CS 222 Computer Organization & Architecture

Lecture 32 [22.04.2019]

DRAM Controllers

John Jose

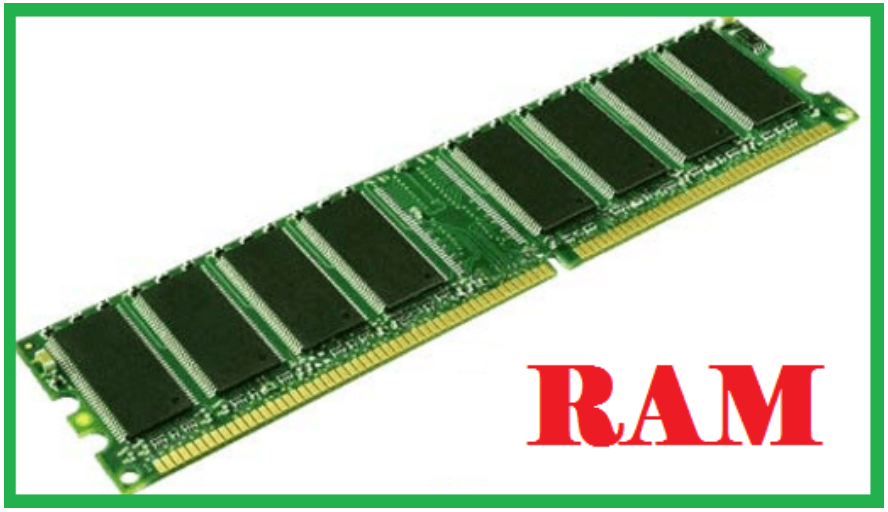
Assistant Professor

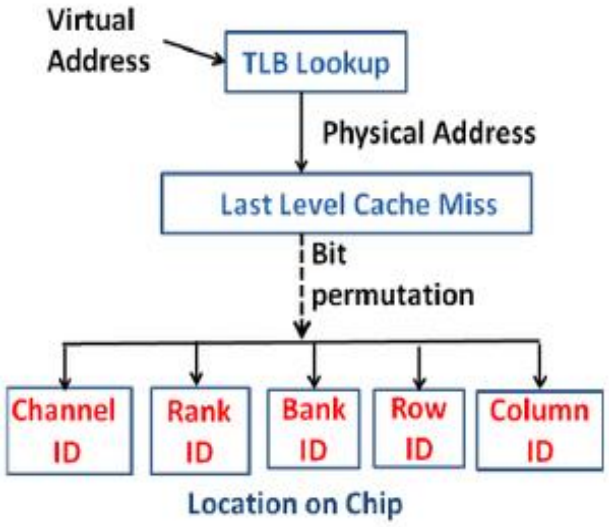
Department of Computer Science & Engineering

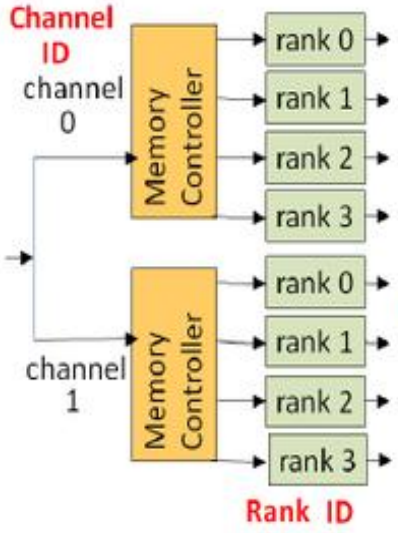
Indian Institute of Technology Guwahati, Assam.

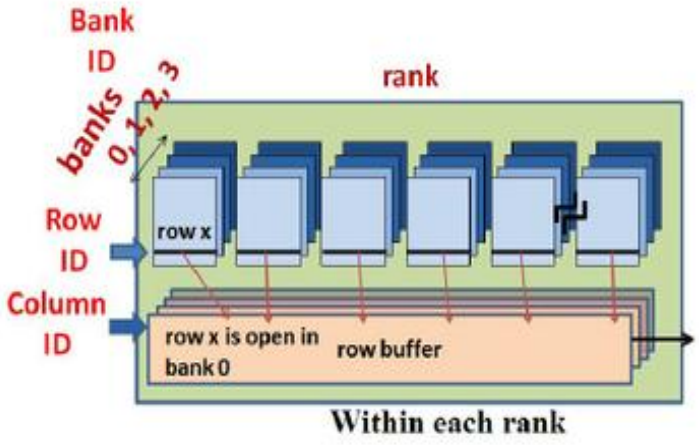
DRAM Subsystem Organization

- Channel
- DIMM
- Rank
- Chip
- Bank
- Row
- Column
- B-Cell









Basic DRAM Operation

- CPU → controller transfer time
- Controller latency
  - Queuing & scheduling delay at the controller
  - Access converted to basic commands
- Controller → DRAM transfer time
- DRAM bank latency
  - Simple CAS (column address strobe) if row is "open" OR
  - RAS (row address strobe) + CAS if array precharged OR
  - PRE + RAS + CAS (worst case)
- DRAM → Controller transfer time
  - Bus latency (BL)
- Controller to CPU transfer time

DRAM Controller: Functions

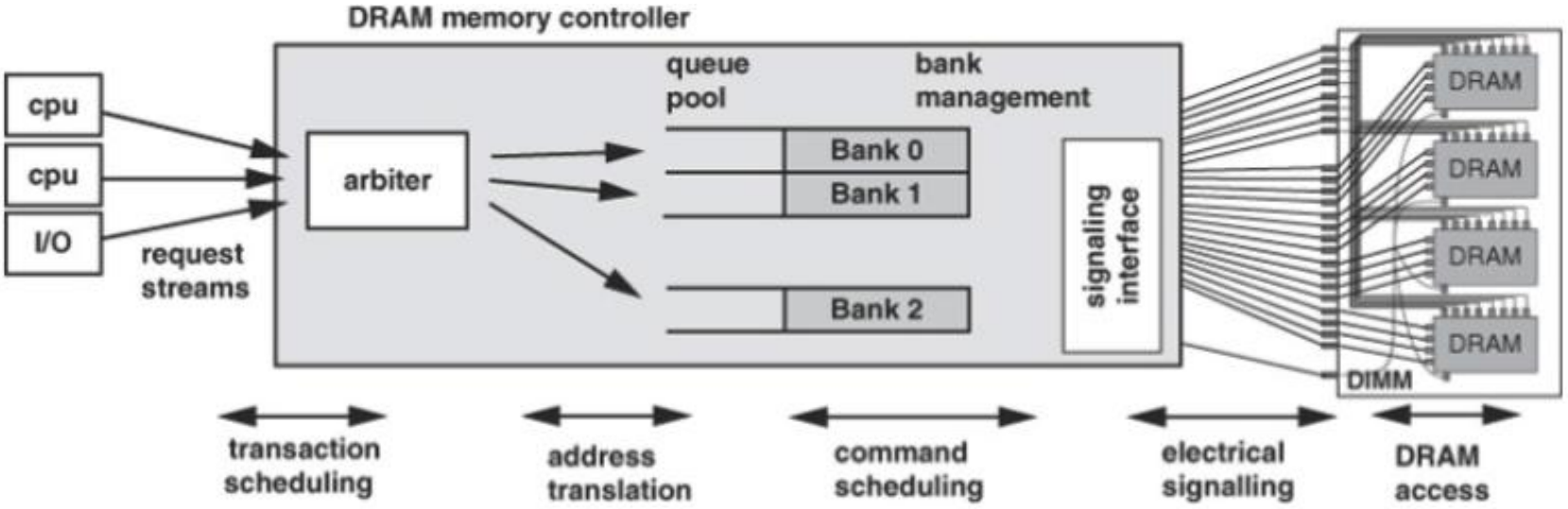
- Ensure correct operation of DRAM
  - Address mapping, refreshing and timing
- Service DRAM requests while obeying timing constraints of DRAM chips
  - Constraints: resource conflicts (bank, bus, channel), minimum write-to-read delays
- Translate requests to DRAM command sequences
- Buffer and schedule requests to improve performance
  - Reordering, row-buffer, bank, rank, bus management
- Manage power consumption and thermals in DRAM
  - Turn on/off DRAM chips, manage power modes

DRAM Controller: Where to Place ?

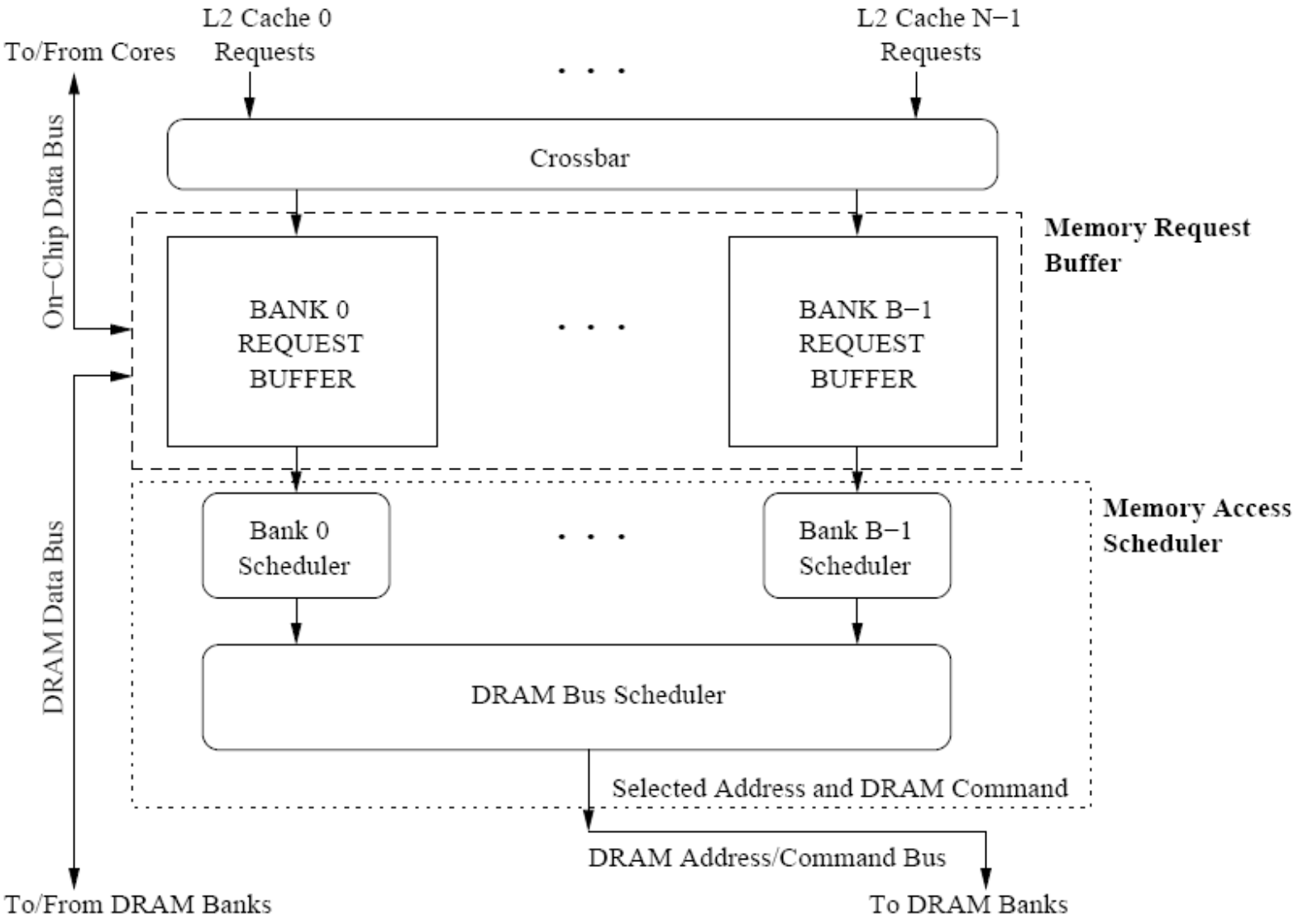
- **In chipset**
- More flexibility to plug different DRAM types
- Less power density in the CPU chip

- **On CPU chip**
- Reduced latency for main memory access
- Higher bandwidth between cores and controller
- More information can be communicated between CPU and controller (criticality of the request)

DRAM Controller Overview



DRAM Controller Logic



DRAM Scheduling Policies

- **FCFS** (first come first served)
  - Oldest request first
- **FR-FCFS** (first ready, first come first served)
  - Row-hit first and then Oldest first
  - Goal is to maximize row buffer hit rate
  - maximize DRAM throughput
- Actually, scheduling is done at the command level
  - Column commands (read/write) prioritized over row commands (activate/precharge)
  - Within each group, older commands prioritized over younger ones

DRAM Scheduling Policies

- A scheduling policy is essentially a prioritization order
- Prioritization can be based on
  - Request age
  - Row buffer hit/miss status
  - Request type (prefetch, read, write)
  - Request mode (load miss or store miss)
  - Requestor Type (CPU, DMA, GPU)
  - Request criticality
    - Oldest miss in the core?
    - How many instructions in core are dependent on it?
    - Will it stall the processor?
  - Interference caused to other cores

Row Buffer Management Policies

- **Open row**
- Keep the row open after an access
- Next access might need the same row → row hit
- Next access might need a different row → row conflict, wasted energy
- **Closed row**
- Close the row after an access (if no other requests already in the request buffer need the same row)
- Next access might need a different row → avoid a row conflict
- Next access might need the same row → extra activate latency
- **Adaptive policies-** Predict whether or not the next access to the bank will be to the same row

Open vs Closed Row Policies

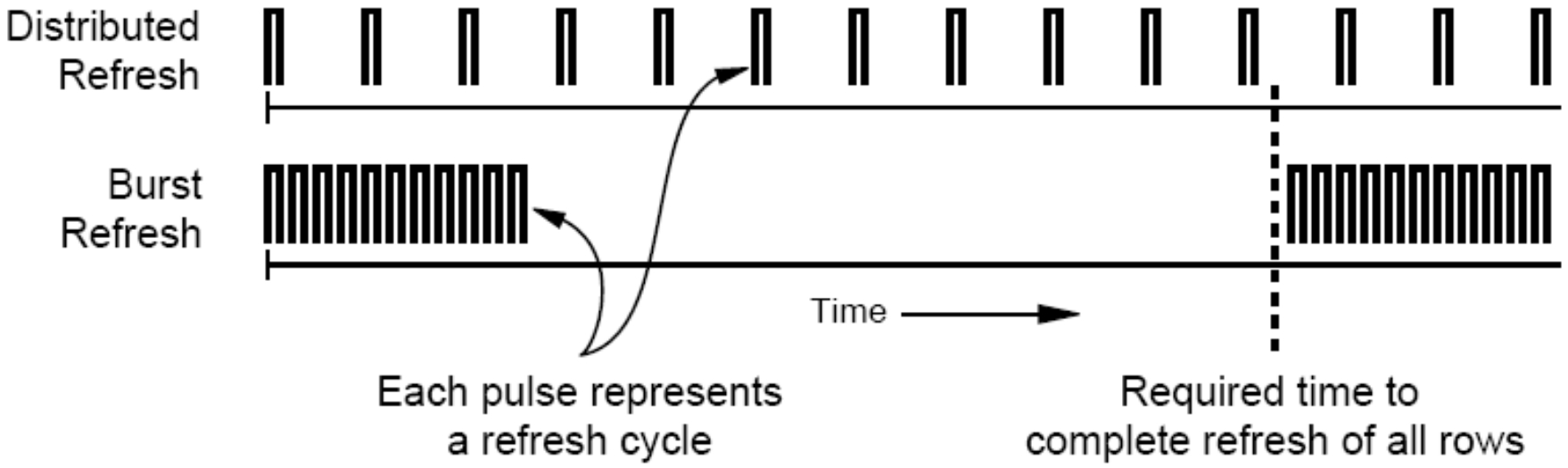
Policy	First access	Next access	Commands needed for next access
Open row	Row 0	Row 0 (row hit)	Read
Open row	Row 0	Row 1 (row conflict)	Precharge + Activate Row 1 + Read
Closed row	Row 0	Row 0 – access in request buffer (row hit)	Read
Closed row	Row 0	Row 0 – access not in request buffer (row closed)	Activate Row 0 + Read + Precharge
Closed row	Row 0	Row 1 (row closed)	Activate Row 1 + Read + Precharge

DRAM Refresh

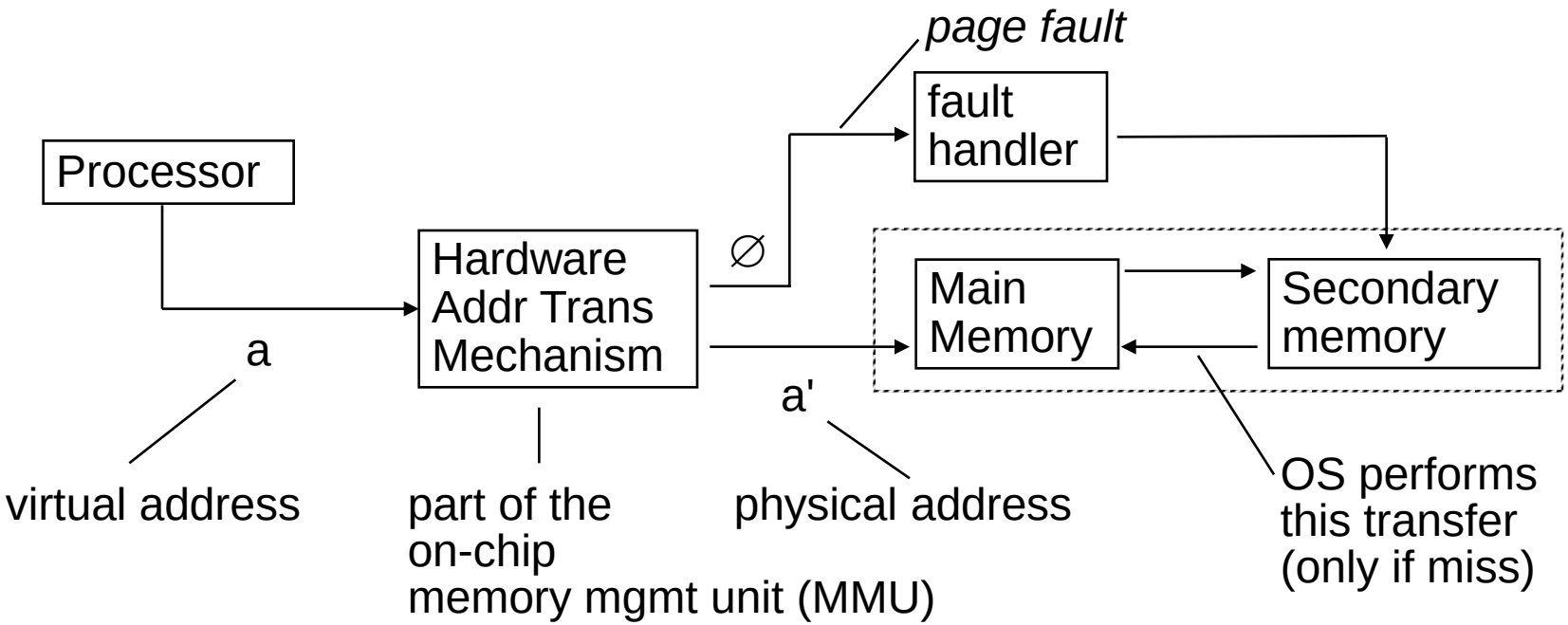
- DRAM capacitor charge leaks over time
- The memory controller needs to read each row periodically to restore the charge
  - Activate + precharge each row every Nms
  - Typical N = 64 ms (Refresh Interval)
- Implications on performance?
  - DRAM bank unavailable while refreshed
  - Long pause times: If we refresh all rows in burst, every 64ms the DRAM will be unavailable until refresh ends

DRAM Refresh

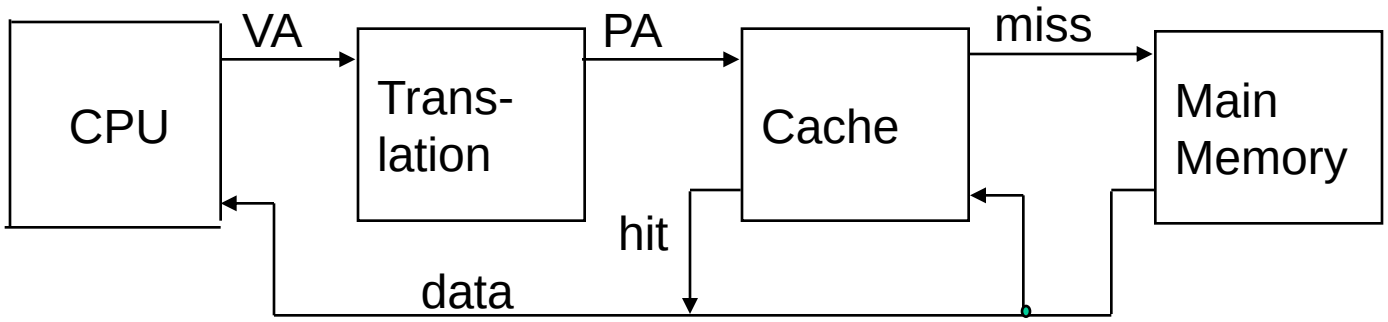
- Burst refresh: All rows refreshed immediately after one another
- Distributed refresh: Each row refreshed at a different time, at regular intervals
- Distributed refresh eliminates long pause times



VM Address Translation



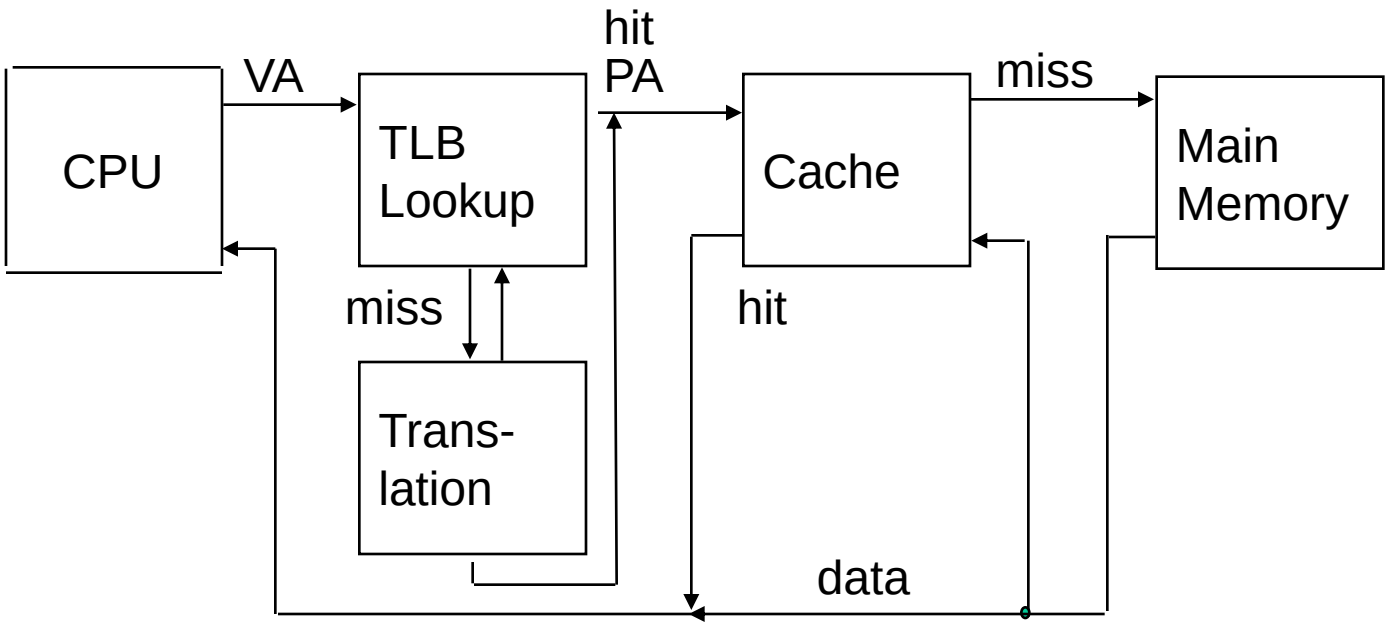
Integrating VM and Cache



- Most Caches are accessed by physical addresses
- Multiple processes can have blocks in cache at same time
- Cache doesn't need to be concerned with protection issues
- Access rights checked as part of address translation
- Perform Address Translation (Page table) Before Cache Lookup - TLB

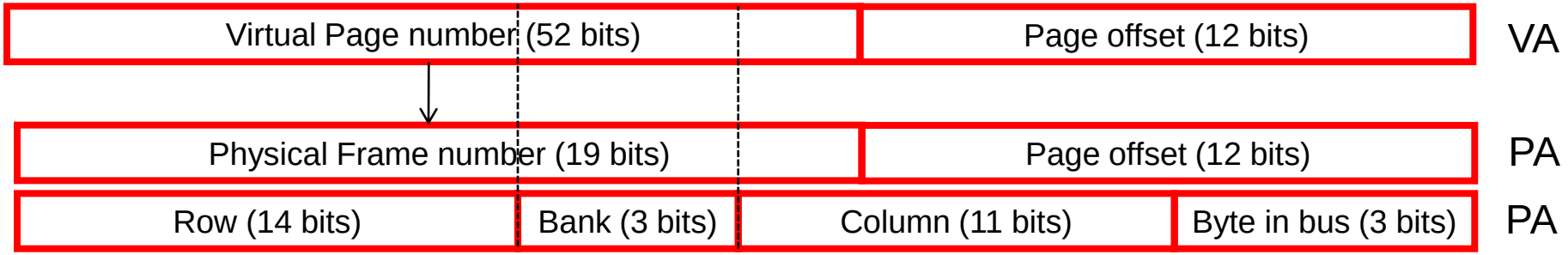
Address Translation with a TLB

- Translation Lookaside Buffer (TLB) is a small hardware cache
- Maps virtual page numbers to physical page numbers
- Contains complete page table entries for small number of pages (recent pages)



Virtual To Physical Address Mapping

- Operating System influences where an address maps to in DRAM



- Operating system can influence which bank/channel/rank a virtual page is mapped to.
- It can perform page coloring to
  - Minimize bank conflicts
  - Minimize inter-application interference

johnjose@iitg.ac.in
http://www.iitg.ac.in/johnjose/