Lecture: Memory Technology Innovations

- Topics: memory schedulers, refresh,
  state-of-the-art and upcoming changes:
  buffer chips, 3D stacking, non-volatile cells, photonics

- Midterm scores: 90+ is top 20, 85+ is top 40,
  79+ is top 60, 69+ is top 80,

- Common errors: SWP, power equations

Row Buffers

- Each bank has a single row buffer

- Row buffers act as a cache within DRAM
  - Row buffer hit: ~20 ns access time (must only move data from row buffer to pins)
  - Empty row buffer access: ~40 ns (must first read arrays, then move data from row buffer to pins)
  - Row buffer conflict: ~60 ns (must first precharge the bitlines, then read new row, then move data to pins)

- In addition, must wait in the queue (tens of nano-seconds) and incur address/cmd/data transfer delays (~10 ns)

Open/Closed Page Policies

- If an access stream has locality, a row buffer is kept open
    - Row buffer hits are cheap (open-page policy)
    - Row buffer miss is a bank conflict and expensive because precharge is on the critical path

- If an access stream has little locality, bitlines are precharged immediately after access (close-page policy)
    - Nearly every access is a row buffer miss
    - The precharge is usually not on the critical path

- Modern memory controller policies lie somewhere between these two extremes (usually proprietary)

Problem 3

- For the following access stream, estimate the finish times for each access with the following scheduling policies:

Req	Time of arrival	Open	Closed	Oracular
X	0 ns			
Y	10 ns			
X+1	100 ns			
X+2	200 ns			
Y+1	250 ns			
X+3	300 ns			

Note that X, X+1, X+2, X+3 map to the same row and Y, Y+1 map to a different row in the same bank. Ignore bus and queuing latencies. The bank is precharged at the start.

Problem 3

- For the following access stream, estimate the finish times for each access with the following scheduling policies:

Req	Time of arrival	Open	Closed	Oracular
X	0 ns	40	40	40
Y	10 ns	100	100	100
X+1	100 ns	160	160	160
X+2	200 ns	220	240	220
Y+1	250 ns	310	300	290
X+3	300 ns	370	360	350

Note that X, X+1, X+2, X+3 map to the same row and Y, Y+1 map to a different row in the same bank. Ignore bus and queuing latencies. The bank is precharged at the start.

Problem 4

- For the following access stream, estimate the finish times for each access with the following scheduling policies:

Req	Time of arrival	Open	Closed	Oracular
X	10 ns			
X+1	15 ns			
Y	100 ns			
Y+1	180 ns			
X+2	190 ns			
Y+2	205 ns			

Note that X, X+1, X+2, X+3 map to the same row and Y, Y+1 map to a different row in the same bank. Ignore bus and queuing latencies. The bank is precharged at the start.

Problem 4

- For the following access stream, estimate the finish times for each access with the following scheduling policies:

Req	Time of arrival	Open	Closed	Oracular
X	10 ns	50	50	50
X+1	15 ns	70	70	70
Y	100 ns	160	140	140
Y+1	180 ns	200	220	200
X+2	190 ns	260	300	285
Y+2	205 ns	320	240	225

Note that X, X+1, X+2, X+3 map to the same row and Y, Y+1 map to a different row in the same bank. Ignore bus and queuing latencies. The bank is precharged at the start.

Address Mapping Policies

- Consecutive cache lines can be placed in the same row to boost row buffer hit rates

- Consecutive cache lines can be placed in different ranks to boost parallelism

- Example address mapping policies:

  row:rank:bank:channel:column:blkoffset

  row:column:rank:bank:channel:blkoffset

Reads and Writes

- A single bus is used for reads and writes
- The bus direction must be reversed when switching between reads and writes; this takes time and leads to bus idling
- Hence, writes are performed in bursts; a write buffer stores pending writes until a high water mark is reached
- Writes are drained until a low water mark is reached

Scheduling Policies

- FCFS: Issue the first read or write in the queue that is ready for issue

- First Ready - FCFS: First issue row buffer hits if you can

- Close page -- early precharge

- Stall Time Fair: First issue row buffer hits, unless other threads are being neglected

Refresh

- Every DRAM cell must be refreshed within a 64 ms window
- A row read/write automatically refreshes the row
- Every refresh command performs refresh on a number of rows, the memory system is unavailable during that time
- A refresh command is issued by the memory controller once every 7.8us on average

Problem 5

- Consider a single 4 GB memory rank that has 8 banks. Each row in a bank has a capacity of 8KB. On average, it takes 40ns to refresh one row. Assume that all 8 banks can be refreshed in parallel. For what fraction of time will this rank be unavailable? How many rows are refreshed with every refresh command?

Problem 5

- Consider a single 4 GB memory rank that has 8 banks. Each row in a bank has a capacity of 8KB. On average, it takes 40ns to refresh one row. Assume that all 8 banks can be refreshed in parallel. For what fraction of time will this rank be unavailable? How many rows are refreshed with every refresh command?

  The memory has 4GB/8KB = 512K rows
  There are 8K refresh operations in one 64ms interval.
  Each refresh operation must handle 512K/8K = 64 rows
  Each bank must handle 8 rows
  One refresh operation is issued every 7.8us and the memory is unavailable for 320ns, i.e., for 4% of time.

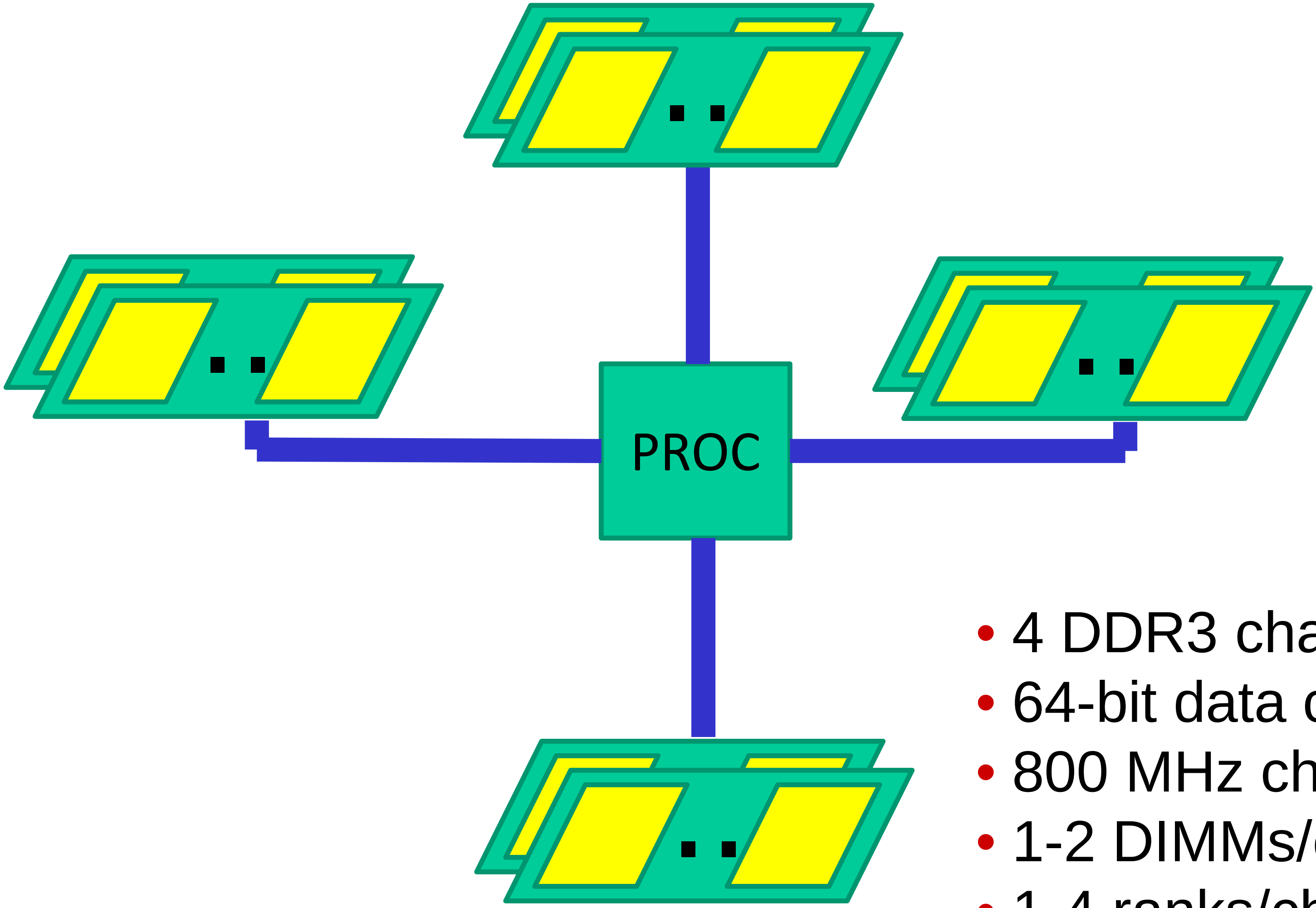
Error Correction

- For every 64-bit word, can add an 8-bit code that can detect two errors and correct one error; referred to as SECDED – single error correct double error detect

- A rank is now made up of 9 x8 chips, instead of 8 x8 chips

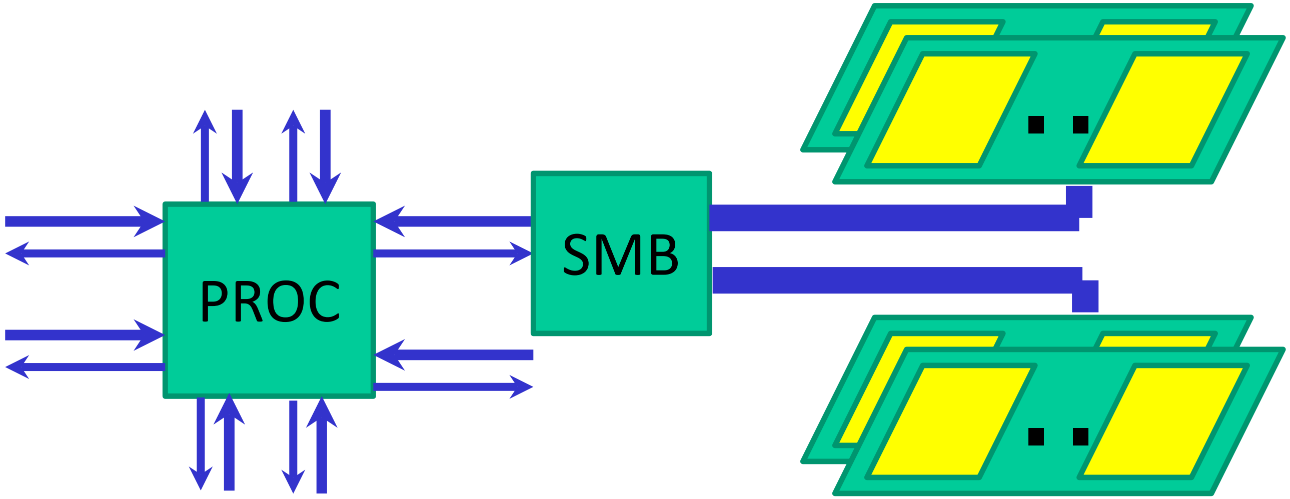
- Stronger forms of error protection exist: a system is chipkill correct if it can handle an entire DRAM chip failure

Modern Memory System



- 4 DDR3 channels
- 64-bit data channels
- 800 MHz channels
- 1-2 DIMMs/channel
- 1-4 ranks/channel

Cutting-Edge Systems



- The link into the processor is narrow and high frequency
- The Scalable Memory Buffer chip is a “router” that connects to multiple DDR3 channels (wide and slow)
- Boosts processor pin bandwidth and memory capacity
- More expensive, high power

Title

- Bullet