GROUP NUMBER – 18

MEMBER'S NAME

SERIAL NO.	NAME	ROLL NO.
1.	JYOTI SUMAN SAHU	201CS122
2.	AJIT KUMAR PANIGRAHI	201CS105
3.	GYANBALLAV SAMIRSAHOO	201CS219
4.	SAHIL KUMAR	201CS149
5.	VISHAL KUMAR PAL	201CS267

PROJECT INFORMATION-

PROJECT NUMBER – 18

PROJECT TITLE -

Problem-18: Detect the SA1 faults on the 4x16 decoder constructed with two 3x8 decoders (See Figure below).

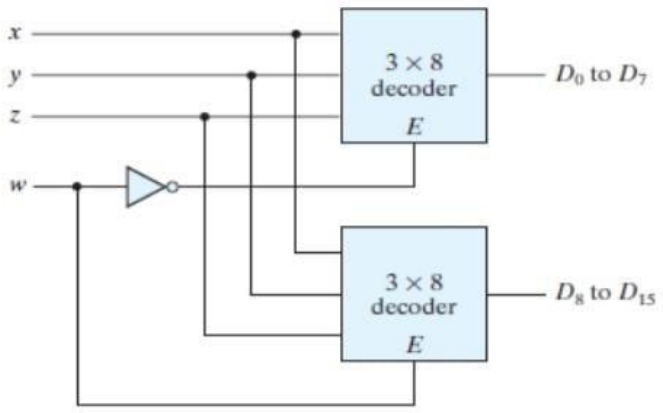


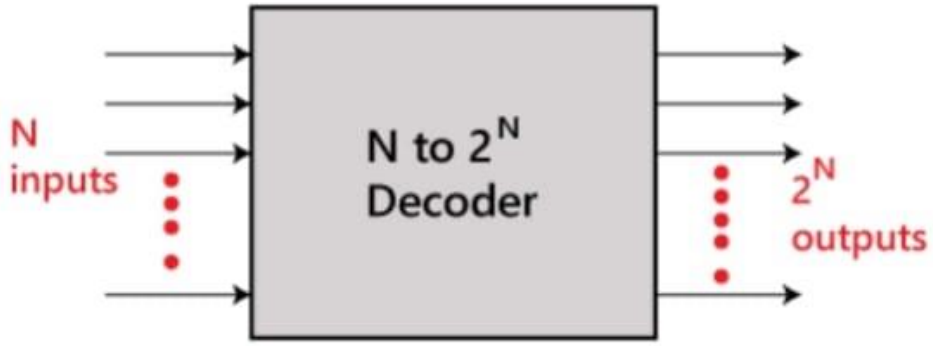
Figure: Logic diagram of a 4x16 decoder.

AIM- Detect the SA1 faults on the 4x16 decoder constructed with two 3x8 decoders.

THEORY

Decoder:

Decoder is a combinational logic circuit that converts the binary information from n coded inputs to a maximum of 2n unique outputs.



Applications of Decoder:

-> Decoder are used to input data to a specified output line as is done in addressing core memory.

-> It is used n code conversion.

-> It is also used for data distribution i.e., demultiplexing

-> It is used as address decoders in CPU memory location identification.

-> It decodes the binary input to activate the LED segments so that the decimal number can be displayed.

-> They can be the application of switching function often with the fewer integrated circuit.

STUCK AT FAULT:

Before direct jumping into Stuck at Fault let's know about what is fault modelling? After manufacturing a chip, the number of physical defects in a chip can be too many i.e., may be infinitely large. So many times,

this is impossible to count and analyze all possible faults. So, while testing a circuit/IC we abstract physical defects and define some logical fault models.

In this way:

-> We reduce the number of faults to be considered.

-> Makes test generation and fault simulation possible.

-> We can quantitatively compare test-sets to minimize the faults as minimum as possible.

Note: Defining fault models doesn’t mean circuit has these faults. We only assume that circuit is behaving like that.

There are different levels of abstraction of fault modelling:

1. Behavioral

2. Functional

3. Structural

4. Switch level

5. Geometrical

Among these, the stuck at fault model comes under structural level fault model and there are 2 types of stuck at fault modelling:

-> Single stuck at fault modelling

-> Multiple stuck at fault modelling

Among these two single stuck at fault modelling is most popular.

why?

1. Due to simplicity of single stuck at model it has been widely used to taste ICs.

2. Some interesting results that A test set that detect all single stuck at fault detect about >95% of multiple stuck at fault.

3. For tree like circuit it detects all multiple stuck at faults.

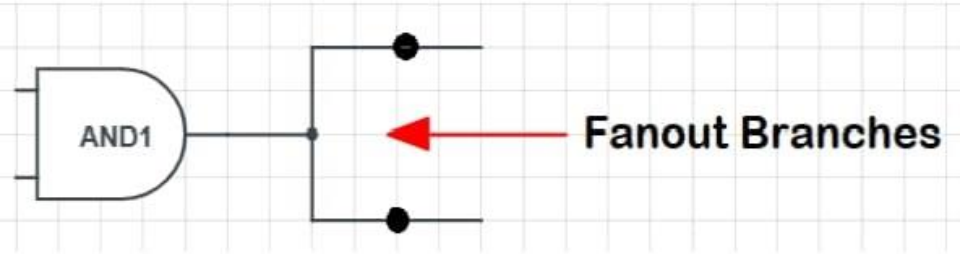
SINGLE STUCK AT FAULT MODELLING:

-> Here we assume the elements (i.e., gates) are absolutely fault free and circuit lines have fault that they are permanently fixed at logic 1 or logic 0 due to some failure. (at only one at a given time)

-> Very popular fault model as it can model realistic physical failures.

-> Faults on line A denoted as “s-a-0” or A/0 similarly for stuck at 1 it is “s-a-1” or A/1.

->Total number of possible stuck at fault position is: No. of gates + No. of input and output + 2*no. of fanout nodes.



However, the testing positions can be minimized by

-> Fault equivalence

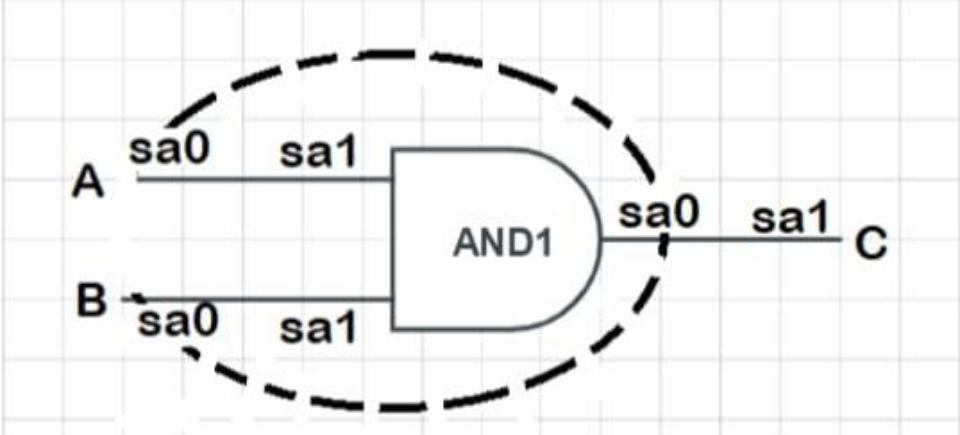
-> Fault dominance

How to minimize stuck at fault position:

1. Fault equivalence:

-> Two faults f1 and f2 are equivalent if all tests that detect f1 also detect f2.

Example:



From total 6 possible faults, we can rule, sa0 at A and sa0 at B and reduce to total 4 possible faults.

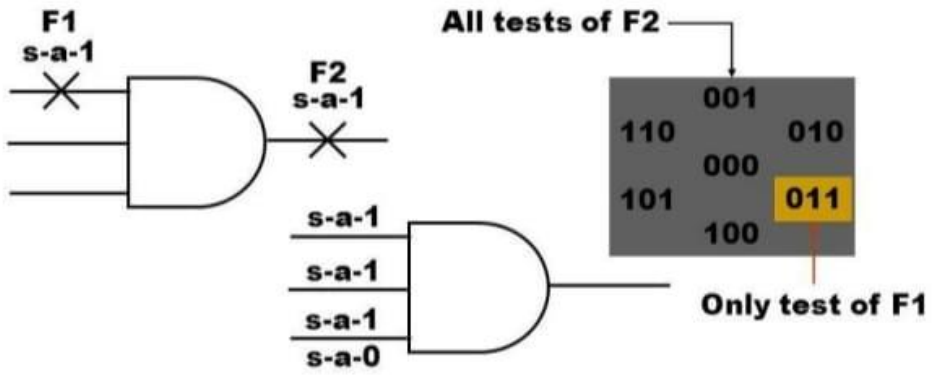
-> If f1 and f2 are equivalent then for both the faulty function is same.

2.Fault Dominance:

-> If all tests for some fault f1 detect another fault f2, then f2 is said to dominate f1.

-> If fault f2 dominates f1, then f2 is removed from the fault list.

Example:



-> If two faults dominate each other then they are equivalent.