A Common Memory Model for SoC Architecture and Software Models using SystemC/TLM2 Common Interface

Ali Poursepanj, Ph. D.

System Architecture Group
Network Components Division
LSI Incorporated

2/28/2011

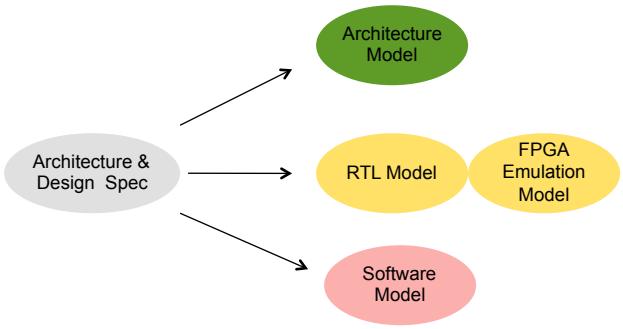
Ali's Bio

- Ali is currently a Distinguished Engineer in the System Architecture Group at LSI Network Computing Division in Austin Texas
- Ali has near 20 years of experience in processors, systems and telecom SoC performance modeling and analysis. He has worked as an architect and lead modeling engineer at Freescale, AMD, IBM, and Motorola.
- Ali obtained his Ph.D. In Electrical and Computer Engineering from the University of Texas at Austin
- Ali has published several papers in the area of performance modeling and analysis in the IEEE, ACM and EE Times publications
- Ali has been representatives of several companies in standard committees such as Power.org, Network Processor Forum(NPF), and EEMBC.
- Ali is married with three daughters and lived in Austin for nearly 30 years
- Ali can be contacted via LinkedIn at poursepanj@yahoo.com

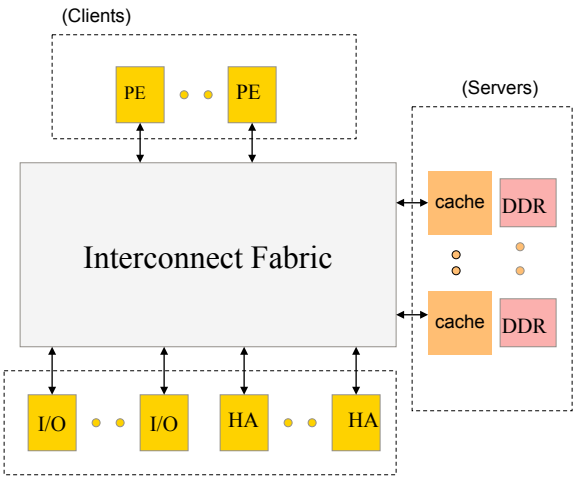
Outline

- Overview
- Architecture Models
- Software Models
- Sharing a Common Memory Model using TLM 2.0
- Summary

Model Evolution in the Chip Design Cycle

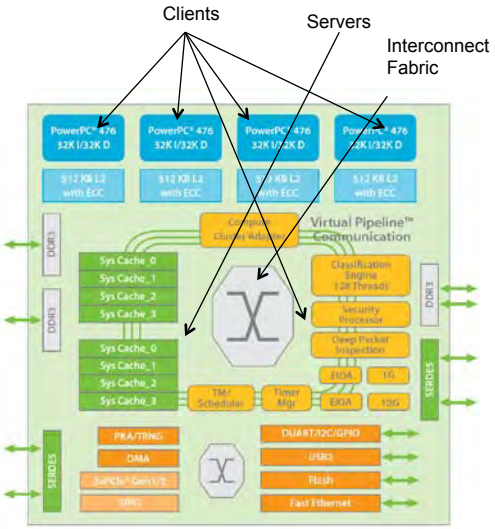


A Typical System on a Chip (SoC) Topology

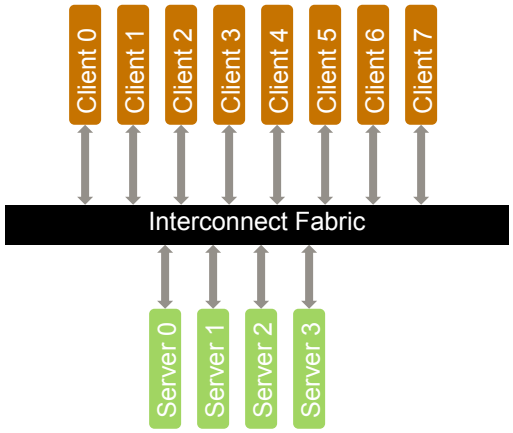


HA = Hardware Accelerator
I/O = IO Devices
PE = Processing Elements

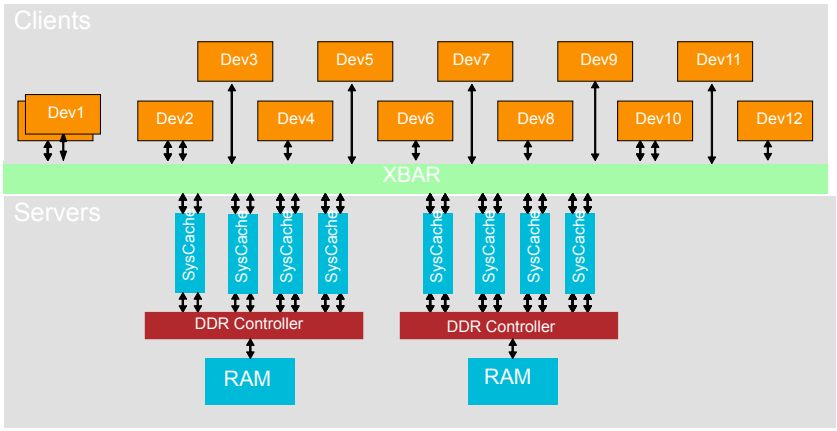
LSI® ACP3400 Multi-Core Communication Processor SoC



An SOC Client–Server Abstract Architecture Model



ACP3400 Multi-Core Communication Processor SoC Architecture Model Block Diagram



DevX = Device X, Client X, Engine X

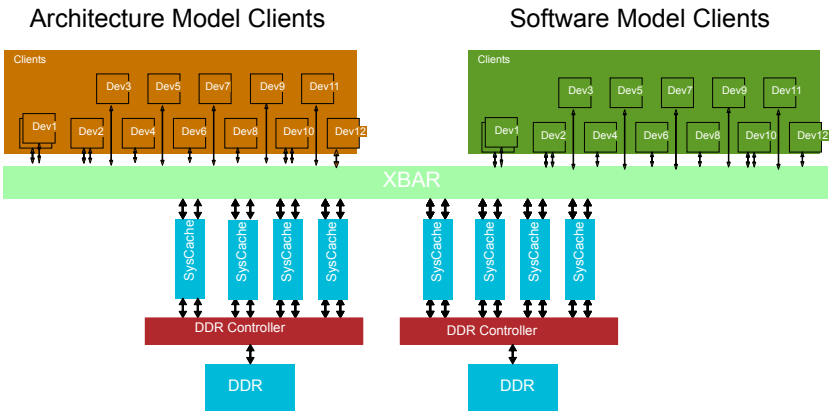
Architecture Model

- Clients: Traffic Generators
- Servers: System Cache, Memory controller, and DDRs
- Interconnect Fabric
- Memory access latency and memory bandwidth utilization are two important performance metrics needed for architecture analysis
- Memory model has to be reasonable accurate
- Memory accesses by clients have to be modeled reasonable accurate
- Client register accuracy in this is not required as it is in the software/functional model
- Clients can be modeled as transaction level traffic generator

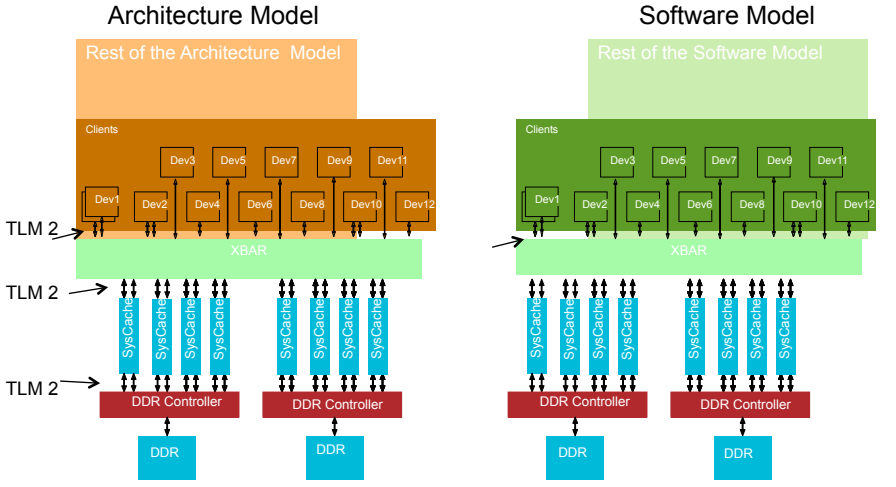
Software Model

- Clients: accurately modeled, functional and register accurate
- Servers: System Cache, Memory controller, and DDRs
- Interconnect Fabric
- Quad PowerPC® 476 processors
- Acceleration Engines
  - Packet Processing Classification
  - Security Engine (Crypto)
  - Stream Editor
  - Deep Packet Inspection
  - Traffic Manager/Scheduler
- Engine Communication Rings
  - Virtual Pipeline™
  - Memory Management
  - Timer Services
- Register accurate and Functionally correct
- More ..

Software and Architecture Models Sharing the System Memory

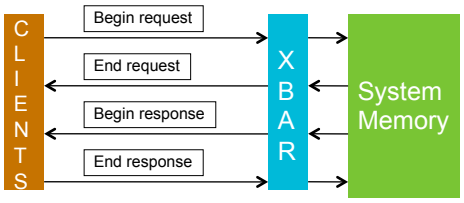


Software and Architecture Models Sharing the System Memory



Implementation

- Software and architecture models were developed for the AXXIA™ Communication Processor by LSI engineers and were used in the production environment
- Models were all developed in SystemC/TLM 2.0
- TLM 2.0 OSCI was used for memory interface. This allowed the reusability of it for both architecture and software model
- Generic 4-phase protocol was used:



TLM 2.0 features and utilities used

- Approximately-timed (AT) protocol-coding style
- Non-blocking transport interface
- Payload event queues
- Using utilities that allow us do many to many multi-socket binding

Summary and Conclusions

- Architecture and software models are developed during the SoC design cycles before the silicon becomes available
- Development of these models takes time and effort
- Reusability of modules across these models reduces the development time
- SystemC/TLM 2.0 common interface is useful for reusability of the modules
- This approach enables modeling of different arbitrary configurations. That is, the number and type of devices/ clients can easily be changed.
- This method enables modeling of many alternatives which is necessary for architecture exploration and analysis