ECE 5730
Memory Systems
Spring 2009

Overview of DRAMs

Cornell University

Announcements

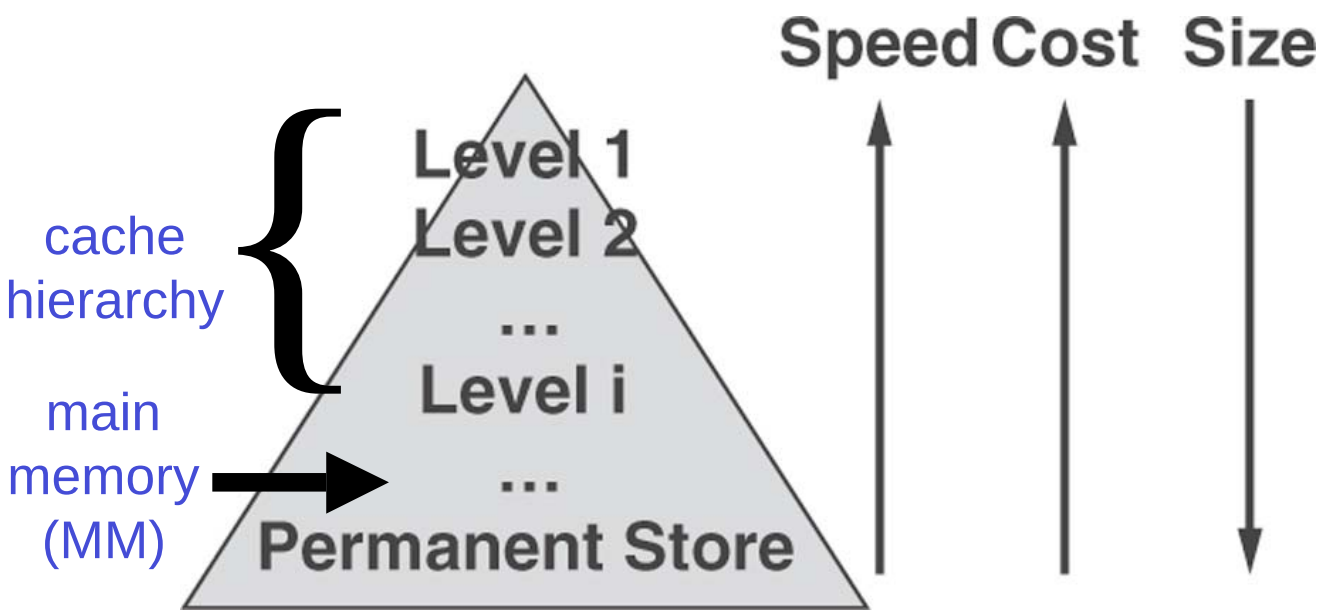
- **Quiz 6 on Tuesday**

- **Quiz 5**
  - **Average = 4.1/10 (ouch!)**

- **Exam I**
  - **Wednesday, March 11, 6:30-9:30pm or 6-9pm or…**
  - **Makeup: Friday, March 13, 12-3pm or…**

Recall the Memory Hierarchy

- **Multiple levels of memory, each optimized for an appropriate cost/performance design point**



[ov.1]

Recall the Memory Hierarchy

- **Multiple levels of memory, each optimized for an appropriate cost/performance design point**

Main Memory

MM storage component

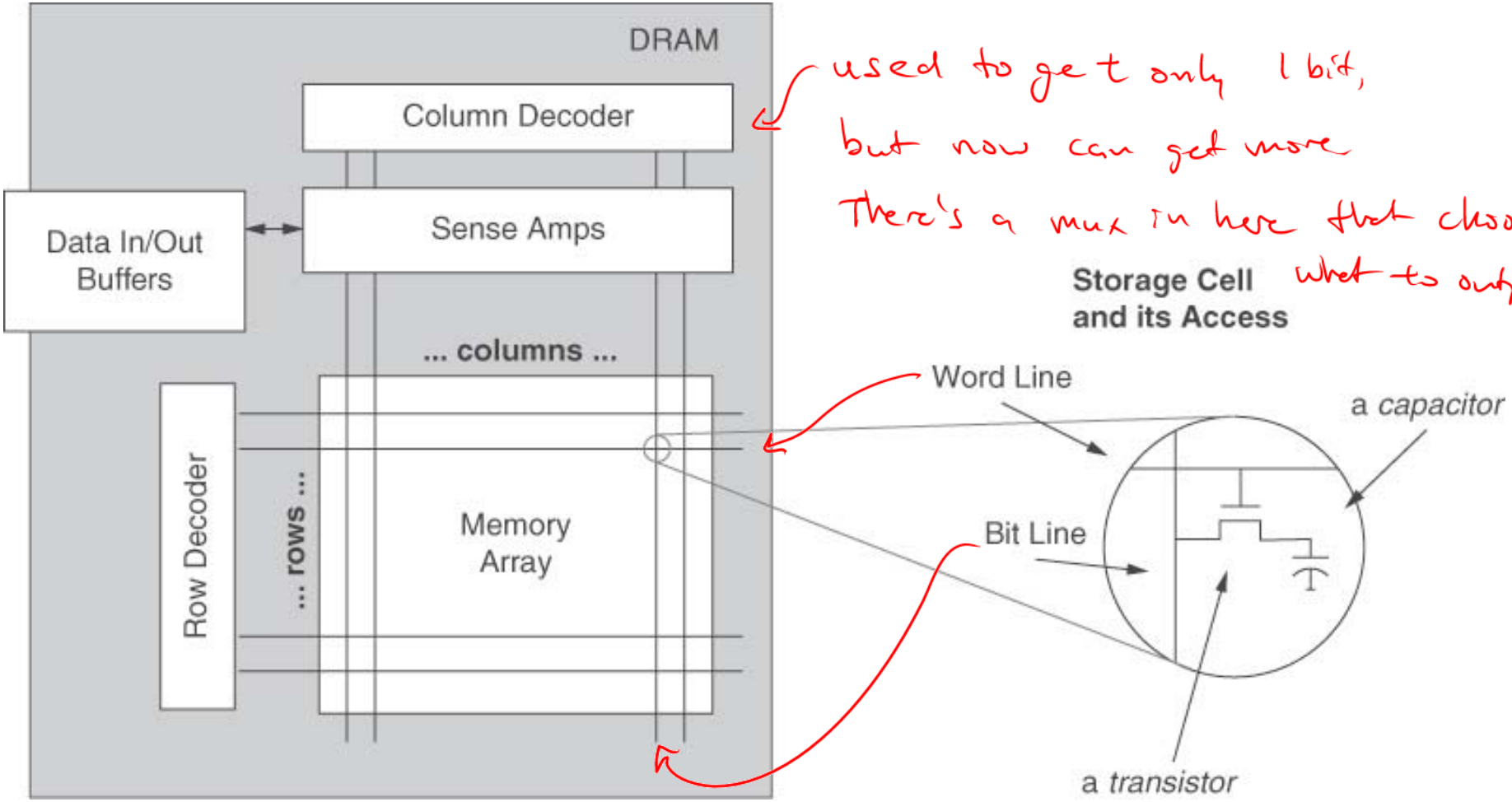
Technology	Bytes per access	Latency per access	Energy per access	Cost per MB
On-chip cache	10	100's of ps	1 nJ	$1-100
Off-chip cache	100	ns	10-100 nJ	$1-10
DRAM	1000 (internally fetched)	10-100 ns	1-100 nJ per device	$0.1
Disk	1000	ms	100-1000 nJ	$0.001

Main Memory

MM Design Tradeoffs

- **Latency** → how long does one random access take?
- **Bandwidth** → how much data can we transfer at a time?
- **Concurrency** → can I do parallel access for different requestors?
  → banking (access different banks simultaneously)
  → need special memory controller
- **Capacity** → space! MORE stuffs!
- **Modularity** → Upgradability. Can I add more RAM later?
- **Power** → low power is good
- **Cost** → typical DRAM optimization is for cost <u>not</u> performance

High-level DRAM Organization

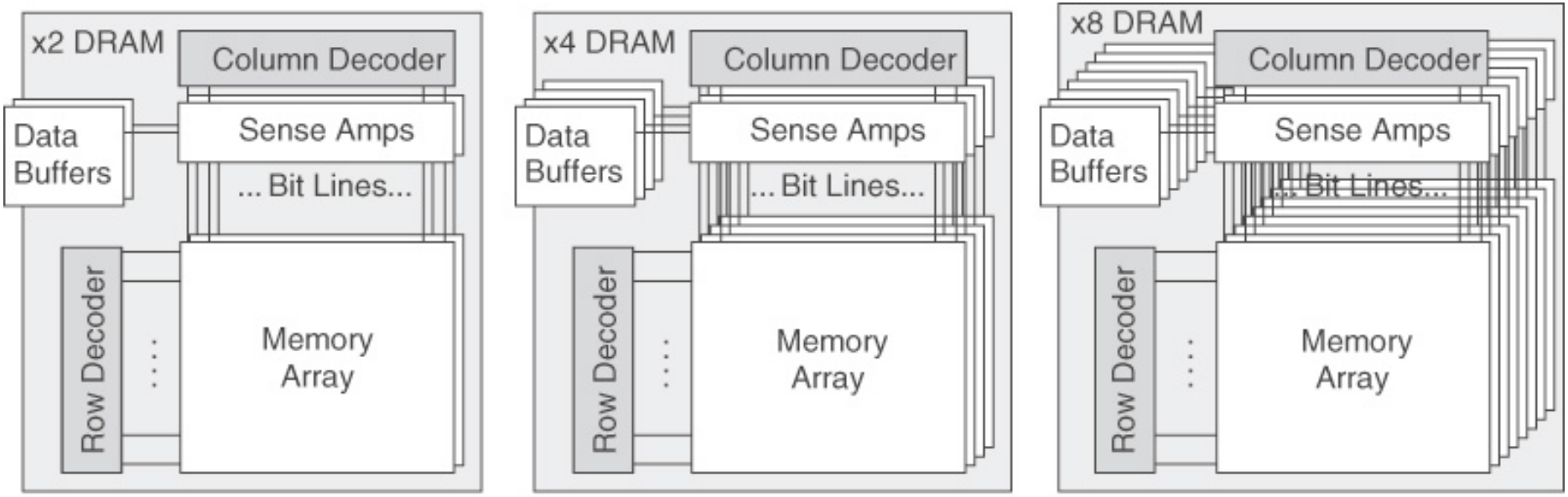


used to get only 1 bit, but now can get more
There's a mux in here that chooses what to output

very simple. complicated @ CMOS level, but still simple @ high level

[7.3]

High-level DRAM Organization



x2 DRAM **x4 DRAM** **x8 DRAM**

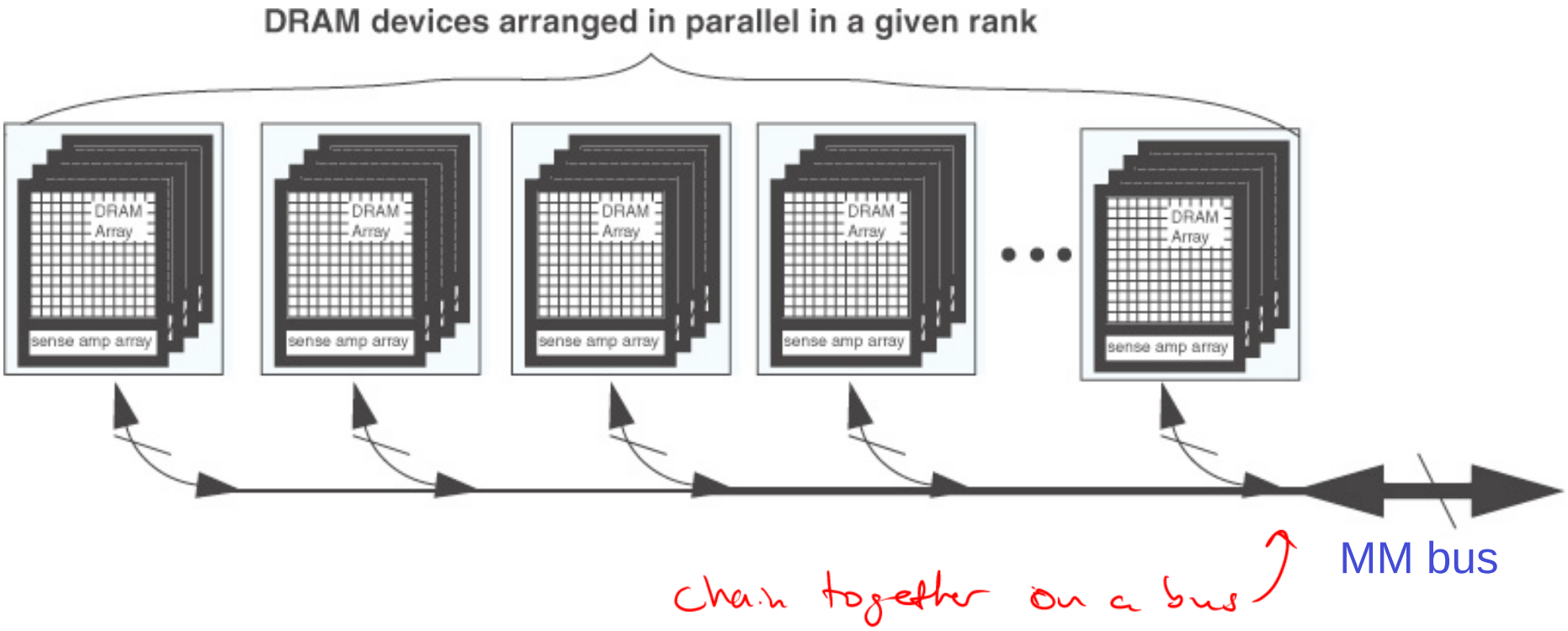
2 arrays

lots of parallel DRAM.

still not enough

[7.4]

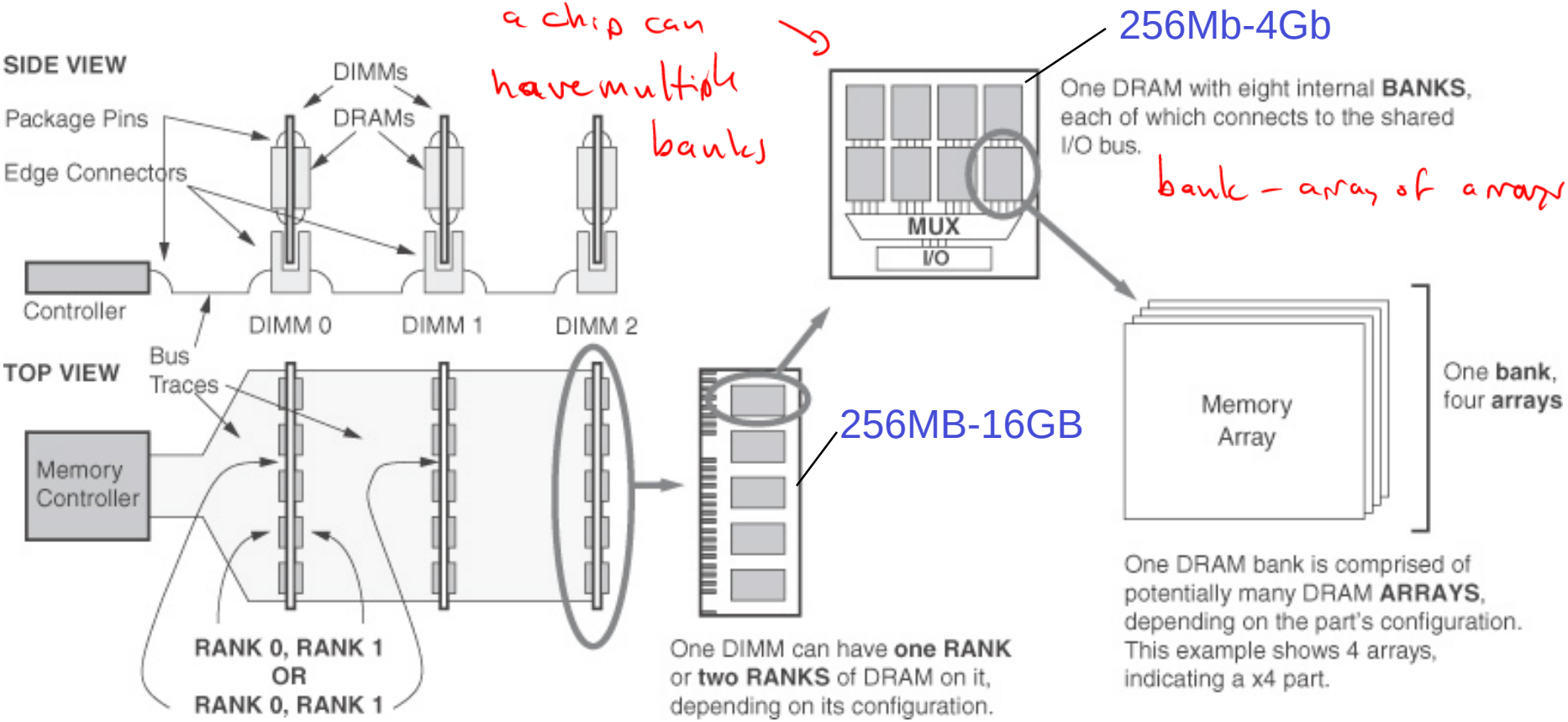
MM Using Multiple DRAMs



[10.8]

MM Using Multiple DRAMs

a chip can have multiple banks



bank – array of arrays

Array: One single matrix of DRAM cells + associated hardware.

Bank: Collection of DRAM *arrays* that are always accessed together

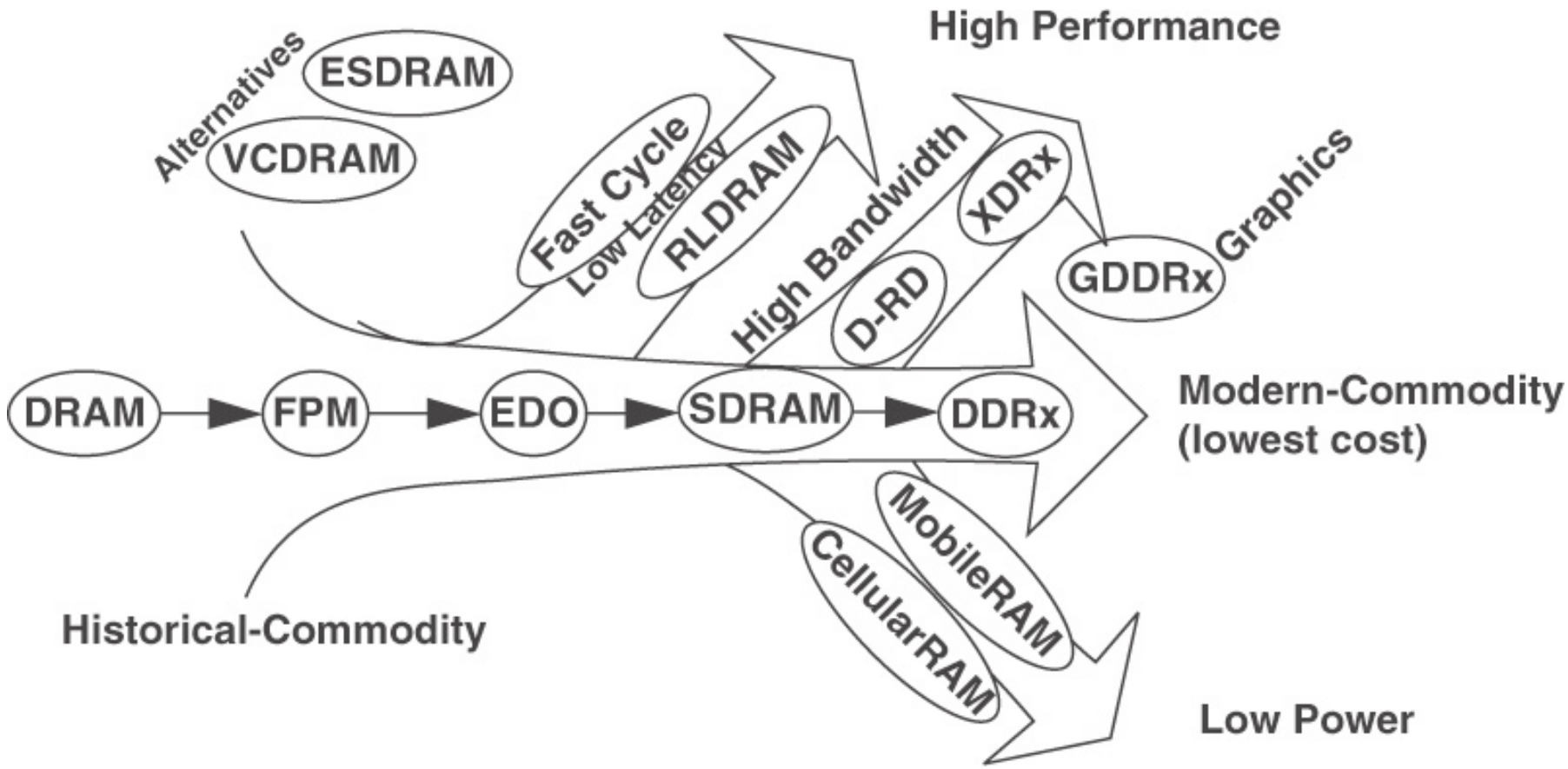
Rank: Collection of DRAM *chips* that are always accessed together

DIMM: dual in-line memory module (contains 1-4 ranks)

different chips hold different parts of data. ⇒ 0 - 15 | 16 - 31 | 0 | 1

[7.5]

The Evolution of the Modern DRAM



[12.1]

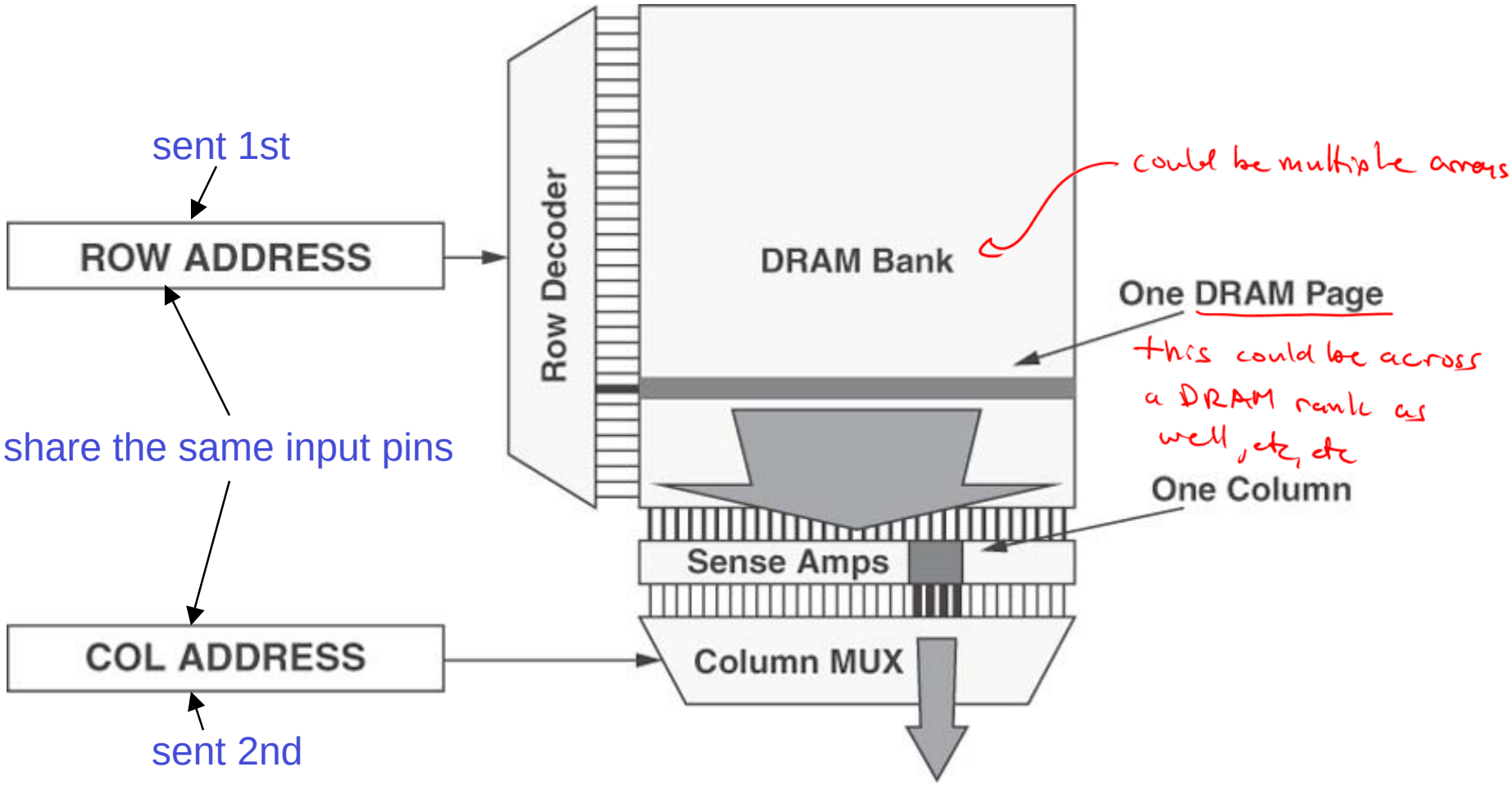
Asynchronous DRAM

- **Really should be called *unclocked* DRAM**

- **Access proceeds in a combinational logic fashion (no flip-flops)**
  → really just a combinatorial logic device.
  → difficult to use due to the memory controller being clocked
- **To save on packaging costs, address is provided in two phases using the same address inputs**
  - ***Row address* to pick the desired row or DRAM *page***
  - ***Column address* to output a subset of those bits**

Asynchronous DRAM

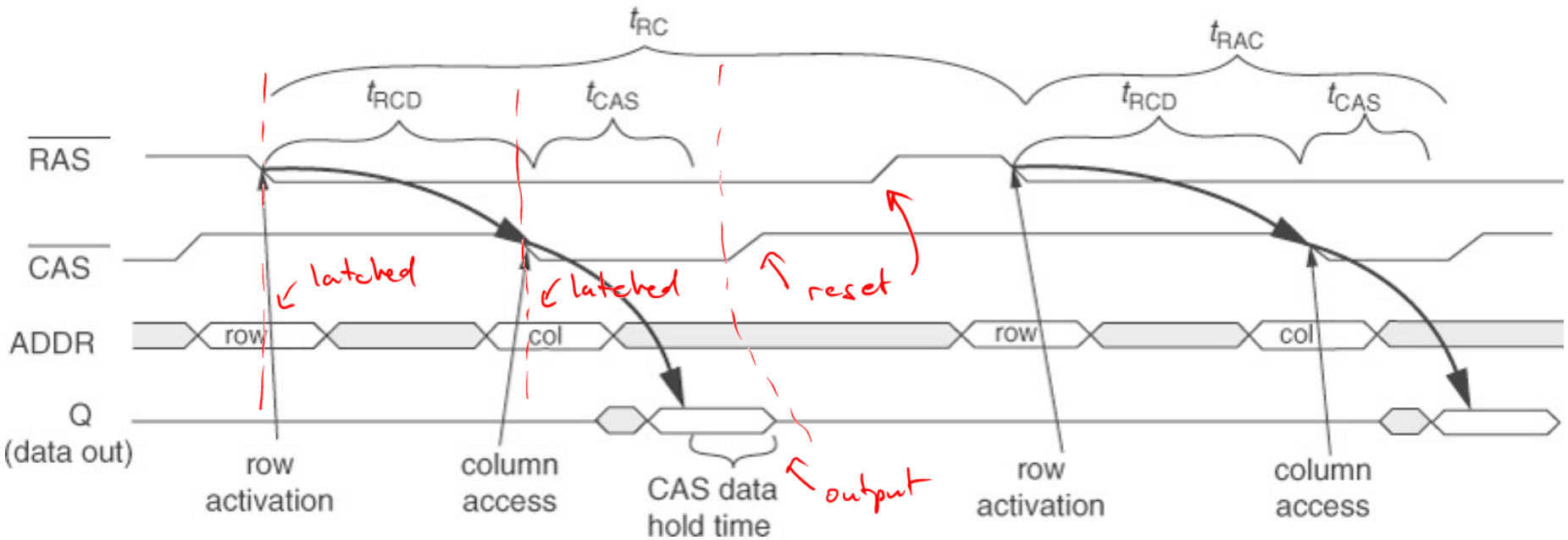
- **Row and column address phases**



[7.8]

Asynchronous DRAM

- **Read access timing**

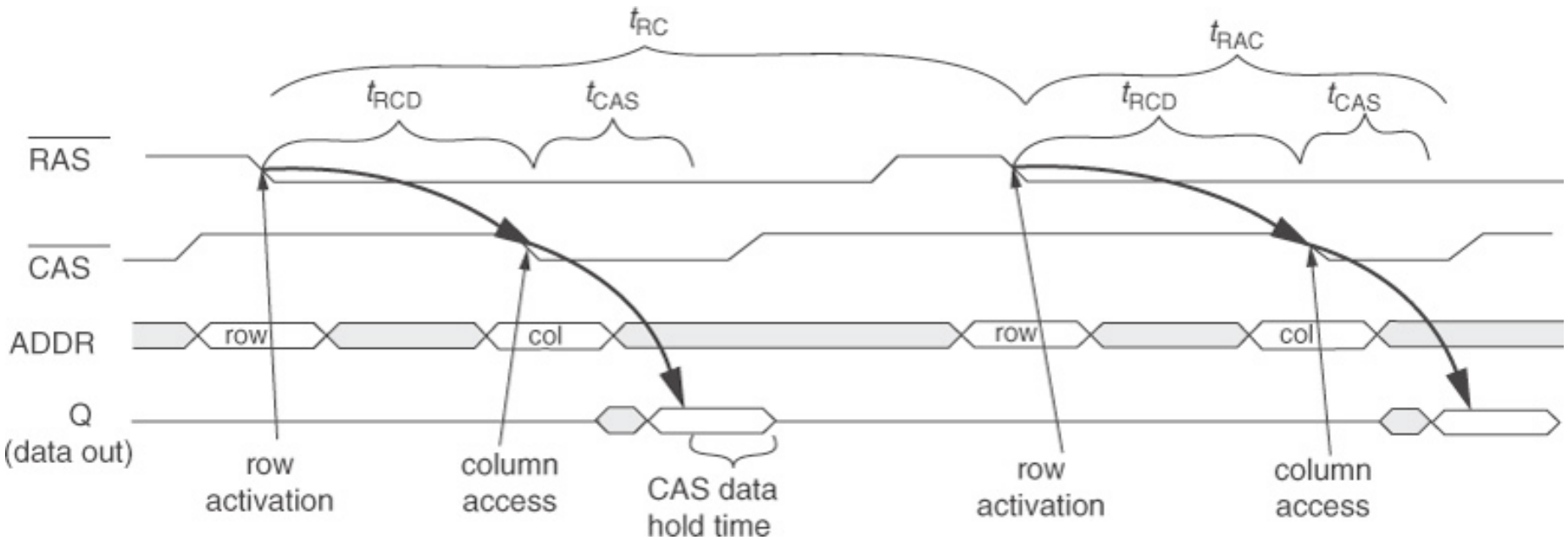


- **Address control signals (active low)**
  - **Row address strobe (RAS): latches row address and activates the selected row**
  - **Column address strobe (CAS): latches column address and serves as the output enable** for tristate driver

[12.7]

Asynchronous DRAM

- **Read access timing**



- **Major timing parameters** ~ 10's of ns
  - **RAS to CAS delay (t_{RCD}): minimum time between assertion of RAS and data available at sense amps**
  - **CAS access time (t_{CAS}): maximum time between assertion of CAS and valid data**

[12.7]

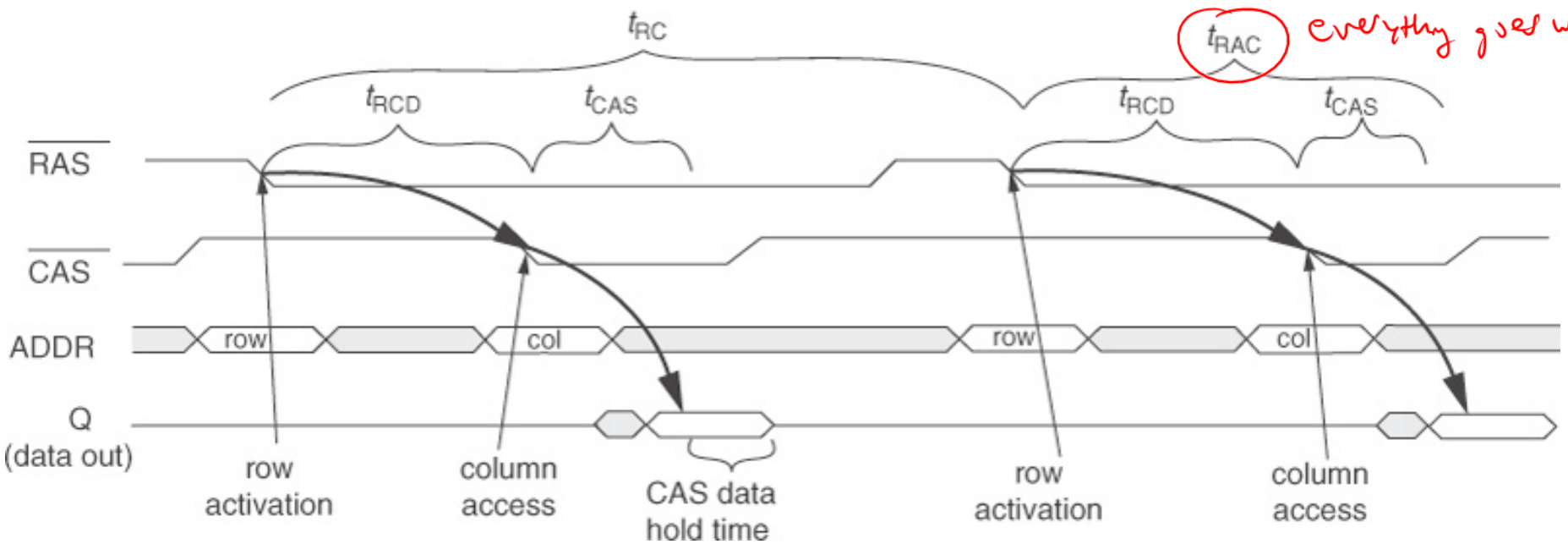
Asynchronous DRAM

Access time once RAS is asserted

$t_{RAC} = t_{RCD} + t_{CAS}$

minimum time from RAS → data if everything goes well

- **Read access timing**

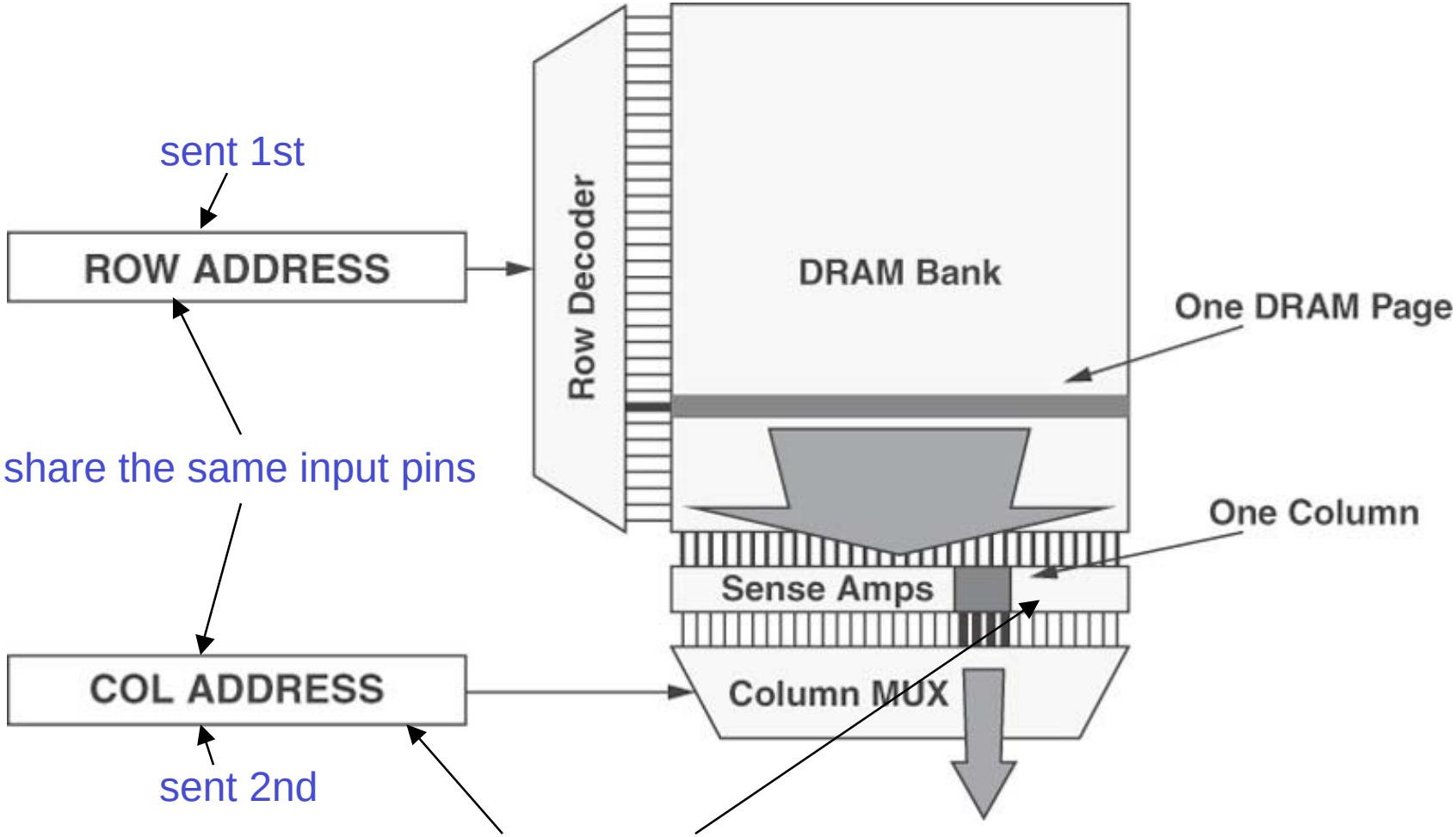


- **Major timing parameters**
  - **RAS cycle time (t_{RC}): minimum required time between two assertions of RAS (includes *precharge time*, t_{RP})**
    for bitlines
  - **CAS hold time: minimum time data remains valid after deassertion of CAS**

[12.7]

Fast Page Mode DRAM

- **Rapid access to bits in the currently *open page***



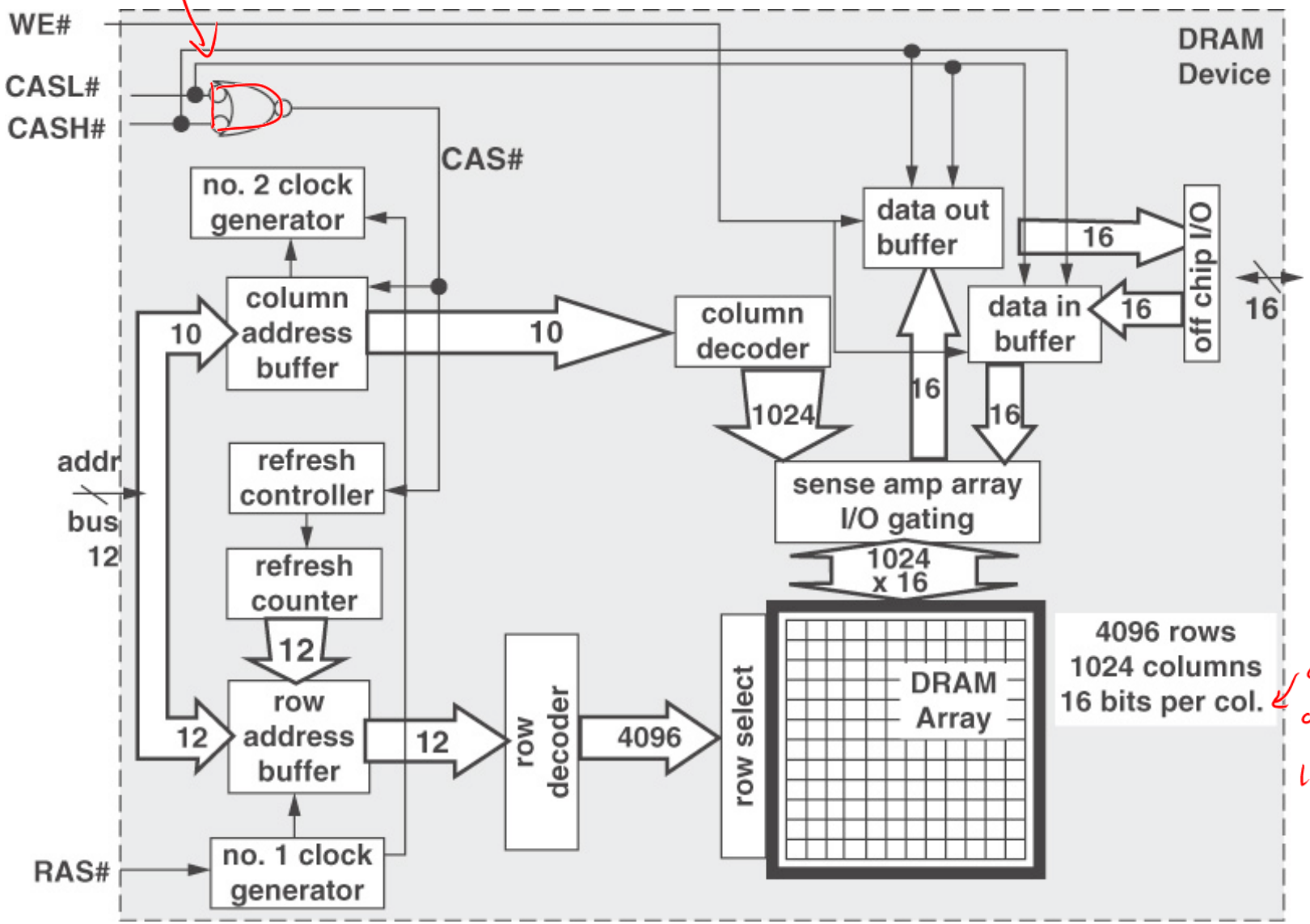
get more bits by ~~just changing~~ the column address!

also must re-assert CAS

[7.8]

they're inverted signals, so it's a selection OR

Fast Page Mode DRAM



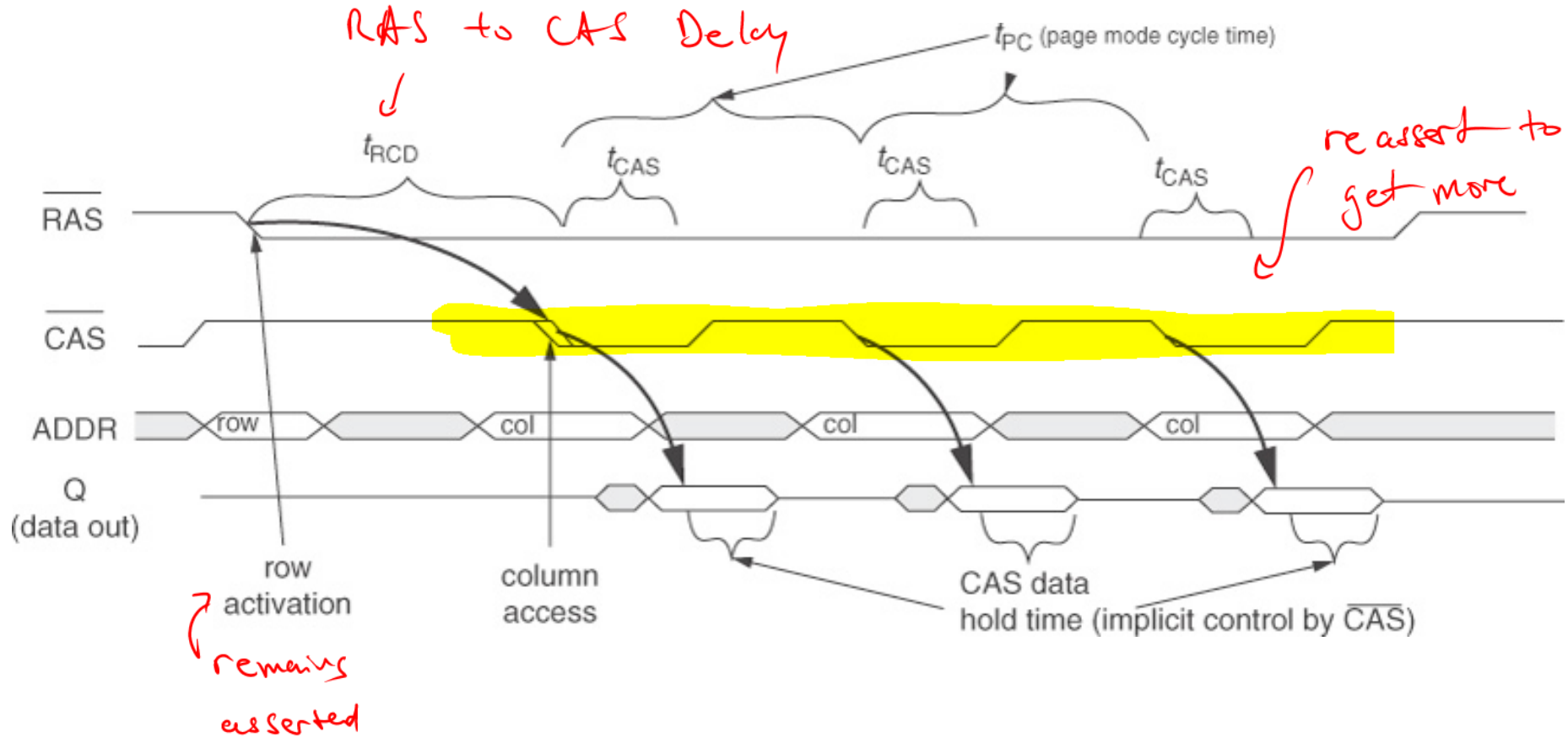
can only access w/ 16 bit granularity

4M x 16 FPM DRAM

4 million, 16 bit ⇒ 64Mb

[12.6]

FPM DRAM Read Accesses



Can use FPM to get a big cache line out of a DRAM easier,
i.e. have a 64-bit bus, want 4x 64-bit words (quadword) /access
toggle CAS 4x to get it via FPM
w/o FPM, have to do it w/ wider bus + multiple chips

[12.7]

Extended Data Out (EDO) DRAM

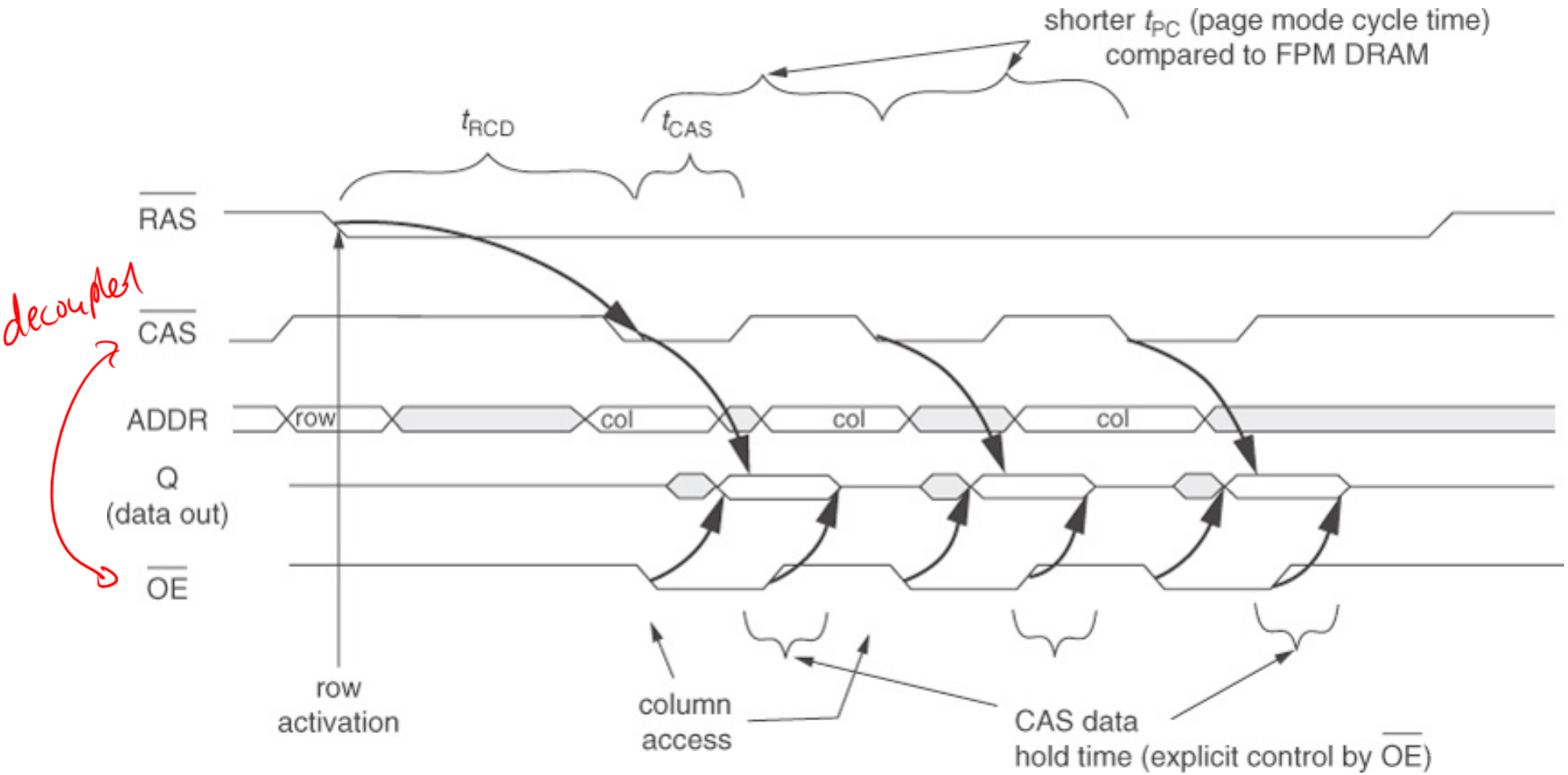
- **Output Enable (OE) signal added to control tristate input on output driver**

- **Permits CAS to be cycled quicker than FPM**

Now the CAS signal doesn't have to drive the tristate output enable. This reduces capacitance on the CAS line.

According to the book, EDO RAM adds latches on the output, holding the data valid. OE is for the latches, thus completely decoupling CAS from the output enable.

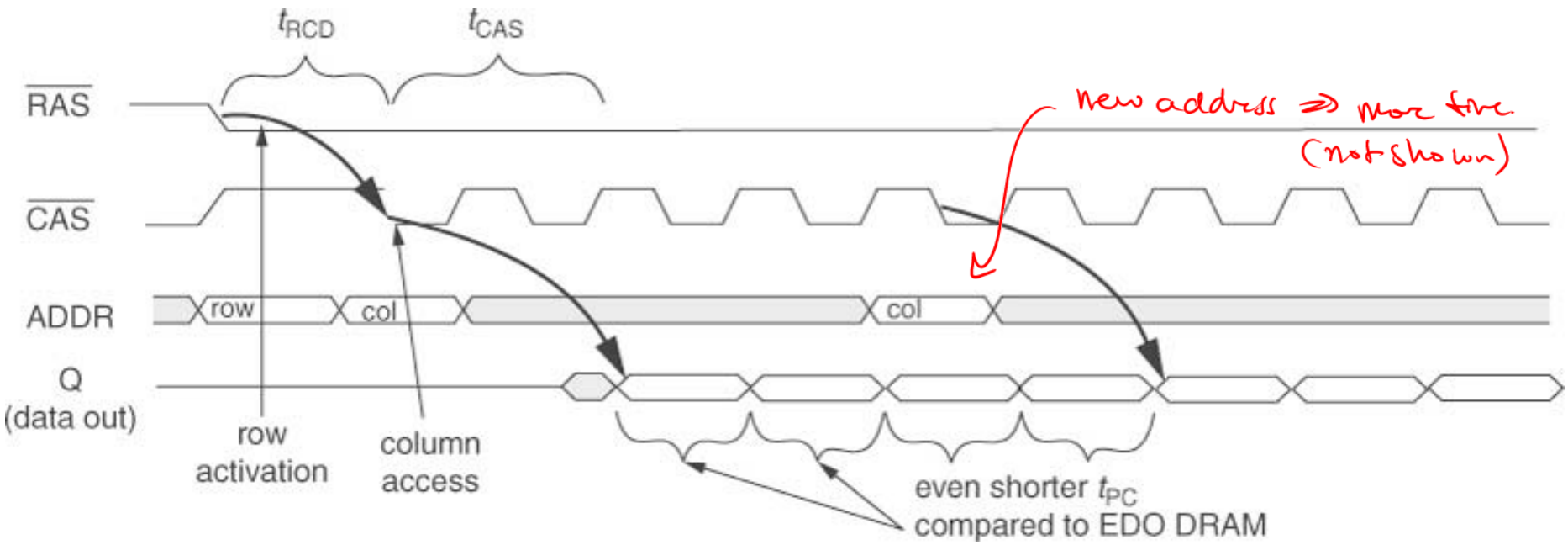
EDO DRAM Read Accesses



[12.8]

Burst EDO DRAM

- **DRAM automatically increments column address internally to burst multiple units of data**



The DRAM will automatically increment the column address when you toggle CAS. No longer have to worry about setup and hold time on the input column address

[12.9]

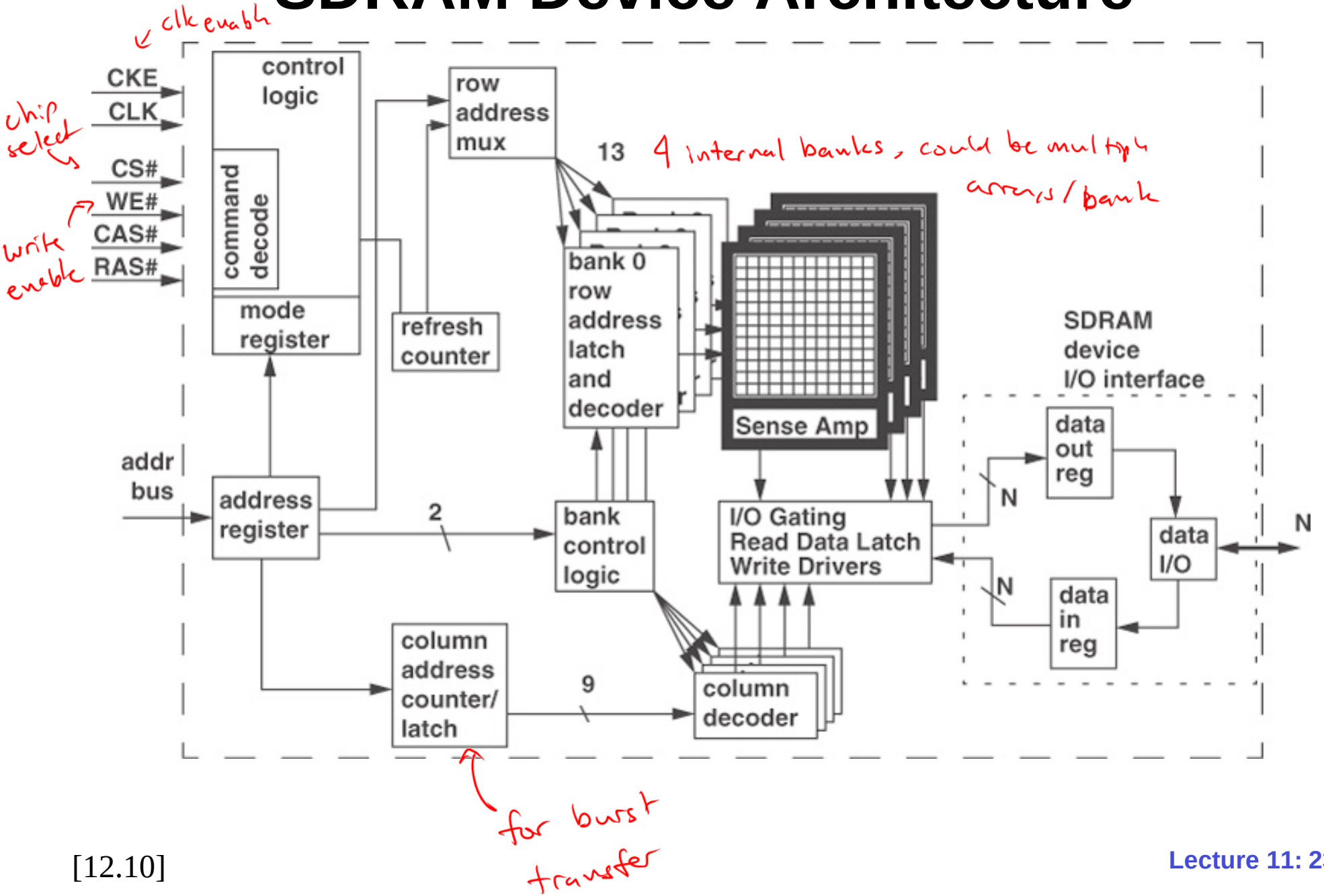
Synchronous DRAM (SDRAM)

- **Asynchronous DRAMs had limited bandwidth and concurrency**
  - **Limited overlap of address and data phases of consecutive accesses**
  - **Single bank of arrays**

- **SDRAMs greatly increase concurrency and bandwidth by**
  - **Registering the inputs and outputs** → add flip flops
  - **Incorporating multiple independent banks** → more banks
  - **Increasing the amount of on-chip control intelligence**

    → control now distributed to SDRAM

SDRAM Device Architecture



[12.10]

SDRAM Commands

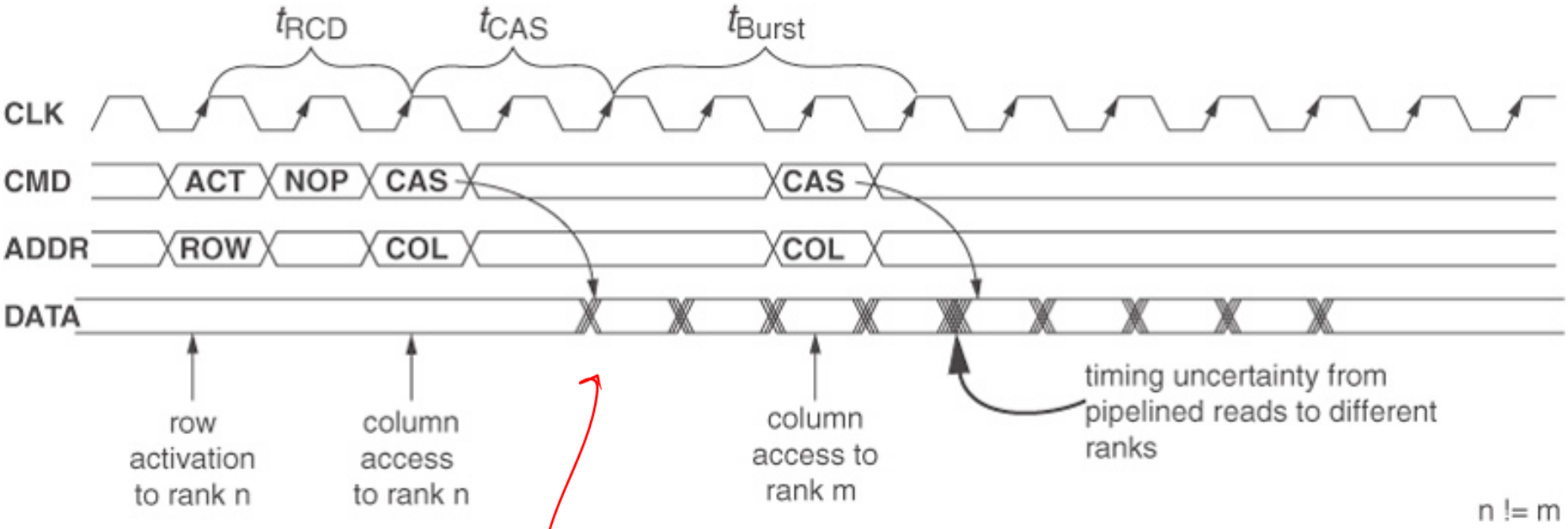
must be low to do anything

today used to select ranks

Name (Function)	CS#	RAS#	CAS#	WE#
COMMAND INHIBIT (NOP)	H	X	X	X
NO OPERATION (NOP)	L	H	H	H
ACTIVE (Select bank and activate row)	L	L	H	H
READ (Select bank and column, and start READ burst)	L	H	L	H
WRITE (Select bank and column, and start WRITE burst)	L	H	L	L
BURST TERMINATE	L	H	H	L
PRECHARGE (Deactivate row in bank or banks)	L	L	H	L
AUTO REFRESH or SELF REFRESH (Enter self refresh mode)	L	L	L	H
LOAD MODE REGISTER ← controls burst length, other timing things	L	L	L	L

[Micron00]

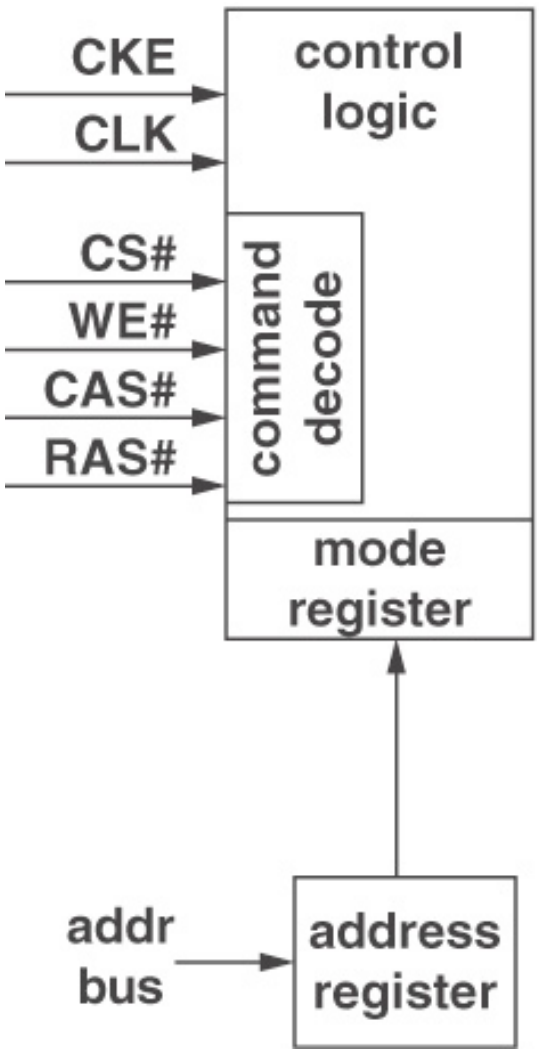
SDRAM Access Protocol

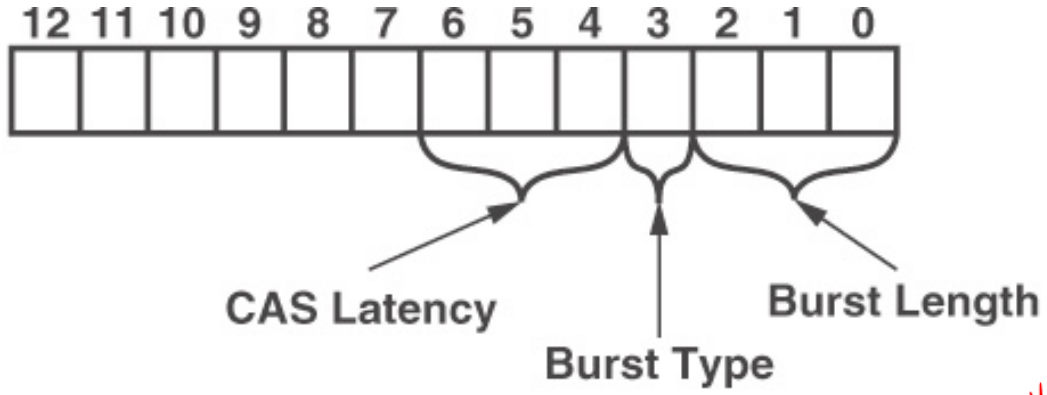


poorly shown. in register. clocked out on rising edge of clock

[12.12]

Programmable Mode Register



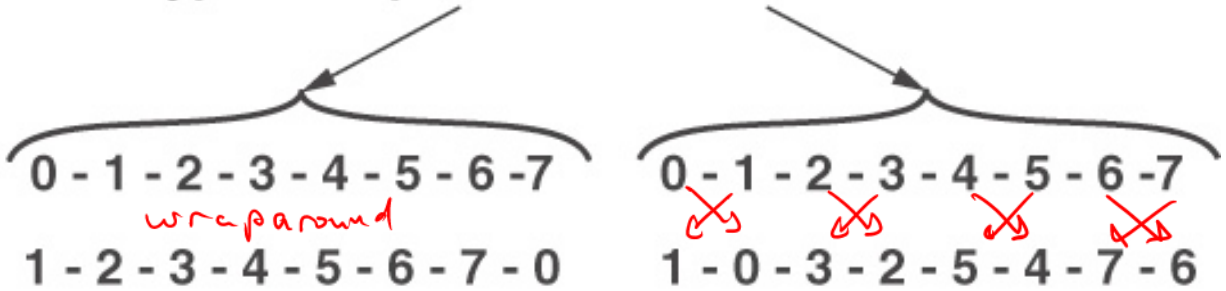


allows for slowing down to match older parts

Burst Length = 1, 2, 4, 8, or Page mode

CAS Latency = 2, 3 (4, 5, etc. in special versions)

Burst Type = Sequential or Interleaved

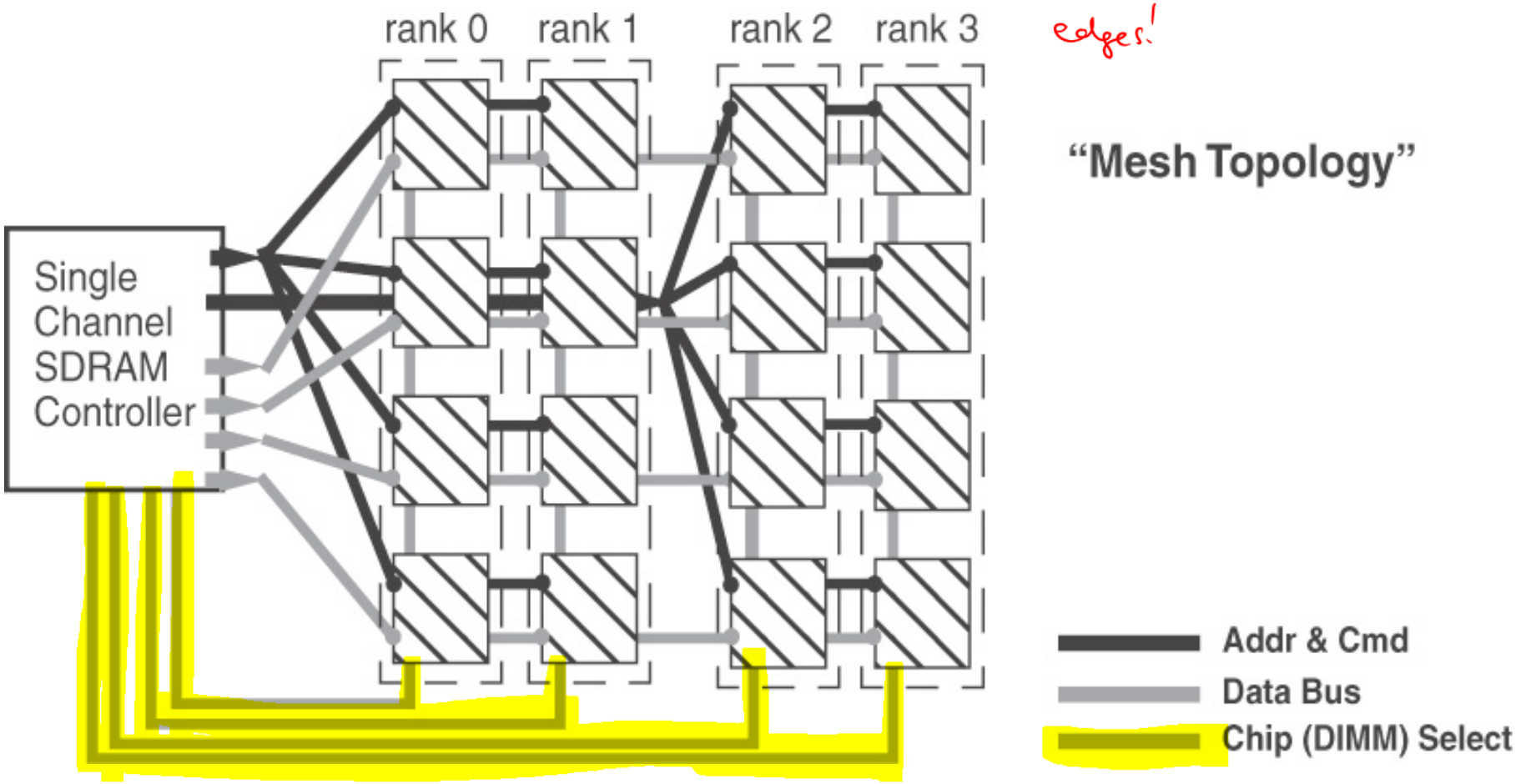


[12.11]

Double Data Rate (DDR) SDRAM

must go to all DRAMS. lots of capacitance

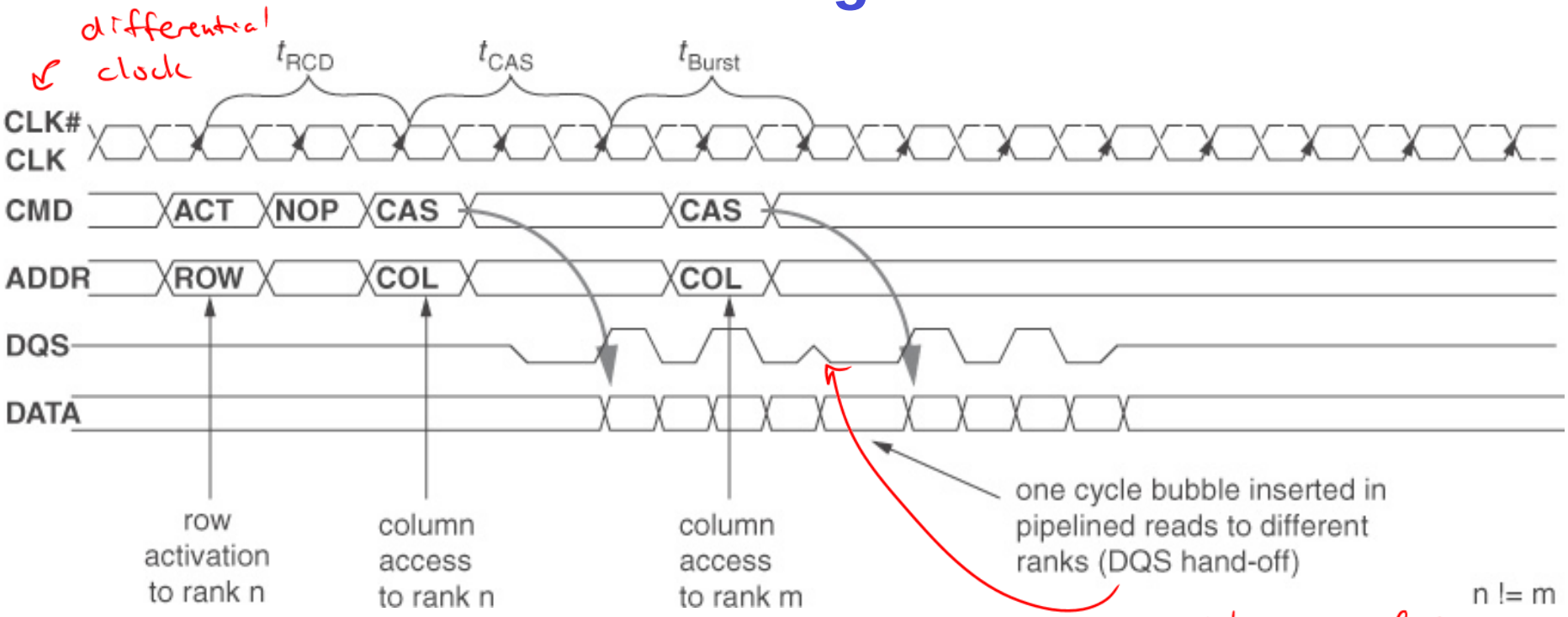
- **Address and Command buses more heavily loaded than data bus** → so run the data bus on both clock edges!



[10.13]

Double Data Rate (DDR) SDRAM

- **Run data bus on both edges of the clock**



need to pass off DQS to prevent fighting drivers

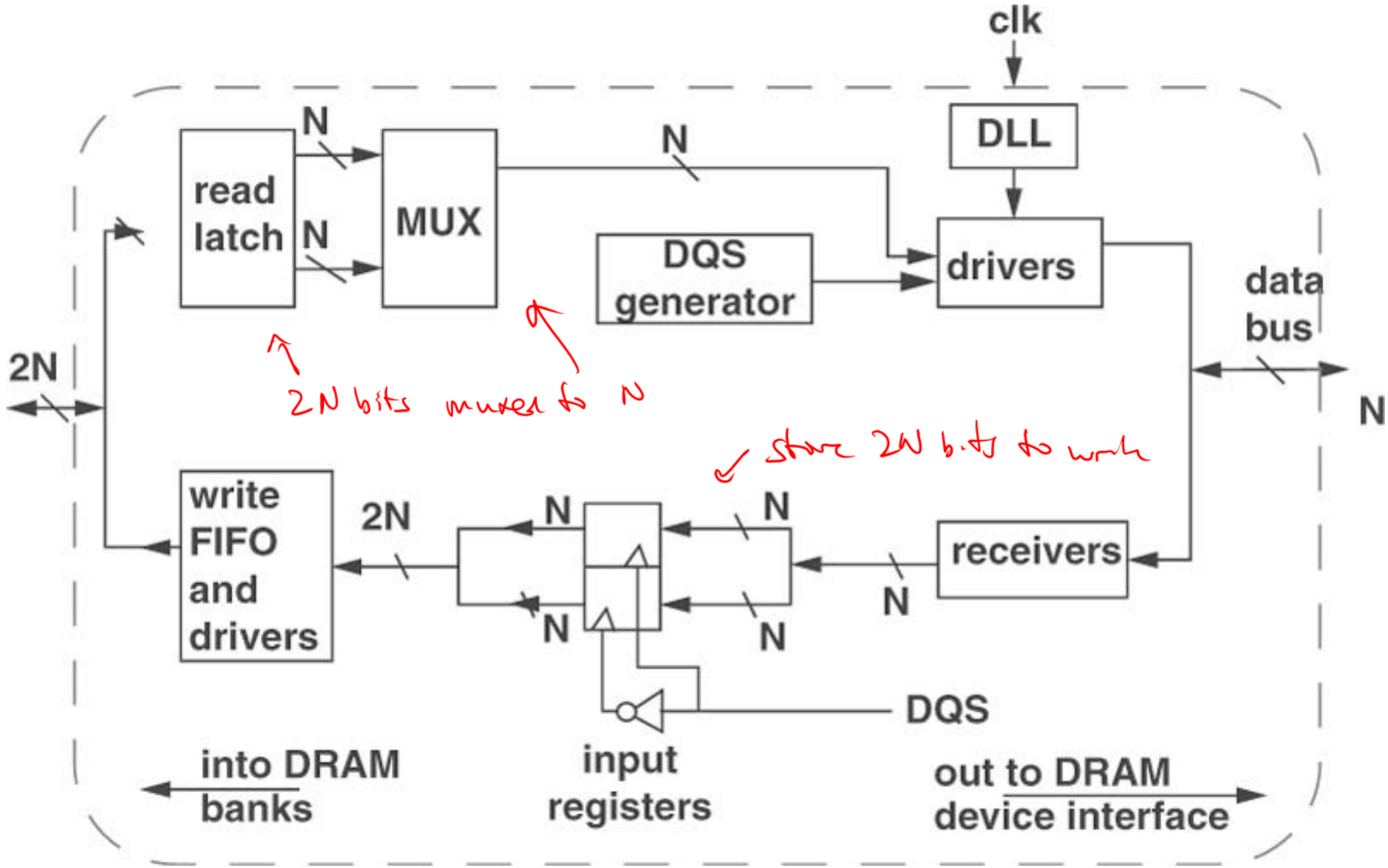
DQS: (shared) data strobe signal controlled by the source of the data

Source-synchronous strobe signal

→ provides a clock that is synced to the data

[12.16]

DDR SDRAM I/O Architecture



2-bit prefetch architecture

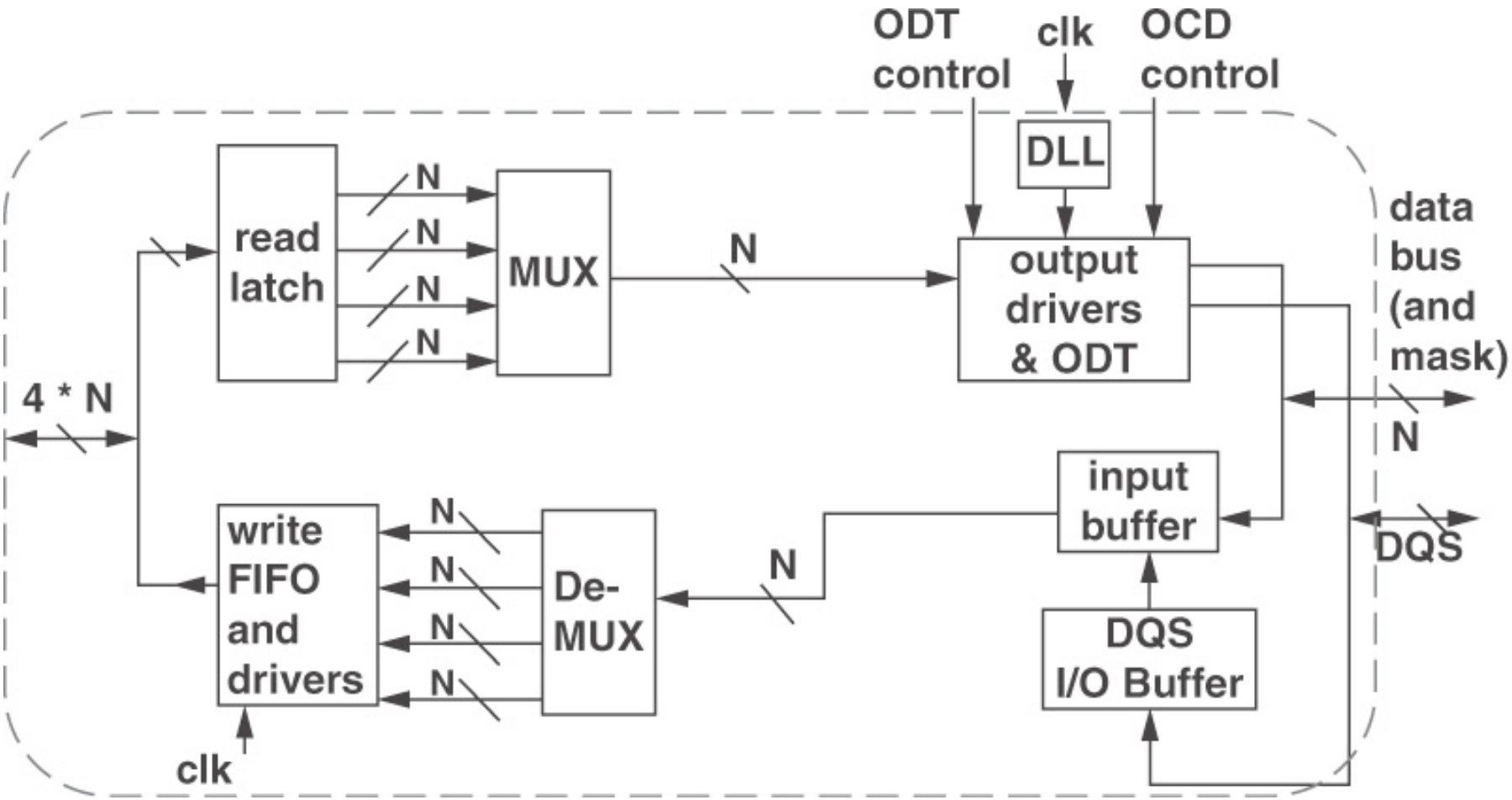
[12.17]

if we have n-bit wide external bus,
internally we have a 2n-bit bus w/ mux

DDR2 and DDR3

- Increased prefetch length
  - 4 in DDR2 and 8 in DDR3
- Some interface signaling changes
  - E.g., differential DQS
- Some additional commands

DDR2 I/O Architecture



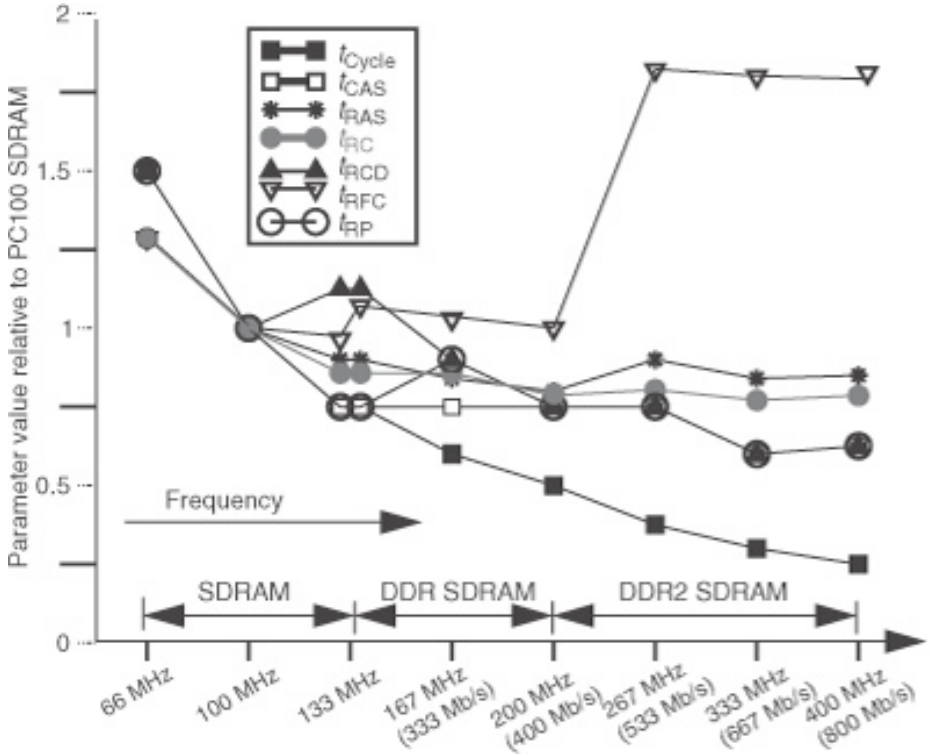
[12.19]

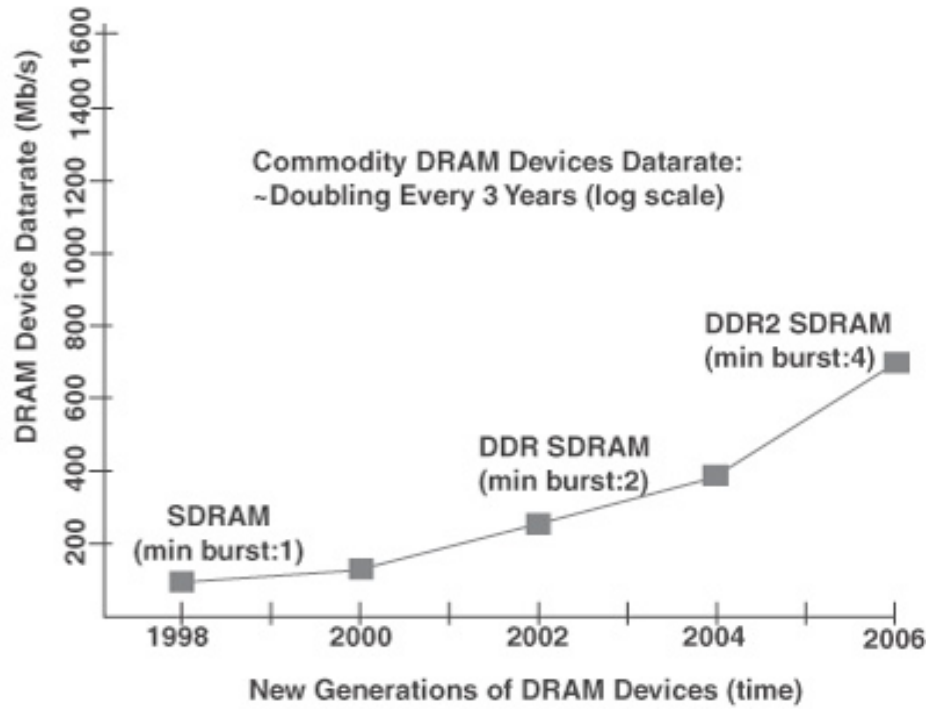
SDRAM and DDRx Comparison

Variables	SDRAM	DDR1	DDR2	DDR3
Clock	100/133/166 MHz	100/133/166/200 MHz	200/266/333/400 MHz	400/533/667/800 MHz
Transfer Data Rate	100/133/166 Mbps	200/266/333/400 Mbps	400/533/667/800 Mbps	800/1066/1333/1600 Mbps
I/O width	x16/x32	x4/x8/x16/x32	x4/x8/x16	x4/x8/x16/x32
Prefetch bit width	1 bit	2 bits	4 bits	8 bits
Clock Input	Single Clock	Differential Clock	Differential Clock	Differential Clock
Burst Length	1, 2, 4, 8, full page	2, 4, 8	4, 8	8, 4 (Burst chop)
Data Strobe	Unsupported	Single data strobe	Differential data strobe	Differential data strobe
Supply Voltage	3.3V/2.5V	2.5V	1.8V	1.5V
Interface	LVTTL	SSTL_2	SSTL_1.8	SSTL_1.5
$\overline{\text{CAS}}$ latency (CL)	2, 3 clock	2, 2.5, 3 clock	3, 4, 5, clock	5, 6, 7, 8, 9, 10 clock

[Choudhary08]

DRAM Scaling Trends





Random row access cycle time increased 7% per year
Transfer data rate doubled every three years

[12.22]

Next Time

More on DRAMs