ALB: Adaptive Lane Borrowing of Hybrid Memory Cube

Xianwei Zhang †, Youtao Zhang † and Jun Yang*

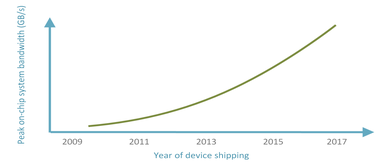
Computer Science Department, University of Pittsburgh †

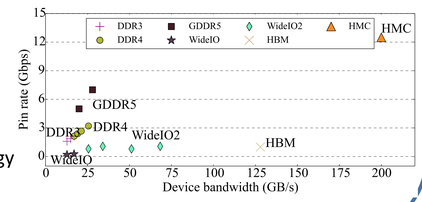
Electrical and Computer Engineering Department, University of Pittsburgh*

†{xianeizhang, zhangyt}@cs.pitt.edu, ⋆juy9@pitt.edu

I. Introduction

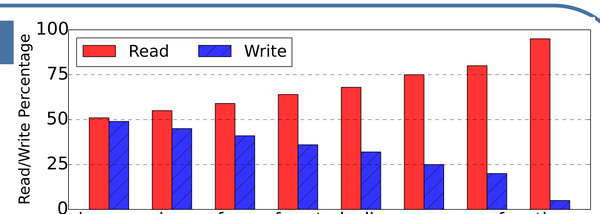
- **Bandwidth Demand Keeps Increasing**
  - HPC, Big Data analysis, image/video processing, etc;
  - Thus, more memory channels, larger bandwidth/ channel.
- **Parallel Buses cannot Meet the Demand**
  - **Signal integrity** is a great challenge in DDRx wide parallel buses;
  - **Pin count** grows fast with #channels, restricting #channels and thus the total bandwidth.
- **High Speed Serial Designs Give a Solution**
  - Failures: FB-DIMM, BoB; overheads on power, energy and latency;
  - Success: HMC; greatly save power, latency and energy.

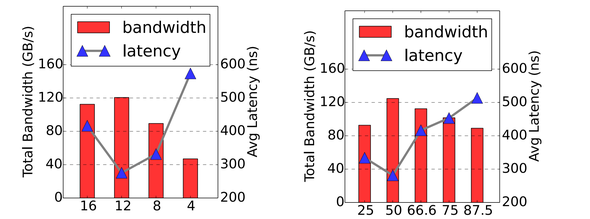




III. Motivation

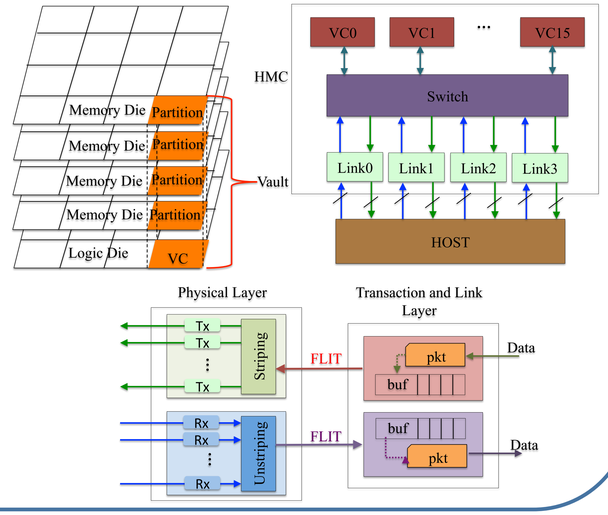
- **HMC Lanes Prefer to Balanced Traffic on Tx and Rx lanes**
  - HMC link is a **balanced** design, 16 lanes for either direction;
  - And, lanes are **uni-directional**, i.e., a lane is dedicated to Tx or Rx;
- **Applications Usually Show Imbalanced Read/Write Requests**
  - Read/Write ratio varies across different applications
- **Synthetic Studies**
  - Vary link configuration by allocating different #lanes for Rx (16/12/8/4), and rest for Tx;
  - even lane partition is sub-optimal, highest bandwidth at **12 Rx and 20 Tx lanes**;
  - Keep even lane partition, vary the portion of read requests from 25% to 87.5%;
  - Maximum bandwidth achieves at roughly **50%** read percentage.





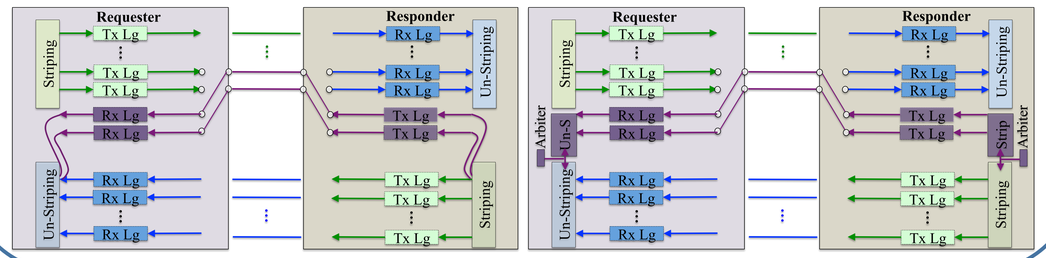
II. Structure of Hybrid Memory Cube (HMC)

- **3D-stacked**
  - Hybrid: bottom logic layer + upper DRAM layers;
  - Memory dies are vertically segmented into **vaults** (channels), with each has a memory controller in the logic base;
  - Each vault contains partitions with several banks (e.g., a 4GB HMC has 4 memory dies organized as 16 vaults, with each has four 2-bank partitions.
- **High Speed Serial Links**
  - HMC has 4 (V2.0) or 8 (V1.0) links, achieving up to **320/480GBps** bandwidth;
  - Each link consists of 16 transmit (TX) and 16 Receive (RX) lanes;
  - Commands and data are transmitted over lanes as packets consists of **128-bit FLITs**.
- **Serial Link Layers (Serialization and Deserialization)**
  - **Transaction and link layer** generate packets, which then are chopped into FLITs and sent to physical layer;
  - **Physical layer** serializes the FLIT and drive it across the lanes.



IV. Adaptive Lane Borrowing (ALB) Schemes

- **Lane Borrowing**
  - A subset of Rx/Tx lanes can be re-configured for Tx/Rx use w.r.t read/write ratio;
  - On **responder** side, add N copies of **Tx** lane supporting logics (Tx Lg), redirect N lanes from Rx to Tx;
  - On **requester** side, add N copies of **Rx** lane supporting logics (Rx Lg), redirect N lanes from Tx to Rx;
  - Acceptable overhead as only physical layer is modified.
- **Possible Implementations**
  - **Wide single link:** integrate extra lanes with existing ones to form a wider single Tx link than the baseline;
  - **Extra narrow link:** construct an extra Tx link from the borrowed lanes.



V. Experimental Results

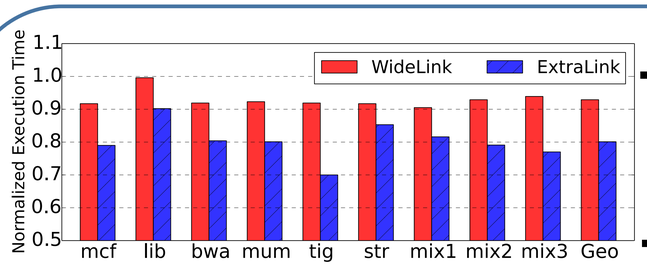


Figure 1: Comparison of execution time

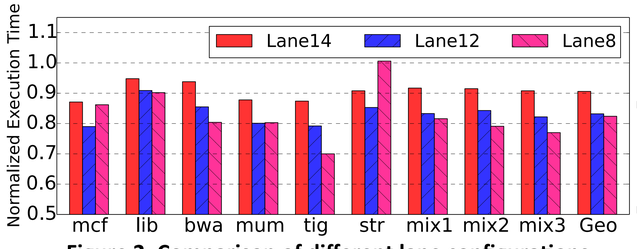


Figure 2: Comparison of different lane configurations

- **Experimental Methodology**
  - **Simulator:** in-house HMC simulator based on USIMM and BOBSim
  - **Processor:** 8 cores, 2.5GHz, ROB size 128, Fetch/Retire width: 4/4, Pipeline depth: 10
  - **HMC:** 1 link with 16Tx/Rx lanes at 10Gb/s, 4GB, 16 vaults, 4 4-bank partitions/vault, 256B page size, packet: 64B data, 16B header
  - **Workloads**: SPEC (mcf/libq/bwaves), BIOBENCH (mummer/tigr), MICRO-BENCH (stream)
  - **Schemes: Baseline** (16 Tx, 16 Rx), **WideLink** (16+N Tx, 16-N Rx), **ExtraLink** (16 Tx, N Tx, 16-N Rx)
- **Impacts on Program Execution Time**
  - On average, **WideLink** reduces the execution time by 7.1% and ExtraLink achieves much larger 19.9% reduction;
  - Compared to **Baseline**, both **WideLink** and **ExtraLink** widen the response path thereby benefiting read accesses;
  - **WideLink** introduces **larger skew delay** which enlarges latency, and 128b FLIT is not efficiently transferred;
  - Different workloads favorite different link configurations (Figure 2), e.g. **8** for *bwa*/*big*, and **12** for *str*.
- **Impacts on Memory Bandwidth**
  - **Baseline** (20GB/s Tx + 20GB/s Rx) has average 19.4GB/s response bandwidth and <10GB/s request bandwidth for most workloads;
  - **WideLink** improves link utilization by 11.6%, and **ExtraLink** by 24%.
- **Sensitivity on #cores and packet size**
  - #cores: 4/8/16 (Figure 3), packet data size: 16/32/64/128B (Figure 4).
  - Bandwidth demanding grows as #cores increases and packet size enlarges, and thus benefits are larger.

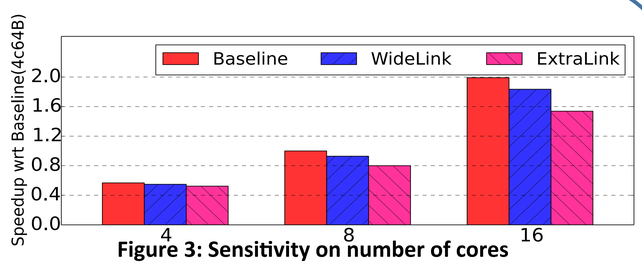


Figure 3: Sensitivity on number of cores

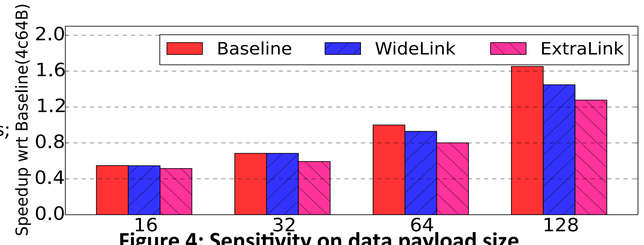


Figure 4: Sensitivity on data payload size

Presented @ 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)