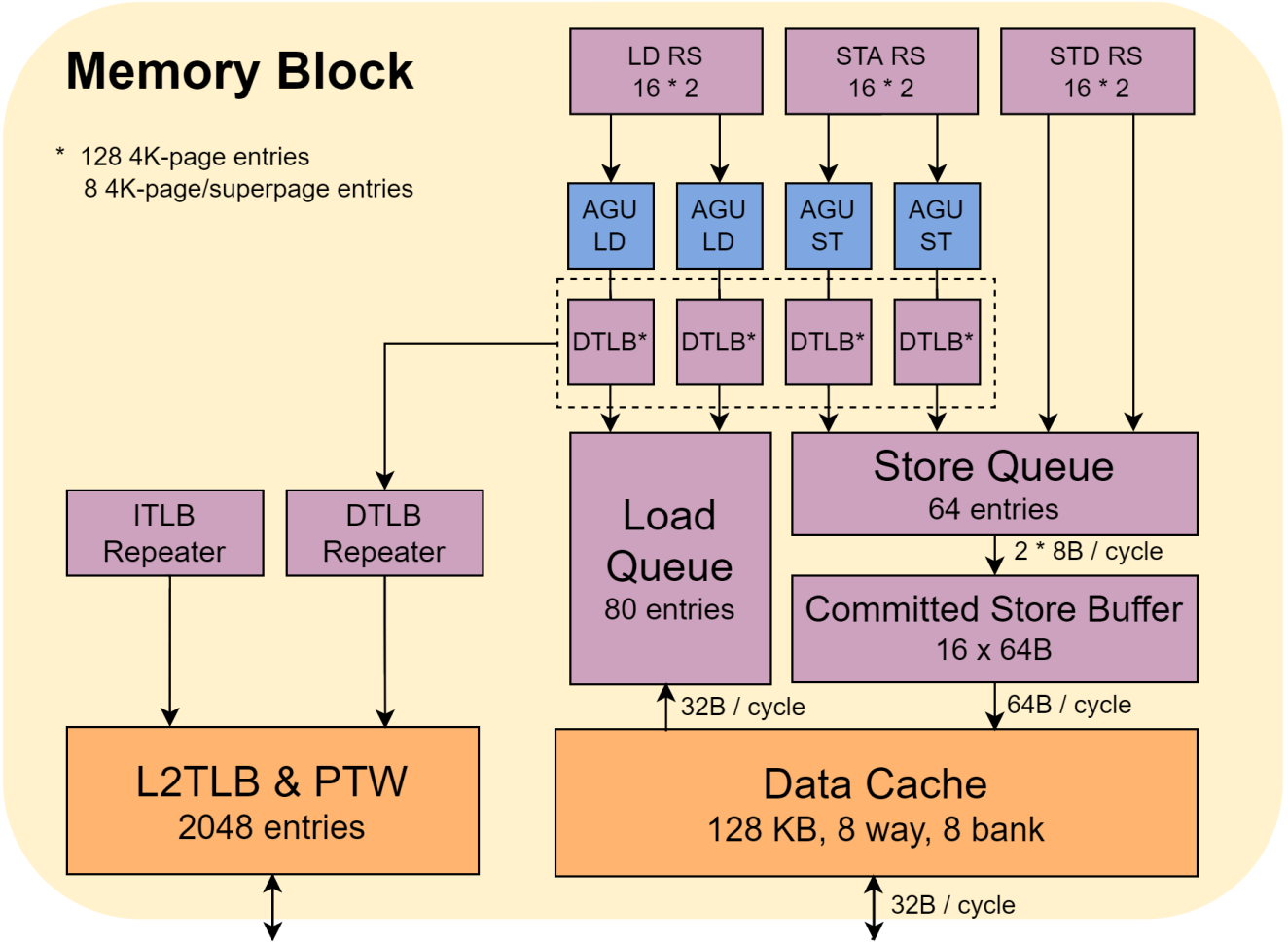
南湖架构访存子系统的设计与实现

王华强 张林隽 张紫飞
中科院计算所
2022年8月25日

香山处理器访存子系统

- 访存单元整体架构
- 相对雁栖湖架构的变化
  - 访存流水线/访存队列
  - 数据缓存设计



参数对比: 雁栖湖->南湖

AGU & Pipeline & Queue	Yanqihu	Nanhu	Improvement
AGUs	2-LD, 2-ST	2-LD, **2-STA, 2-STD**	**拆分 store 数据地址处理**
L1 LD hit latency	3	**4**	**调整流水线长度优化时序**
L1 LD hit bandwidth	2x8B/cycle	2x8B/cycle	
Store data bandwidth	2x8B/cycle	2x8B/cycle	
Load Queue Entry	64	**80**	**+16项,优化分配逻辑**
Store Queue Entry	48	**64**	**+16项,优化分配逻辑**
Data Cache	Yanqihu	Nanhu	Improvement
写合并缓冲 (Store Buffer)	16 cachelines	16 cachelines	
L1 Data cache	32KB, 8way	**128KB**, 8way, **8banks**	**+96KB**
Data cache Probe Queue	16	**8**	
Data cache Miss Queue	16	16	
Data cache Write-back Queue	16	**18**	
软件预取指令	-	**支持**	**添加软件预取支持**

雁栖湖架构访存子系统的瓶颈

- Store to Load Forward
  - 关键路径
- L1D miss latency
  - 为了初版时序考虑, miss latency 比较大, 影响性能
- L1D 容量
  - 复制 SRAM 限制了 L1D 的容量
- Store 的数据地址未分离
  - 限制了并行度
  - 不利于访存违例的回避
- 各队列和 SRAM 的读写时序

→ 使用虚地址的前递设计

→ 调整设计降低 L1D miss latency

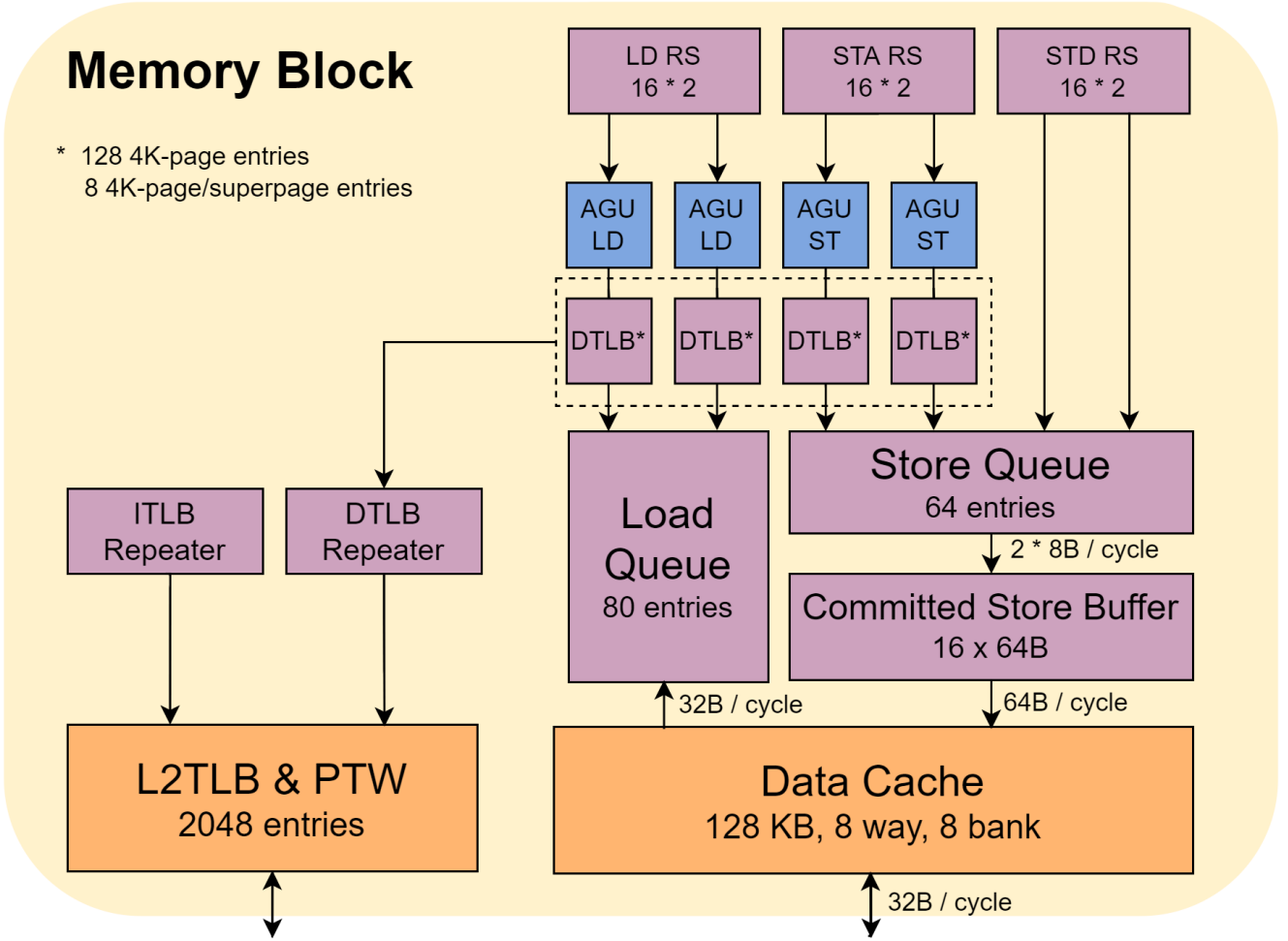
→ 使用分 bank 的方式存放 L1D 的数据

→ 更新相关架构, 分离 store addr 和 data

→ 优化相关架构

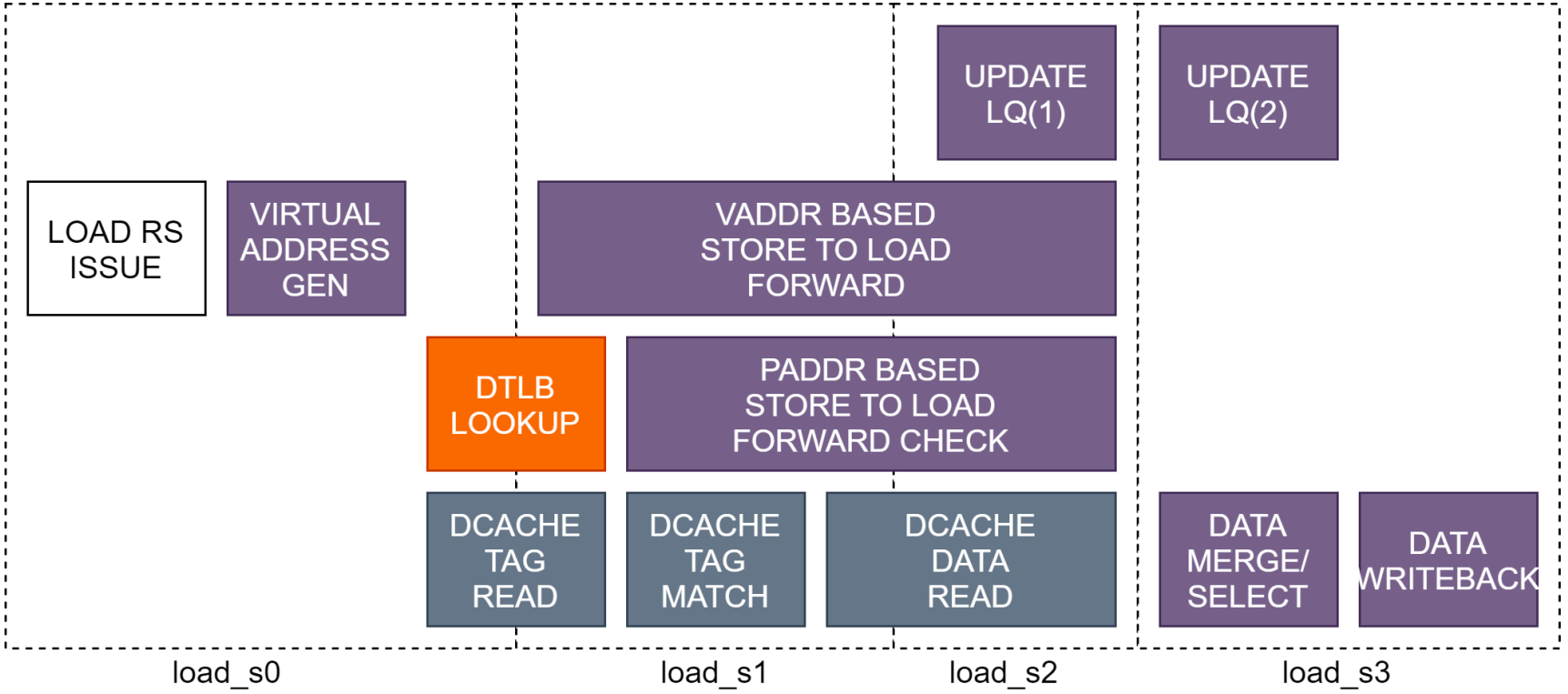
LSU 整体架构 (流水线和队列)

Load 流水线	2
Store 地址流水线	2
Store 数据流水线	2
L1 LD hit latency	4
L1 LD hit bandwidth	2x8B/cycle
Store data bandwidth	2x8B/cycle
Load Queue Entry	80
Store Queue Entry	64
写合并缓冲 (Store Buffer)	16 cachelines



Load 流水线调整

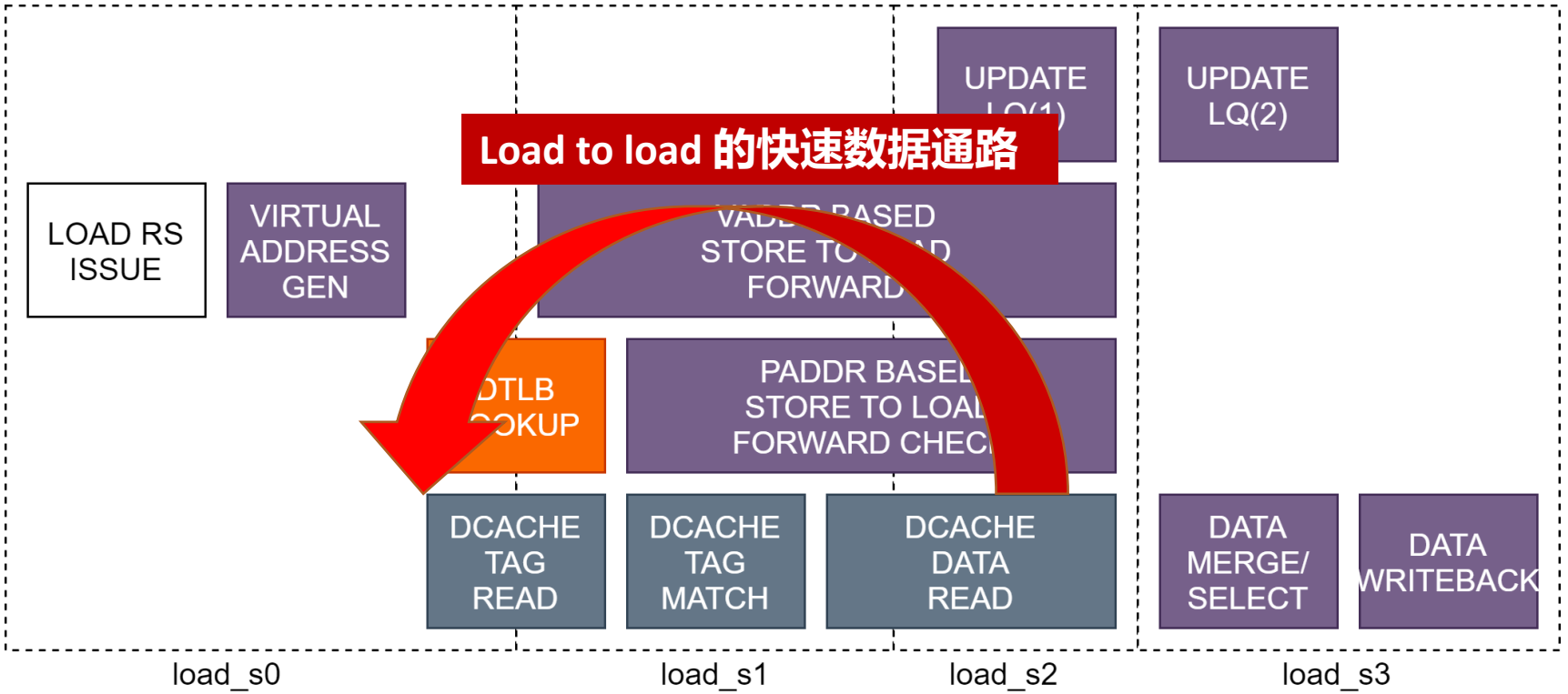
- Load 流水线长度变化: 调整到 **4** 拍
- Tag / Data / LQ 读写相关逻辑调整
- 到保留站的反馈逻辑调整



Load 流水线调整: Load to load 优化

- 仅限后续的 load 指令为 ld 时进行
- 推测 imm 不会引发读 tag index 变化
- 使用推测的 index 进行读取

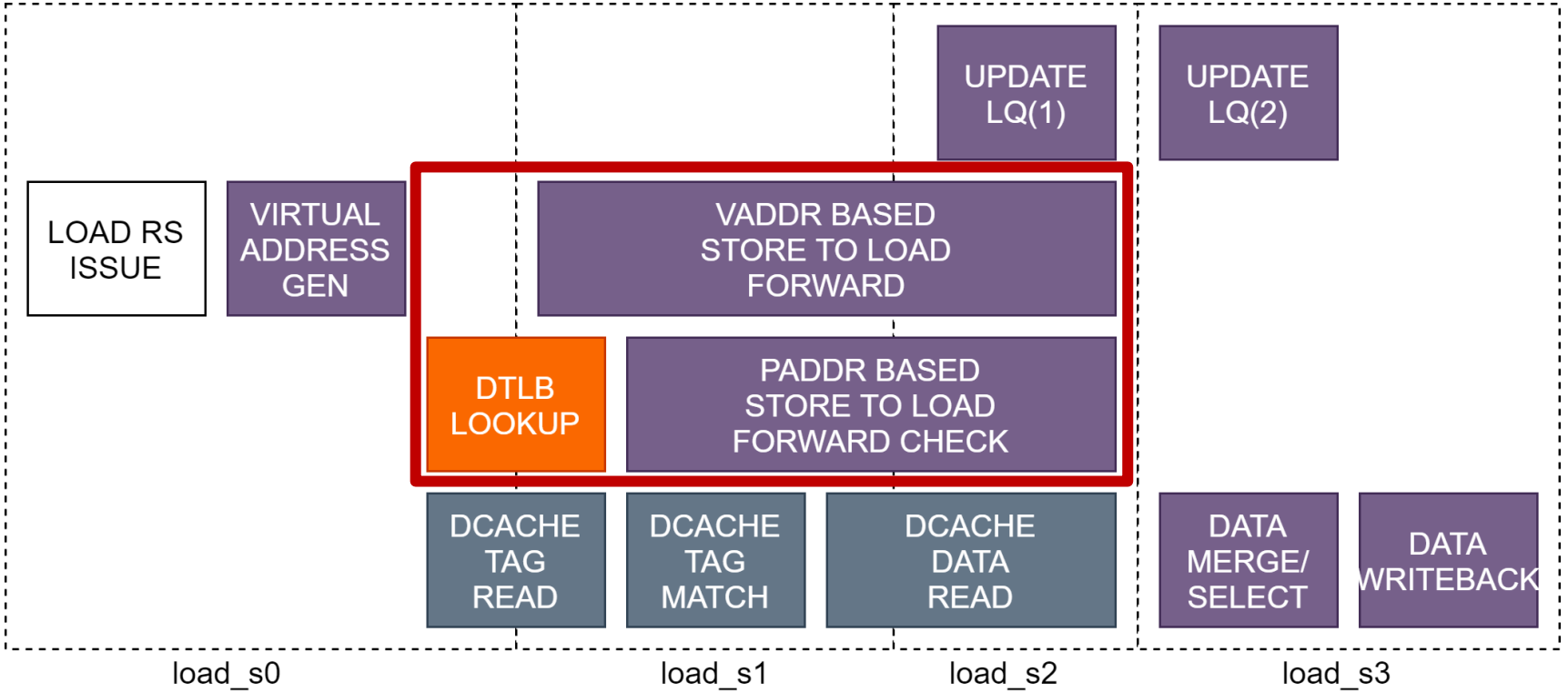
若推测成功，l2l 延迟为 3 拍
若推测失败，延迟退化为 4 拍



Load 流水线调整: 前递机制优化

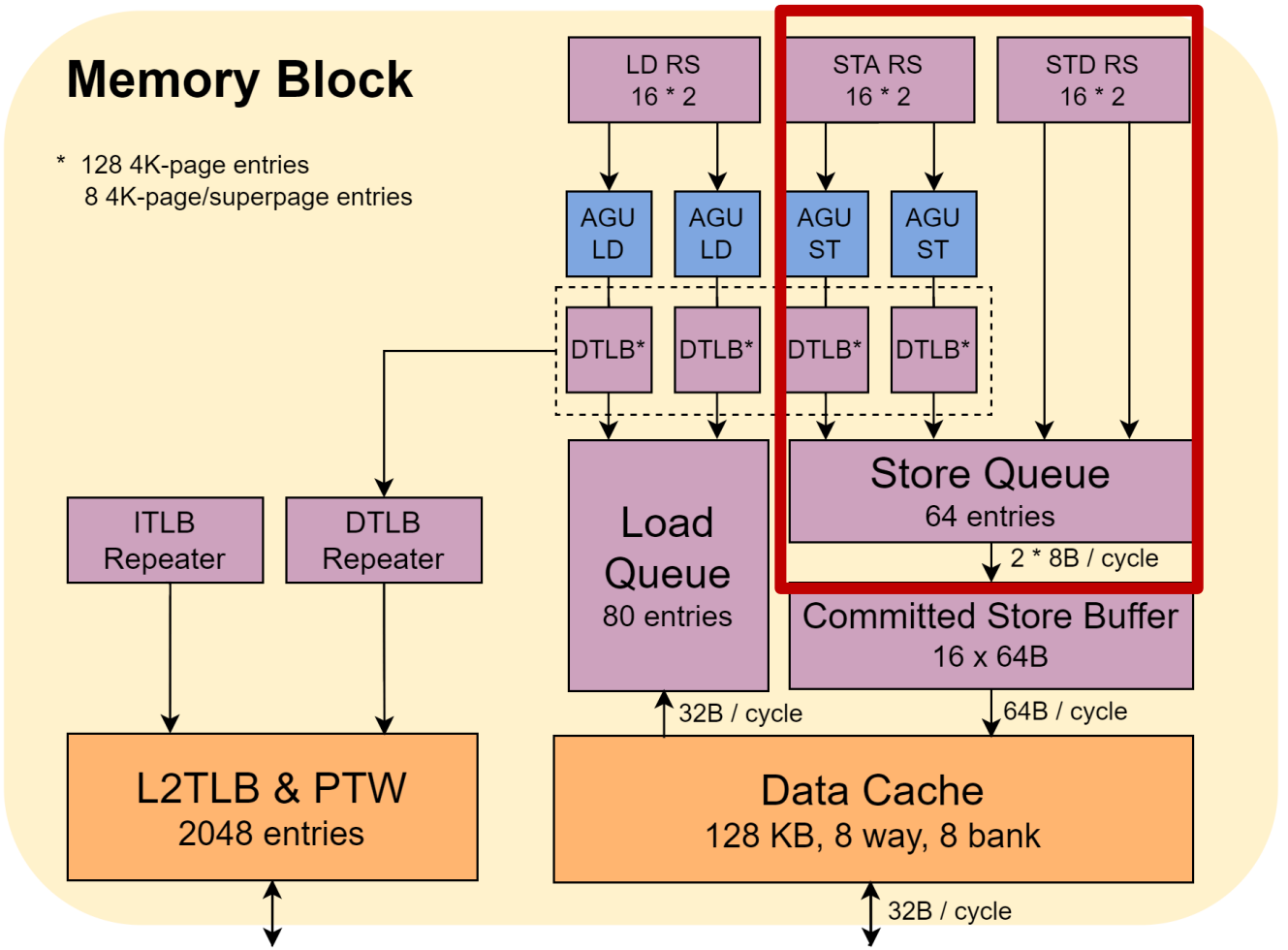
- 调整前递机制以优化时序: 推测性虚地址前递
  - 目的: 在数据通路上排除DTLB的影响
  - 使用**虚地址**进行前递时的地址比较
  - 使用**物理地址**判断前递结果是否正确

虚实地址前递结果不匹配的频率很低
检测到前递错误时清空流水线重新执行



Store 流水线调整

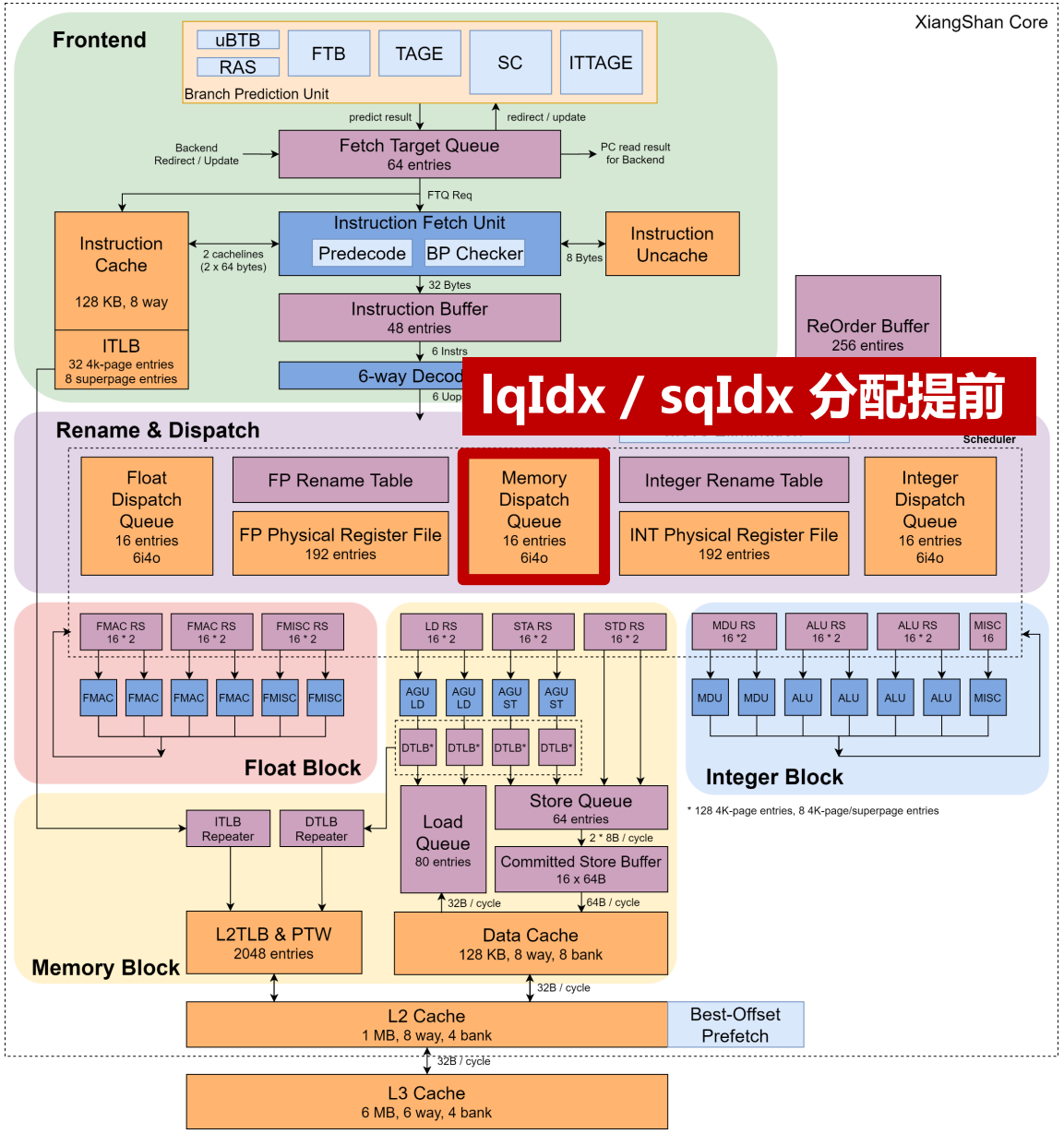
- STA/STD 拆分
- STA 2 拍
  - 物理地址写入 Store Queue 使用两拍
  - 随后违例检查继续进行
  - Store 等待到其违例检查完成才提交
- STD 2 拍
  - 数据写入 Store Queue 占用两拍



LQ / SQ / Sbuffer 调整

- LSQ: 分配逻辑调整
  - 提前分配 LQ/SQ 的 lqIdx/sqIdx
  - 从而优化 LSQ 分配的时序

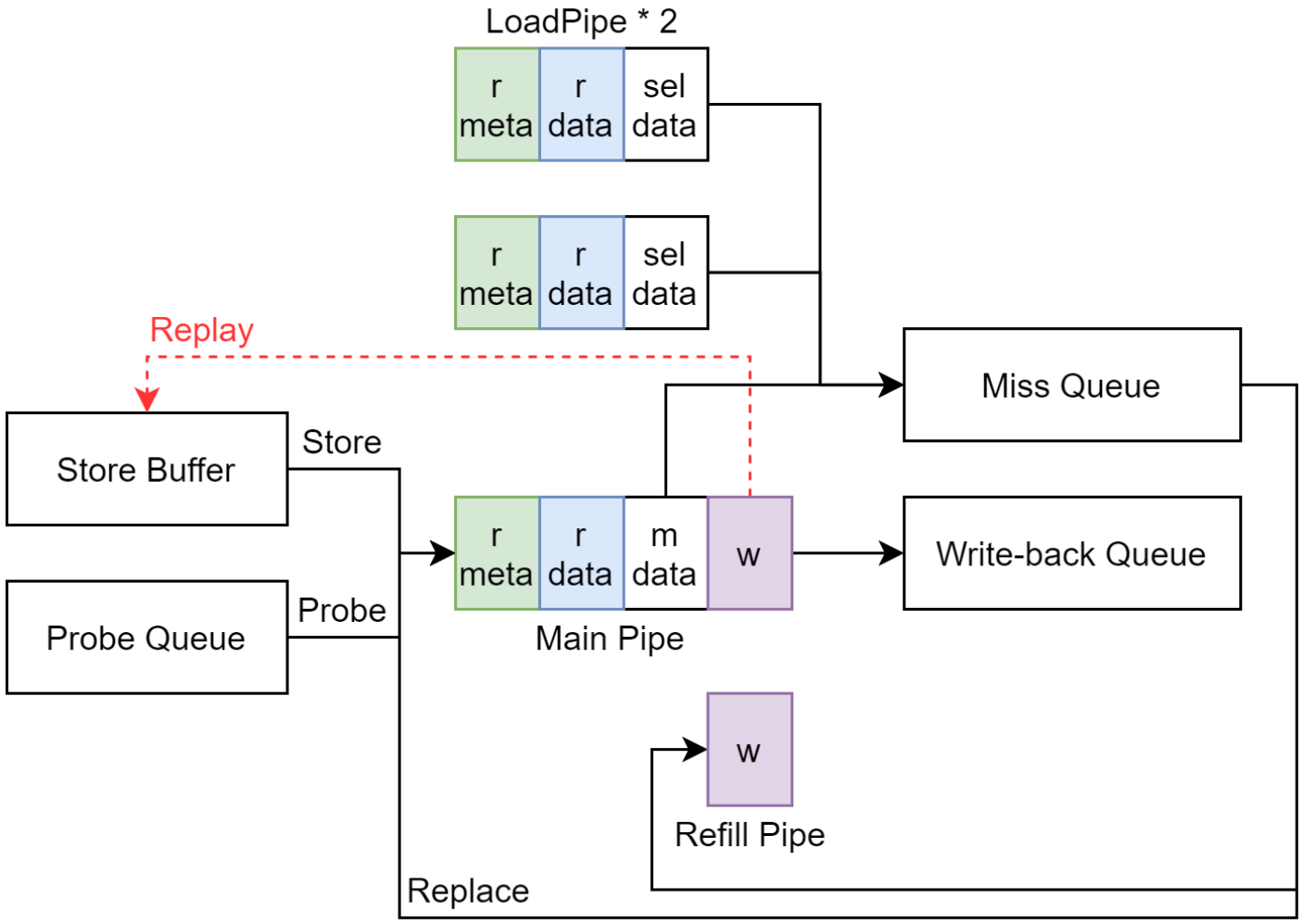
- LSQ, Store Buffer: 读写时序优化
  - 存在大位宽, 大项数的 buffer
  - 数据相关的写操作 2 拍完成



南湖架构中 lqIdx、sqIdx首次分配的位置

L1 数据缓存

L1 Data cache size	128KB
L1/L2 Bus Width	256bit
Store Buffer	16
Probe Queue	8
Miss Queue	16
Write-back Queue	18
软件预取指令	**支持**
ECC	**只检不纠错**



L1 数据缓存架构概述

缓存主要模块及功能

执行流水线

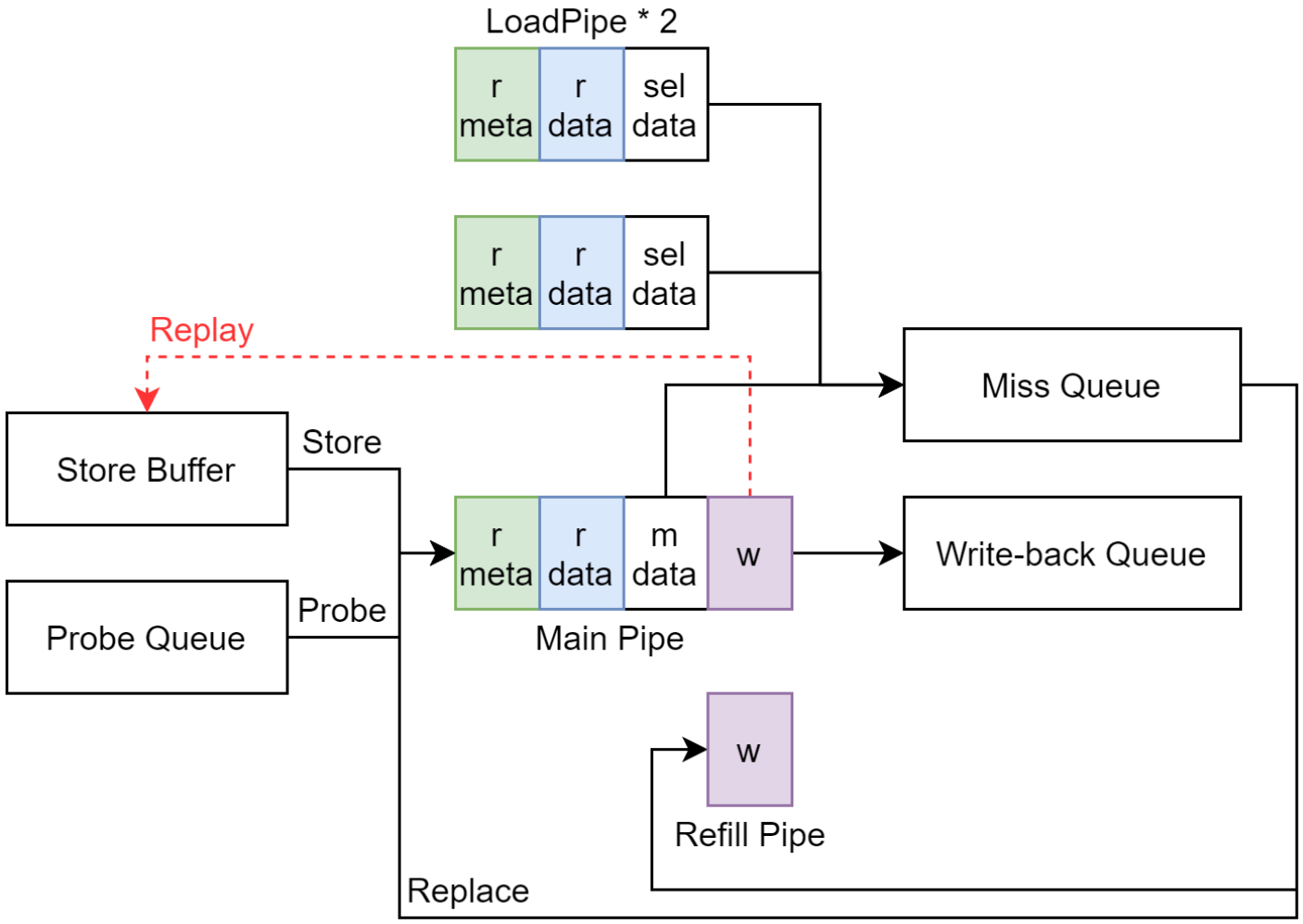
- Load Pipe: 处理 load 请求
- Refill Pipe: 处理 L2 提供给 L1 的数据
- Main Pipe: 处理其他请求

与 load、store 指令交互

- Miss Queue
- Store Buffer

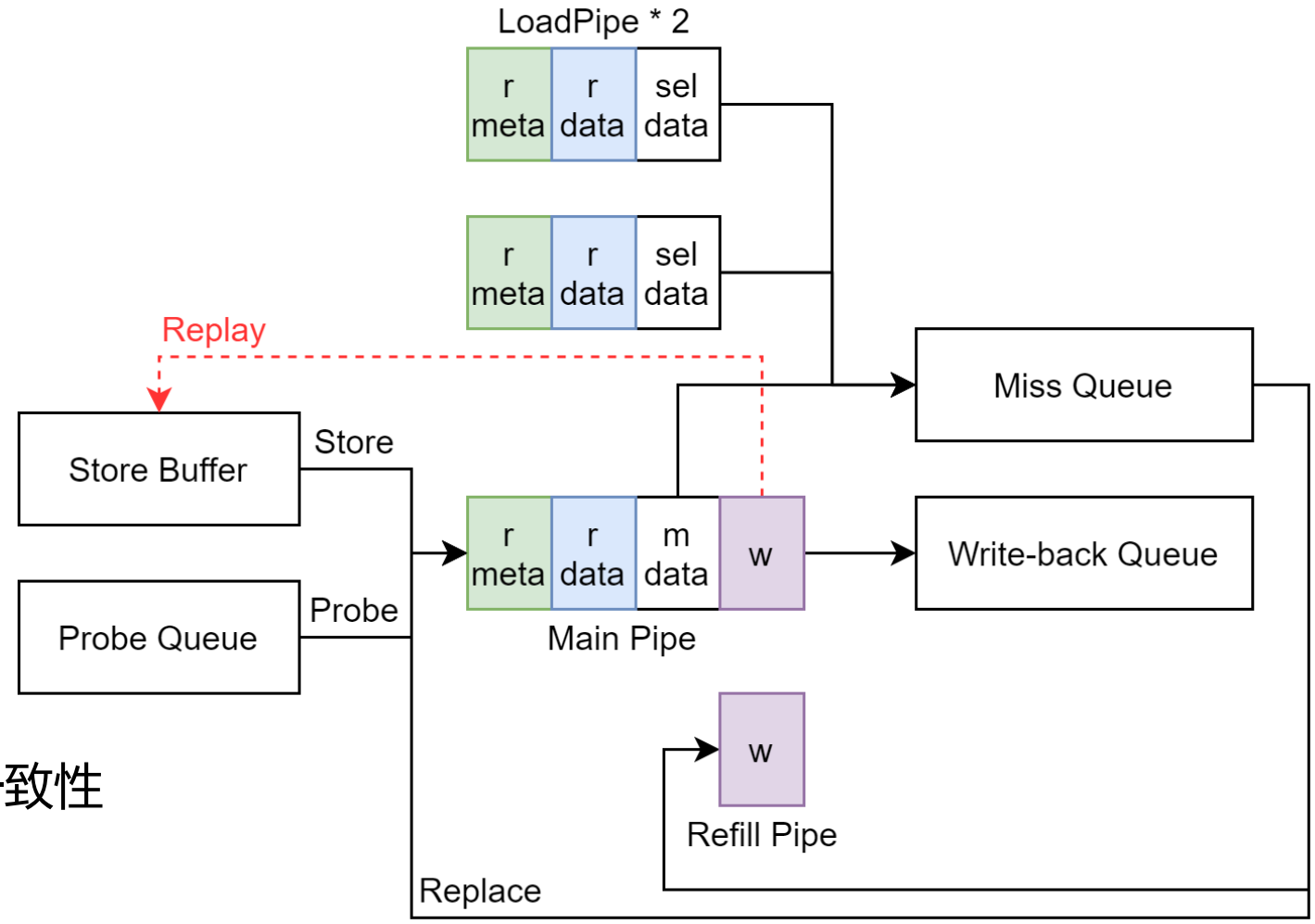
与 TileLink 总线交互

- Probe Queue
- Miss Queue
- Writeback Queue



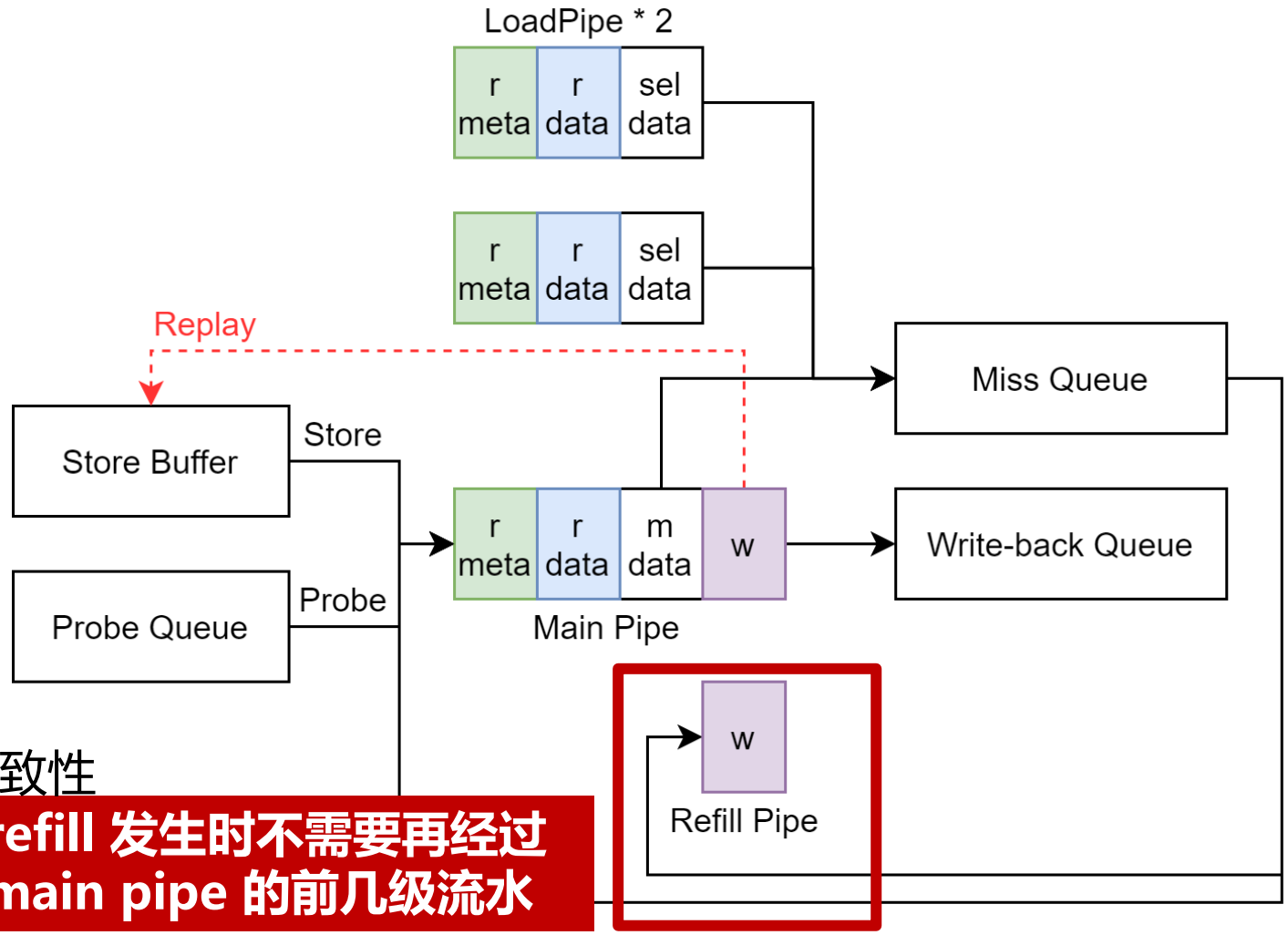
L1 数据缓存架构改进

- 容量变动
  - 由最高 32K 提高到最高 **128K**
- 架构变动
  - 缓冲区调整
    - 合并上一代的多级 store buffer
    - 缓冲区大小调整
    - 更优的分配和合并逻辑
  - miss 处理调整
    - 减少 miss 延迟: 独立的 Refill Pipe
    - Evict 延后到 Fill Data 到达才开始
- 一致性变动
  - 使用 TileLink 协议, 支持 ID Cache 一致性



L1 数据缓存架构改进

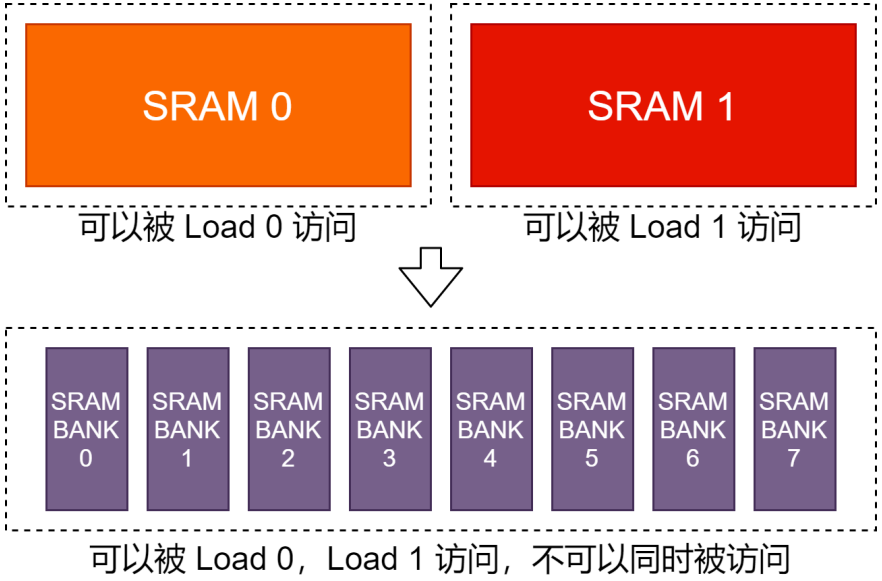
- 容量变动
  - 由最高 32K 提高到最高 **128K**
- 架构变动
  - 缓冲区调整
    - 合并上一代的多级 store buffer
    - 缓冲区大小调整
    - 更优的分配和合并逻辑
  - miss 处理调整
    - **减少 miss 延迟: 独立的 Refill Pipe**
    - Evict 延后到 Fill Data 到达才开始
- 一致性变动
  - 使用 TileLink 协议, 支持 ID Cache 一致性



新添加的快速 refill pipe

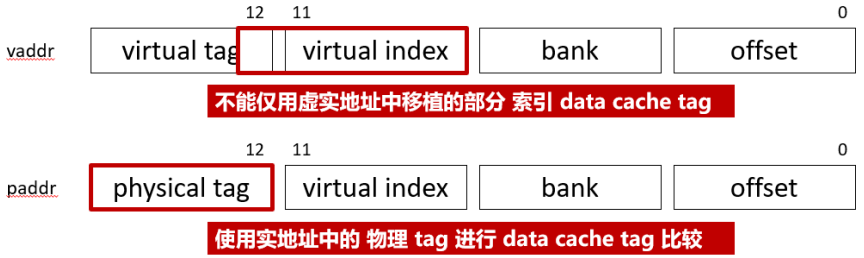
数据缓存容量变动

- 分 Bank 取代 SRAM 复制以实现多端口读
- 节省出的面积用于增大缓存容量



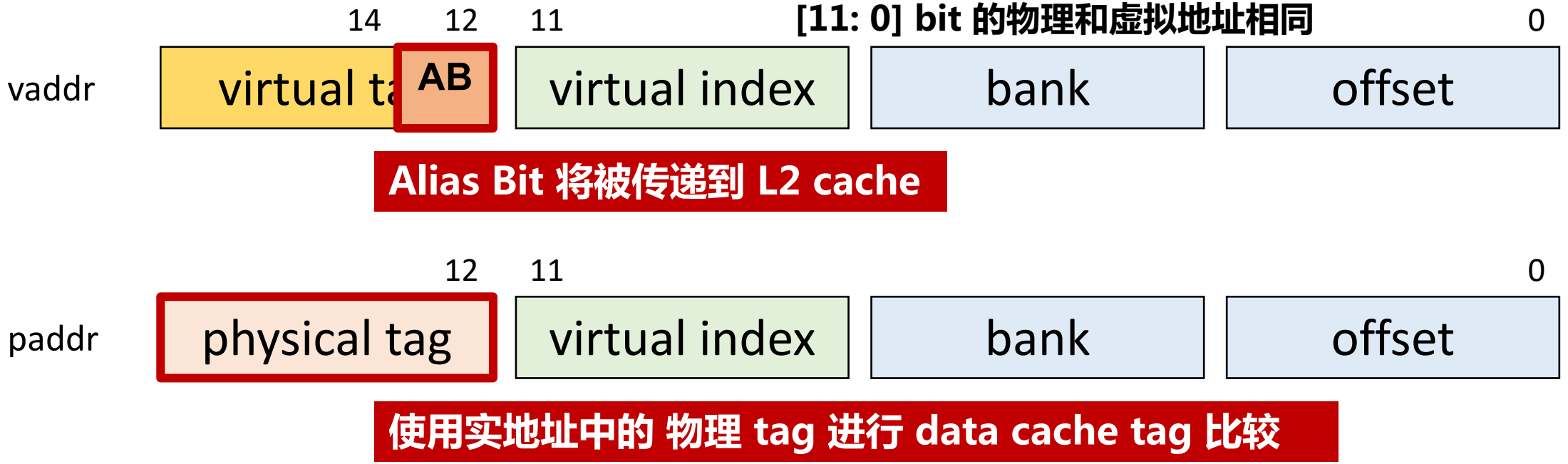
- 容量上限由 32K 提高到 128K

- 缓存容量增大到 32K 以上会出现别名问题



数据缓存容量变动

- 解决办法: Alias Bit (2 bit)
- L2 cache 保证, 对同一 virtual index, **相同的 Alias Bit 不会存在于 L1 cache 中**
- L2 缓存的介绍中会介绍细节



128KB DCache 的虚实地址使用

谢谢！
请批评指正