DDFPGA PROJECT

On

Memory Built in Self-Test (MBIST)

By

Ashish Kumar Srivastava(172VL005)

&

Vishakh V(172VL025)

(M. Tech VLSI Design)

Under the guidance of Dr. Aparna P

Dept. of Electronics & Communication Engineering

National Institute of Technology Karnataka, Surathkal

Content

Introduction

A mechanism that allows a machine to test itself is called built-in self-test (or BIST). It can generate patterns based on a variety of algorithms, each focused on a particular type of circuitry or fault type. Comparison function has a number of unique implementations including actual comparators as well as signal analyzers.

BIST structures generate patterns and compare output responses for a dedicated piece of circuitry. You can implement BIST on entire designs, design blocks or structures within design blocks. Pattern generation as well as output-comparison circuitry can vary depending on the design.

A basic BIST block diagram is shown in Figure

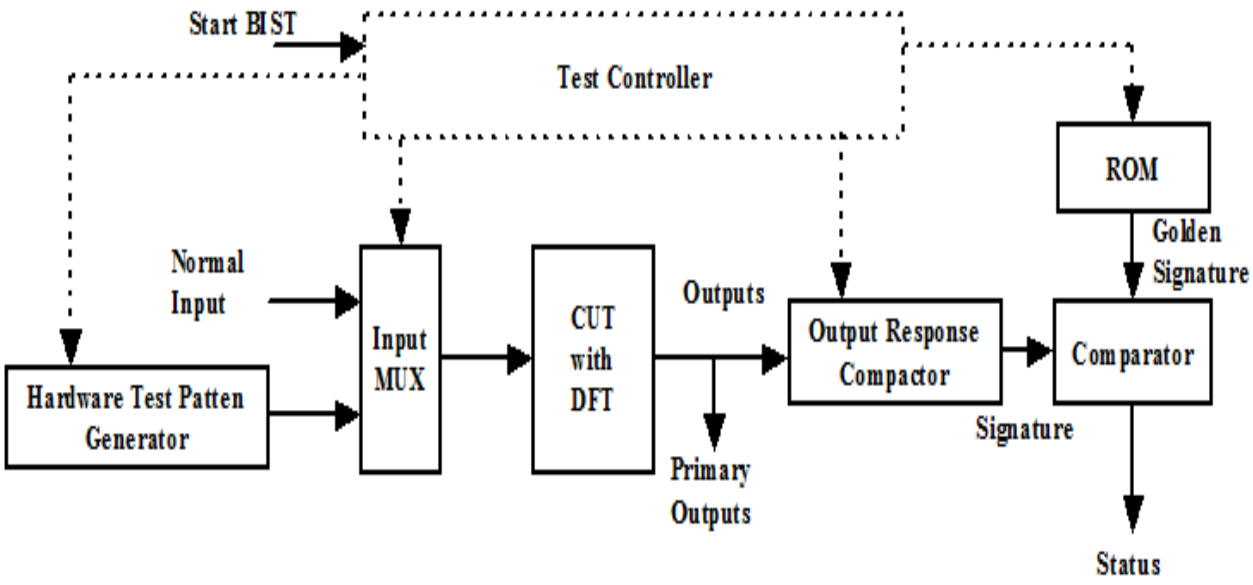


Fig. Basic BIST block diagram

Memory BIST

A manufacturing defect is a physical problem that occurs during the manufacturing process, causing device malfunctions of some kind.

The basic types of memory faults include stuck-at, transition, coupling and neighborhood pattern-sensitive. In this project we will design BIST for memory stuck-at fault model, which means that a memory fails if one of its control signals or memory cells remains stuck at a particular value. It also checks Transition fault.

Stuck-at faults model this behaviour, where a signal or cell appears to be tied to power (stuck-at-1) or ground (stuck-at-0). Transition fault is the fault of cell in which cell fails to make a(0 t0 1) or a (1 to 0) transition when it is written.

Device testing requires stimulus, a mechanism to apply stimulus to the device or circuit under test (CUT), and some means to analyse or compare the device's responses with a known good (non-faulty) response.

Classical testing uses external test patterns as stimulus and applies the patterns to the device via a tester. The tester examines the device's response, comparing it against the known good response stored as part of the test pattern data.

In our project we use March Test to test memory.

March Test

It is widely used memory testing technique. There are four step uses to test memory using March Test.

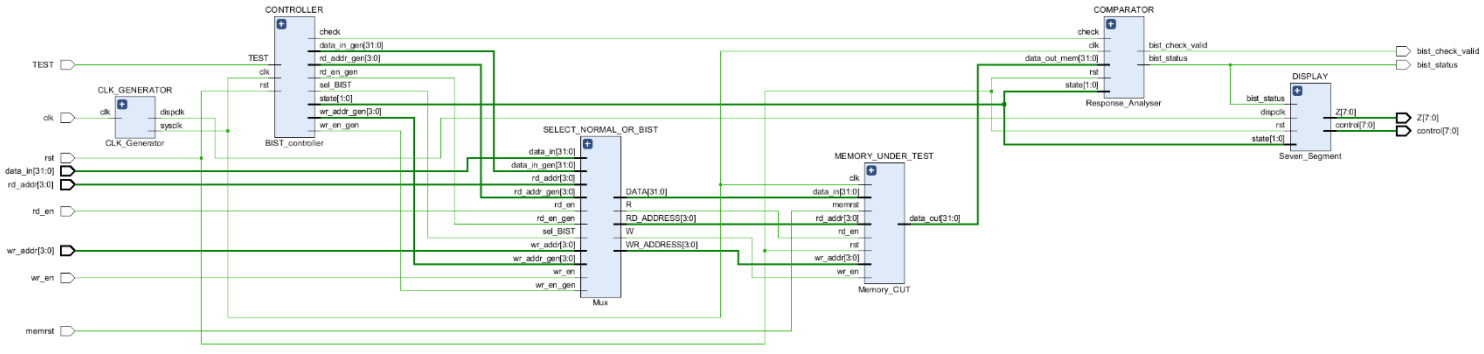
1. In increasing order of address of memory cells, write 0s to the cells

2. In decreasing order of address of the memory cells, read the cells (expected value 0) and write 1 to the cells

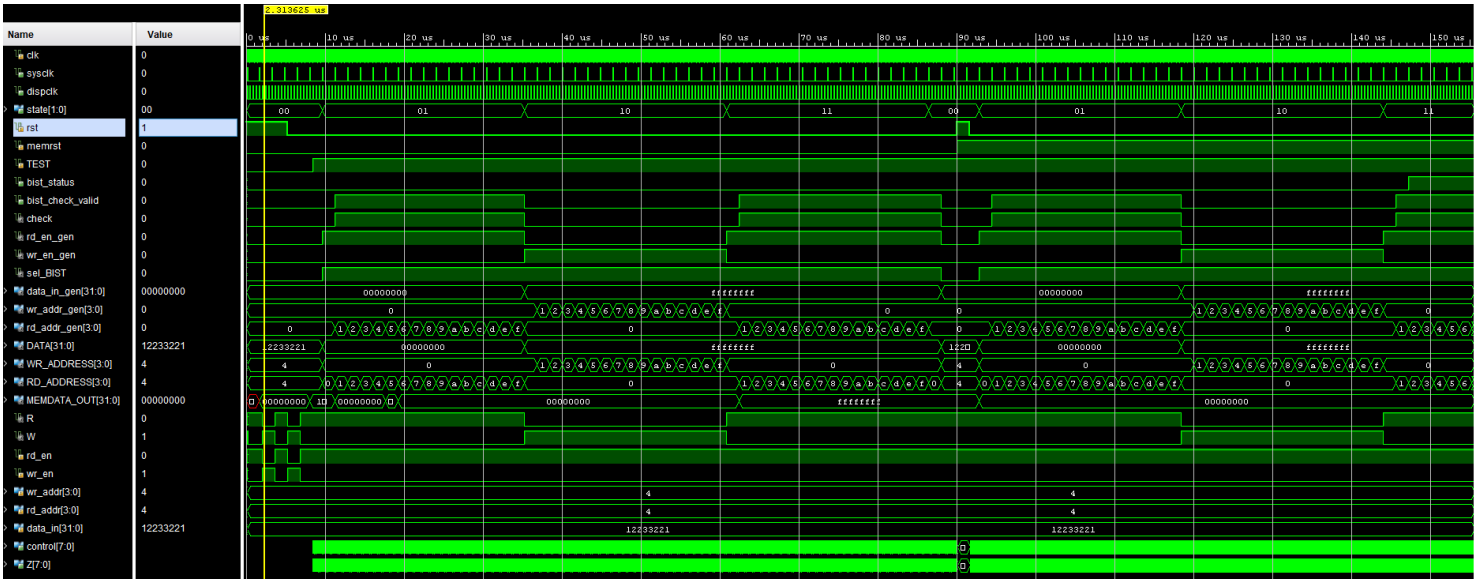
3. In increasing order of address of the memory cells, read the cells (expected value 1) and write 0 to the cells

4. In decreasing order of address of the memory cells, read the cells (expected value 0)

RTL Design



Simulation result



Resource utilization

Resource	Utilization	Available	Utilization %
LUT	256	63400	0.40
FF	644	126800	0.51
IO	64	210	30.48
BUFG	2	32	6.25