DDR-MEMORY CONTROLLER UVM PROJECT

Design implementation document

-BY DANISH RANA

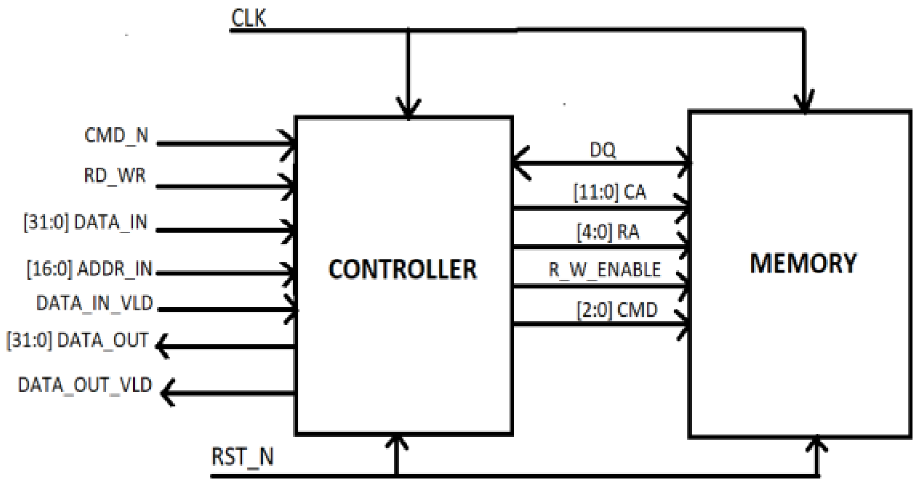
CONTENTS

1 INTRODUCTION

Memory controllers contain the logic necessary to read and write to DRAM, and to "refresh" the DRAM. Without constant refreshes, DRAM will lose the data written to it as the capacitors leak their charge within a fraction of a second.
Reading and writing to DRAM is performed by selecting the row and column data addresses of the DRAM as the inputs to the DRAM, where the DRAM uses the converted inputs to select the correct memory location and return the data.
There is a bidirectional bus between controller and DRAM which takes data_in to DRAM and takes out data_out to controller output.

2 TOP LEVEL IMPLEMENTATION

2.1 TOP LEVEL DIAGRAM



2.2 BASIC FUNCTIONALITY

- It supports command ACT, READ, WRITE, REFRESH,PRE.
- Read and write are external commands to this controller while REFRESH is the self
- generated command for each row from controller after every 3.2us.
- ACT is command internally generated by controller to activate a row.
- Whenever there is change in ROW a new ACT command must be issued.
- No two rows can be activated at the same time. If a row is activated and a new row needs to be activated a PRE command must be issued to precharge the previous activated row.
- ACT command should have a valid RA.
- PRE command should have a valid RA.
- READ/WRITE command should be having a valid CA.
- REFRESH command should also be having valid RA.

2.3 SPECIFICATION

- Command delays from controller are as follows
- READ to READ :2tCK
- READ to WRITE :4tCK
- WRITE TO READ :4tCK
- REFRESH to READ/WRITE:5tCK
- ACT to READ/WRITE :5tCK
- READ/WRITE to PRE: 4tCK
- Write data should come with command while read data should be there AFTER 2tCK of READ command issued.

2.4 SYNCHRONIZATION

- First give rst_n to reset memory to default.

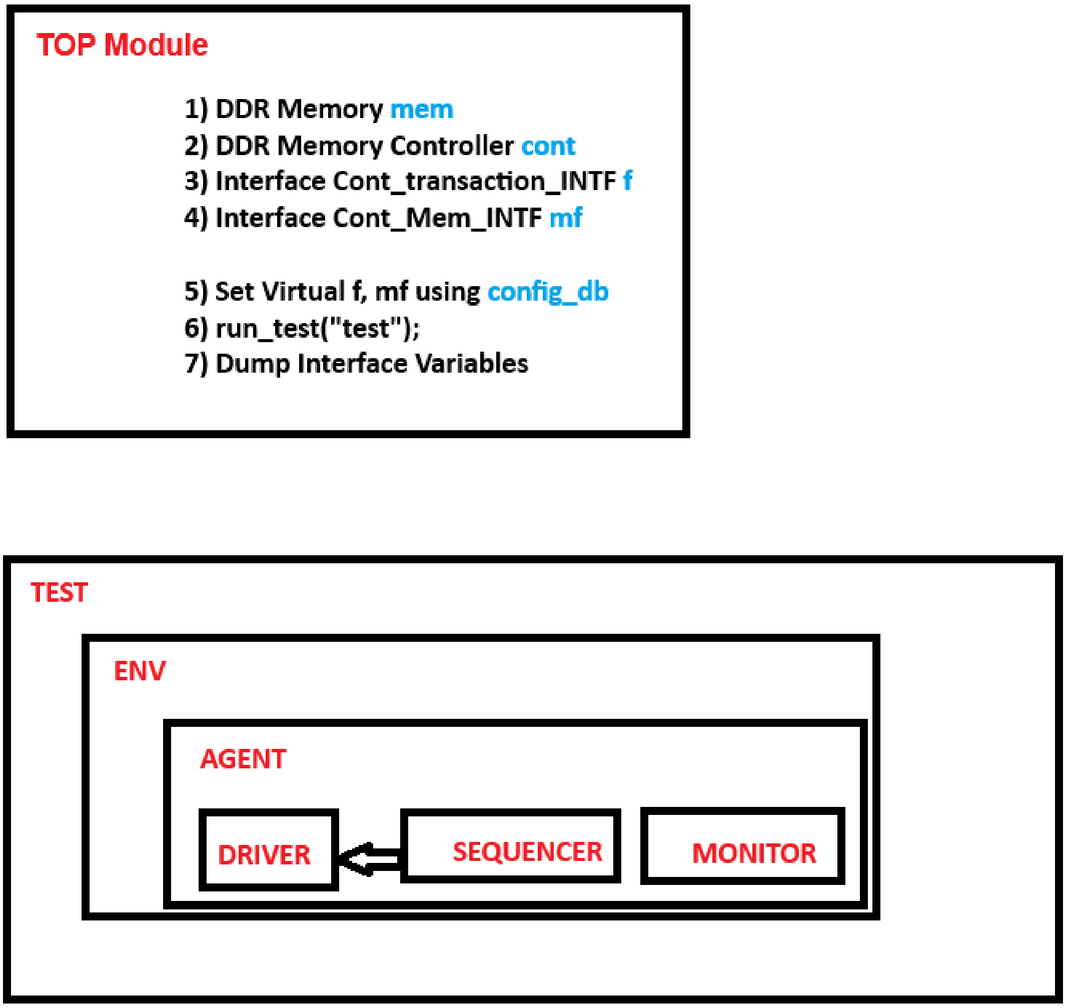
2.5 ASSERTIONS

- Bidirectional bus should never be unknown.
- After read command from controller , nop command should be issued by controller for 2 clock pulses.
- After read command , there should be valid data on bidirectional bus after 2 clk pulses.
- After write command , there should be valid data on bidirectional bus within same clk pulse.
- After read command, there should be read command after 2 clk pulses or pre charge after 4 clk pulses or write after 4 clk pulses.
- After write command, there should be read command after 4 clk cycles or pre charge after 4 clk cycles or write in next clk pulse.

2.6 TEST CASES

- 100 rd/wr commands were issued by controller. Used following sequences to check DUT.
- Fixed single address in transation class to 16’hffff.
- Given unique address everytime for read/write.
- Given random address evrytime for read/write.
- Given corner address like 16’h0, 16’hffff, 16’hfff etc
- Given asynchronous reset in between to check memory set to default.
- Checked refresh command is issued after every 640 clock pulse, by setting refresh condition to 20 clock pulse, 40 clock pulse etc.

3 TESTBENCH DIAGRAM



TOP Module

1) DDR Memory mem
2) DDR Memory Controller cont
3) Interface Cont_transaction_INTF f
4) Interface Cont_Mem_INTF mf

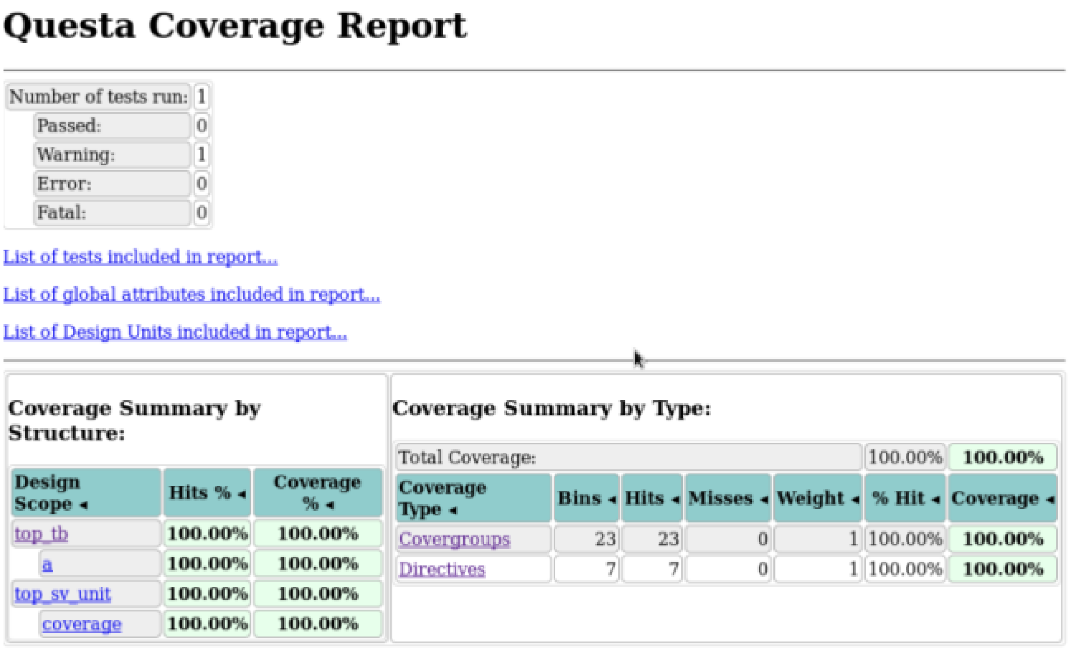
5) Set Virtual f, mf using config_db
6) run_test("test");
7) Dump Interface Variables

4 COVERAGE COLLECTOR

Detail of covergroup and cover bins

2.5 COVERAGE COLLECTOR

Detail of covergroup and cover bins



Questa Coverage Report

Number of tests run:	1
Passed:	0
Warning:	1
Error:	0
Fatal:	0

List of tests included in report...

List of global attributes included in report...

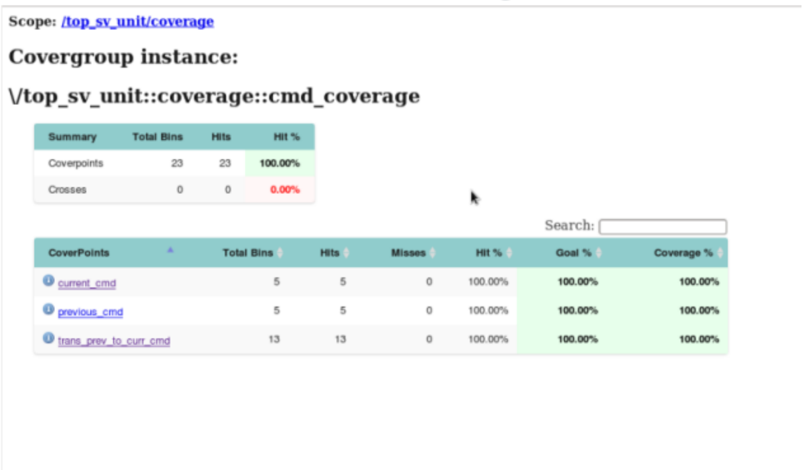
List of Design Units included in report...

Coverage Summary by Structure:

Design Scope ◂	Hits % ◂	Coverage % ◂
top_tb	100.00%	100.00%
a	100.00%	100.00%
top_sv_unit	100.00%	100.00%
coverage	100.00%	100.00%

Coverage Summary by Type:

Total Coverage:					100.00%	100.00%
Coverage Type ◂	**Bins ◂**	**Hits ◂**	**Misses ◂**	**Weight ◂**	**% Hit ◂**	**Coverage ◂**
Covergroups	23	23	0	1	100.00%	100.00%
Directives	7	7	0	1	100.00%	100.00%



Scope: /top_sv_unit/coverage

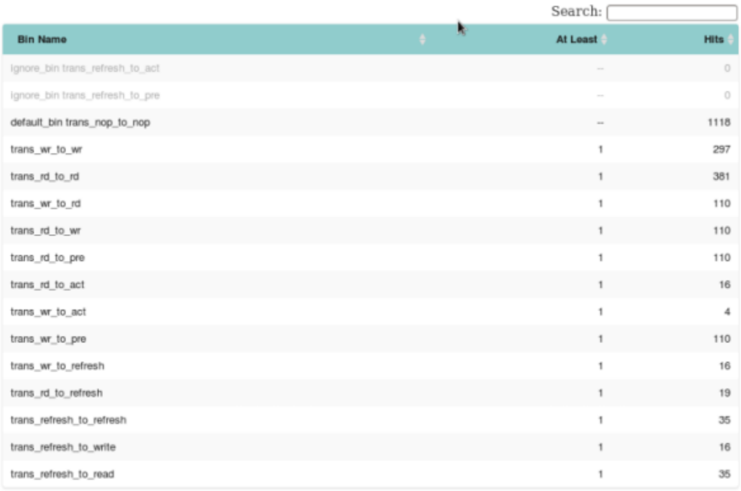
Covergroup instance:

\/top_sv_unit::coverage::cmd_coverage

Summary	Total Bins	Hits	Hit %
Coverpoints	23	23	100.00%
Crosses	0	0	0.00%

Search:

CoverPoints	Total Bins	Hits	Misses	Hit %	Goal %	Coverage %
current_cmd	5	5	0	100.00%	100.00%	100.00%
previous_cmd	5	5	0	100.00%	100.00%	100.00%
trans_prev_to_curr_cmd	13	13	0	100.00%	100.00%	100.00%



Search:

Bin Name	At Least	Hits
ignore_bin trans_refresh_to_act	--	0
ignore_bin trans_refresh_to_pre	--	0
default_bin trans_nop_to_nop	--	1118
trans_wr_to_wr	1	297
trans_rd_to_rd	1	381
trans_wr_to_rd	1	110
trans_rd_to_wr	1	110
trans_rd_to_pre	1	110
trans_rd_to_act	1	16
trans_wr_to_act	1	4
trans_wr_to_pre	1	110
trans_wr_to_refresh	1	16
trans_rd_to_refresh	1	19
trans_refresh_to_refresh	1	35
trans_refresh_to_write	1	16
trans_refresh_to_read	1	35