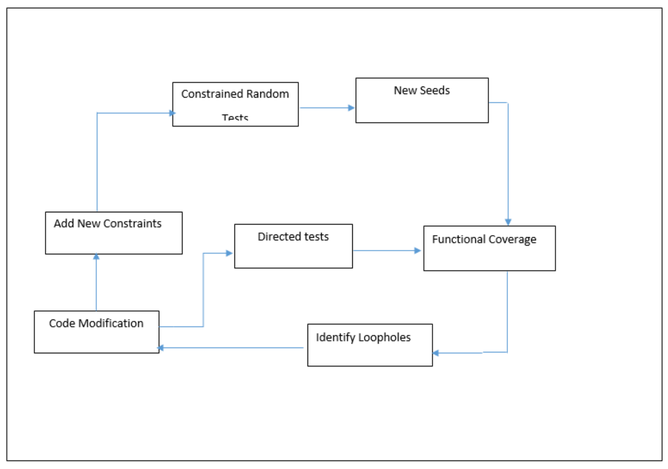
Adaptive Verification Methodology for a System on a Chip (SOC)

Prithvin Kumble and Mayur Madhusoodan
Department of Electrical Engineering, San Jose State University, San Jose, California 95192

Introduction

The process of verification most of the times is carried out simultaneously with the design of the System on a Chip (SOC). Writing directed test [1] cases to verify such a complex design and manually reviewing the log files or the waveform is not feasible. Thus a different verification methodology is required to verify the functioning of the Design under Test (DUT).

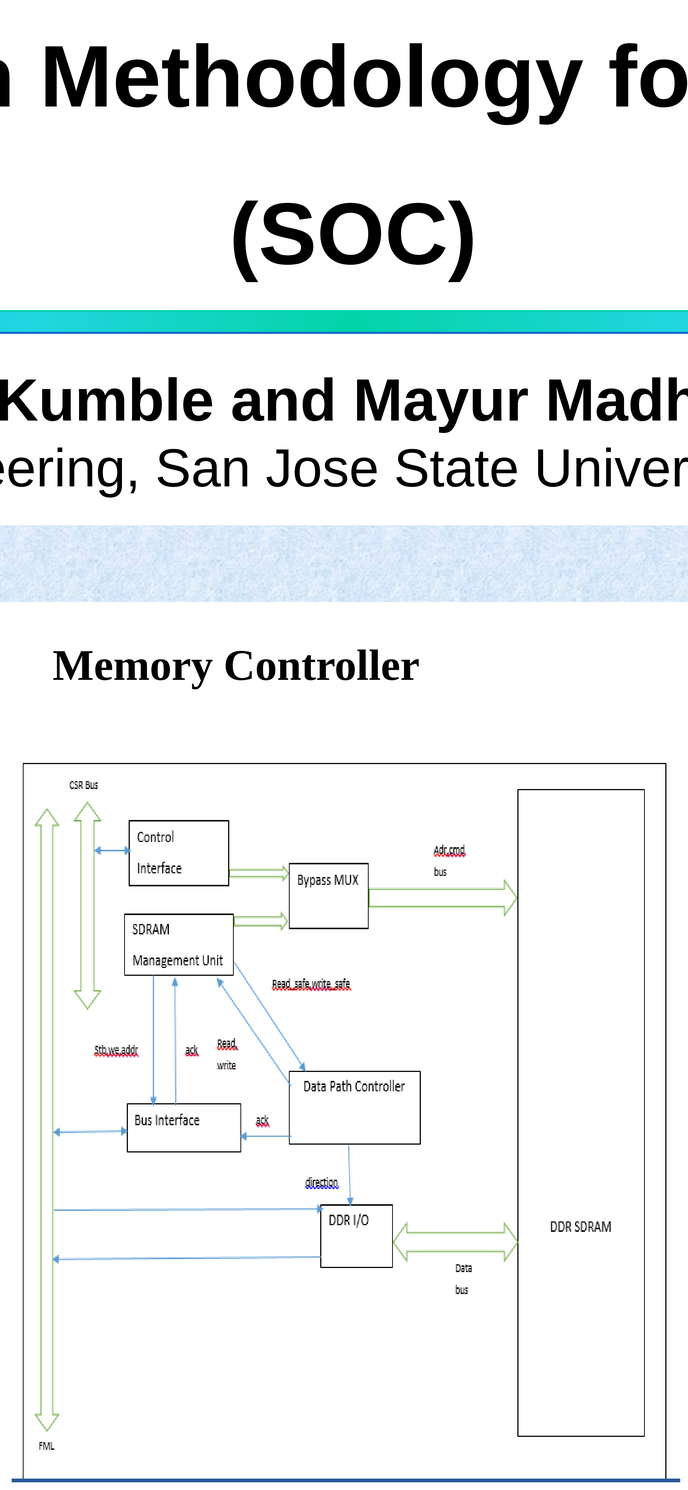


The test bench is first started by writing the constrained random test cases. The test cases are applied to the design under test and the working of the design is first verified by identifying the loopholes in the design. If a satisfactory coverage [2] is not obtained, new test cases are added in addition to writing directed test cases to cover corner cases

Test Bench Design

- Transactor is the class where the signals that are fed to the design are listed. It is used to perform high level operations
- Stimulus Generator class is used to generate the stimulus which are later sent to the driver class either using queues or mailboxes.
- Driver class accepts the randomized object of the generator and then feeds it to the design through an interface.
- Virtual Interface provide a mechanism to separate the actual signals that drive the design and the abstract high level objects.
- Monitors are used to identify the protocol violations and to identify the different transactions taking place in the verification environment
- Scoreboard is used to store the expected output of the design

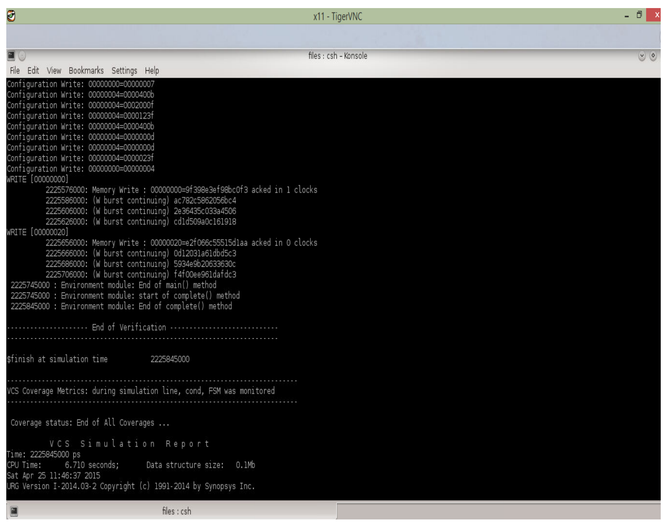
Memory Controller



- Control Interface of the memory controller used sets timing constraints, controls the operating mode of the core and to initializes the SDRAM
- System Registers like Timing Register, Bypass Register, Delay Register control the timing delays generated and mode of operation of the memory controller.
- Management Unit is a state machine that controls the address and the command buses.
- Management Unit has complete control over the controller and will relinquish command only in bypass mode.
- Data Path Controller decides the direction of of Data Strobe signal and introduces suitable delays [3] between read, write and pre-charge commands.
- Bypass Mux selects either the output of the control interface during bypass mode or the output of Management Unit during normal mode.
- The memory controller also consists of the Configuration and Status Register Bus whose main function is to write and read data into and from the status
- The memory controller consists of a Fast Memory Link Bus(FML) that acts as an interface between the memory controller and the peripherals to access data.
- The FML is synchronized with the system clock and supports burst transfer.

Results

The objective of the project was to achieve a functional coverage of 100%. The above snapshot shows that the expected coverage is achieved.



The above figure shows the single read and write bursts. The signals sent to the memory controller and the output value obtained from the monitor and the scoreboard are shown in the above screen grab.

Conclusions

The working of the High Performance Memory Controller was verified by creating a verification environment in System Verilog. All the features of the memory controller were verified since a functional coverage of 100% was achieved. The verification environment can be further enhanced by including other coverage metrics like code, fsm coverage to obtain a holistic coverage. Assertion can be written to check the temporal functioning of the design with respect to the system clock.

Key References

[1] Th. Kropf. Introduction to formal hardware verification ,Springer, 2009.

[2] M. Zambaldi and W. Ecker, “A Tester-Related Simulation Environment,” *Proc. GI/ITG/GMM Workshop Methods and Specification Languages for Modeling and Verification of Circuits and Systems*, Soc. for Computer Science (GI), Information Technical Soc. in VDE (ITG), and Soc. for Microelectronics, Micro and Fine Mechanics (GMM), 2004.

[3] D. Abts, N. D. Enright Jerger, J. Kim, D. Gibson, and M. H. Lipasti. Achieving predictable performance through better memory controller placement in many-core CMPs. In ISCA-36,2009

Acknowledgments

We would like to thank Prof. Morris Jones for his guidance which helped us to complete the project

For further information

Please contact *prthvinkumble@@gmail.com or mkmayur@hotmail.com* for the code, simulation results or the coverage scripts .