Multi-Buffer Management

A Folded Approach

Shepard.Siegel@atomicrules.com

Designer's Development Notes

2009-06-06 ; revised 2009-11-24

Problem Statement

- Managing Multi-Buffers (Buffer Queues) requires careful synchronization
- Role- and Mode- based specialization is notorious for introducing hazards

So…

- Construct a base-model that can be reused across many roles and modes!

Multi-Buffer Utility

- Maximize concurrency between producer and consumer operations
- Overlap communication with compute
- Minimize inter-message “dead” time
- Maximize throughput, even with high-latency transports, by as-soon-as-possible data advance semantics

Roles and Modes

- Fabric Producer or Fabric Consumer ?
- Locally Passive, Move the Message, or Send Flowcontrol Info?
- Where Applicable: PIO or DMA?

Local Role	 Fabric-Producer (FP)	 Fabric-Consumer (FC)
Passive	1. Partner polls status and reads message data by any means	2. Partner polls status and writes message data by any means
Active Message Movement	3. We receive fabric flowcontrol for far-side "partner-buffer-free"; we **produce message data by Active-Push** DMA	4. We receive fabric flowcontrol for far-side "partner-buffer-full"; we **consume message data by Active-Pull** DMA
Active Flowcontrol Transmission	5. We **Transmit** DMA "local-full" flowcontrol to our partner; Partner reads message data by any means	6. We **Transmit** DMA "local-free" flowcontrol to our partner; Partner sends message data by any means

PIO

	PUSH	PULL
Locally-Active Fabric-Consumer	1. DMA “local free” doorbells to partner	5. Receives far-side full flowcontrol , pull DMA
Locally-Passive Fabric-Consumer	2. Partner polls status and pushes message data	6. Infeasible
Locally-Active Fabric-Producer	3. Receives flowcontrol for far-side free, push message data by DMA	7. DMA “local full” flowcontrol to far-side, partner reads
Locally-Passive Fabric-Producer	4. Infeasible	8. Partner polls status and pulls message data

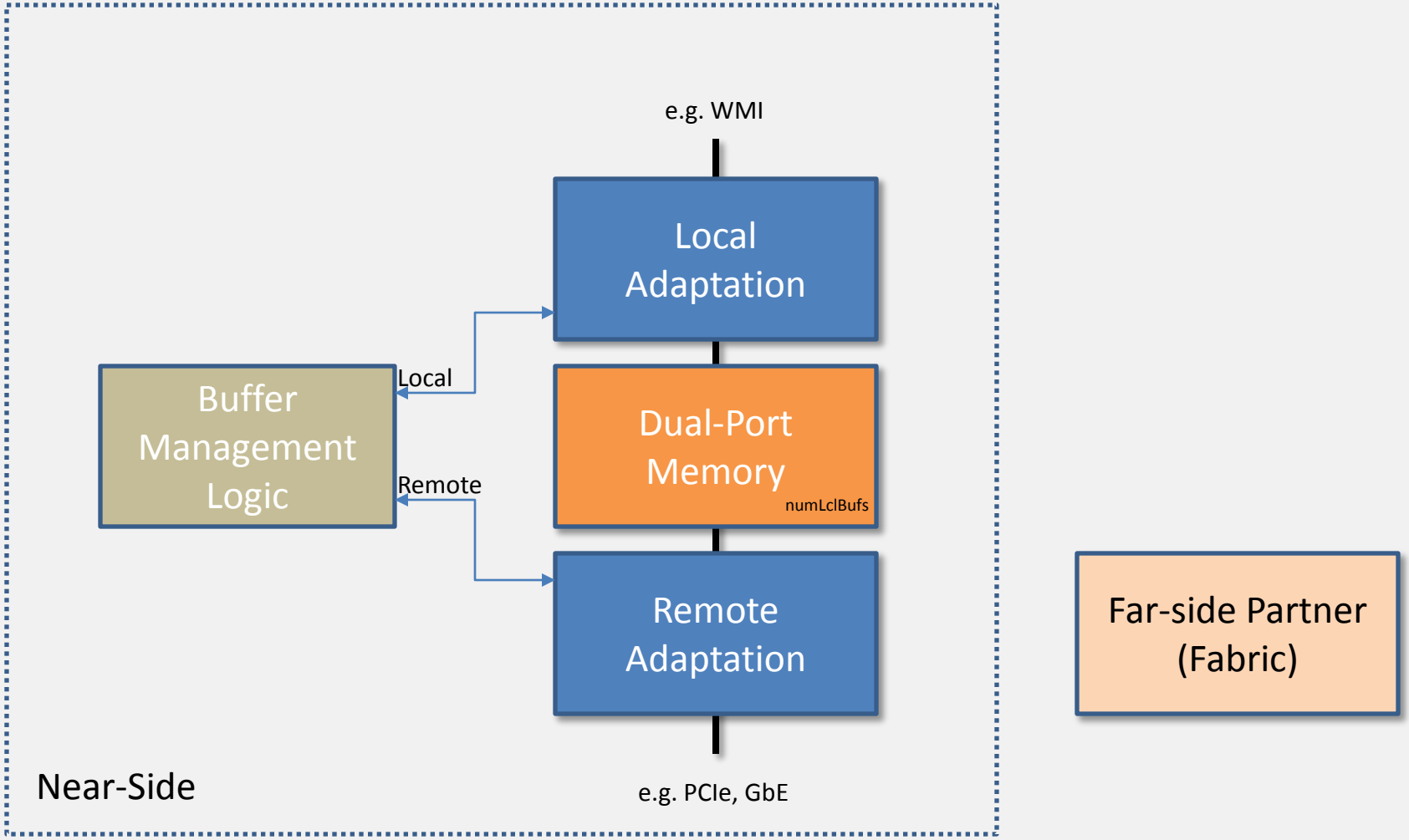
Push-DMA

	PUSH	PULL
Locally-Active Fabric-Consumer	1. DMA “local free” doorbells to partner	5. Receives far-side full flowcontrol , pull DMA
Locally-Passive Fabric-Consumer	2. Partner polls status and pushes message data	6. Infeasible
Locally-Active Fabric-Producer	3. Receives flowcontrol for far-side free, push message data by DMA	7. DMA “local full” flowcontrol to far-side, partner reads
Locally-Passive Fabric-Producer	4. Infeasible	8. Partner polls status and pulls message data

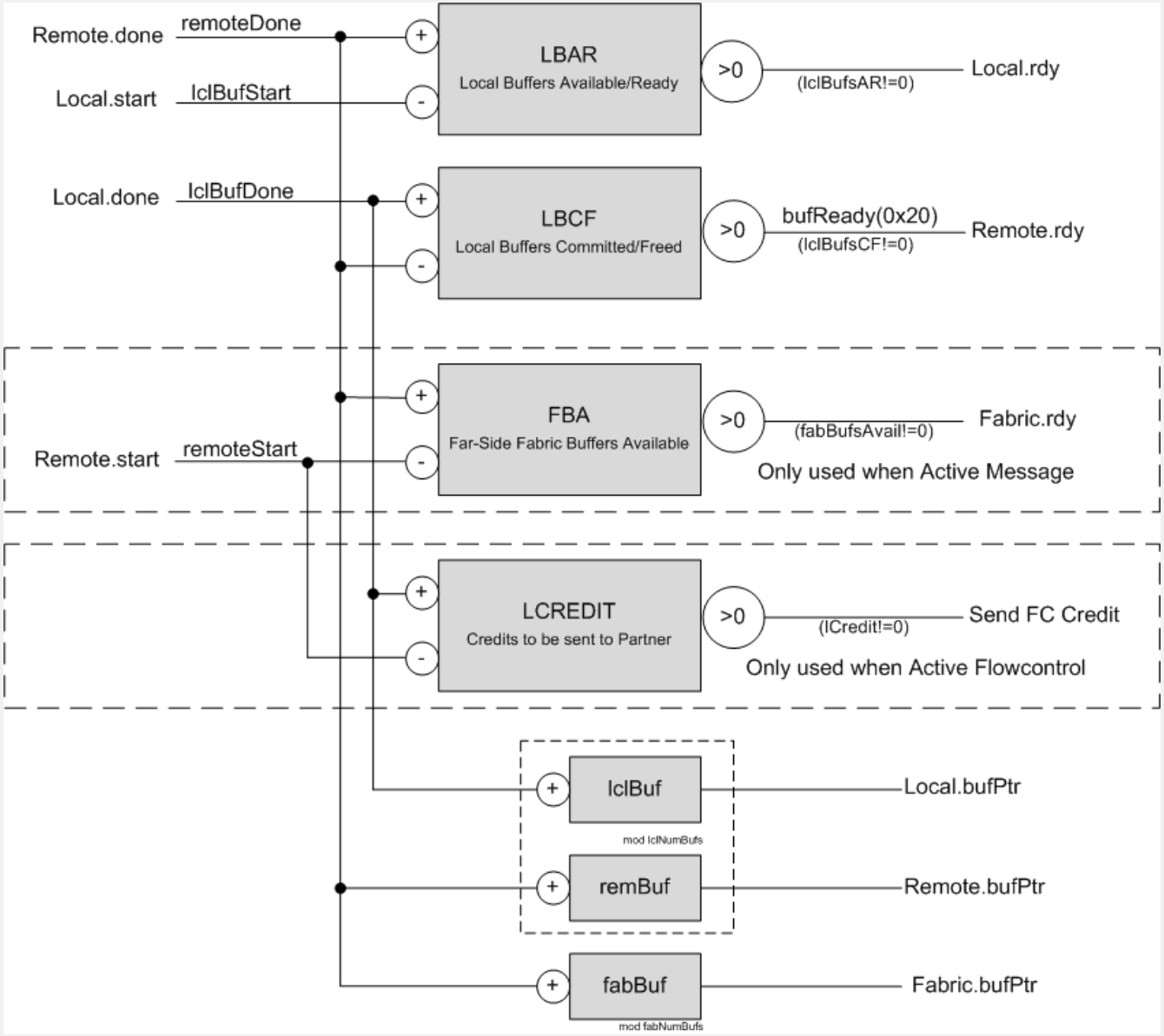
Pull-DMA

	PUSH	PULL
Locally-Active Fabric-Consumer	1. DMA "local free" doorbells to partner	5. Receives far-side full flowcontrol , pull DMA
Locally-Passive Fabric-Consumer	2. Partner polls status and pushes message data	6. Infeasible
Locally-Active Fabric-Producer	3. Receives flowcontrol for far-side free, push message data by DMA	7. DMA "local full" flowcontrol to far-side, partner reads
Locally-Passive Fabric-Producer	4. Infeasible	8. Partner polls status and pulls message data

Near and Far Adaptation



Schematic


Remote.done
remoteDone
Local.start
lclBufStart
LBAR
Local Buffers Available/Ready
>0
(lclBufsAR!=0)
Local.rdy
Local.done
lclBufDone
LBCF
Local Buffers Committed/Freed
bufReady(0x20)
(lclBufsCF!=0)
Remote.rdy
Remote.start
remoteStart
FBA
Far-Side Fabric Buffers Available
(fabBufsAvail!=0)
Fabric.rdy
Only used when Active Message
LCREDIT
Credits to be sent to Partner
(lCredit!=0)
Send FC Credit
Only used when Active Flowcontrol
lclBuf
mod lclNumBufs
Local.bufPtr
remBuf
Remote.bufPtr
fabBuf
mod fabNumBufs
Fabric.bufPtr