Implementation of Verification IP for ACE Cache Coherency Protocol
Sameeksha, Sivaprasanth Iluppur Thiyagarajan
Department of Electrical Engineering, San Jose State University, San Jose, California 95192.

INTRODUCTION
- ACE cache coherency protocol is a complex protocol to verify.
- Verification IP are hard for customers to understand.
- Goal of the project is to reduce effective time involved in verification and achieve early time to market.
- Enable plug and play feature to make the VIP importable to any cache coherency protocol system.
- Project increases scalability and reusability by adopting Universal Verification Methodology.

FEATURES

- **Reusable Components**
  - ace master agent
    - Has a driver, sequencer, monitor and a predictor.
    - Can be made active or passive through agent configuration class.
  - ace master driver
    - Converts packets into interface signals.
    - Has a handle for virtual interface(vif).
  - ace master monitor
    - Reads pin level signals and transfers into packets.
    - Has an analysis port which transfers packet to the scoreboard.
  - Each components are connected together through TLM ports.
  - TLM ports enhances reusability and portability by making components independent of each other.
  - Predictor is used for unit level verification when single master is used.

SYSTEM ARCHITECTURE

- Supports 128 masters.
- Has a virtual sequencer to route packets between 2 different agents.
- System configuration class is used for environment configuration.
- Sequences randomizes the sequence item and transmits to the sequencer.
- Driver receives the packet from the sequencer and drives it to the interface.
- System level scoreboard is used to check transaction across different masters.

BUS FUNCTIONAL MODULE

- Driver has 2 ports – one for sending coherent transactions and another for snoop transactions.
- Supports both AXI and ACE cache coherency Interface.
- Agents can be active or passive.
- Cache model is associated with each agent.
- scoreboard is used for both unit level testing and block level testing.

RESULTS AND OBSERVATIONS

- Snoop master gives a shared status to the Interconnect.
- Interconnect doesn’t provides data to the Initiating master.

CONCLUSION

- Supports up to 128 masters.
- Supports both unit level and block level testing.
- Cache coherency is demonstrated and verified using block level testing.
- Supports all kinds of coherent and snoop transactions.
- Designer can easily modify the VIP according to his specification needs.
- Protocol checks are done using SystemVerilog Assertions(SVA).

FUTURE WORK

- To support DVM and barrier Transactions.
- To support snoop filter.
- To support all kinds of cache.
- To implement a cache controller.

REFERENCES


ACKNOWLEDGEMENTS

The authors wish to thank and appreciate the valuable guidance of Dr. Thuy T. Le throughout the duration of our project. His time and advice helped us narrow down and overcome the issues we faced while working on our project.

---

**Introduction:**

ACE cache coherency protocol is a complex protocol to verify. Verification IP are hard for customers to understand. The goal of the project is to reduce effective time involved in verification and achieve an early time to market. The project enables plug and play feature to make the VIP importable to any cache coherency protocol system. The project increases scalability and reusability by adopting Universal Verification Methodology.

**Features:**

**Reusable Components**

- **Ace master agent**
  - Has a driver, sequencer, monitor and a predictor.
  - Can be made active or passive through agent configuration class.
- **Ace master driver**
  - Converts packets into interface signals.
  - Has a handle for virtual interface(vif).
- **Ace master monitor**
  - Reads pin level signals and transfers into packets.
  - Has an analysis port which transfers packet to the scoreboard.
- Each components are connected together through TLM ports.
- TLM ports enhances reusability and portability by making components independent of each other.
- Predictor is used for unit level verification when single master is used.

**System Architecture**

- Supports 128 masters.
- Has a virtual sequencer to route packets between 2 different agents.
- System configuration class is used for environment configuration.
- Sequences randomizes the sequence item and transmits to the sequencer.
- Driver receives the packet from the sequencer and drives it to the interface.
- System level scoreboard is used to check transaction across different masters.

**Bus Functional Module**

- Driver has 2 ports – one for sending coherent transactions and another for snoop transactions.
- Supports both AXI and ACE cache coherency Interface.
- Agents can be active or passive.
- Cache model is associated with each agent.
- Scoreboard is used for both unit level testing and block level testing.

**Results and Observations**

- Snoop master gives a shared status to the Interconnect.
- Interconnect doesn’t provides data to the Initiating master.

**Conclusion**

- Supports up to 128 masters.
- Supports both unit level and block level testing.
- Cache coherency is demonstrated and verified using block level testing.
- Supports all kinds of coherent and snoop transactions.
- Designer can easily modify the VIP according to his specification needs.
- Protocol checks are done using SystemVerilog Assertions(SVA).

**Future Work**

- To support DVM and barrier Transactions.
- To support snoop filter.
- To support all kinds of cache.
- To implement a cache controller.

**References**


**Acknowledgements**

The authors wish to thank and appreciate the valuable guidance of Dr. Thuy T. Le throughout the duration of our project. His time and advice helped us narrow down and overcome the issues we faced while working on our project.