

Performance Modeling Using TLM-2.0

ZHU ZHOU, ATUL KWATRA

Acknowledgements: Trevor Wieman, Brian Mears

Outline

- Architectural Exploration and Performance Modeling
- TLM-2.0 AT Modeling Experiences
- Modeling Split Transactions
- LT/AT Switching Mechanism
- OSCI Standard Gaps
- Summary

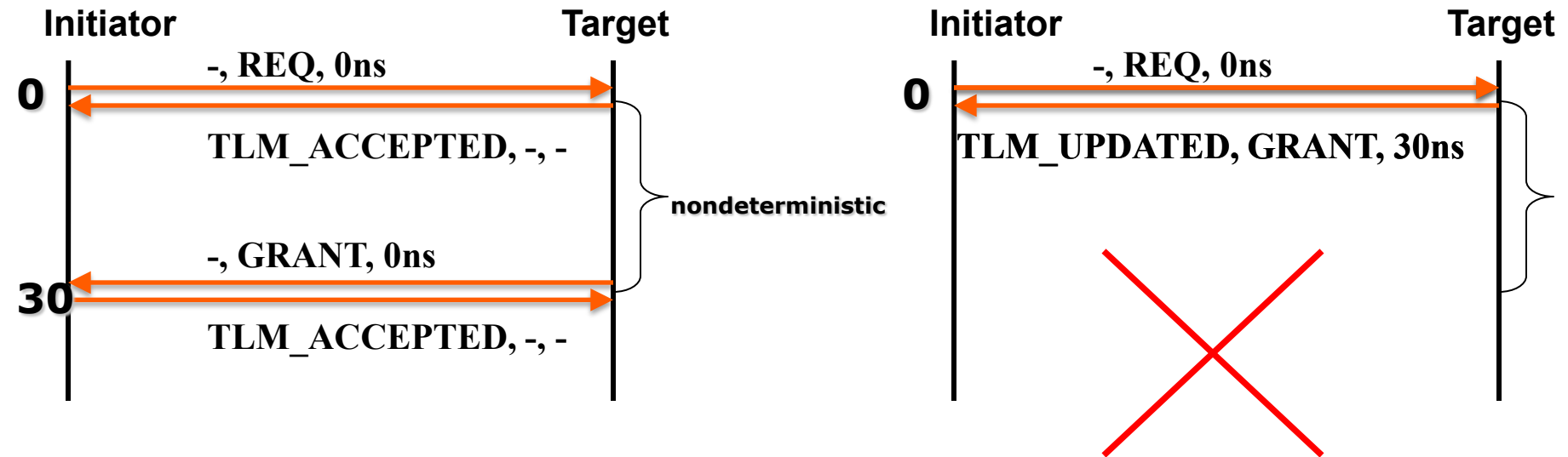
Architectural Exploration & Performance Modeling

- Architectural exploration is performed to optimize power/performance/cost
- Requirements
 - Optimizations best done in parallel for best trade-off analysis
 - Modularity and configurability for micro-architecture fine-tuning and studies of system level scalability
 - Use real workloads to prevent overdesign or underdesign
 - Standard approach to integrate likely 3rd party IPs for most SoCs
- Performance modeling
 - Predicts throughput and latency for various micro-architecture configurations
 - Accuracy is #1 priority followed by performance and interoperability
 - Timing accuracy required for:
 - Interconnect protocol, buffering, arbitration, ordering, flow control, deadlock avoidance, back pressure, pipelining, etc.

TLM-2.0 AT Modeling – Interoperability vs. Extensions

- Interoperability enabled by having a common base protocol
 - Easy to achieve for LT but not AT
 - AT Interoperability loses meaning if accuracy not achieved
- AT payload & phase extensions allow customized protocol
- Interoperability should be addressed by standardizing payload and extensions for a given bus standard
 - i.e. bus specific extensions for PCI-Express, AXI, OCP, etc.
 - Build model repositories to support those protocols
 - Only way to achieve accuracy and interoperability

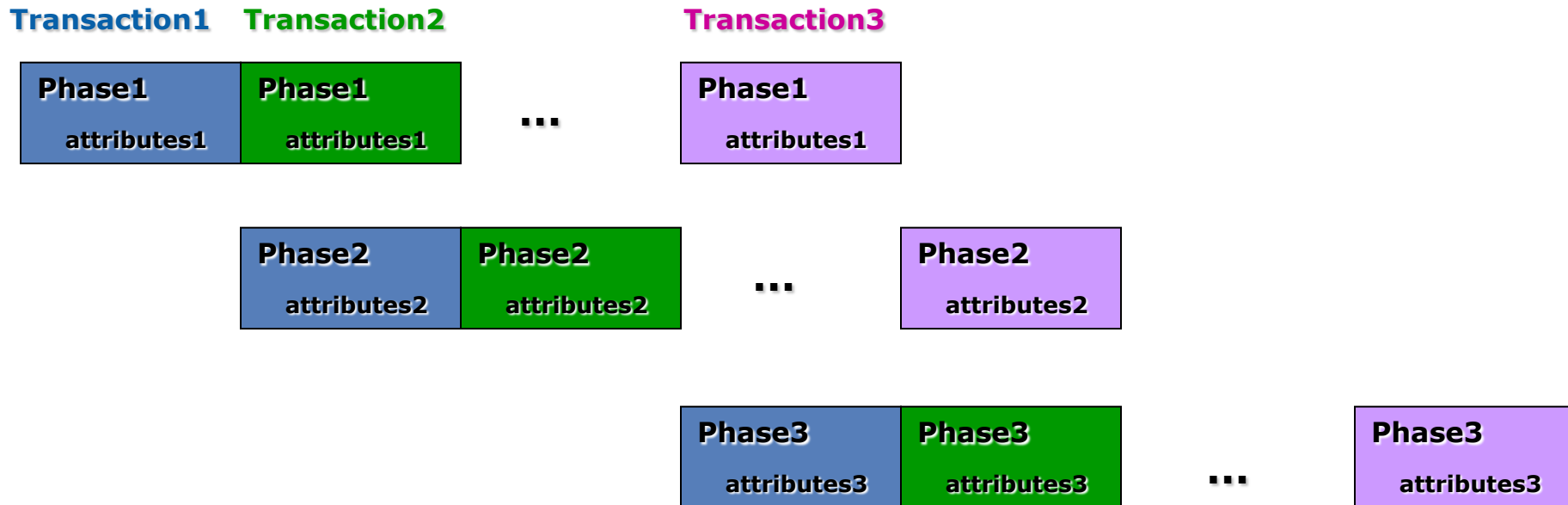
TLM-2.0 AT Modeling – No TLM_UPDATED



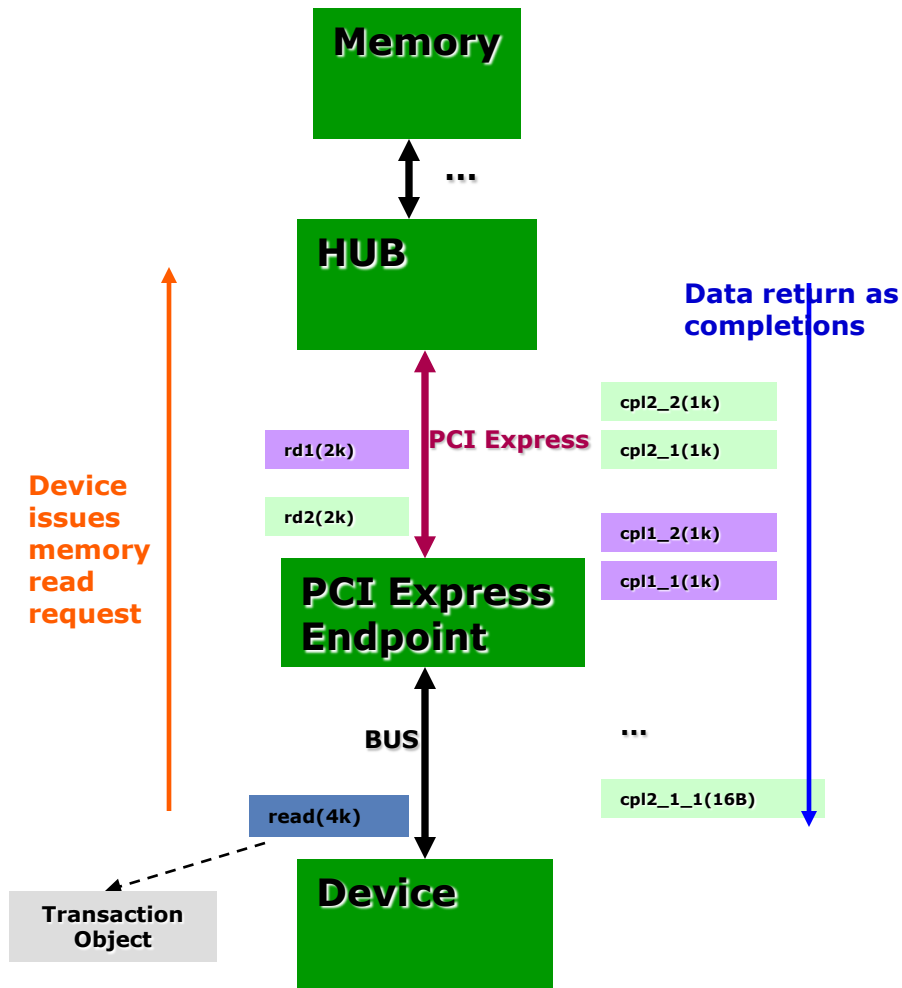
- Phase used to define the nondeterministic timing point caused by dynamic behavior such as arbitration, ordering, etc.
- No use of return path to approximate elapsed time

TLM-2.0 AT Modeling – Debug/Validation

- SCV based transaction recording heavily used for debugging and protocol validation
- Pipe trace visualization necessary to verify timing and protocol logics (arbitration, etc.)



Modeling Split Transactions



- Completions modeled as separate transactions (mimic hardware)

- Performance
- Easier switch to LT

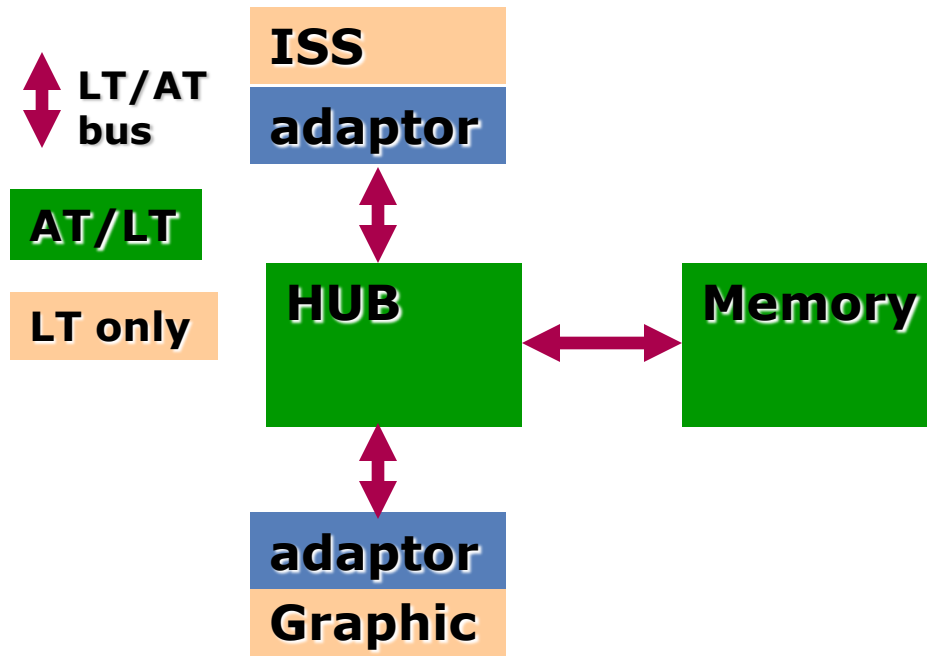
- Challenges

- Address and size different for each burst
- Out-Of-Order
- backward socket path

- Solution

- Add a vector element in the extension for each burst
- Pass-by-value for the index to the specific extension when Out-Of-Order happens

LT/AT Switching Use Case



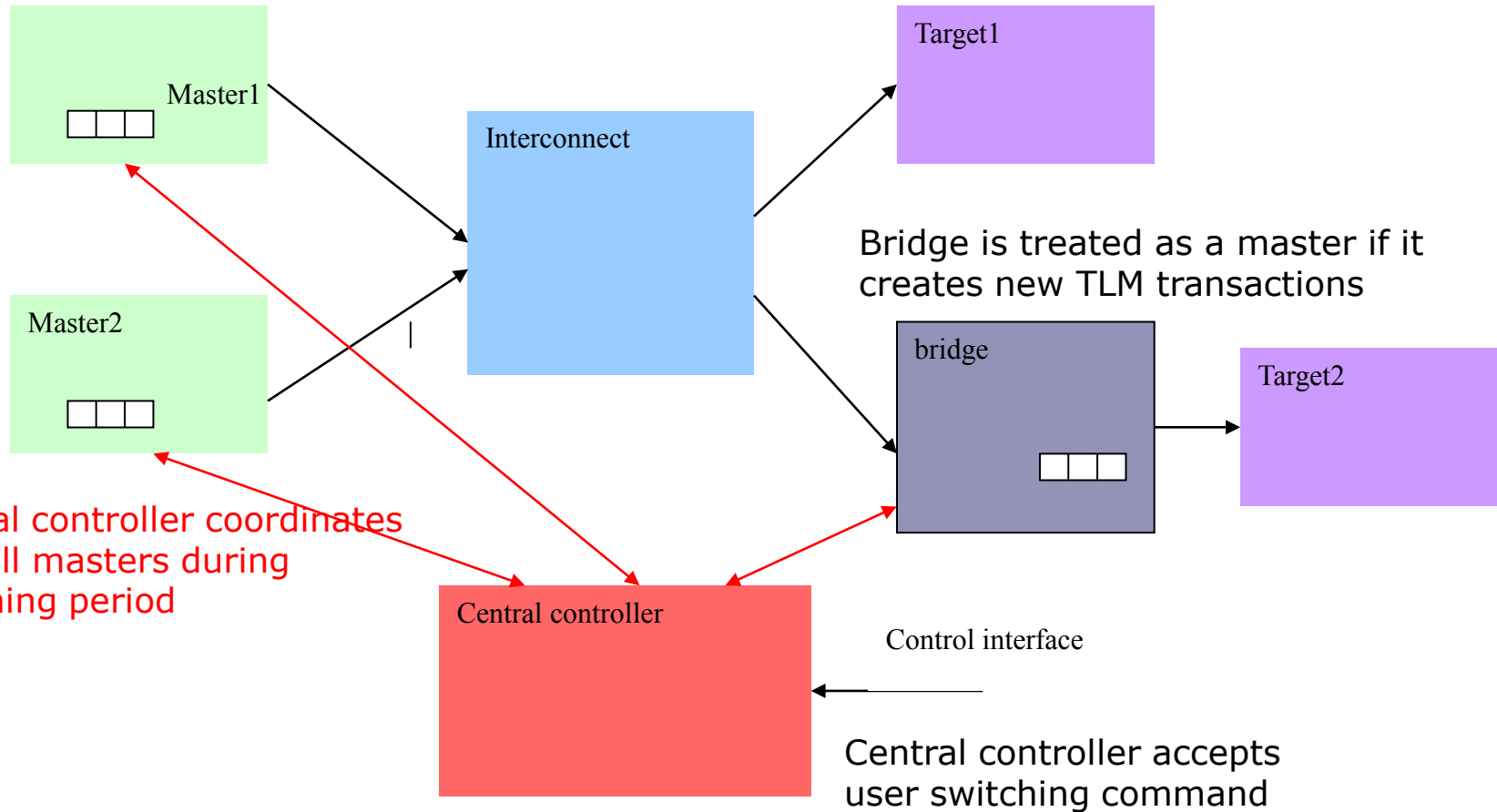
Switching Scenario:

1. System starts with LT mode
2. Switch-able units (HUB, Memory) together with adaptors switch to AT. Performance data collected
3. Switch back to LT, functionality remains
4. Loop through step 2 and 3 if needed

- Today: Traffic generator with statistical patterns used
- Mixed LT/AT provides realistic traffic sources and better synchronization for architectural optimization
- Today: Performance model and functional model normally separate
- TLM-2.0 enables combination of two models into one for efficiency and better correlation

Generic LT/AT Switching Mechanism

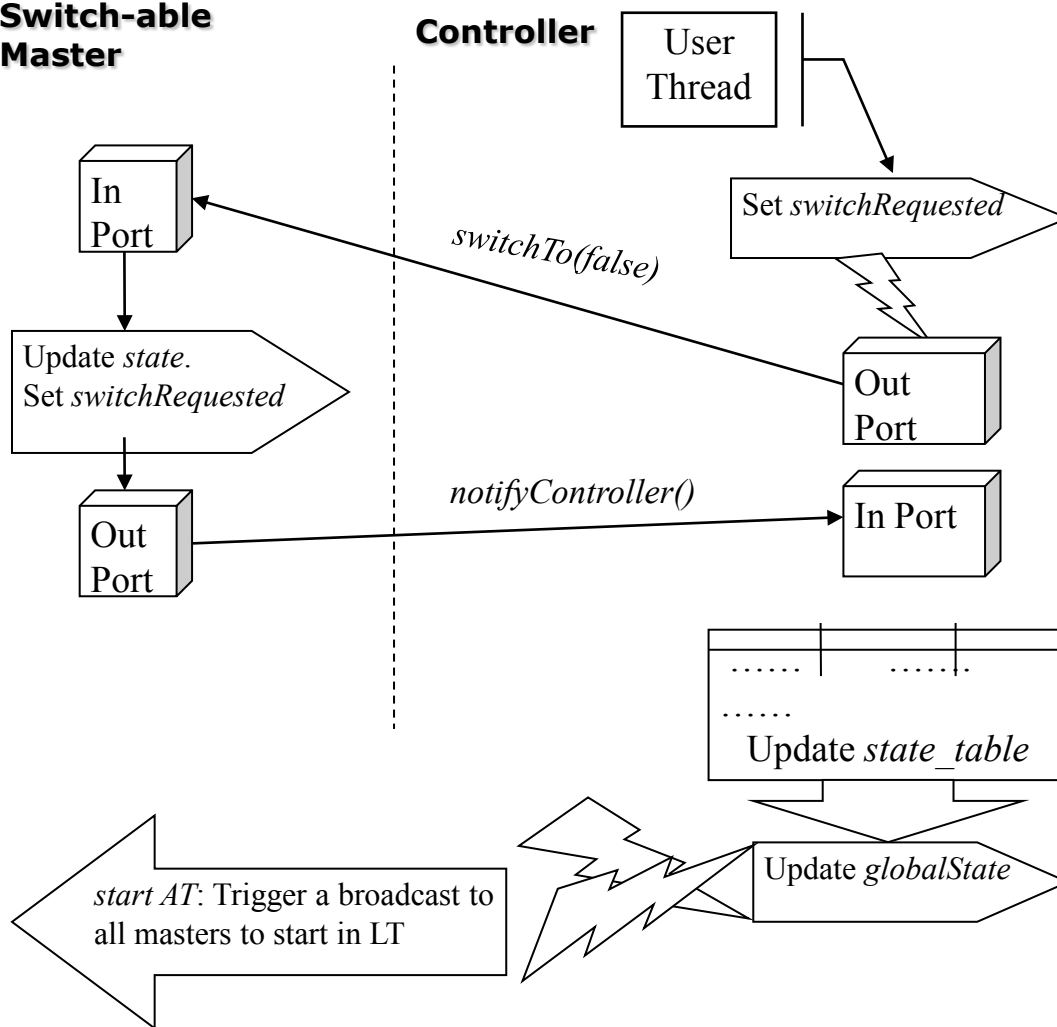
Each master registers itself to controller during elaboration



Switching Protocol (LT->AT)

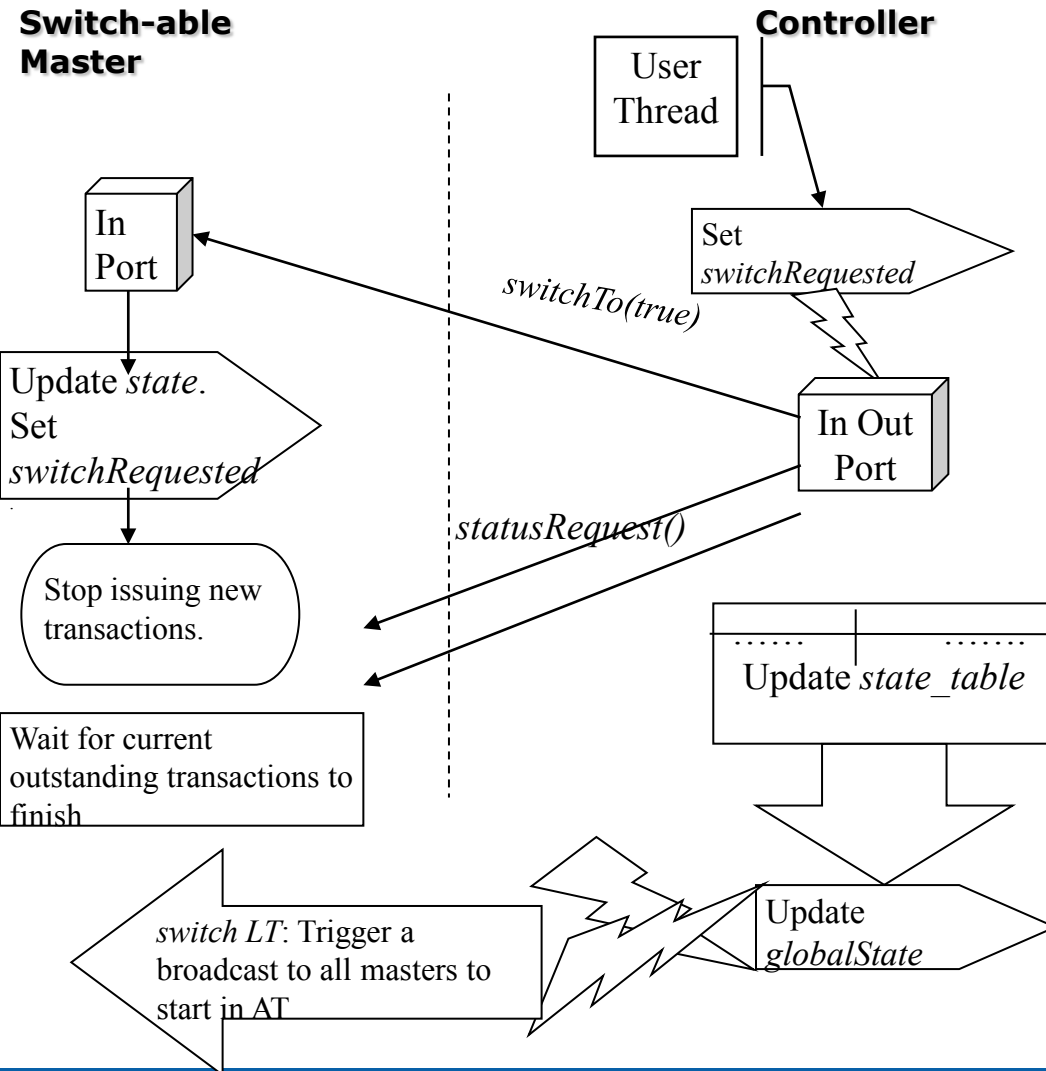
Switch-able Master

Controller



1. User enters LT->AT switch command
2. Controller broadcasts to all masters
3. Each master finishes its blocking call then notifies the controller
4. Controller updates its internal table
5. Only when all masters finish LT calls, controller broadcasts switching instruction
6. All masters start to issue AT transactions in non-blocking fashion

Switching Protocol (AT->LT)



1. User enters AT->LT switch command
2. Controller broadcasts to all masters
3. Each master stops issuing new transaction while continuing to finish outstanding AT transactions
4. Controller periodically polls each master
5. Only when all masters complete outstanding AT transactions, controller broadcasts switching instruction
6. All masters start to issue LT transactions

OSCI Standard Gaps

- Standard interfaces for configurability, controllability, analysis, visibility and register modeling
 - These topics are being addressed by OSCI's recently formed Configuration, Control and Inspection (CCI) WG; visit OSCI's web site for details
- Standard Power and Cost (die size) modeling interface/infrastructure and generic methodology
- Methodology to handle mixed LT/AT system and LT \leftrightarrow AT switching handling
 - Coding guideline for switch-able model
- Comprehensive verification methodology on TLM models
 - Temporal assertions (protocol level checker)
 - Formal Verification
 - Code/Datapath/functional coverage

Summary

- TLM-2.0 strengths include Flexibility, Performance, Usability and Documentation
- Experience gained in using TLM-2.0 AT for
 - Internal TLM-2.0 AT modeling guidelines
 - Split transaction modeling
- AT/LT switching scheme developed
- Future standard focus areas include:
 - Infrastructure – standard model access interfaces
 - Generic modeling methodologies such as LT/AT Switching, TLM verification, etc.