Chapter 08: The Memory System

Lesson 07:

Page mode and fast-page mode DRAMs

Objective

- Learn total access time computation in asynchronous DRAM operations in page modes

Page mode design of asynchronous DRAM

Alternative memory chip design (page mode)

- If the contents of the row could be kept in or near the multiplexer, it would be possible to read other bits within the same row by just sending a different column address (in subsequent accesses after the first) to the DRAM, rather than doing a full RAS-CAS cycle

Page Mode and fast page mode Access DRAMs

- DRAMs that do the RAS remains active during the sequence of column address transfers using CAS before the start of each new read or write are called page-mode access DRAMs
- A page means an entire row of data
- DRAM that latches latch DRAM row data so that for each new column address sent the row outputs are available instantaneously are called fast-page mode access DRAM

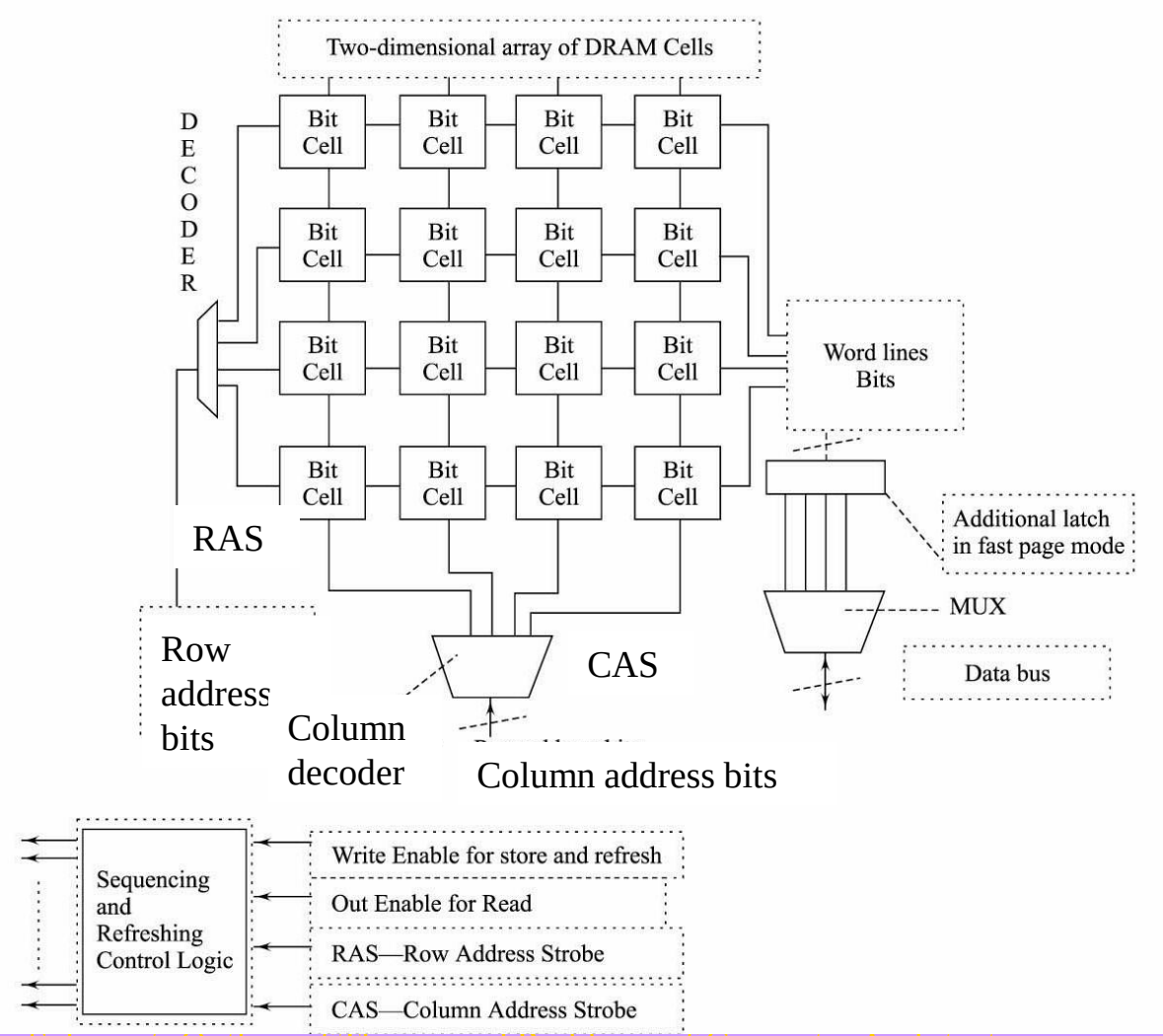
Page Mode and fast page mode Access DRAMs

- DRAM that use latch to latch the DRAM row data so that for each new column address sent the row outputs are available instantaneously are called fast-page mode access DRAM

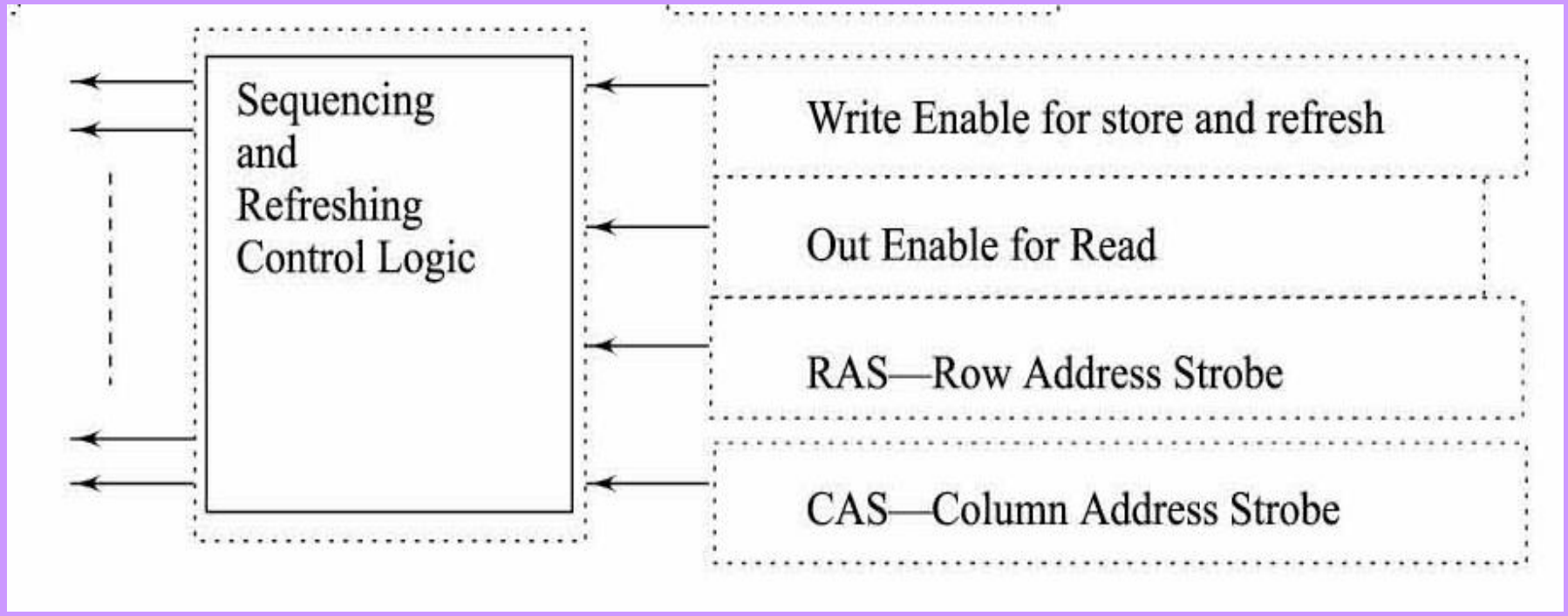
Page Mode Access DRAMs

- RAS remains active to load the row addresses
- RAS remains active during the sequence of column address transfers using CAS before the start of each new read or write
- RAS remains active to load the row addresses
- Entire row defines one page and column address defines different pages

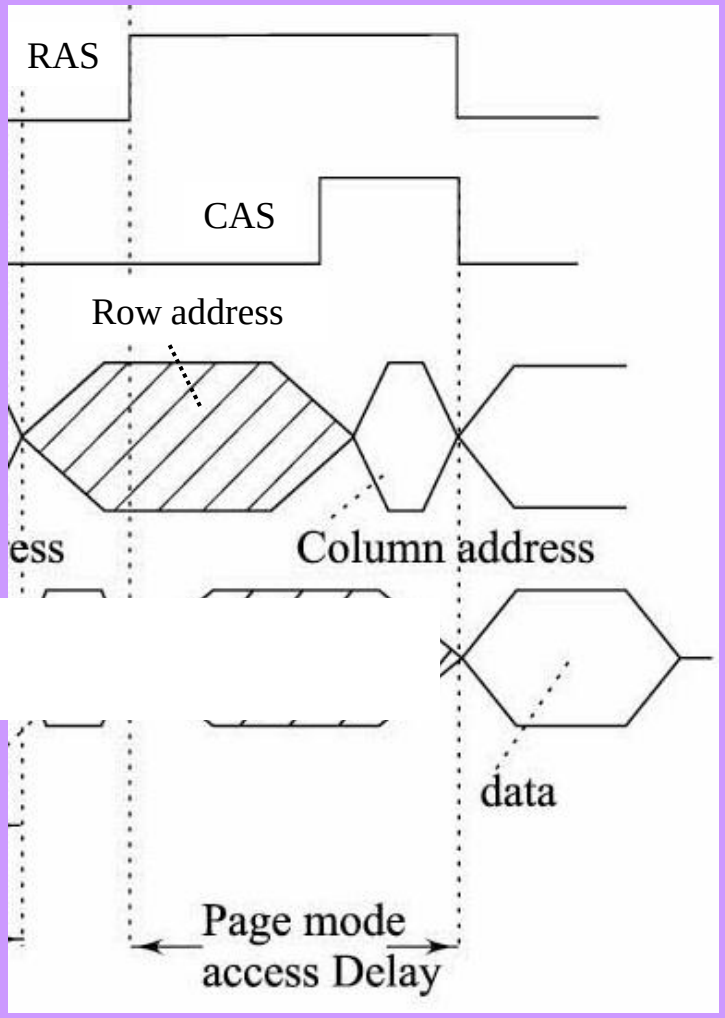
Page mode DRAM with a latch in fast page mode DRAM



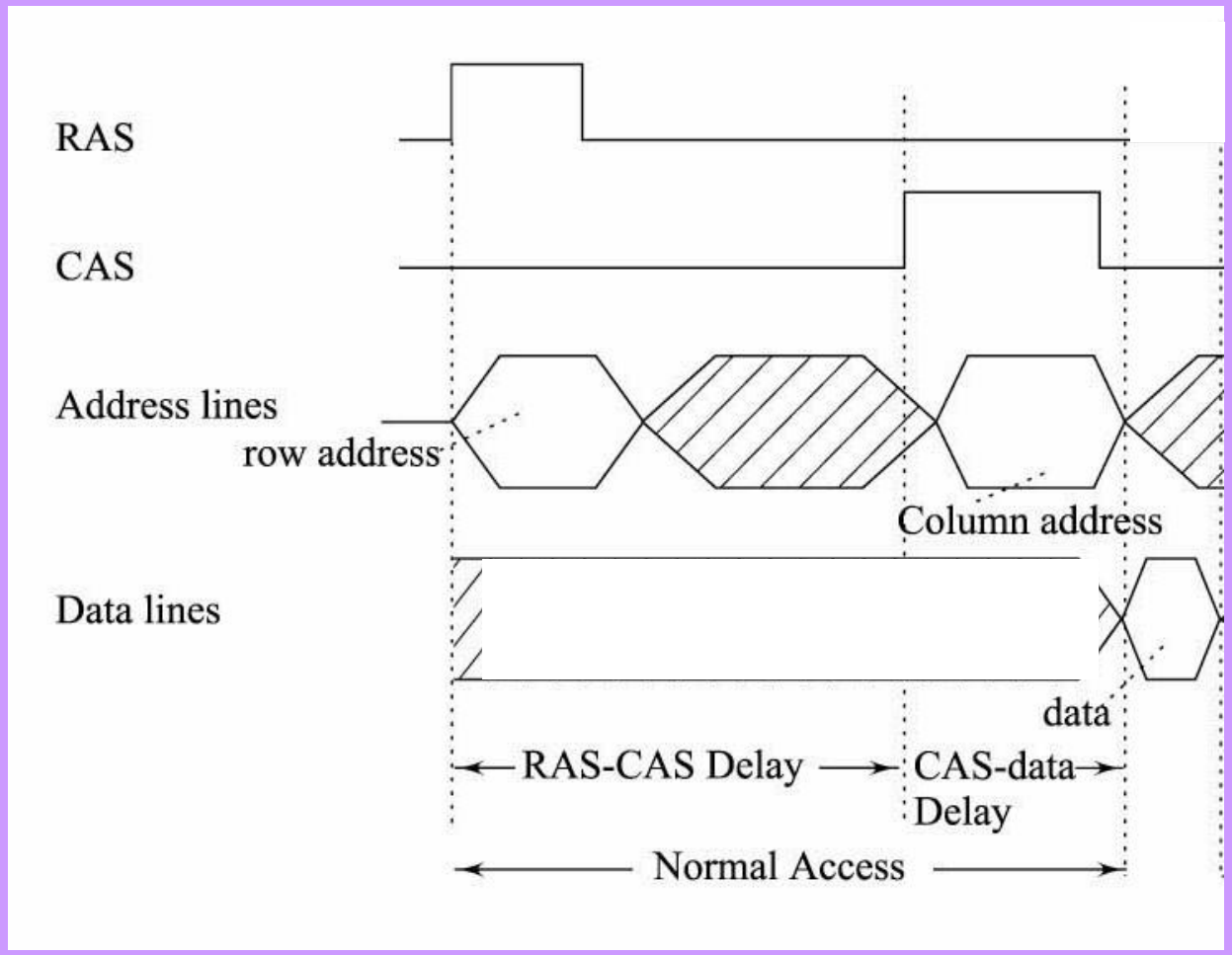
Sequencing and refresh control logic



Page mode DRAM Access (RAS remain active to load row addresses during sequence of column address transfer



Normal mode DRAM Access



Access Times in Page Mode

DRAM Access timings in page access mode

- Whenever a row address is sent to the DRAM, the entire contents of the row are stored in the latch, because RAS remains active
- This allows subsequent accesses that reference a column within the same row to simply send a second column address to the DRAM

DRAM Access timings in page access mode

- It greatly reducing the time required to fetch a contiguous block of data from the DRAM

Drain Access Timings

- RAS (row address strobe) between t0 and t1
- and CAS (column address strobe) between t0 + Δt0" and t1
- RAS and CAS signals overlaps

Drain Access Timings

- Row address being sent between t0 – Δt0' and t1 + Δt1'
- Column addresses being sent between t0 + Δt0"– Δt0' and t1 + Δt1'
- Data available from during t1 – Δt1 to next RAS-CAS cycle

Data Delay Timings

- Time between t0 and t1 + Δt1 is page mode access delay data

Total DRAM taken to Read Time

- Total time to read the DRAM is the t1 – t0

Page Mode Example 1

- Suppose that a processor's memory constructed out of page-mode DRAMs
- RAS-CAS delay and CAS-data delay = 50ns each
- The main memory returns 1 word of data for each request from the processor

...Example 1

- If page-mode DRAMs used to implement the main memory of a system whose cache uses 8-word blocks
- Assume─ blocks aligned such that a block's data always lies within the same row of the DRAM and that the memory returns one word of data per access

Solution for RAS-CAS delays are saved to fetch each time a block of memory into cache

- Using page mode, we only need to do a full RAS-CAS cycle to fetch the first word of the block
- Subsequent words can be fetched using only a CAS
- Therefore, (8 – 1)= 7 RAS-CAS delays are saved in page mode each time a block is fetched

Example 2

- Suppose that a processor's memory constructed out of page-mode DRAMs whose RAS-CAS delay and CAS-data delay are 50ns each
- The main memory returns 1 word of data for each request from the processor

... Example

- If the block size of the processor's cache is 16 words
- Find fraction of the memory requests from the processor will be able to use page mode
- Find what will be the reduction in the time to transfer a block from the main memory because of the use of page mode

Solution

- Assume─ blocks are aligned so that they always lie completely within one row of the DRAM
- Assume─ successive blocks lie in different rows of the DRAM
- Also assume─ that the DRAM does not require any precharge time between requests

Solution

- Since the main memory returns 1 word per request from the cache, 16 requests are required to fetch each block
- Successive blocks lie in different rows of the DRAM, but each block is completely contained within a DRAM row, so the first request for each block will be unable to use page mode, but the rest will be able to use page mode

...Solution

- Therefore, 15/16 = 93.8 percent of the requests will use page mode
- If the DRAM did not support page mode, each request would take 100 ns (50 ns RAS-CAS delay + 50 ns CAS-data delay), so transferring a block into the cache would take 1600 ns

...Solution

- With page mode, the first request still takes 100ns, but the next 15 take only 50ns (just the CAS-data delay), so the total time to transfer a block is 850 ns, a saving of 47 percent

Summary

We learnt

- Access cycles and time in Asynchronous DRAM in page mode DRAMs
- First block takes same time as normal mode = RAS + CAS delay
- Successive blocks takes CAS delay only in page mode
- Fast page mode DRAM accesses fast due to use of a latch at the MUX output

End of Lesson 07 on

Page mode and fast-page mode DRAMs