

MPSoC '07

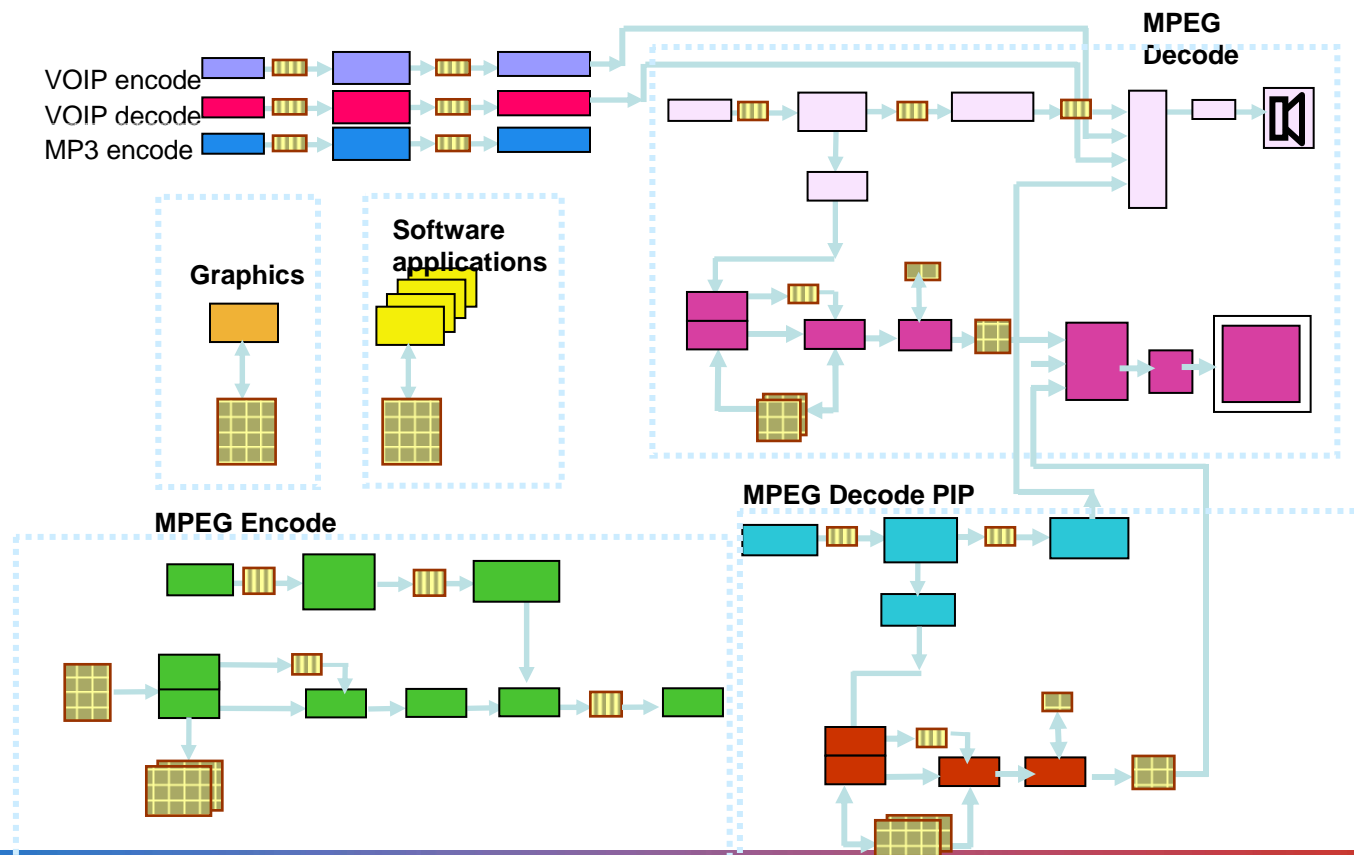


High-performance Multithreaded Memory Subsystems for MPSoC's

Drew Wingard, Sonics

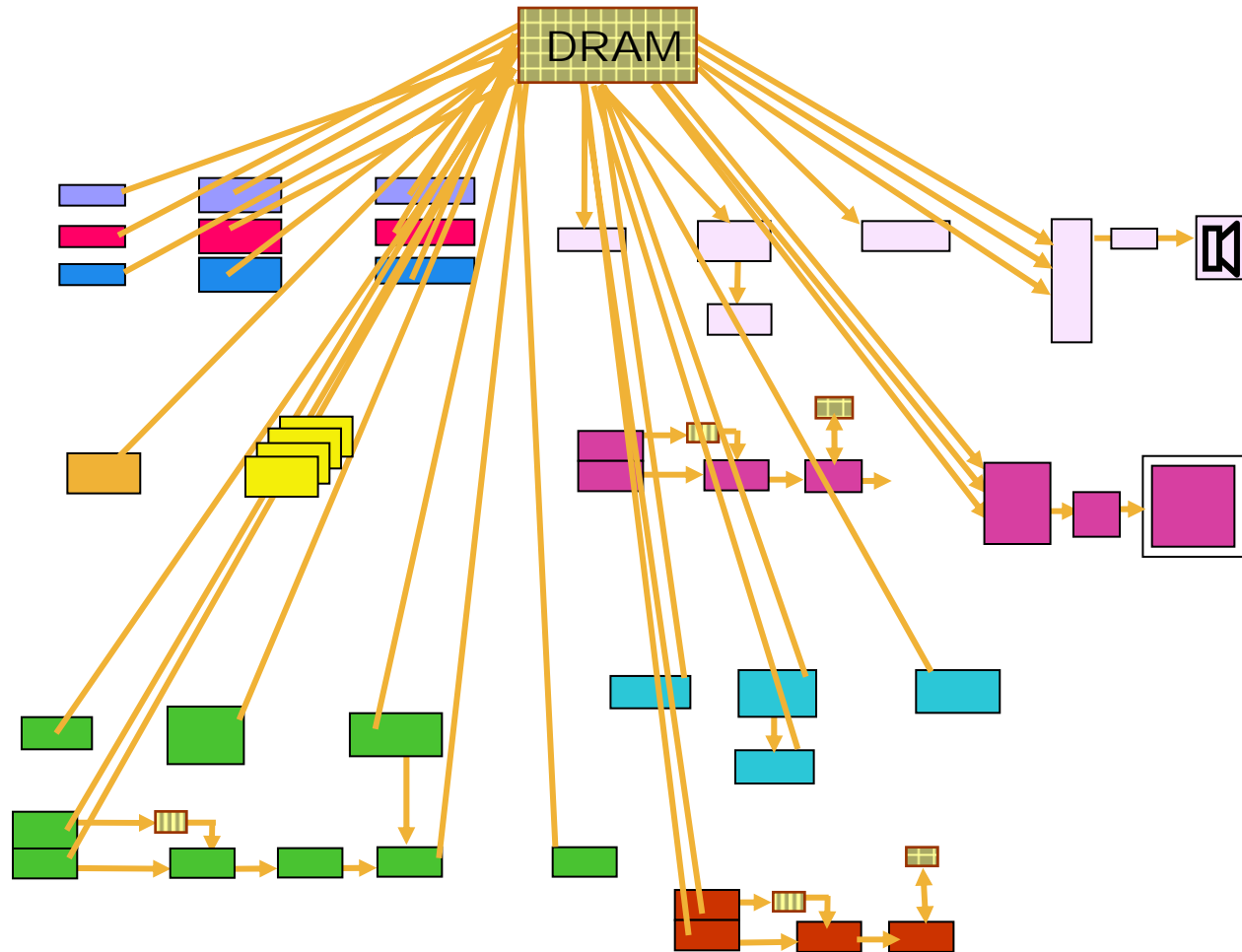
Interconnect-level Concurrency in SoC

- Consumer MPSoC's process lots of data in parallel, but communicate...



Interconnect-level Concurrency in SoC

- Consumer MPSoC's process lots of data in parallel, but communicate...

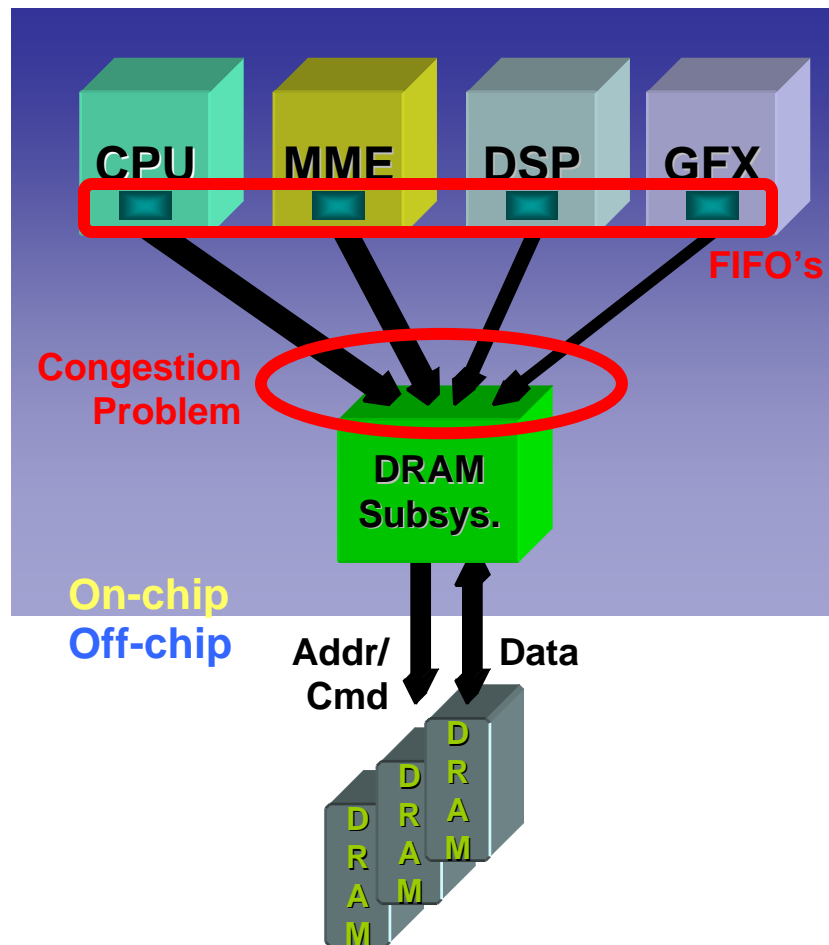


Interconnect-level Concurrency in SoC

- Consumer MPSoC's process lots of data in parallel, but communicate...
- Assertion: SoC applications have >> 50% of traffic to DRAM
- Implications:
 - Most of network is a fanin tree to a single DRAM
 - Maximizing delivered DRAM efficiency is key

Star Topology Memory Subsystems

Traditional Approach



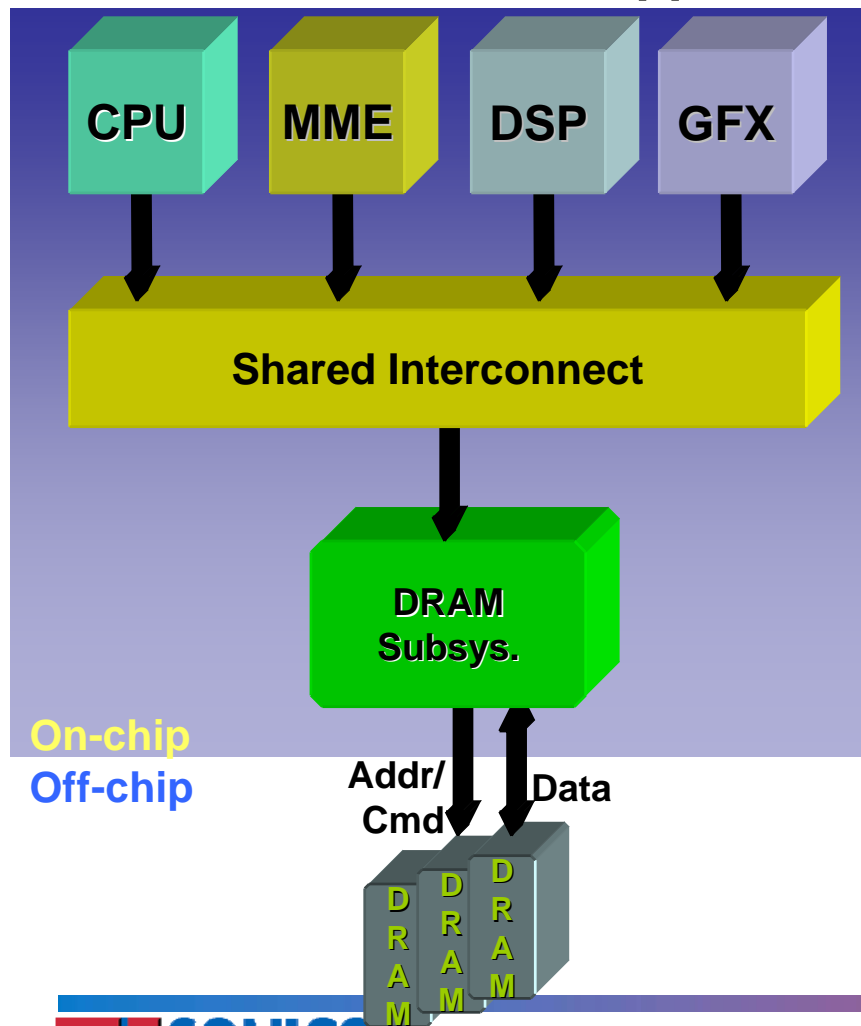
- Initiators present requests in parallel to multi-port scheduler
- FIFO's at initiators provide
 - Rate decoupling
 - Service jitter tolerance
- DRAM subsystem needs no FIFO, only pipelining
- System performance limited only by traffic & scheduler

But:

- LOTS of wires/congestion
- Lots of small/inefficient FIFO's
- Large part of system must be BW matched to DRAM

Single-ported Memory Subsystems

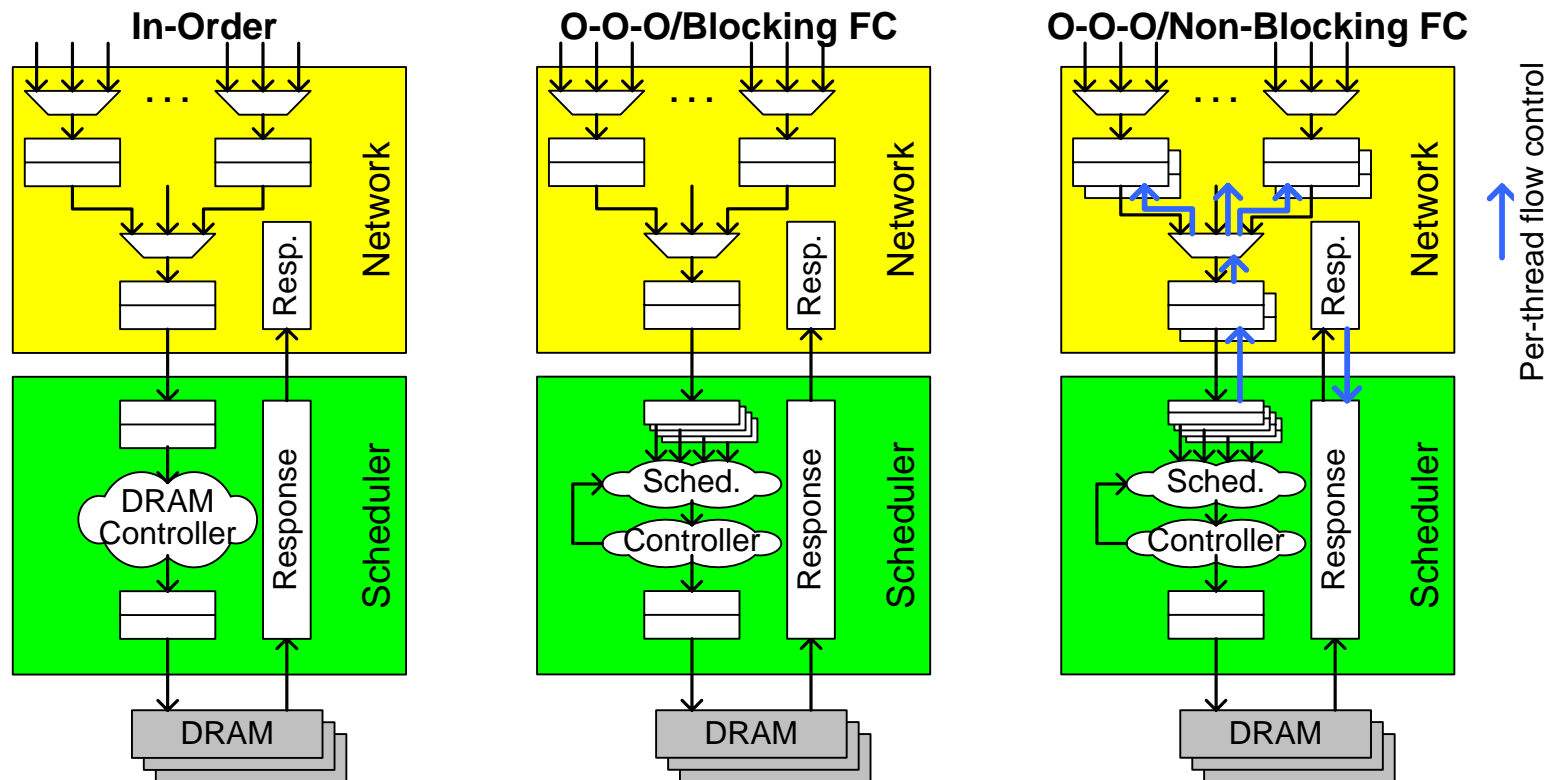
Shared Interconnect Approach



- Interconnect presents requests in series to single-port scheduler
 - Saves wires/congestion
- But:
- Interconnect arbitration impacts scheduling
 - Risks lower utilization
 - May not meet deadlines
 - Where do FIFO's live?
 - How much of system is BW-matched to DRAM?
 - System performance also limited by communication system

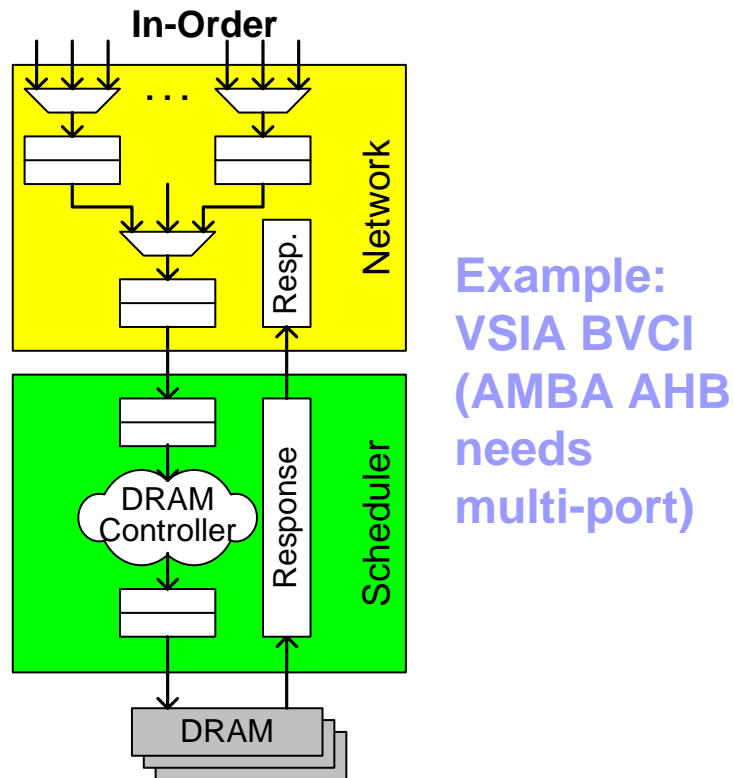
Single-port DRAM Protocols

- Interconnect and subsystem must support multiple outstanding requests (cover DRAM pipeline depth)



In-order Protocol

- Interface protocol supports multiple outstanding burst requests, but all service matches request order

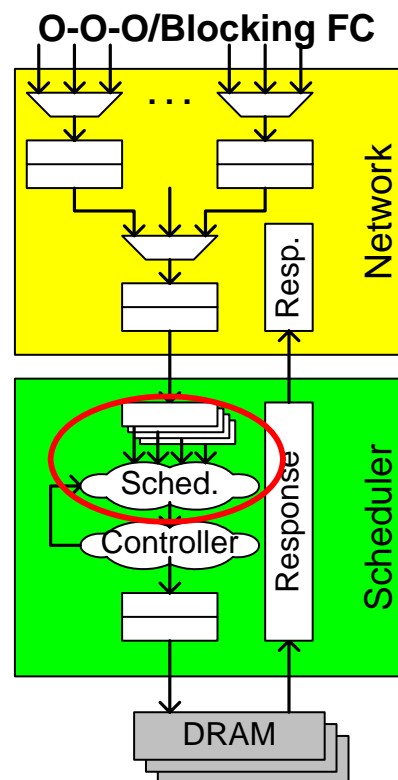


- Simplest scheme (lowest hardware cost)
- Service order determined by interconnect arbitration
- Scheduler can only optimize pipeline (looking ahead for page misses to other banks)
- High efficiency requires long bursts, leads to high latency (poor QoS)

Out-of-order Protocol with Blocking FC

- Interface protocol provides ordering tags to allow scheduler to reorder some requests, but flow control is shared across all tags

Examples:
AMBA AXI
OCP Tags



- Interconnect presents requests in order
- Scheduler queues requests & chooses order to optimize throughput and QoS

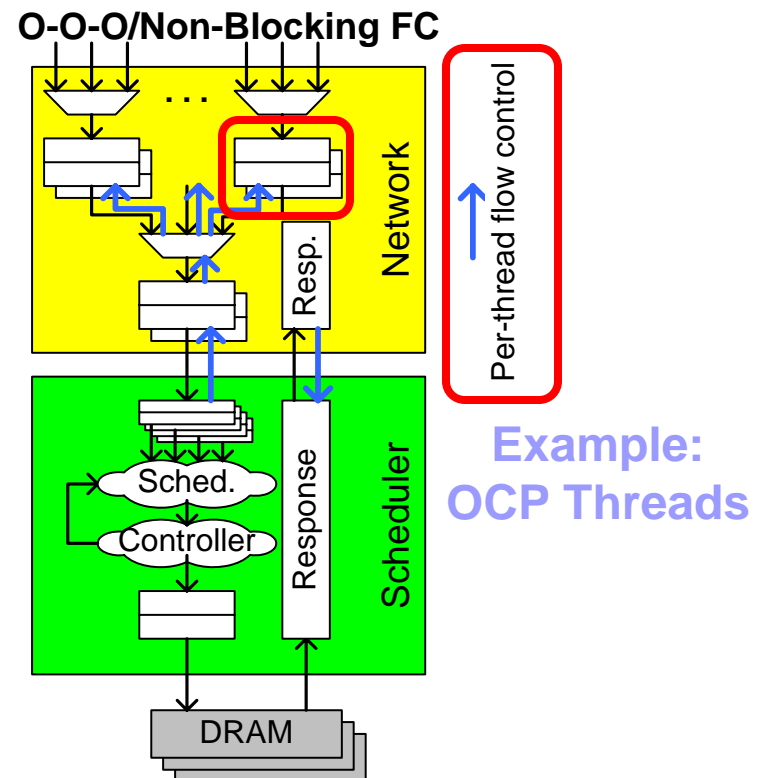
But

- Bursty flows can fill queues, hurting latency & BW for others
- Full queues block into network
- Frequency scales poorly with queue depth

Out-of-order Protocol with Non-blocking FC

- Interface protocol provides per-thread ID's and flow control, enabling re-ordering while preventing blocking

- Interconnect maps initiator threads into target threads
- Scheduler queues requests & chooses order to optimize throughput and QoS **on per-thread basis**
- Non-blocking (per-thread) flow control minimizes inter-thread interactions
- Per-thread queues inherently ordered, implemented as compiled SRAM
- Result: lower latency, BW guarantees, higher guaranteed throughput



Single-port DRAM Subsystem Protocols

Ordering/ flow control	In-order/ blocking	Out-of-order/ blocking	Out-of-order/ non-blocking
Peak BW limited by	DRAM	DRAM	DRAM
Ordering flexibility	None	High	High
Queuing	None	Shared	Per-thread
Compiled RAM-friendly	No	No	Yes
Init. BW==DRAM BW	Yes	Yes	No
DRAM efficiency	Medium	High	High
Max. CPU latency	High	Medium	Low
Data interleaving	None	Minor	High

Example: Toshiba Super Companion Chip

Cell companion chip gets hot demo

By Michael Kanellos CNET News.com

Published on ZDNet News August 15, 2005

Toshiba showed off a "super companion chip," for the Cell that can record 48 separate MPEG 2 streams at once.

Example: Toshiba Super Companion Chip

TOSHIBA

Key Architecture (1/2)

■ Internal Bus Architecture

- Quality of Service (QoS)
 - Bandwidth allocation by bus arbitration

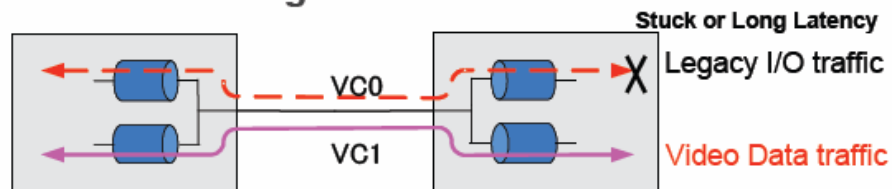
e Cell

TOSHIBA

Key Architecture (2/2)

■ Other Features

- Virtual Channel Mechanism
 - To avoid blocking for data flow.



- Multiple thread mechanism
 - Multi/Single threads are alternative.
- Pipelined data processing
- Data ordering mechanism

Courtesy: Toshiba

Example: Toshiba Super Companion Chip

MOUNTAIN VIEW, Calif.—February 6, 2007— Sonics Inc., ... today announces that Toshiba Corp. has selected Sonics' SMART Interconnect solution and MemMax memory scheduler as the foundation for their Super Companion Chip for the microprocessor "Cell" that Toshiba jointly developed with IBM and Sony Group.

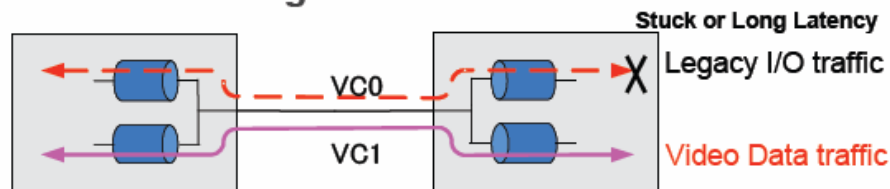
- Quality of Service (QoS)
 - Bandwidth allocation by bus arbitration

TOSHIBA

Key Architecture (2/2)

Other Features

- Virtual Channel Mechanism
 - To avoid blocking for data flow.



- Multiple thread mechanism
 - Multi/Single threads are alternative.
- Pipelined data processing
- Data ordering mechanism

Courtesy: Toshiba



...ce multithreaded
or MPSoC's

Summary

- Multi-threaded DRAM subsystem with non-blocking flow control offers equivalent performance to multi-port subsystems
 - With much less routing congestion and more efficient buffering
- Benefits only apparent when non-blocking maintained end-to-end across interconnect
 - Virtual channel re-mapping should focus on more than deadlock & congestion avoidance