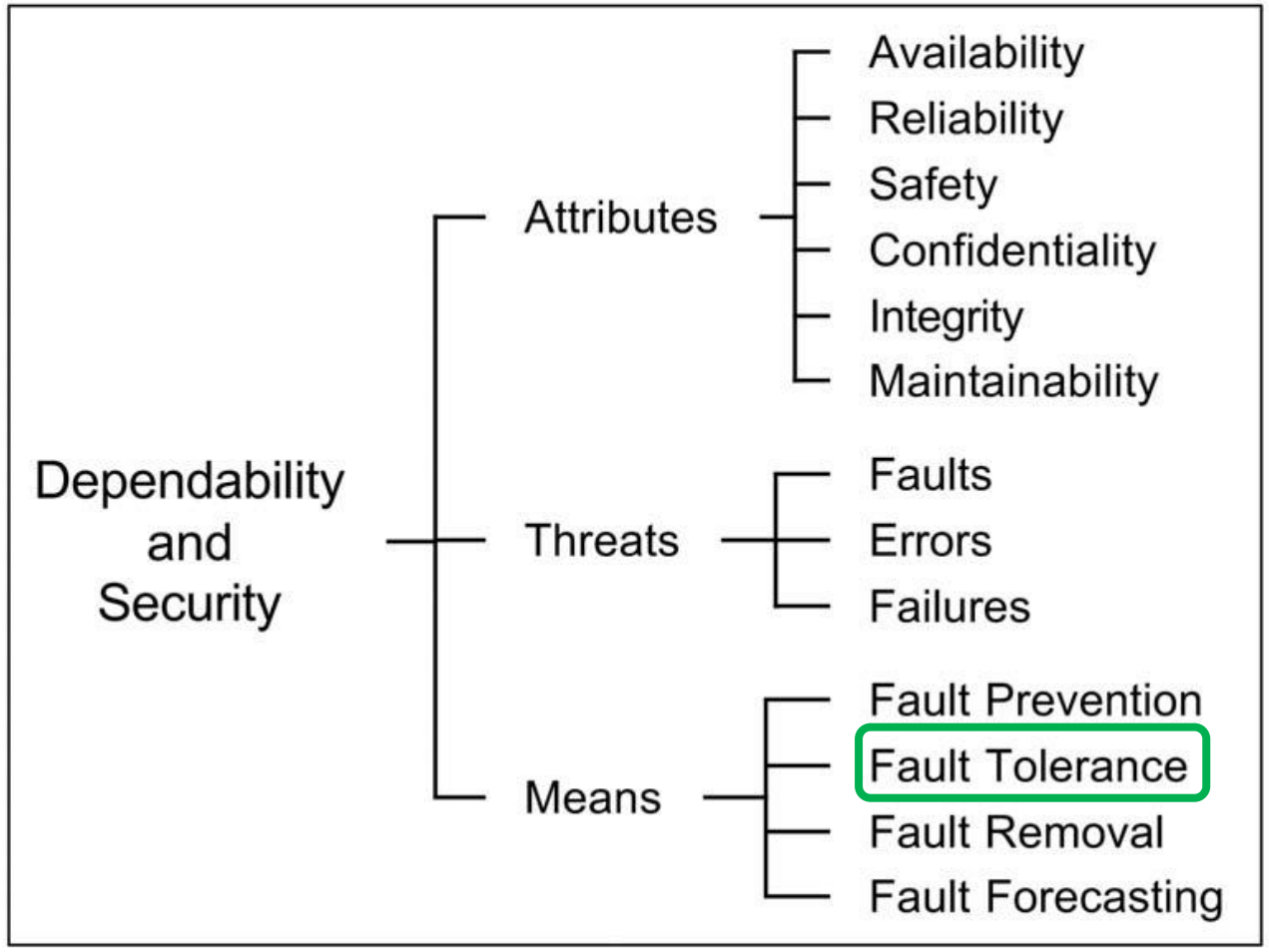
Means for dependability

A. Avizienis, J.C. Laprie, B. Randell, C. Landwehr
Basic Concepts and Taxonomy of Dependable and Secure Computing
IEEE Transactions on Dependable and Secure Computing, Vol. 1, N. 1, 2004

Dependability tree



From [Avizienis et al., 2004]

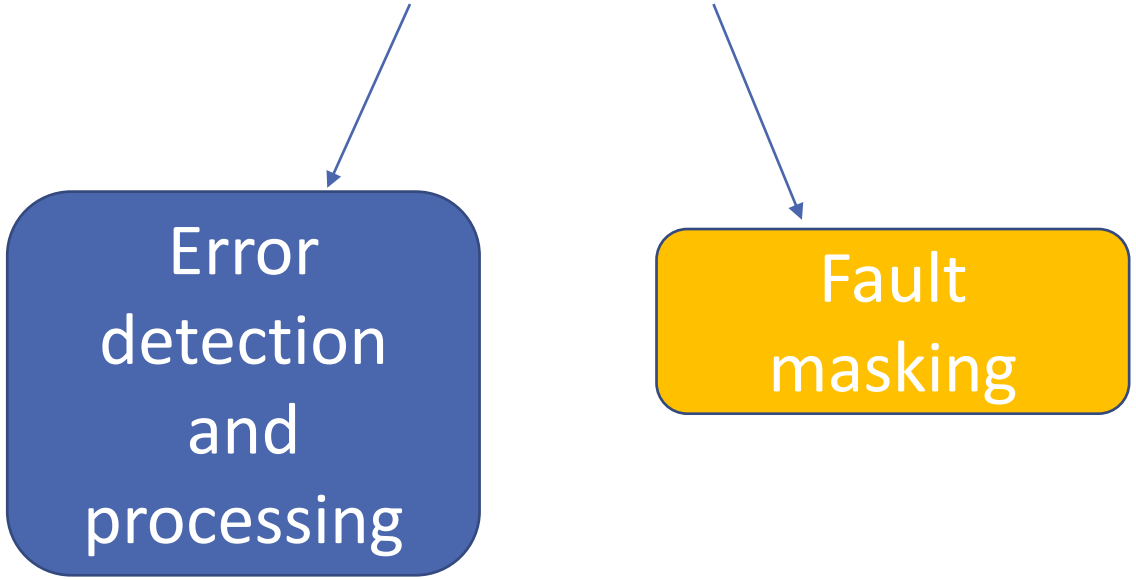
Fault tolerance techniques

Fault Tolerance

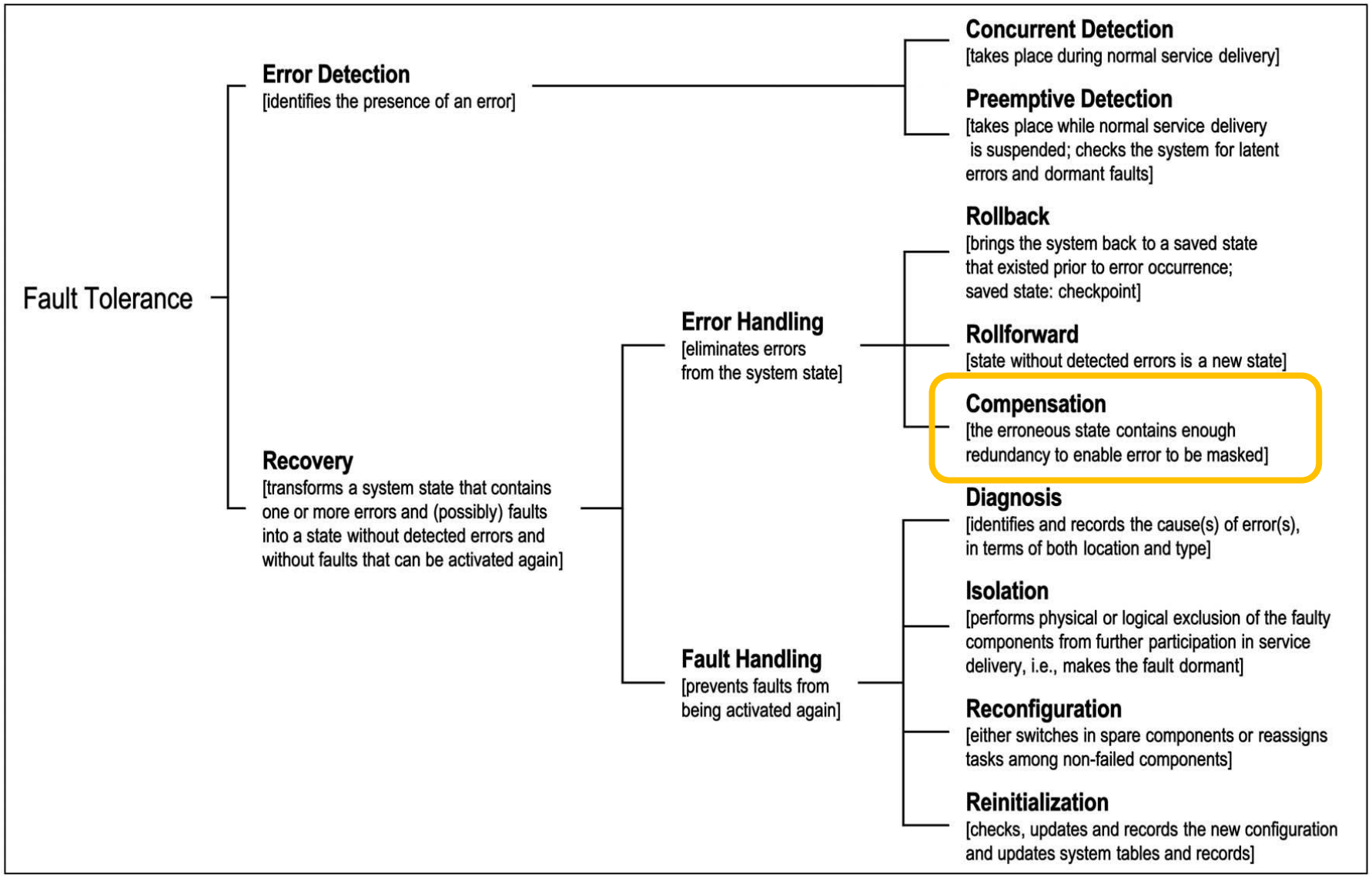
deals with faults at run-time

(zero faults not possible)

deliver correct service in presence of activated faults and errors



Organisation of fault tolerance



From [Avizienis et al., 2004]

Error compensation

the system contains enough redundancy to enable errors to be masked

⇒ faults are masked

fault masking

A general method to achieve fault masking is to perform multiple computations through multiple channels, either sequencially or concurrently and then apply majority vote on the outputs

Hardware faults

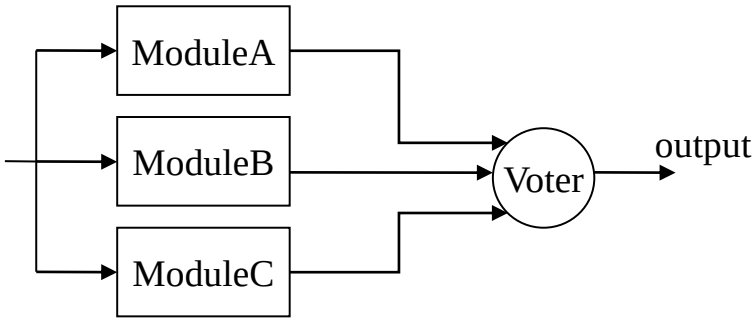
- **Hardware components fail independently**
- replicas of the hw component

Software faults

- **Replicas of the same sw do not fail independently**
- Versions of the sw that implement the same function via separate designs and implementations(**design diversity**)

Passive HW fault tolerance technique:TMR

1. Triple Modular Redundancy (TMR) – fault masking



TMR: tolerates 1 faulty module

Triplicate the hw (processors, memories, ..) and perform a majority vote to determine the output

- 2/3 of the modules must deliver the correct results
- effects of faults neutralised without notification of their occurrence
- masking of a failure in any one of the three copies at a time

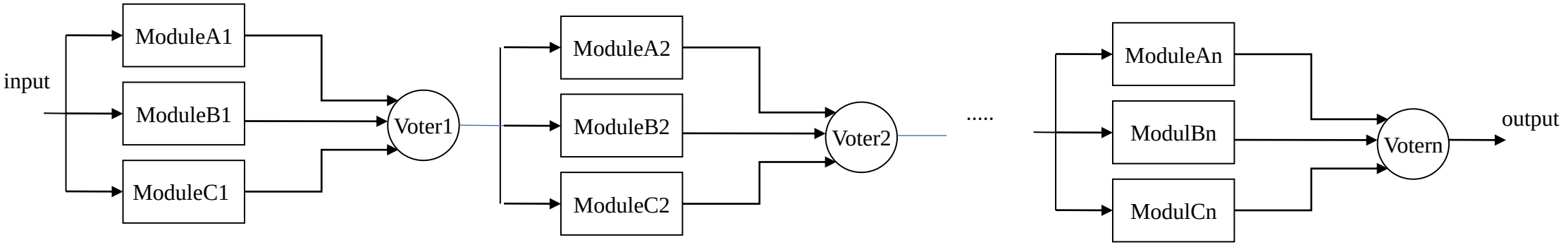
For permanent faults, since the faulty module is not isolated, the fault tolerance decreases

Good for transient faults

In some cases, two faulty modules *are tolerated*

e.g. memory location 127@ModuleA, memory location 153@ModuleB

Cascading TMR with triplicated voters



Cascading series of TMR modules

Series of TMR configurations.

The effect of partitioning of modules (A, B, C) is that the design can withstand more failures than the solution with only one large triplicated module.
One faulty module for each element of the series.

Voter is a single point of failure. Reliability of the Voter is very important.

TMR: the Voter

Majority voting is normally performed on a bit-by-bit basis

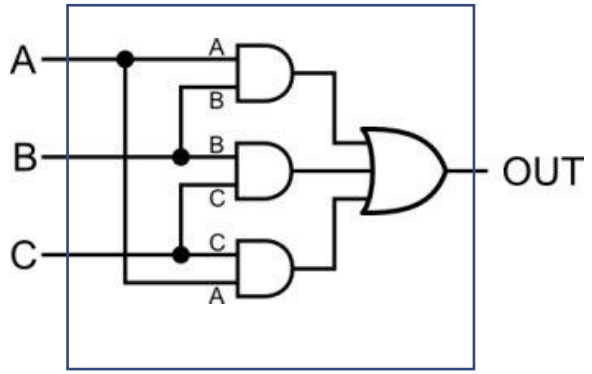
AND - OR circuit

the output is 1 if at least two inputs are 1

the output is 0 if at least two inputs are 0

OUT = AB + BC + AC

1 bit Voter on 3 input bits



Difficulties

Cascading TMR

Delay in signal propagation (decrease in performance):
- due to the voter
- due to multiple copies synchronisation

Trade-off : achieved fault tolerance vs hw required

N-Modular Redundancy

2. NMR – extension of the TMR concept to N Modules

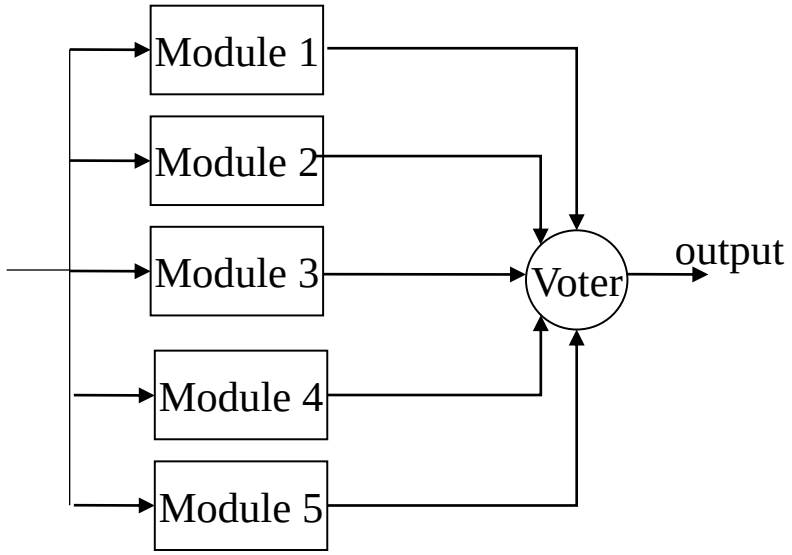
N is made an odd number

Coverage:
m faulty modules, with N = 2m +1

5MR: tolerates 2 faulty modules

7MR: tolerates 3 faulty modules

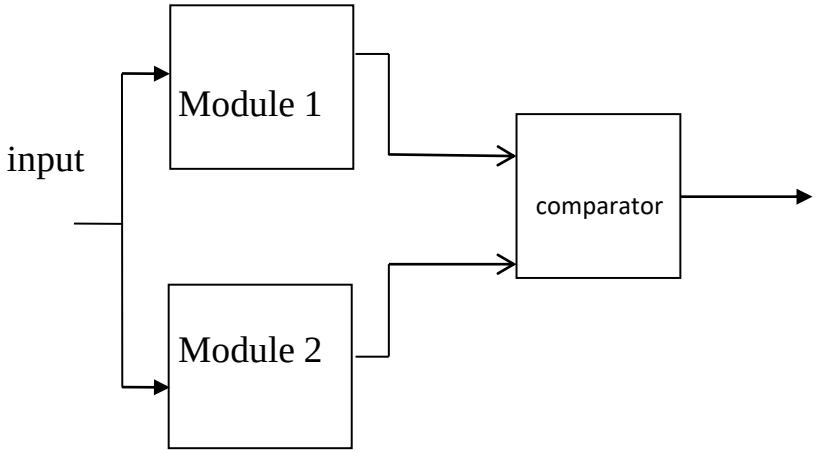
……..



1. Duplication with comparison scheme (Error detection)

Two identical pieces of hw (Module1 and Module 2) are operated in parallel

- when a failure occurs, the two outputs are no more identical and a simple comparison detects the fault
- Only disagreement can be determined and an error can be signalled by the comparator



The entire system must be considered faulty

Assumption:
the two copies must be unlikely to be corrupted together in the same way

Active hw redundancy: the comparator

UNIVERSITÀ DI PISA

Problems

- faults in the comparator may cause
  - an error indication when no error exists (false postive) or
  - possible faults in duplicated modules are never detected (false negative)

Coverage

- detects all single faults except those of the comparator

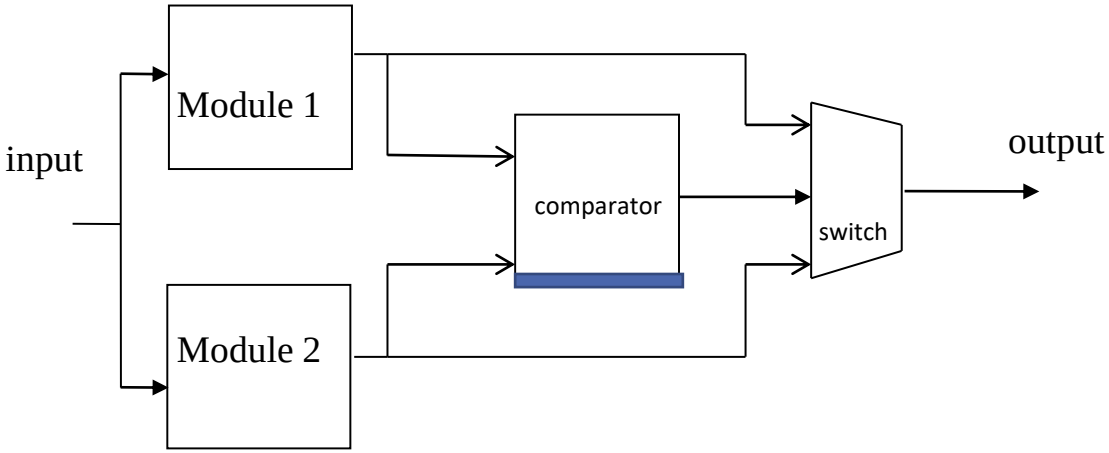
Advantages

- simplicity, low cost, low performance impact of the comparison technique, applicable to all levels and areas

2. Reconfigurable Duplication
(Error detection, disconnet the faulty module and disable the comparison)

Two identical pieces of hw (Module1 and Module 2) are operated in parallel

- when a failure occurs, the two outputs are no more identical and a simple comparison detects the fault
- the comparator (hw component) must select the correct output if a disagreement is detected



Dual-modular redundancy
(also Duplex system)

Active hw redundancy: the comparator

The comparator applies checks to select the correct output

Types of checks

-Coding
-Self-checking components
-Reversal Checks
-Reasonableness Checks
-Specification checks
-….

Ability to determine which of the two modules is faulty

Ability to disconnect the faulty module and to disable the comparator

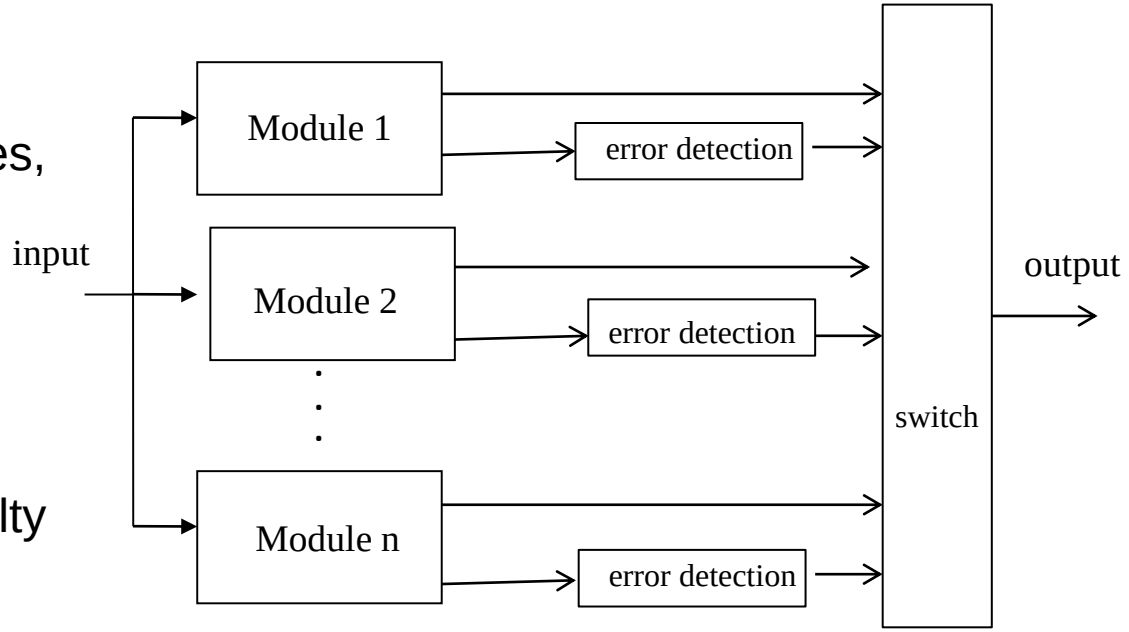
Active HW redundancy

3. Stand-by sparing

(error detection, identification of the faulty module, reconfiguration)

Each module is extended with an error detection module. Part of the modules are operational, part of the modules are spares modules (used as replacement modules). The switch can decide no longer use the value of a module (fault detection and localization). The faulty module is removed and replaced with one of the spares.

- *hot spares*
  the spares operate in synchrony with the on line modules, and they are prepared to take over
- warm spares
  the spares are running but receive inputs only after switching
- *cold spares*
  the spares are unpowered until needed to replace a faulty module



As long as the outputs of the operational modules agree, the spares are not used

Different schemes can be implemented

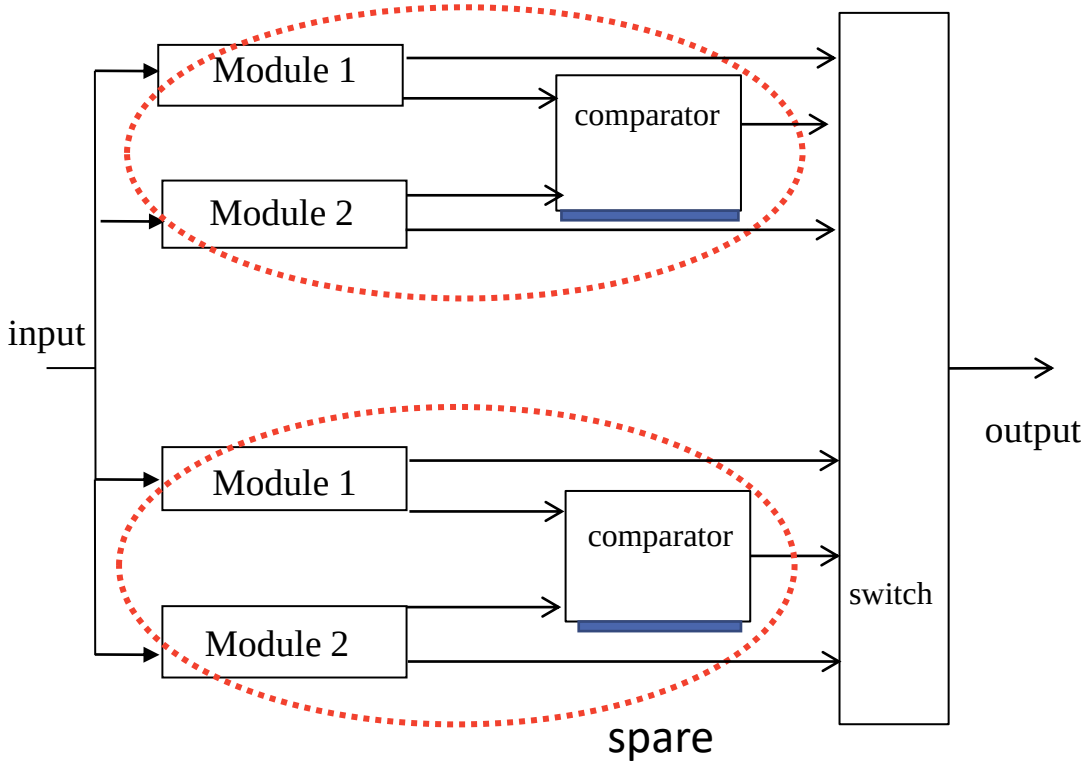
Pair-and-spare approach

- A module is a Duplex system, pairs connected by a comparator

- Duplex systems are connected to spares by a switch

- As long as the two outputs agree, or the comparator can detect the right value, the spare is not used.

- Otherwise, the comparator signals the switch that it is not able to compute the right value and the switch operates a replacemnet using the spare.



Pair results are used in a spare arrangment. Spare components at coarser granularity.

Not all four copies must be synchronised (only the two pairs)

Hybrid HW approaches

Combine both the active and passive approaches

Very expensive in terms of the amount of hw required to implement a system

Applied in commercial systems, safety critical system (aviation, railways, …)

NMR disadvantage: fault masking ability deteriorates as more copies fail
- Replace failed copies with unused spares (hybrid redundancy)

Reconfigurable NMR

Modules arranged in a voting configuration
- spares to replace faulty units
- rely on detection of disagreements and determine the module(s) not agreeing with the majority

Reconfigurable NMR

- N redundant modules configuration (active modules)

- Voter (votes on the output of active modules)

- The Fault detection units
  1) compares the output of the Voter with the output of the active modules
  2) replaces modules whose output disagree with the output of the voter with spares

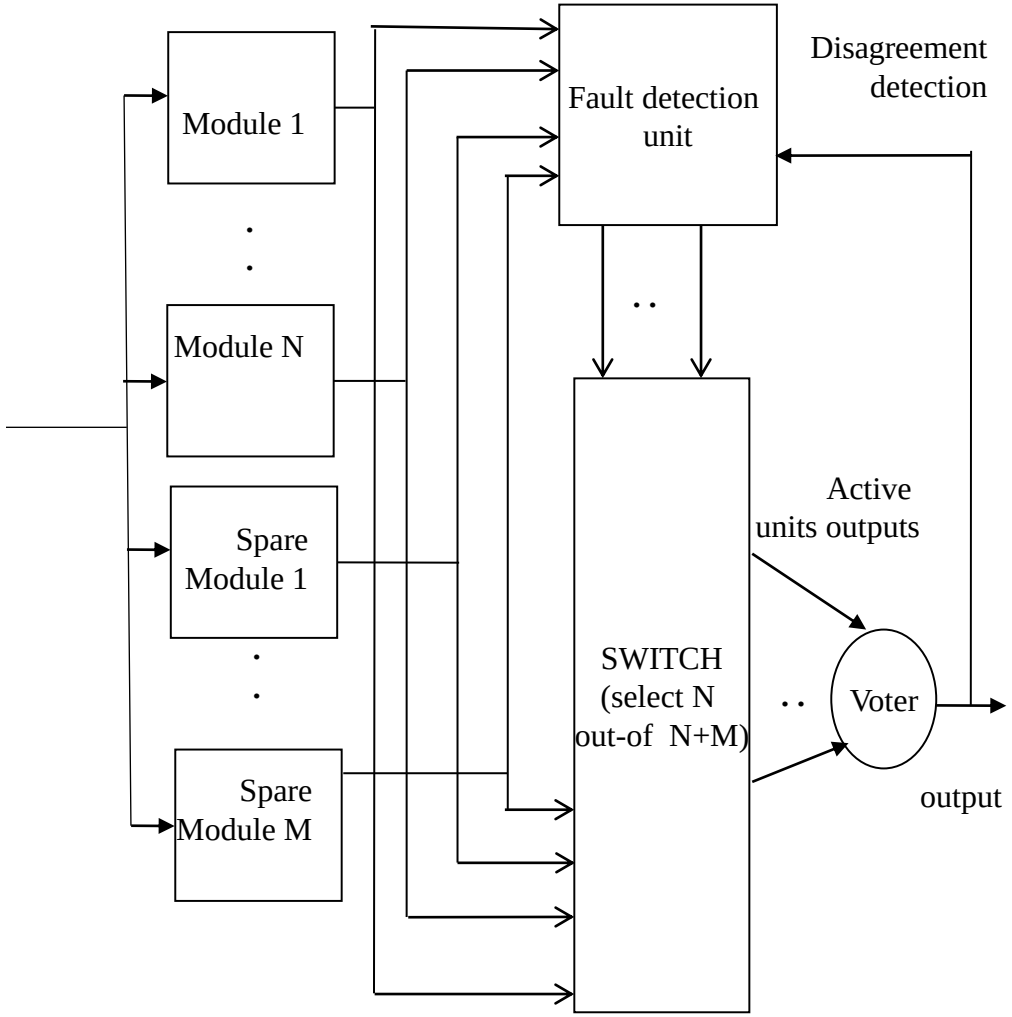
Reliablity

as long as the spare pool is not empty

Coverage

TMR with one spare can tolerate 2 faulty modules

(mask the first faulty module; replace the module; mask the second faulty module)



Hw redundancy techniques: summary

UNIVERSITÀ DI PISA

Key differences

Passive: rely on fault masking
Active: rely on error detection, fault location and recovery
Hybrid: emply both masking and recovery

- **Passive** provides fault masking but requires investment in hw (5MR can tolerate 2 faulty modules)
- **Active** has the disadvantage of additional hw for error detection and recovery, sometimes it can produce momentary erroneous outputs
- **Hybrid** techniques have the highest reliability but are the most costly (3MR with one spare can tolerate 2 faulty modules)

Self checking circuitry

Necessity of reliance on the correct operation of **comparators** and **voters** that are used as core for fault tolerant architectures

Self-checking circuit

given a set of faults, a circuit that has the ability to automatically detect the existence of the fault and the detection occurs during the normal course of its operations

Typically obtained using *Coding* techniques

D. P. Siewiorek R.S. Swarz,

Reliable Computer Systems Prentice Hall, 1998, pp.124-126

https://archive.org/details/reliablecomputer00siew

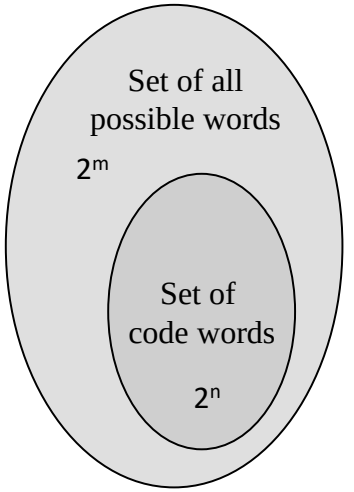
Coding

Coding: application of redundancy to information

Information is represented with more bits that strictly necessary: says, an n-bit information chunk is represented by

n+c= m bits

Among all the possible 2^m configurations of the m bits, only 2^n represent acceptable values (code words)

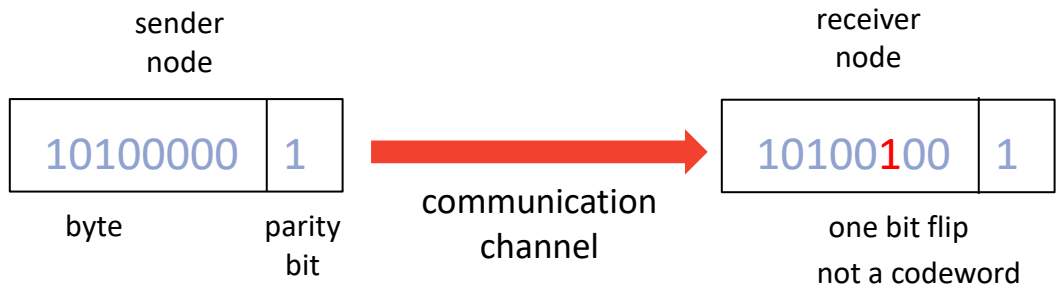


Invalid representations: $2^m - 2^n$

if a non-code word appears, it indicates an error in transmitting, or storing, or retrieving …

Parity code – odd parity

for each unit of data, e.g. 8 bits, add a parity bit so that the total number of 1's in the resulting 9 bits is odd



Two bit flips are not detected

Coding

Codes

encoding: the process of determining the c bit configuration for a n bit data item

decoding: the process of recovering the original n bit data from the m total bit

Separable code: a code in which the original information is appended with new information to form the code word. The decoding process consists of simply removing the additional information and keeping the original data

Nonseparable code: requires more complicated decoding procedures

Parity code is a separable code

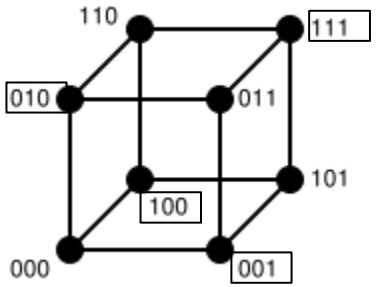
Additional information can be used for error detection and for error correction

Examples of codes

Parity-code (odd parity)

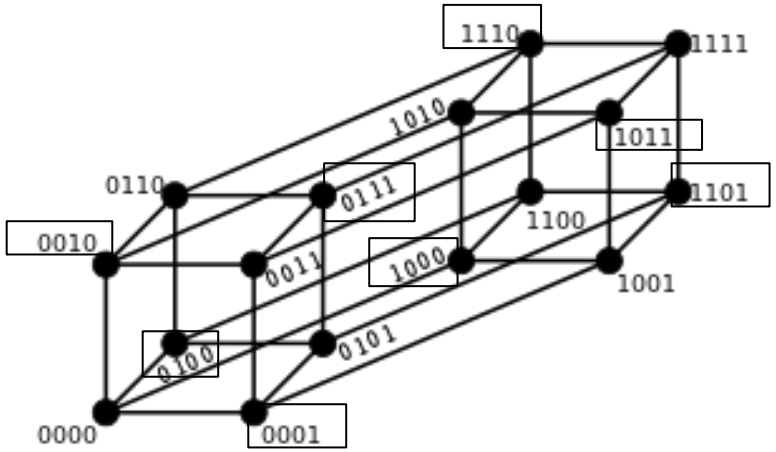
boxed words are code words in the figures

n=2, m=3



3-bit words
8 possible words
4 code words

n=3, m=4



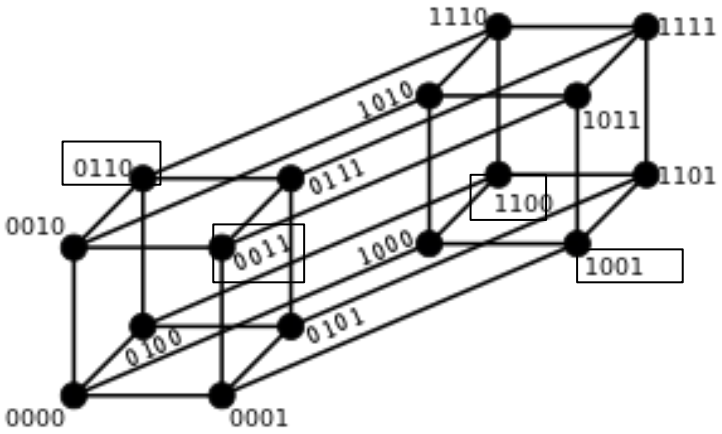
4-bit words
16 possible words
8 code words

4-bit words – 8 code words

Examples of codes

CD - complemented duplication

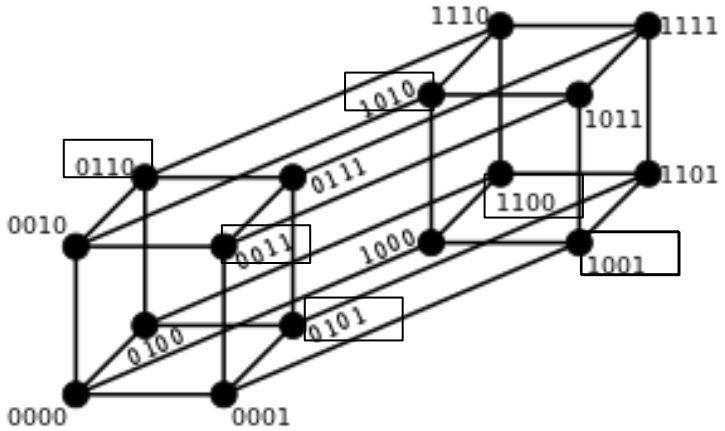
join n bit value with its complement: n complement(n)
the second half is the complemented duplication of the first half



4-bit words
16 possible words
4 code words:
{0011, 0110, 1001, 1100}

m/n code - m bit equal to 1

2/4 code



4-bit words
16 possible words
6 code words:
{1001, 1010, 1100, 0110, 0011, 0101}

Hamming distance

Hamming distance

- the number of bit positions on which two code words differ

Minimum Hamming distance found between any two code words
is the number of independent single bit errors that the code can detect

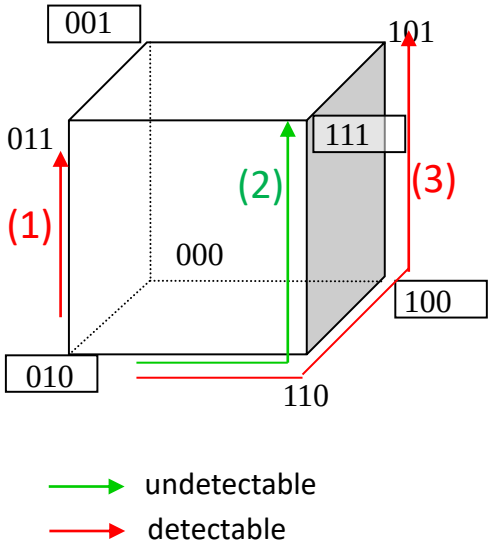
A code such that the Hamming distance between two code
words is equal to k will detect all errors up to k-1 bits

Memories of computer systems.
Parity bit added before writing the memory. Parity bit is checked when reading.

Useful distance measures depend on type of data and faults

Bank account numbers should be such that mistyping a digit
does not credit the wrong account.

each edge of the cube represents a distance-1 transition



Parity-code
Hamming distance 2

Codes for error correction

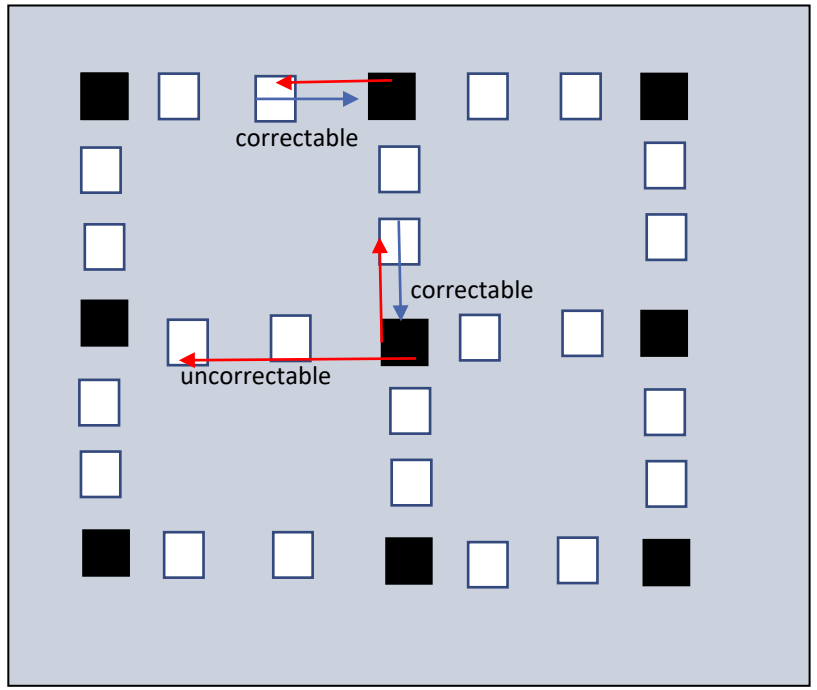
Minimum Hamming distance:
minimum distance between two code words

A code with the minimum Hamming distance is k

- detect up to k-1 single bit errors

- correct up to d errors, where k = 2d +1

Hamming distance 3:
detects 1 or 2 bits errors
correct 1 bit error



The corrupted data is closer to the correct code word than to any other code word

Self checking circuitry

Self-checking circuit

given a set of faults, a circuit that has the ability to automatically detect the existence of the fault and the detection occurs during the normal course of its operations

Basic idea:

- circuit inputs and outputs are encoded (also different codes can be used)
- fault free + code word input -> output: correct code word
- fault + code word input -> output: (correct code word) or (non code word)

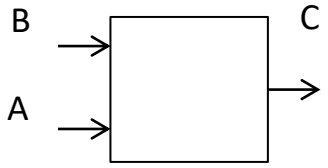
Self checking circuitry

- **Self-testing circuit:** if, for every fault from the set, the circuit produces a non code word output for at least one code word input (each single fault is detectable)
- **Fault-secure circuit:** if, for every fault from the set, the circuit never produces a incorrect code word output for a code word input
- **Totally self-checking (TSC):** if the circuit is self-testing and fault-secure

two-input TSC comparator

two signal input comparator (A, B)

output is 0 if inputs are equal; output is 1 otherwise



A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

Fault assumption:

- single fault
- stuck-at-1/stuck-at-0 of each line in the circuit

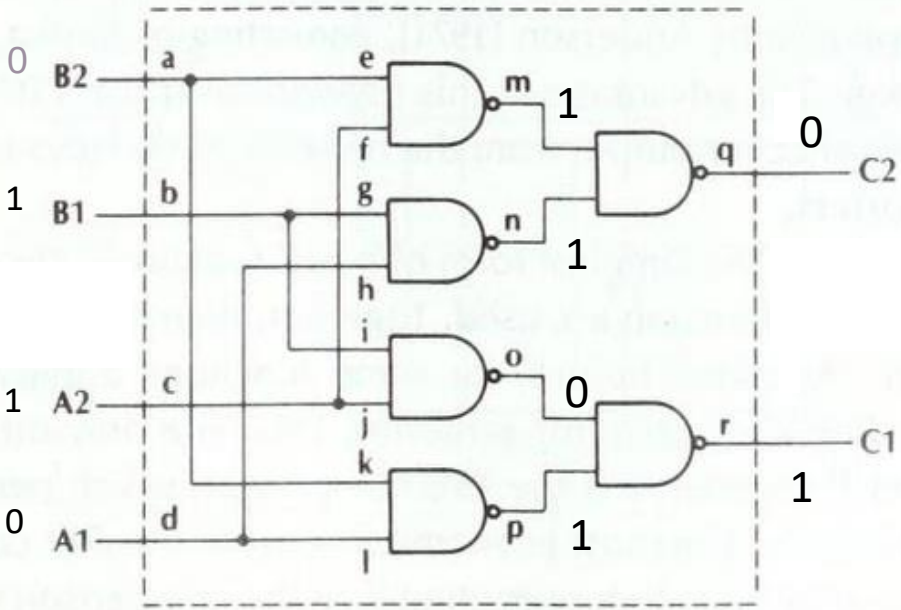
Coding: complemented duplication
(dual-rail signal: coded signal whose two bits are always complementary)

A	:	A1	A2
0	:	0	1
1	:	1	0

two-input TSC comparator

output 0 if inputs are equal; 1 otherwise

Fault free
A =0, B =1 different input
m=1, n =1, q=0
o = 0, p=1, r= 1
c2=0
c1=1
c1c2: code word
Output = c1 = 1 correct

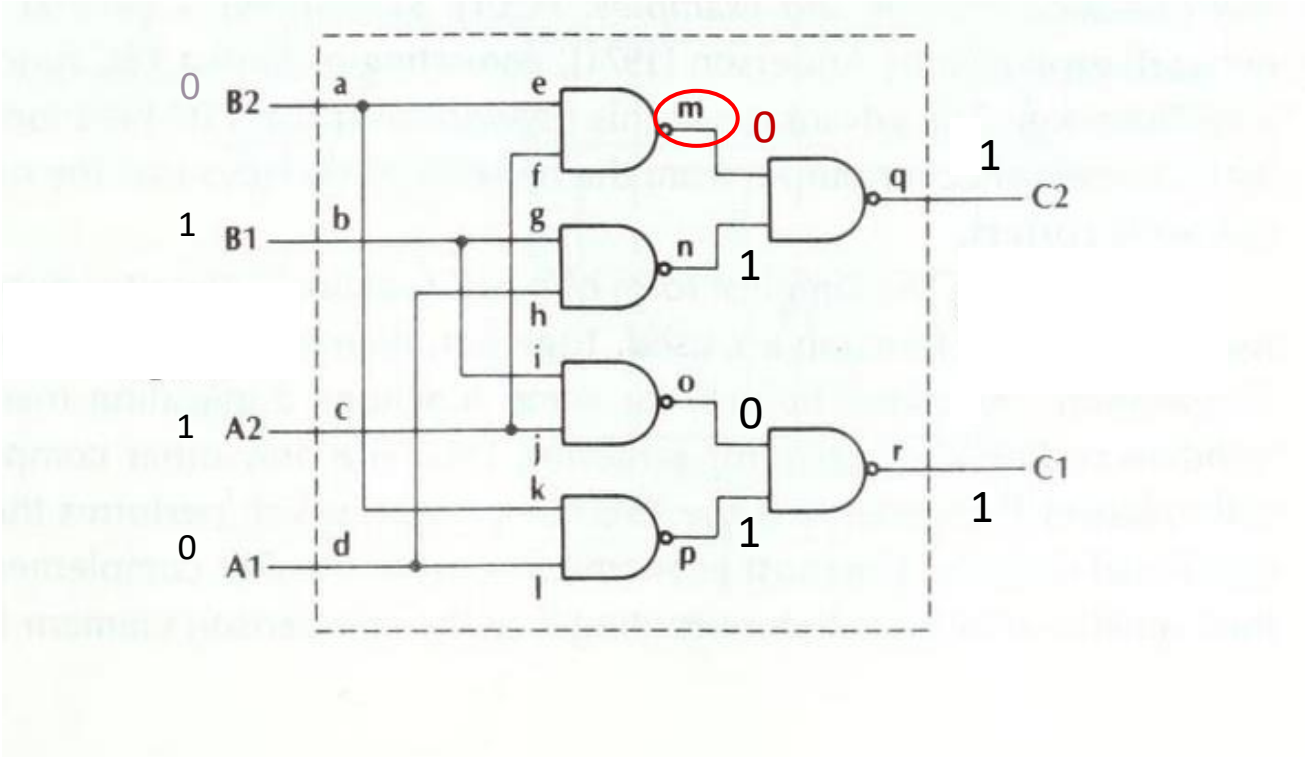


Taken from:[Siewiorek et al., 1998]

two-input TSC comparator

output 0 if inputs are equal; 1 otherwise

Faulty:
A=0, B=1 different input
m: stuck-at-0
c2 = 1
c1 = 1
c1c2: non code word
Output = error

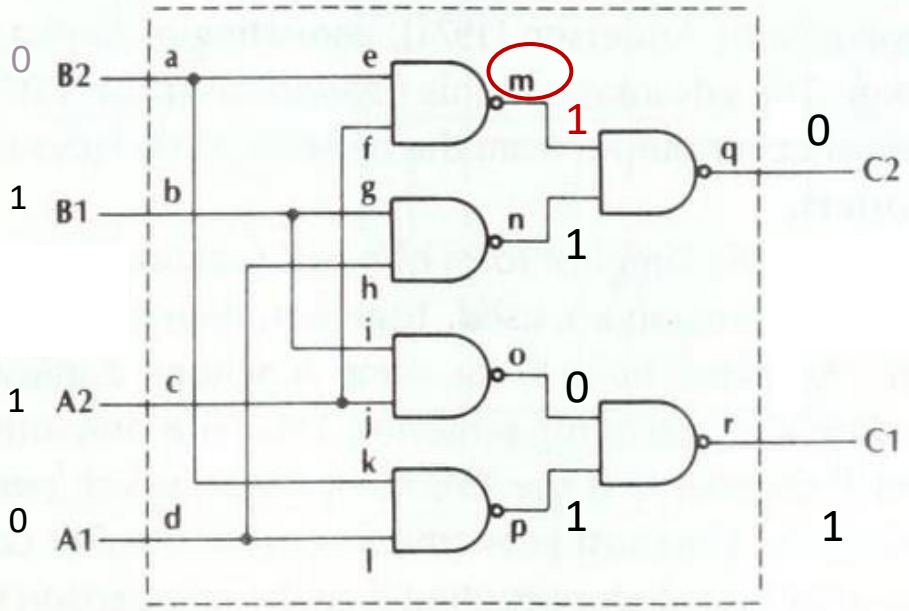


Taken from:[Siewiorek et al., 1998]

two-input TSC comparator

output 0 if inputs are equal; 1 otherwise

Faulty:
A=0, B=1 different input
m: stuck-at-1
c2=0
c1=1
c1c2: code word
output = c1 = 1 correct



Taken from:[Siewiorek et al., 1998]

two-input TSC comparator

Inputs		Normal	Outputs C2C1 Resulting from Single Stuck-at-1 Faults																
B2B1	A2A1	Output	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
01	01	10	**11**	10	**11**	10	10	10	10	10	10	**11**	**11**	10	10	**00**	10	10	10	**11**
01	10	01	**11**	01	01	**11**	**11**	01	01	**11**	01	01	01	01	01	01	**00**	01	**11**	01
10	01	01	01	**11**	**11**	01	01	**11**	**11**	01	01	01	01	01	01	01	01	**00**	**11**	01
10	10	10	10	**11**	10	**11**	10	10	10	10	**11**	10	10	**11**	**00**	10	10	10	10	**11**

Inputs		Normal	Outputs C2C1 Resulting from Single Stuck-at-0 Faults																
B2B1	A2A1	Output	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
01	01	10	10	**00**	10	**00**	10	10	**00**	**00**	10	10	10	10	10	10	**11**	**11**	**00**	10
01	10	01	01	**00**	**00**	01	01	01	01	01	**00**	**00**	01	01	**11**	**11**	01	01	01	**00**
10	01	01	**00**	01	01	**00**	01	01	01	01	01	01	**00**	**00**	**11**	**11**	01	01	01	**00**
10	**10**	10	**00**	10	**00**	10	**00**	**00**	10	10	10	10	10	10	10	10	**11**	**11**	**00**	10

Taken from:[Siewiorek et al., 1998]

- For each fault, there exists at least one input configuration such that the output is a non code word
- If the output is a code word, the output is correct

n-input TSC comparator

- n-input TSC comparator:
  tree of two input self checking comparators