HY425 Lecture 15: DRAM Technology

Dimitrios S. Nikolopoulos

University of Crete and FORTH-ICS

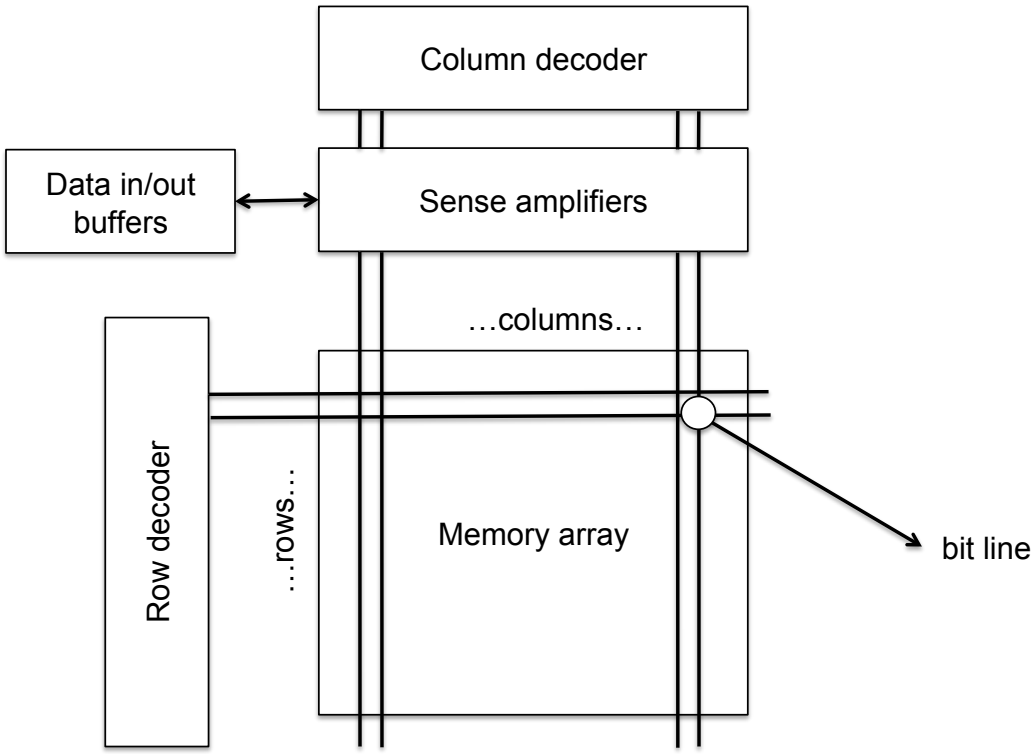
December 2, 2011

DRAM

Fundamentals

- Random-access memory using one transistor-capacitor pair per bit
- Capacitors leak, needs refresh
- Composed of one or more memory arrays
  - Organized in rows and columns
  - Need sense amplifiers to compensate for voltage swing

DRAM cell

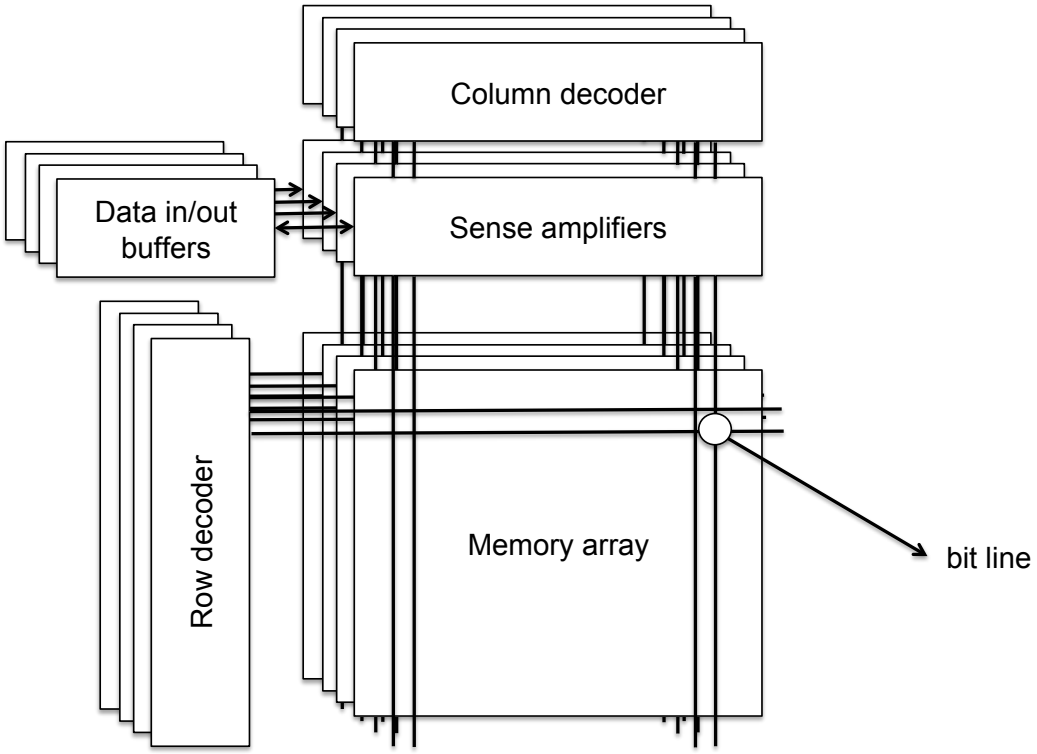


DRAM

Fundamentals

- Each DRAM memory array outputs one bit
- DRAMS use multiple memory arrays to output multiple bits at a time
  - $\times$N indicates DRAM with N memory arrays
  - $\times$16, $\times$32 DRAMS typical today
- Each collection of $\times$N arrays forms a DRAM bank
- Banks can be read/written independently

×4 DRAM

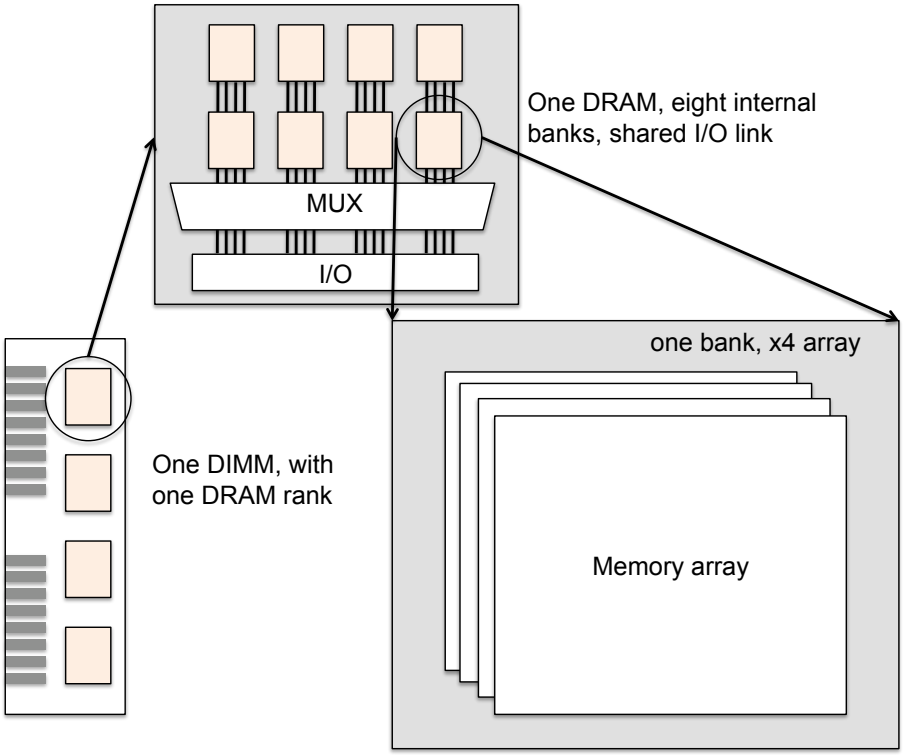


Interleaved DRAM

DRAM memory bandwidth

- Limited bandwidth from one DRAM bank
- Increase bandwidth by delivering data from multiple banks
  - Processor DRAM interconnect (e.g. bus) with higher clock frequency than any one DRAM
  - Bus control switches between multiple DRAM banks to achieve high data rate

DIMMs and Ranks



Modern DRAM organization

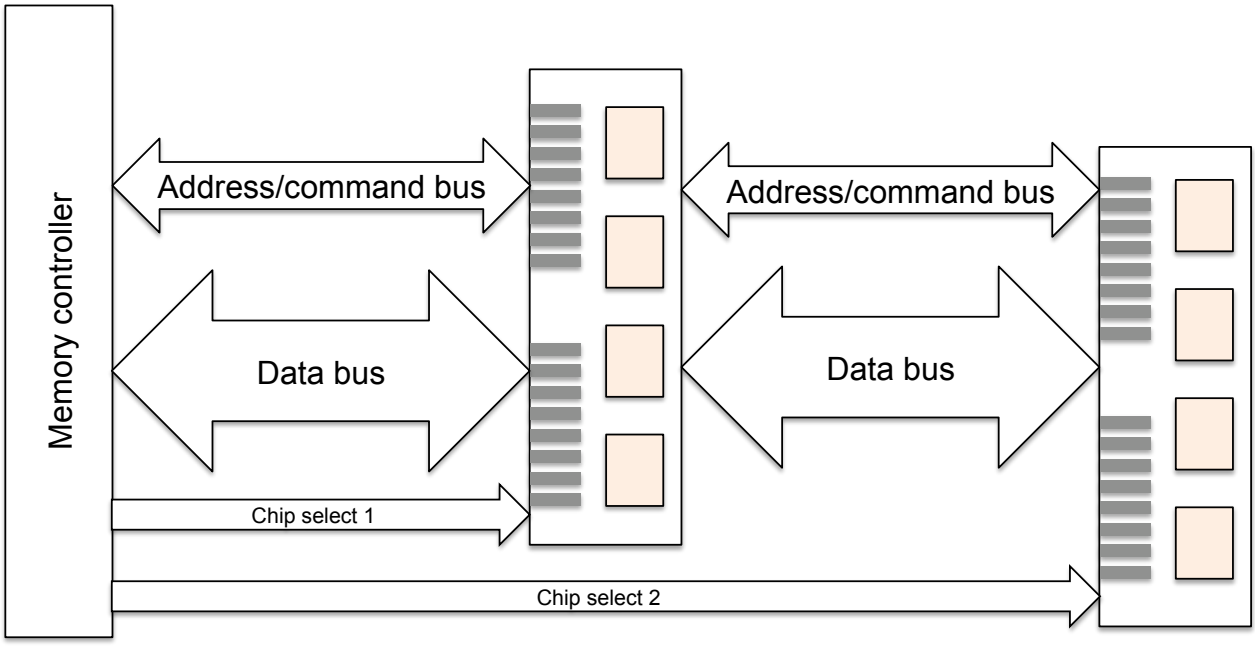
Hierarchy of DRAM memories

- A system has multiple DIMMs
- Each DIMM has multiple DRAM devices in one or more ranks
- Each DRAM device has multiple banks
- Each bank has multiple memory arrays
- Concurrency in ranks and banks increases memory bandwidth

Processor–DRAM interconnect

- Buses
  - Address/command lines
  - Data lines (wide, $>=$ 64 bits in leading processors)
  - Chip select lines
- Recent systems adopt increasingly more scalable solutions
  - Point-to-point, crossbar interconnects
  - Hypertransport, Intel CSI/QuickPath

Processor–DRAM bus organization



Memory controller

Controller operation

- Device executing processor memory requests
- Separate off-processor chip in earlier systems
- Integrated on-chip with the processor in modern systems
- Bus, point-to-point, crossbar interconnect with processor

Lifetime of a memory access

Steps in memory access

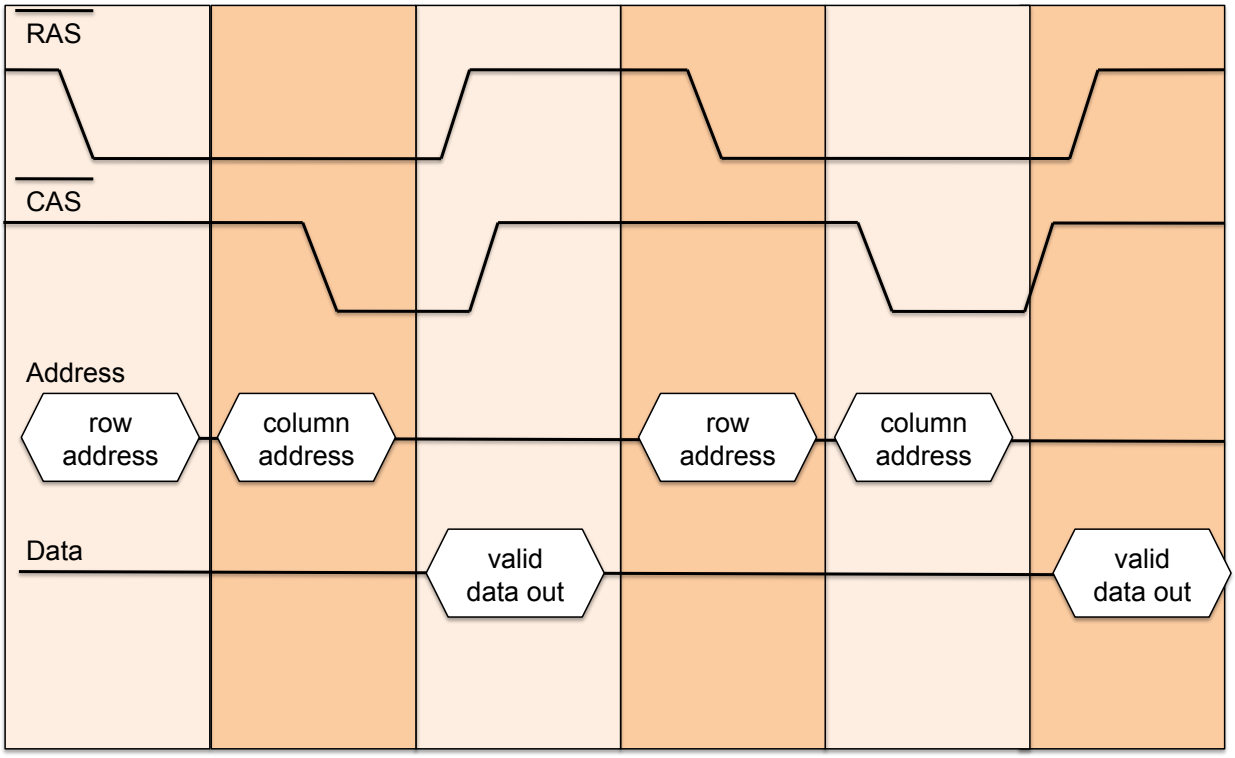
1. Processor orders and queues memory requests
2. Request sent to memory controller
3. Controller queues and orders requests
4. For request in head of queue, controller waits until requested DRAM ready
5. Controller breaks address bits into rank, bank, bank row, bank column fields
6. Controller sends chip-select signal to select rank
7. Selected bank at selected rank precharged to activate selected row

Lifetime of a memory access

Steps in memory access

8. Activate row in DRAMs of selected bank in selected rank
   - Use RAS (row-address strobe signal)
9. Send entire row to sense amplifiers
   - Sense amps may already have a valid row
10. Select desired column using CAS (column-address strobe)

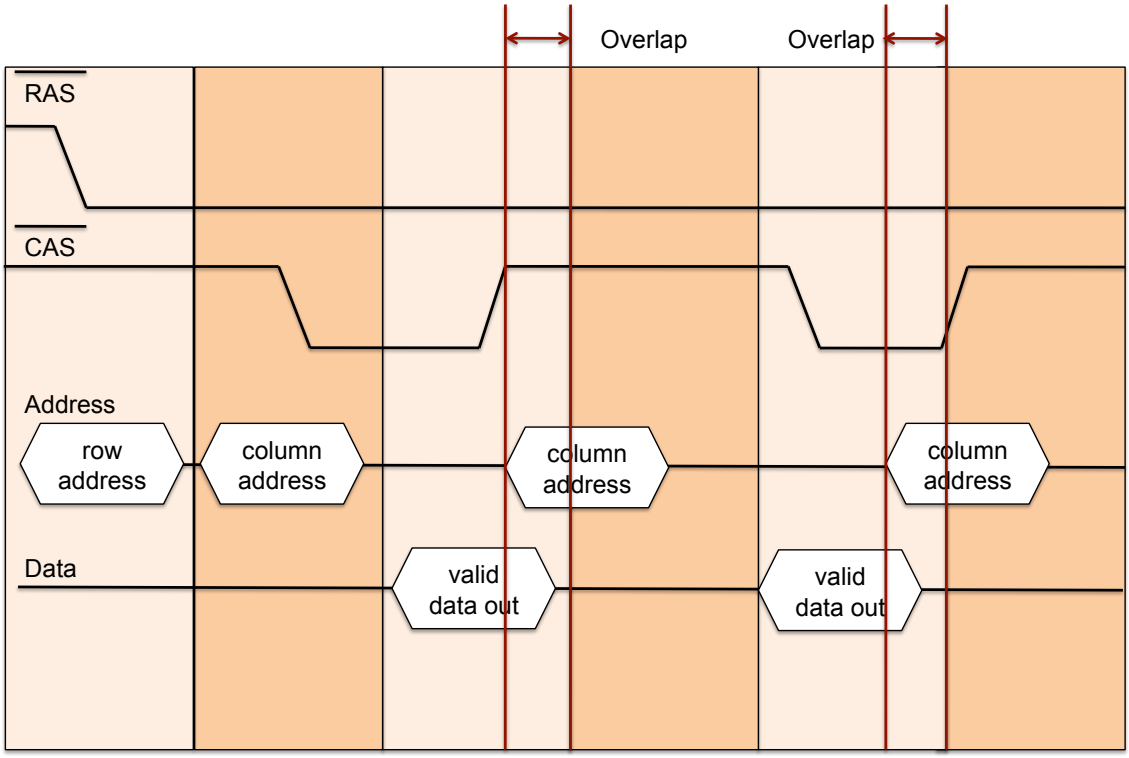
Asynchronous DRAM timing



Fast Page Mode

- Allow row to remain available (open) for multiple column accesses
- Holds row data in sense amplifiers for longer period
- Memory controller holds RAS signal while changing CAS signal
- Sense amplifiers function as ”cache” for DRAM rows
- Multiple CAS signals can access multiple words in same row
- Exploits spatial locality via successive accesses to same row

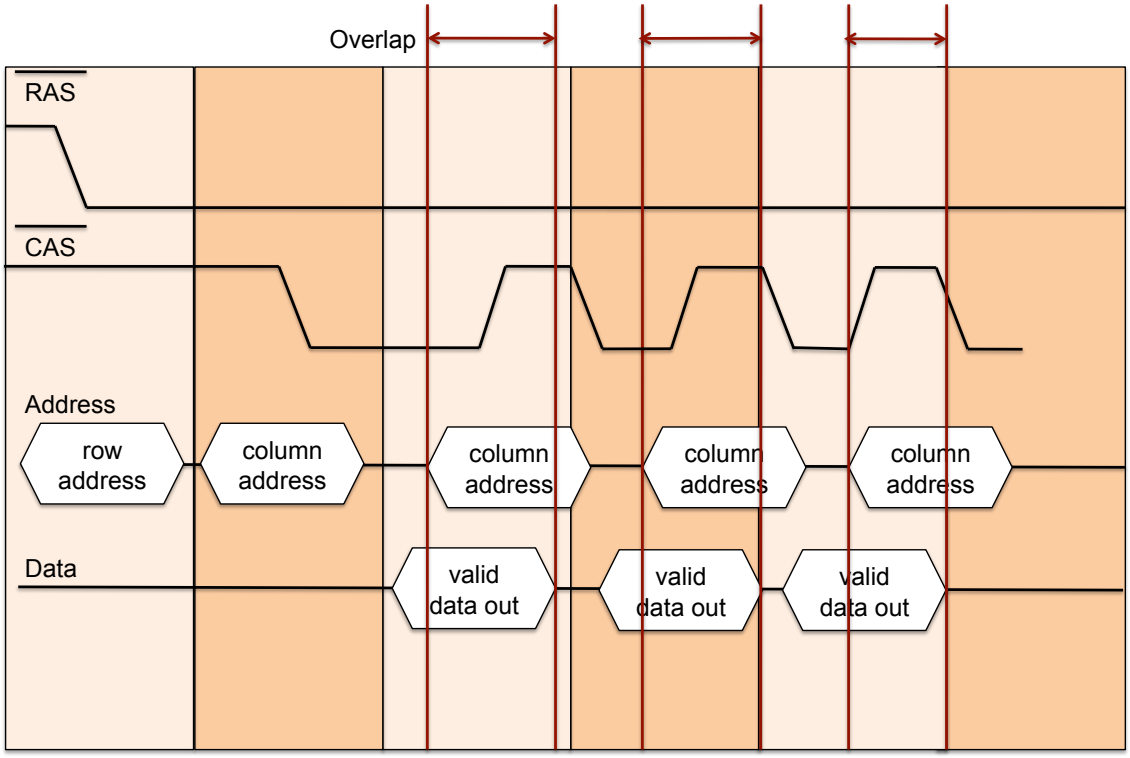
FPM DRAM timing



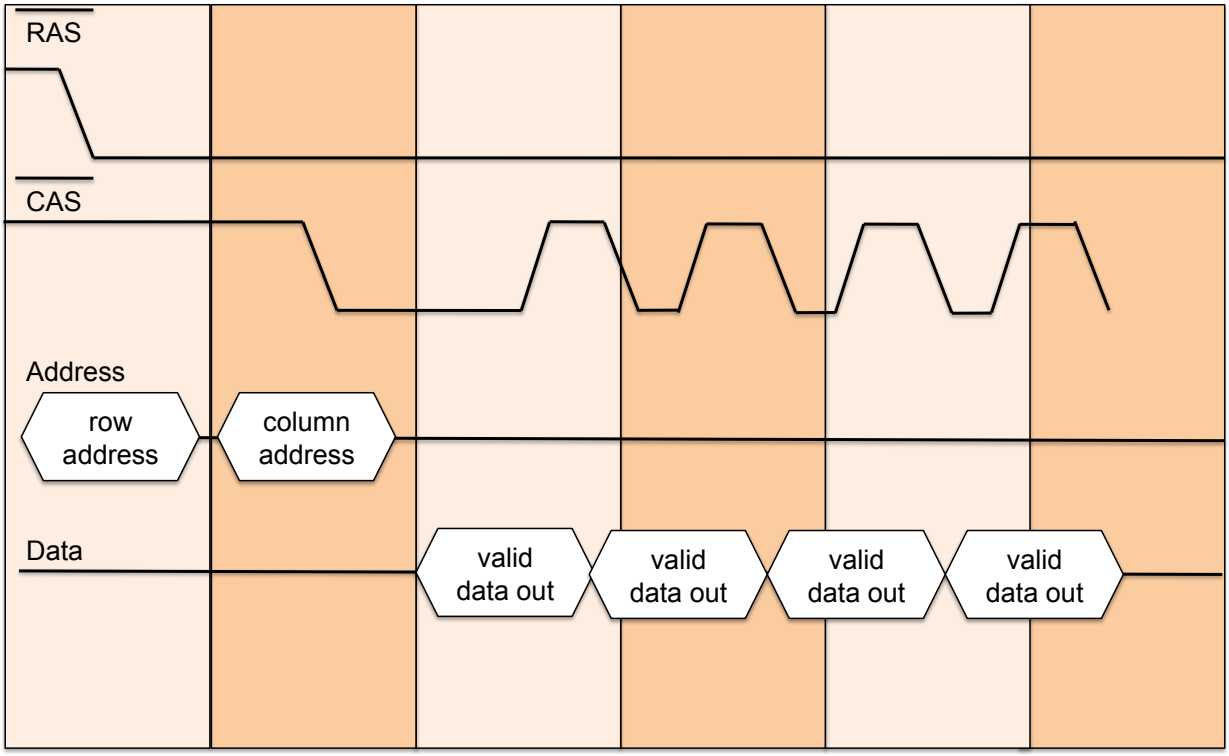
EDO DRAM

- Adds latches to FPM DRAM to permit rapid CAS deassertion
- Accelerates precharging for output
- Latches allow also row in output to remain valid longer
- 10%–15% shorter access time than FPM

EDO DRAM timing



Burst mode EDO DRAM timing

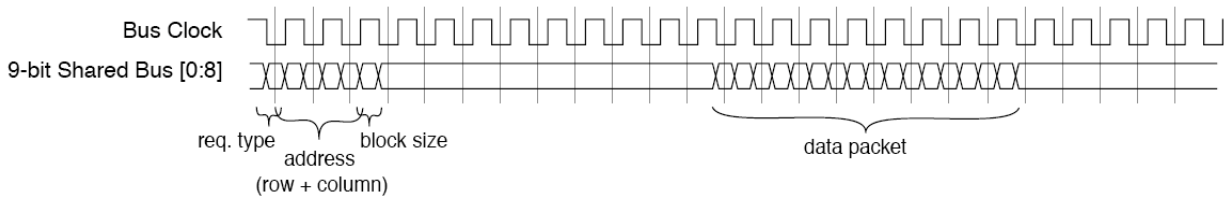


Synchronous DRAM

- Asynchrony in DRAM due to RAS and CAS signals arriving at any time
- Synchronous DRAM uses clock to deliver requests at regular intervals
- More predictable DRAM timing
- Less skew, faster turnaround on requests
- Synchronous DRAMs support burst mode accesses
- Initial performance similar to BEDO DRAM
- Clock scaling enabled higher performance later

Rambus DRAM (RDRAM)

- Fully multiplexed, narrow bus replaces, control, data, address bus
  - 8-bit bus at 250 MHz, delivers 500 MB/s
- Split request-response protocol resembling network protocols



Concurrent Rambus DRAM

- Split bus into address, command and data segments
- 1-byte data segment, 1-bit address segment, 1-bit control segment
  - Later extended to 2 bytes data, 5 bits address, 3 bits control
  - Frequency also increased to 500 MHz
- Perform simultaneous command, address, data transmit on bus

Modern DRAM designs

- Double Data Rate (DDR) SDRAM
  - Double data transfer rate by transferring at both clock edges
  - Otherwise almost identical to single data rate DRAM
- Virtual Channel Memory SDRAM
  - Adds a real cache (SRAM) to buffer large data blocks
  - Increased read/write latency on miss
- Fully Buffered DIMM
  - Channel speed improving at the expense of channel capacity
  - Memory controllers on DIMMS
  - Replace shared bus with point-to-point connections between controllers and DRAMs
  - Higher storage capacity without sacrificing bandwidth

Virtual Memory 101

Why VM?

- Share a physical address space among many processes
- Providing protection between processes
- Handle efficiently processes with sparse address spaces
- Load physical memory on-demand
- Load programs anywhere in physical memory (relocation)
- Run programs too large to fit in physical memory

Virtual Memory 101

VM terminology

- Page or segment correspond to block
  - Pages are fixed-size, segments are variable-size blocks
- CPU produces virtual addresses translated to physical addresses

VM versus caches

- Replacement controlled by operating system versus hardware
- Memory miss penalty huge compared to cache miss penalty
  - Makes replacement decision extremely important

Cache vs. VM parameter comparison

Parameter	First-level cache	Virtual memory
Block (page) size	16–128 bytes	4096–65,536 bytes
Hit time	1–3 clock cycles	50–150 clock cycles
Miss penalty (Access time) (Transfer time)	8–150 clock cycles (6–130 clock cycles) (2–20 clock cycles)	1,000,000–10,000,000 clock cycles (800,000–8,000,000 clock cycles) (200,000–2,000,000 clock cycles)
Miss rate	0.1–10%	0.00001–0.001%
Address mapping	25–45 bit physical address to 14–20 bit cache address	32–64 bit virtual address to 25–45 bit physical address

Design choices

Block placement

- Miss penalty huge compared to cache
- OS designer opts for lower miss rate
- Fully associative placement
  - Exception: page coloring
  - Page consecutive VM in consecutive physical frames pages to avoid cache conflicts
  - Requires knowledge of cache organization and cache mapping scheme

Design choices

Finding the block in memory

- Page tables or segment tables or segmented paging
  - Common optimizations: inverted page tables, multi-level page tables
- TLB for fast address translation

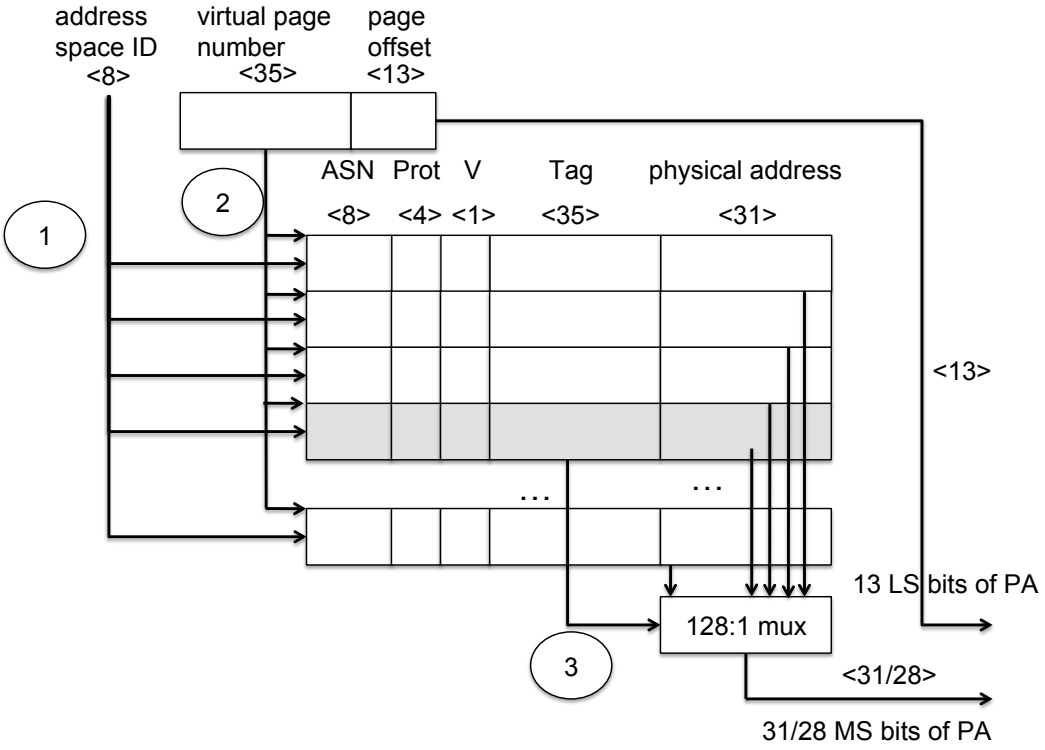
Selecting block for replacement

- Approximations of LRU with one or more use and reference bits

Write policy

- Always write-back due to disk latency

Alpha 21264 TLB example



Alpha TLB in detail

Design choices

- Virtually addressed TLB
  - Uses address space identifier (PID)
  - Avoids flushes on context switches
- No use or reference bit
  - System periodically clears permission bits (read, write)
  - Recorded reads, writes serve as reference/use bits
  - No need to write to TLB during normal memory accesses

Selecting page size

Trade-off's

- Larger page size means smaller page tables
- Larger page size can enable a larger virtually-indexed, physically-tagged L1 cache
- Transferring large pages from disk can be more efficient (latency lags bandwidth)
- Less TLB entries, more memory mapped in the TLB
- Smaller page size means less memory waste due to internal fragmentation