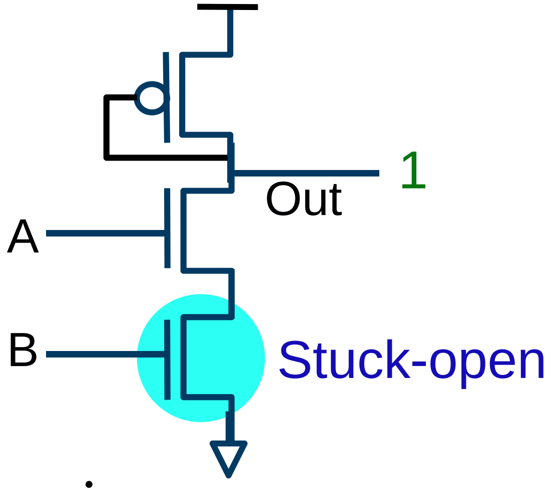
- If multiple cells are accessed simultaneously with a single address, as in fault D, a read operation can yield one of the following two results
  - All addressed cells have the same contents
  - Not all addressed cells have the same contents Two cases can be distinguished
    - The memory returns a value that is a deterministic function of the contents of all addresses cells
    - The memory returns a random or pseudorandom result
- In general, it is safest for AF tests to expect the memory to return a random value, except for the case in which all cells contain the same value
  - The tests should be designed such that they are not influenced by random results

Specific Address Decoder Fault

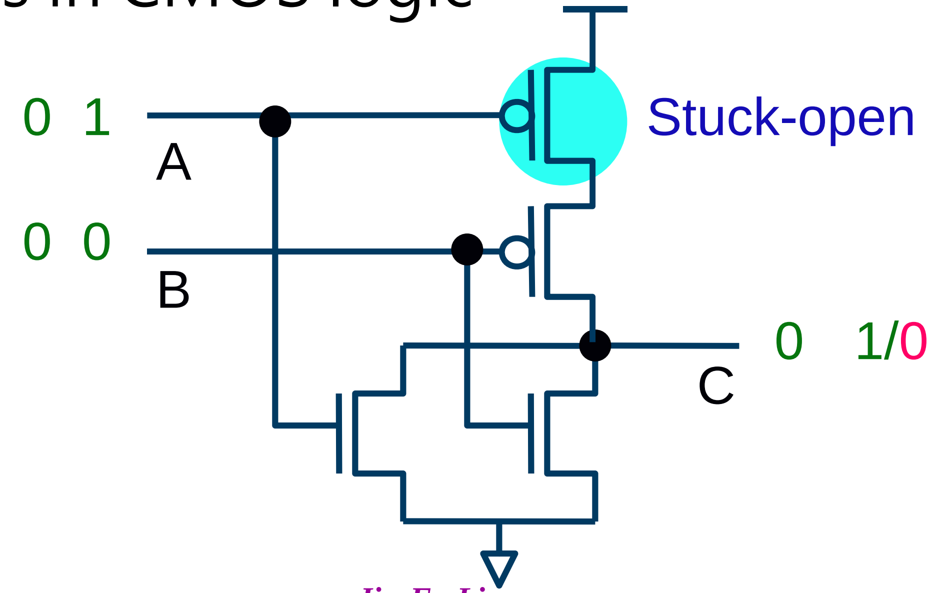
- The address decoder can be implemented with different technologies
  - NMOS, CMOS, etc.
- Open defects in NMOS logic can be mapped to SAFs
- Only a subset of open defects in CMOS logic can be mapped to SAFs
- A subset of open defects in CMOS logic causes sequential behavior in logic circuit
  - These defects in address decoders can not be detected by conventional tests

Specific Address Decoder Fault

- Open defects in NMOS logic

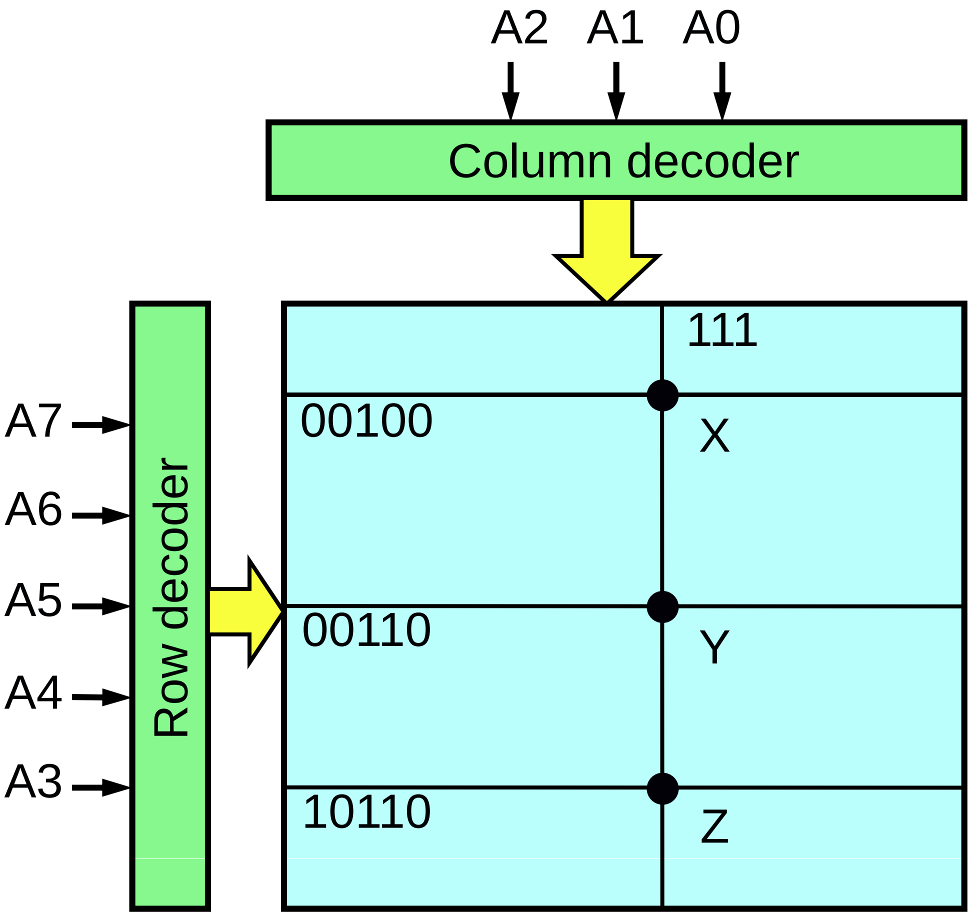


- Open defects in CMOS logic



Specific Address Decoder Fault

- A graphical representation of the open defects causing sequential fault behaviors



Let the PMOS connected to A7 has a stuck-open defect

1. Write address Z(10110 111) with logic 1
2. Write address X(00100 111) with logic 0
3. Read address Z: result is logic 1, that is correct
4. Write address Y(00110 111) with logic 0
5. Read address Z: result is logic 0, that is wrong!

Specific Faults for Multi-Port RAMs

- The memory cell array faults can be detected as the single-port RAMs
- Multi-port RAM specific faults should be covered
  - *Inter-port faults*
  - *Intra-port faults*
- To reduce the test complexity, usually, the testing of inter-port and intra-port faults is based on the following assumption
  - Physical layout is known

References

- [1] A. J. van de Goor and C. A. Verruijt," An Overview of Deterministic Functional RAM Chip Testing" , ACM Computing Surveys, vol. 22, no. 1, March 1990.
- [2] A. K. Sharma," Semiconductor Memories – Technology, Testing, and Reliability" , IEEE PRESS, 1997.
- [3] R. Dekker, F. Beenker, and L. Thijssen," Fault Modeling and Test Algorithm Development for Static Random Access Memories" , Int. Test Conf., pp. 343-352, 1988.
- [4] M. Sachdev," Defect Oriented Testing for CMOS Analog and Digital Circuits" , Kluwer Academic, 1998.
- [5] B. F. Cockburn," Tutorial on Semiconductor Memory Testing" , JETTA, pp. 321-336, 1994.