Energy-Aware Computing

Lecture 9: Memory structures and caches

Outline

- Memory circuits
  - SRAM
  - CAM
- Memory hierarchy energy efficiency
- Cache organisation

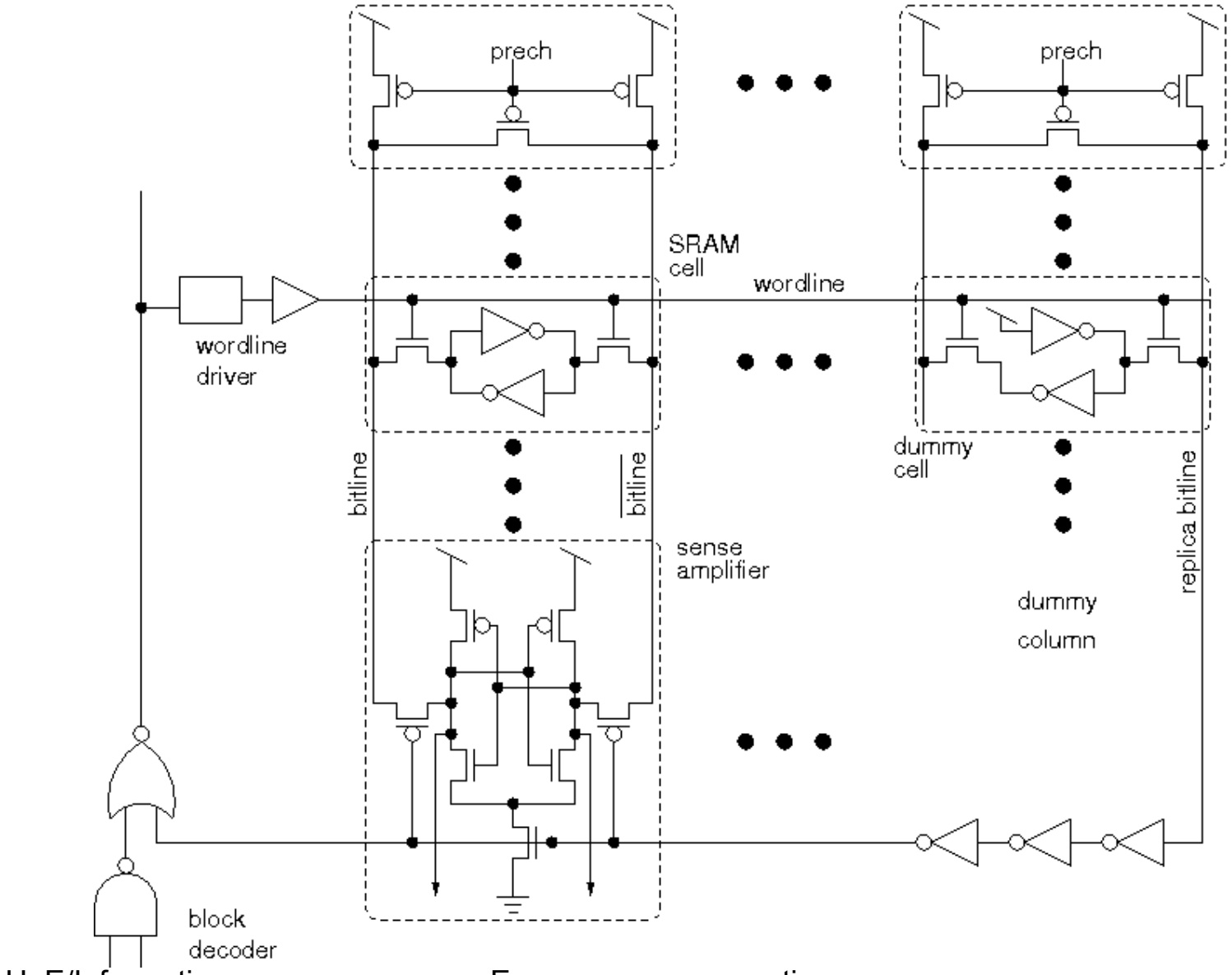
Next lecture:

- Value compression in memories
- Reducing cache parallel activity
- Controlling cache idle capacity

SRAM

- Used for holding state:
  - Caches, branch predictor tables, register files, …
- Can have multiple ports for reading/writing
  - E.g. register file
- Much smaller and regular than normal registers (e.g. used for pipeline latches)
  - 6 transistors per bit only
- Compared to DRAM
  - A lot faster
  - A lot larger (DRAM 1 transistor + capacitor per bit)
  - Static: does not need refreshing

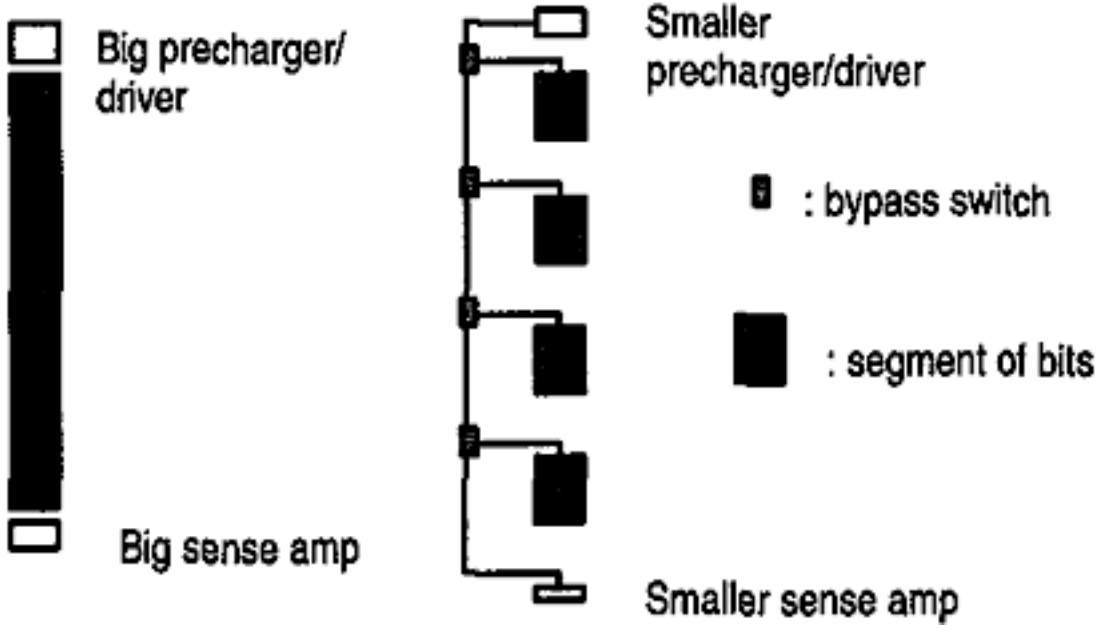
Static RAM


prech
prech
SRAM
cell
wordline
wordline
driver
dummy
cell
bitline
bitline
replica bitline
sense
amplifier
dummy
column
block
decoder

SRAM speed/power

- Operation sequence (read)
  - Pre-decode, decode, drive WL
  - Cell pulls down BL
  - Sensamp fires
- Speed/power depends on
  - Wordline length (cells per row)
  - Bitline length (cells per column)
  - Periphery (row decoder, col mux, …)
- Smaller arrays are faster/low power

Bit-line segmentation



- Reduce the effective C by isolating segments
  - The array row to be accessed is known
  - 1/Nth of the drain capacitance of access transistors
  - 2x the wire C for active segment, some C from the switches, large C for driving switches on/off

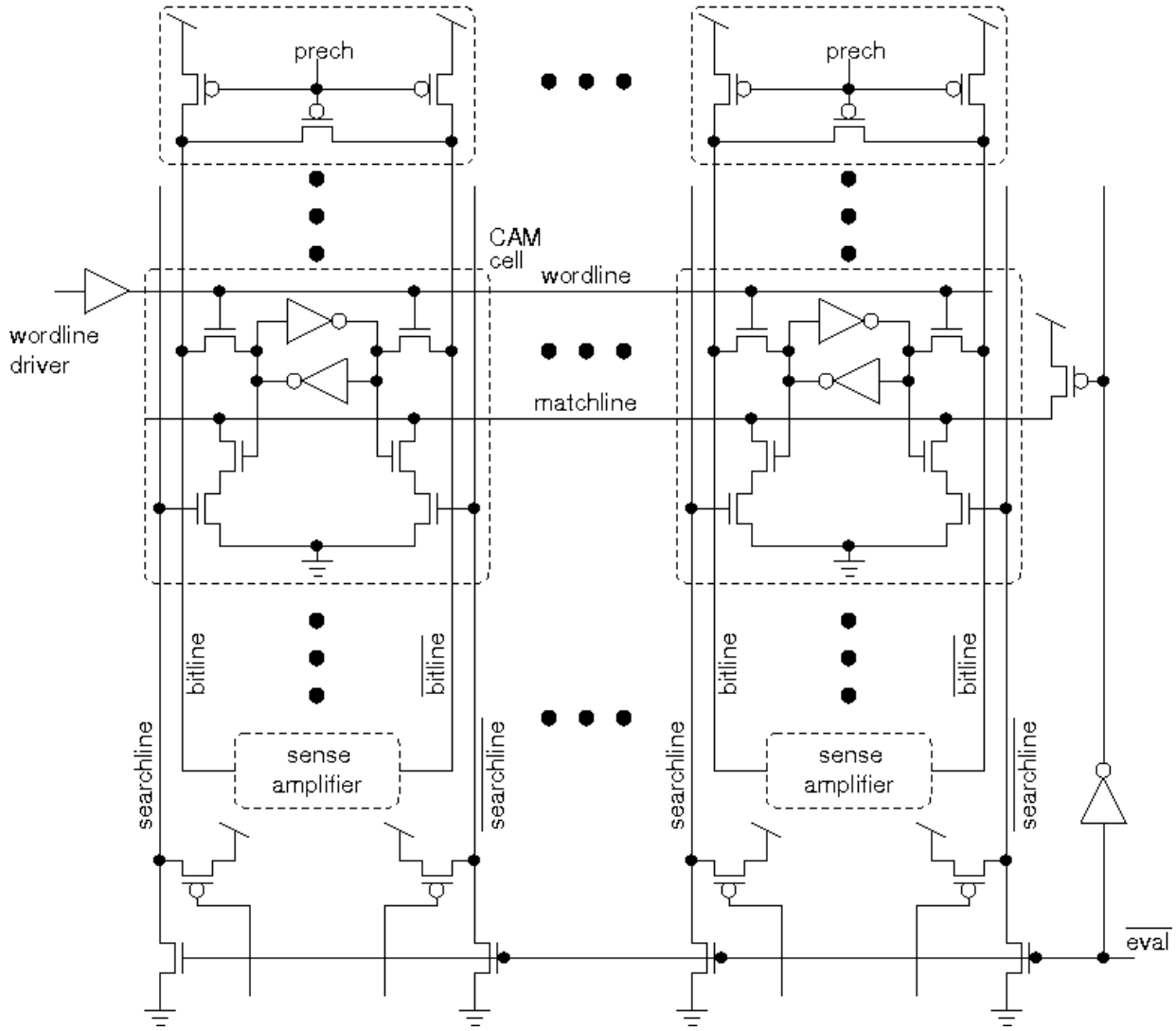
Sub-banking

- Sub-banking term used for many techniques
  - Column multiplexing
  - Column disabling
  - Using multiple, independent small arrays
    - Allows pseudo multi-porting
- The case for sub-banking:
  - Energy/access increases with SRAM array size
  - With sub-banking the actual size accessed remains constant
  - Routing power increases with total storage capacity
  - There's an optimal array size (sub-bank)

Content addressable memory

- Similar to SRAM
  - Organisation, read, write operations
- Able to perform parallel searches
  - Is this value in the memory?
- Very useful for:
  - Tags of highly associative caches
  - Processor issue queues
  - TLBs

CAM circuit


prech
prech
CAM
cell
wordline
wordline
driver
matchline
bitline
bitline
bitline
bitline
searchline
searchline
searchline
searchline
sense
amplifier
sense
amplifier
eval

Power/energy efficiency of caching

Energy efficiency of memory hierarchy:

- Smaller memory is more power efficient
- Access to main memory is very expensive in power consumption
  - Main mem is off chip
- Disadvantage:
  - Power associated with tag handling, line replacement, etc.

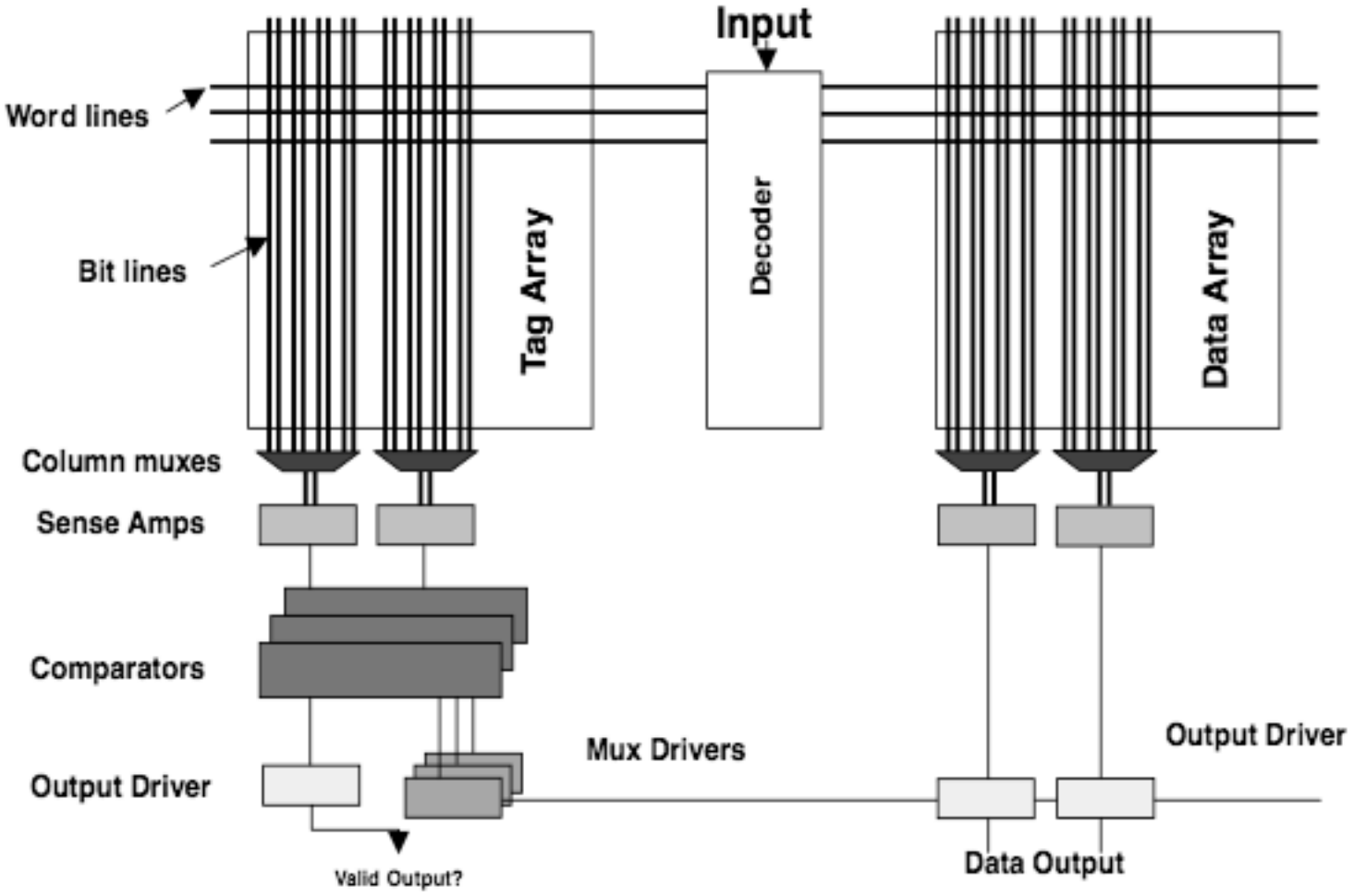
Filter cache

- Tiny cache positioned at “level 0”
  - 128-256 bytes
- Filters out most accesses
  - Around 60%
- Misses in filter cache take longer
- Classic Energy x Delay problem
  - Smaller filter cache more efficient
  - But with worse hit rate -> slower system

Line buffering

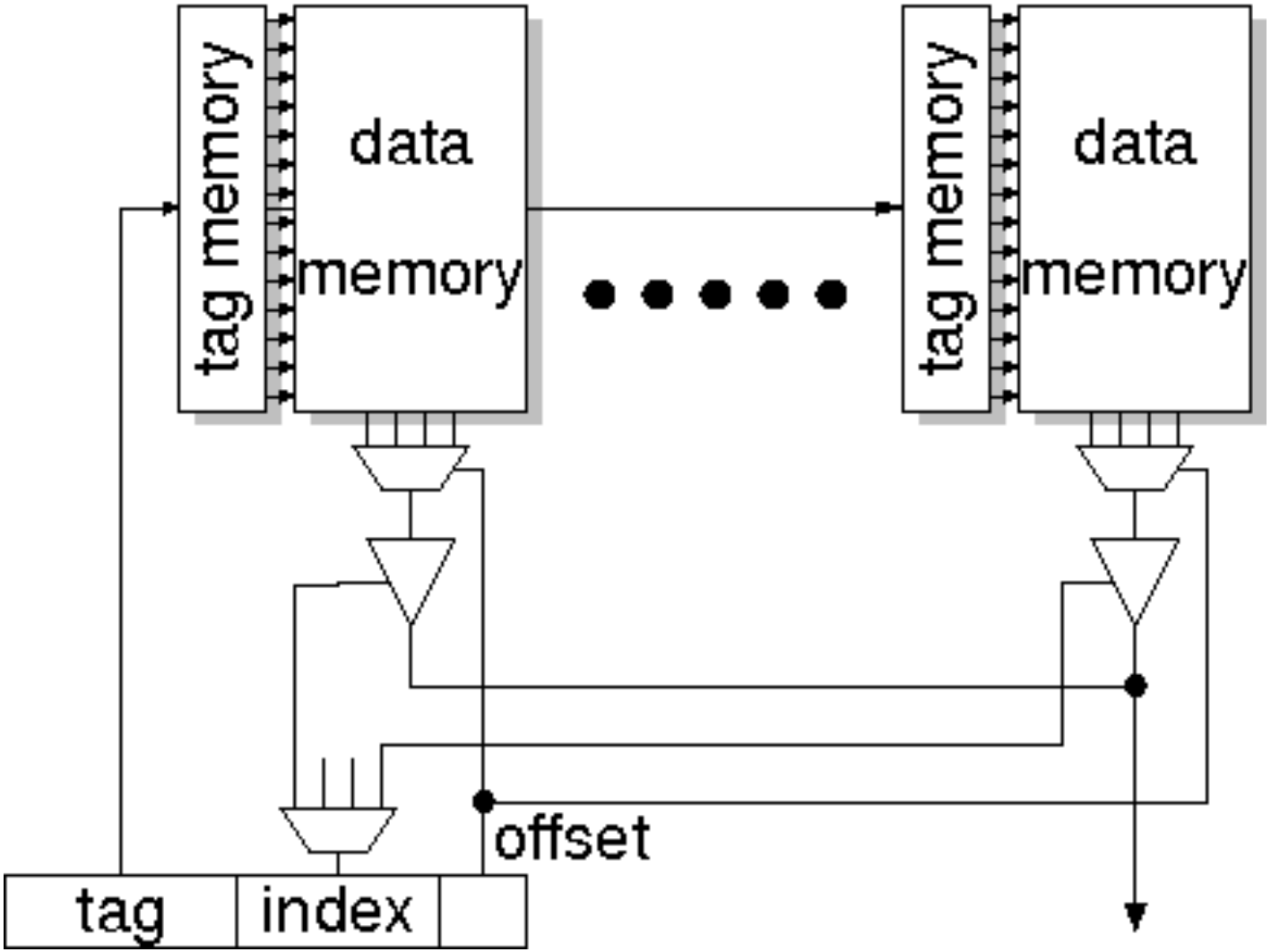
- Similar to filter cache but integrated in L1 cache
- A cache block read out is usually stored in an internal register
- Use this register as a “filter cache”
  - Or keep the last few ones
- Access to LB in parallel to L1 access
  - If LB hit, stop L1 (recall pre-computation idea)
  - No slow-down as filter-cache but some extra energy per access for LB misses

Cache organisation: RAM-tags

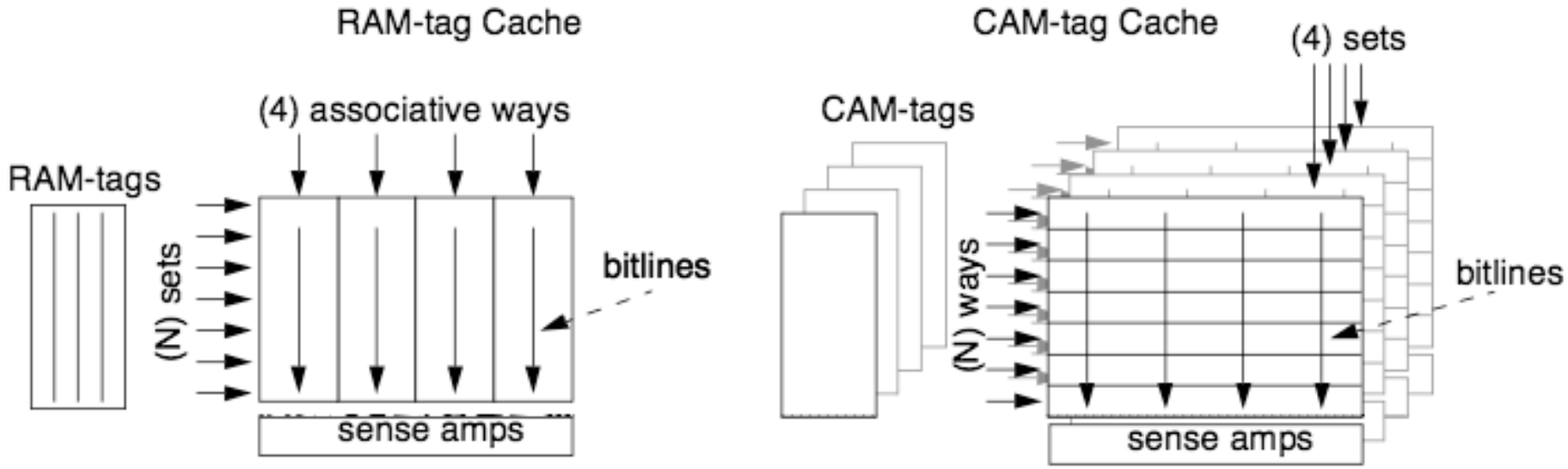


Src: P. Shivakumar, N. Joupi, Cacti3.0, WRL tech rep 2001/2

Cache organisation: CAM tags



Associative cache organisation



- Ways: locations available to store a cache block
- Set: The collection of all the ways where a block can be stored

Summary

- SRAM organisation
- SRAM power a function of size (rows, columns)
- CAM organisation / power
- Memory hierarchy
  - Power efficient
  - Improvements: filter cache, line buffering
- Cache organisation
  - RAM tags
  - CAM tags