

DDR Controller IP Evaluation Studies using Trace Based Methodology

Abhilash Nair, Texas Instruments, Bangalore, India

Amit Nene, Texas Instruments, Bangalore, India

Ritesh Sojitra, Texas Instruments, Dallas, USA

Prashant Karandikar, Texas Instruments, Bangalore, India

Prajakta Bhutada, Texas Instruments, Bangalore, India

Abstract— This paper talks about DDR controller evaluation using Traces. The approach talk about how you can extract approximated traces for an application which can be used to evaluate Controller from various vendors. Also the paper shares result obtained from few vendors

I. INTRODUCTION

DDR controller and DDR is becoming a key factor influencing the success of SOC. With the increase in bandwidth requirement from different Initiators, getting the best out of DDR is very important. There are various factors which should be considered and compared before making the right selection viz Performance, Power, Cost and Area. This paper presents a “Trace based Methodology” for DDR controller evaluation.

DDR controller performance are mainly driven by

- Workload characteristics → CPU driven workloads are different than Multimedia workloads with real-time initiators. Hence DDR controller requirement for Initiator would vary based on the access pattern. Initiator could be Bandwidth critical or Latency sensitive or Latency critical and each of them would have different requirement [1].
- Scheduling Algorithm → Selecting the right access over a set of access from same or different masters. Selections are typically done based on access Priority/Type/Master/Address.
- DDR page management → DDR allows the controller to keep a specific number of Pages open. Keeping the right Page open helps to reduce the overhead from opening a new Page.
- Address bit mapped to BANK/ROW/COL for the DDR → DDR controller could remap the address to reduce bank conflict and hence increase DDR utilization.

II. ALTERNATE METHOD FOR DDR CONTROLLER EVALUATION

A. RTL simulation with the new DDR controller

- Advantage
 - Accurate and reliable result
- Issues
 - Selection is very late in the cycle and any surprise can result in disastrous designs with little room for revisiting the design
 - Typical selection process for IP’s like DDR controller are done very early in design cycle. So availability of both complete Design and Software is practically impossible

B. C++/SystemC based Simulation platform with the cycle accurate model of DDR controller

- Advantages
 - Accurate and reliable result (Less compare to the RTL simulation approach)
 - The initiators can be Traffic/Trace or Register model. This could enable running abstracted out software rather than the real software

- Issues
 - Availability of SystemC/RTL model from the IP vendor is difficult and mostly during the Evaluation phase information about the IP are shared as datasheet

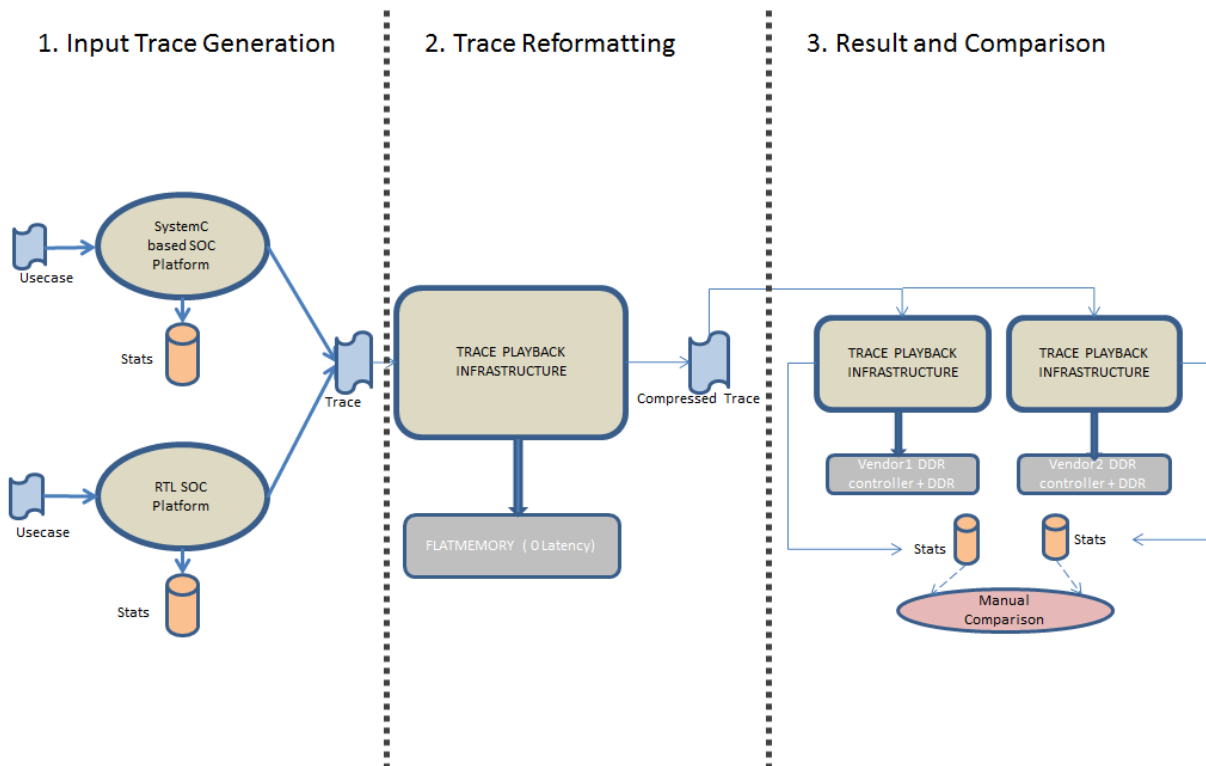
III. TRACE BASED SIMULATION INFRASTRUCTURE FOR DDR CONTROLLER EVALUATION

Trace based approach has all the advantage of C++/SystemC based Simulation platform with Cycle accurate DDR controller. Additionally,

- It enables to perform evaluation without transferring proprietary between the interested parties, avoiding NDA/licensing issues during Evaluation phase.
- Traces can be obtained from different simulation platform like SystemC/C++ simulation or RTL simulation or Real Board(As long as the trace can be extracted at the proper IP boundary)
- Trace methodology allows us to map traffic patterns to any system bus (VBUS, AXI, OCP, etc.), and therefore, can be used to evaluate multiple vendor IP's with the same trace/pattern.
- Traces can be compressed and trimmed to bring out the worst case scenario. It enables to reduce the simulation time and study critical section of the Usecase
 - Trace can be handcrafted to create worst case scenario which is difficult to reproduce in real application.

This methodology can be used for any IP, not just DDR controller.

IV. DETAILS ON TRACE BASED SIMULATION INFRASTRUCTURE



A. Step1 → Input Trace Generation

This involves running the new Usecase on the existing Simulation Platform. The existing Simulation Platform can be SystemC or RTL based. The Simulation Platform would be using the existing DDR controller. The DDR controller trace is captured at the input interface of the DDR controller from the System. Each transaction would be capture as below packet in the trace

* Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors*).

| Transaction Start Time | Master ID | Access Type (Read/Write) | Address | Size | Priority |
|------------------------|-----------|--------------------------|---------|------|----------|
|------------------------|-----------|--------------------------|---------|------|----------|

B. Step2 → Trace Reformatting

The trace generated by Step1 is taken as input to Trace Playback Infrastructure. It would use the below logic to compress the traffic for each packet.

- It would honor the Transaction start time , if there is a large delay between the current and previous transaction
- else it would push the transaction as soon as the memory is ready to take new request.

This logic is important to remove delays caused due to the existing DDR controller which was used to generate the trace. The idea is, if the transactions are close then they would appear back to back to the DDR controller and hence they should be compressed in the trace file. If the transactions are separated by large distance then they are isolated transaction and should not appear back to back to the DDR controller

C. Step3 → Result and Comparison

The trace generated by Step2 is send across to each DDR controller Vendor. The Vendors can run this traffic over SystemC/RTL model of their IP. Results are shared as “Response Trace” and “Summary Tables”.

Summary Table contained below details

- Bandwidth in Mega Bytes per second (MBps) (Overall and for individual master).
- Average, Minimum and Maximum latency (Overall and for individual master).
- DDR utilization percentage

Response Transaction in response trace contained below information (It is equivalent to Input trace with extra information on End time)

| Transaction Start Time | Transaction End Time | Master ID | Access Type (Read/Write) | Address | Size | Priority |
|------------------------|----------------------|-----------|--------------------------|---------|------|----------|
|------------------------|----------------------|-----------|--------------------------|---------|------|----------|

Response trace can be used in Trace analyzing tools to do more detail study. Some of the charts generated along with their purpose are given below

V. CONCLUSION

This method was successfully used in evaluating DDR controller from multiple vendors for the Next generation SOC device in Texas Instruments (TI) for Automotive and Communication Infrastructure space. It helped to share information with vendor in a more effective way rather than just numbers on datasheet and documents. It enabled validation of numbers shared by the vendors over the datasheet and also helped to evaluate the controller with real usecase relevant for the TI Business. Some of the usecase was replaced with handwritten trace to mimic worst case scenario which was difficult to generated using applications

REFERENCES

- [1] Ashley Stevens, “QoS for High-Performance andPower-Efficient HD Multimedia”