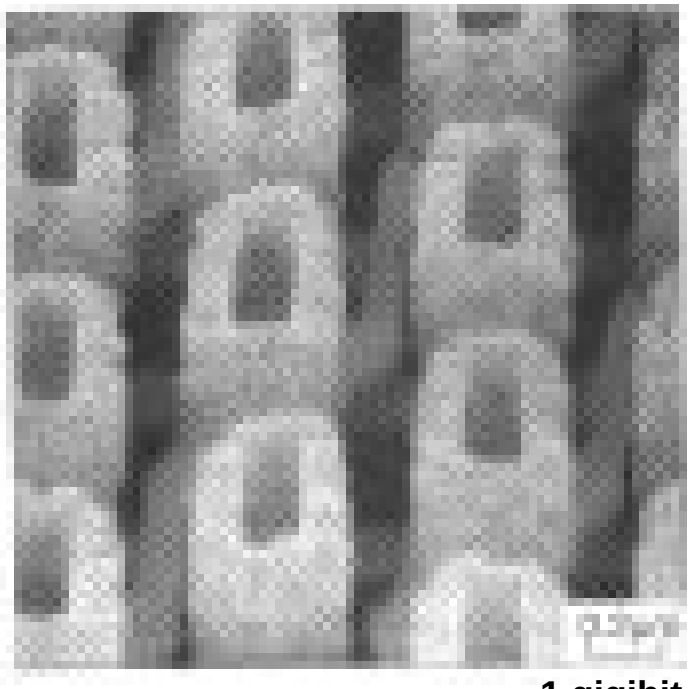
Northeastern University Computer Architecture Research (NUCAR)

Basic System Level DRAM Design

(asynchronous timing interface)



1 gigibit

student **David Morano**
advisor **Professor David Kaeli**

NUCAR seminar 99/06/11

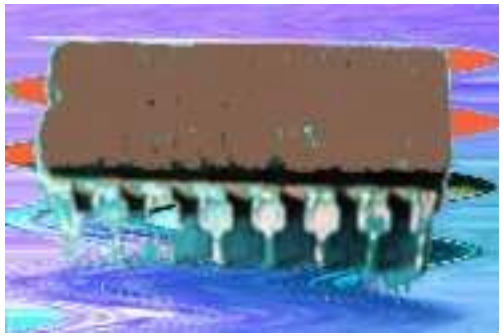
Outline

- background
  - history
  - technology comparison
- classic DRAM control
  - basic circuit
  - basic timing (asynchronous DRAM interface)
  - refresh
- DRAM design optimizations
  - internal refresh counters
  - within-page access
  - delaying RAS precharge on purpose
- summary / future

Background -- History

- DRAM idea invented at IBM in 1968 by Robert H. Dennard
- research continued in several companies immediately
- in 1970 Intel (prompted by Honeywell) invents the world's first DRAM chip (*i1102* reported in ISSCC 1970 -- 1k bit PMOS)
  - first production IC was *i1103* -- 1k bits in PMOS (October 1970)
  - John Reed was principal engineer
  - first commercial computer with them was the HP 9800
  - by 1972 it became the world's best selling IC in history
- introduction of an all 5 Volt design
- MOSTEK 16k bit (circa 1980)
- nibble-mode and page-mode variants
- internal refresh counters
- video variants
- AT&T Bell Laboratories introduces 1M bit
- synchronous & burst variants

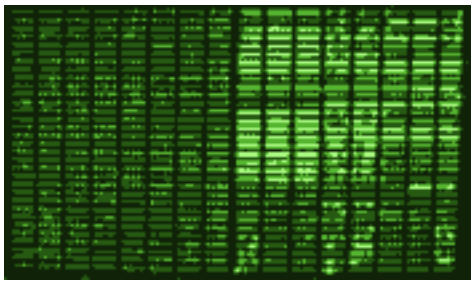
i1102



Background -- Technology Comparison

- **DRAMs are the fifth major computer system memory technology**
  - **vacuum tube**
  - **sonic delay line**
  - **Williams tube**
  - **magnetic core (about 1955 - 1978)**
  - **DRAM (about 1979 - present)**

Williams tube

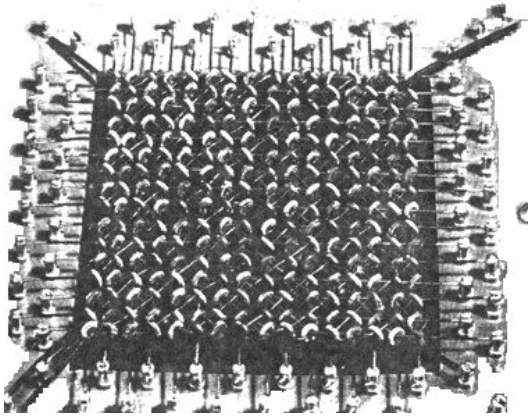


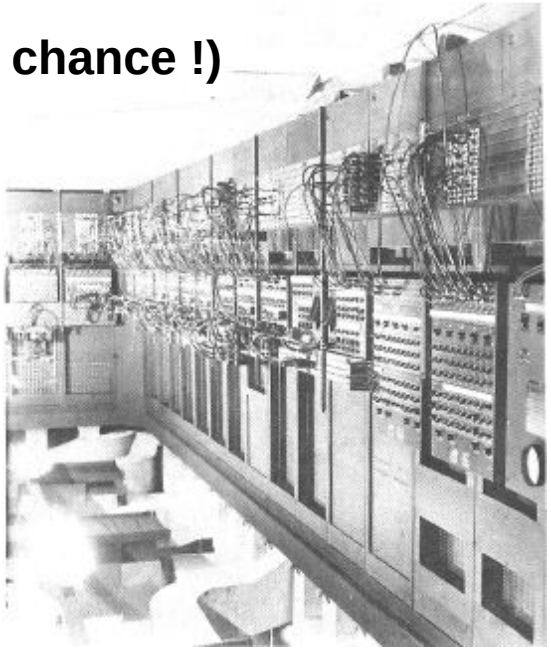
- **SRAM never really got a clear chance to be a dominant main memory technology (except for super computers)**
- **SRAMs usually use six transistors per bit cell (can be four per bit cell), DRAMs use one transistor per bit cell**
- **DRAMs are usually at least 4 times larger in bit storage than SRAMs of the same technology generation**
- **DRAMs historically have used multiplexed addresses for access**

Background -- Technology Comparison (cont)

- **SRAMs (and ROMs) can be powered down, DRAMs are always “up”**
- **DRAMs have numerous timing parameters (50-60 parameters typically)**
- **SRAMs can be left alone and they retain their data, DRAMs need constant attention (refresh !)**
- **DRAMs are slow (and increasingly slower) compared to SRAMs of the same size**
- **DRAMs tend to need more bypass than SRAMs (sensitive analog comparison circuitry inside)**
- **magnetic bubble ? (never had a chance !)**

core

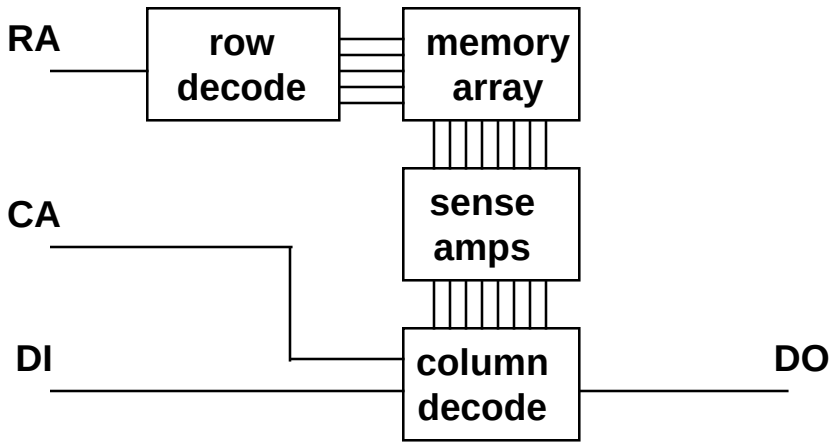




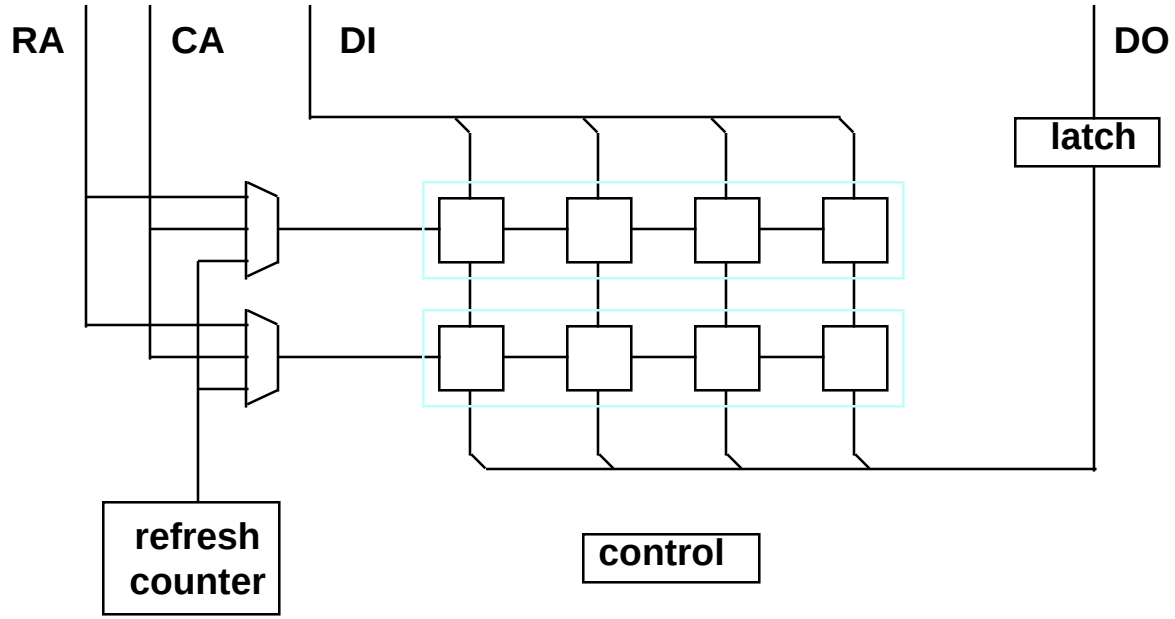
Eniac

vacuum tubes !!

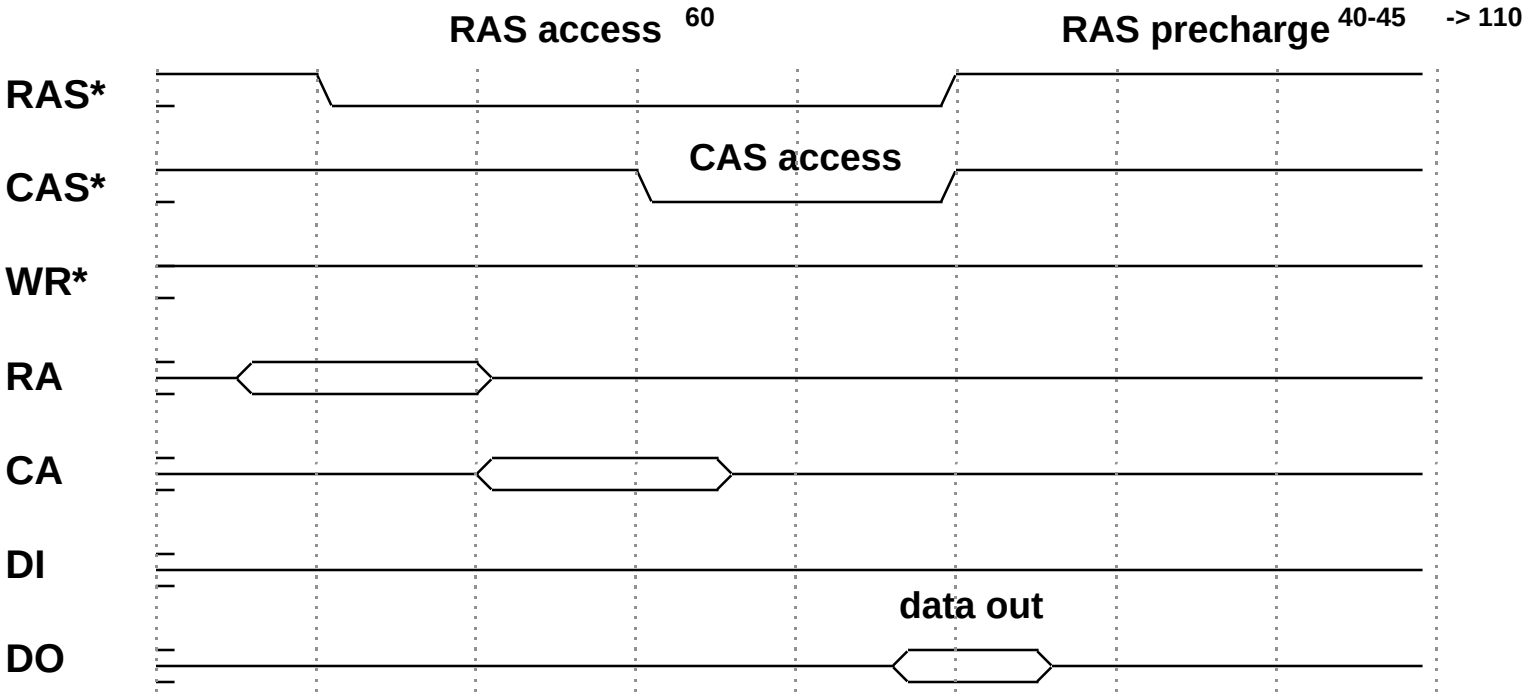
Basic DRAM Integrated Circuit



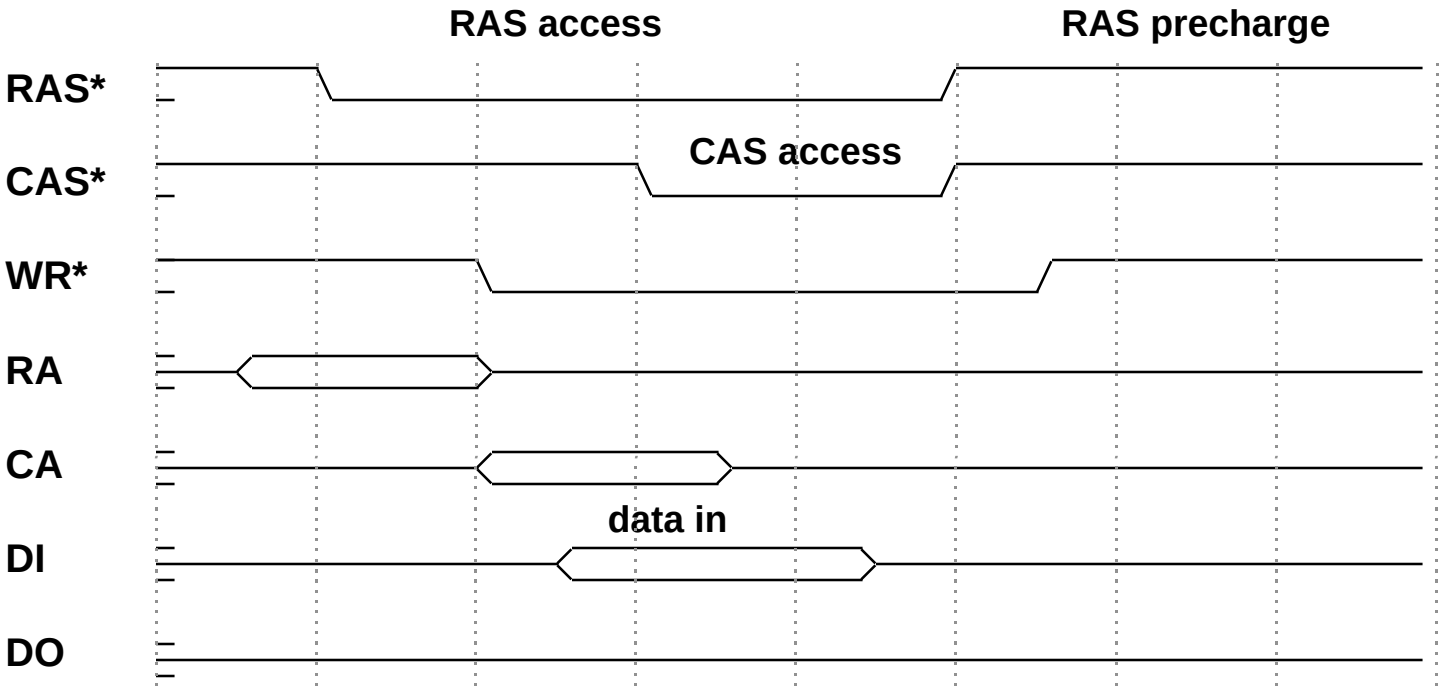
Basic System DRAM Circuit



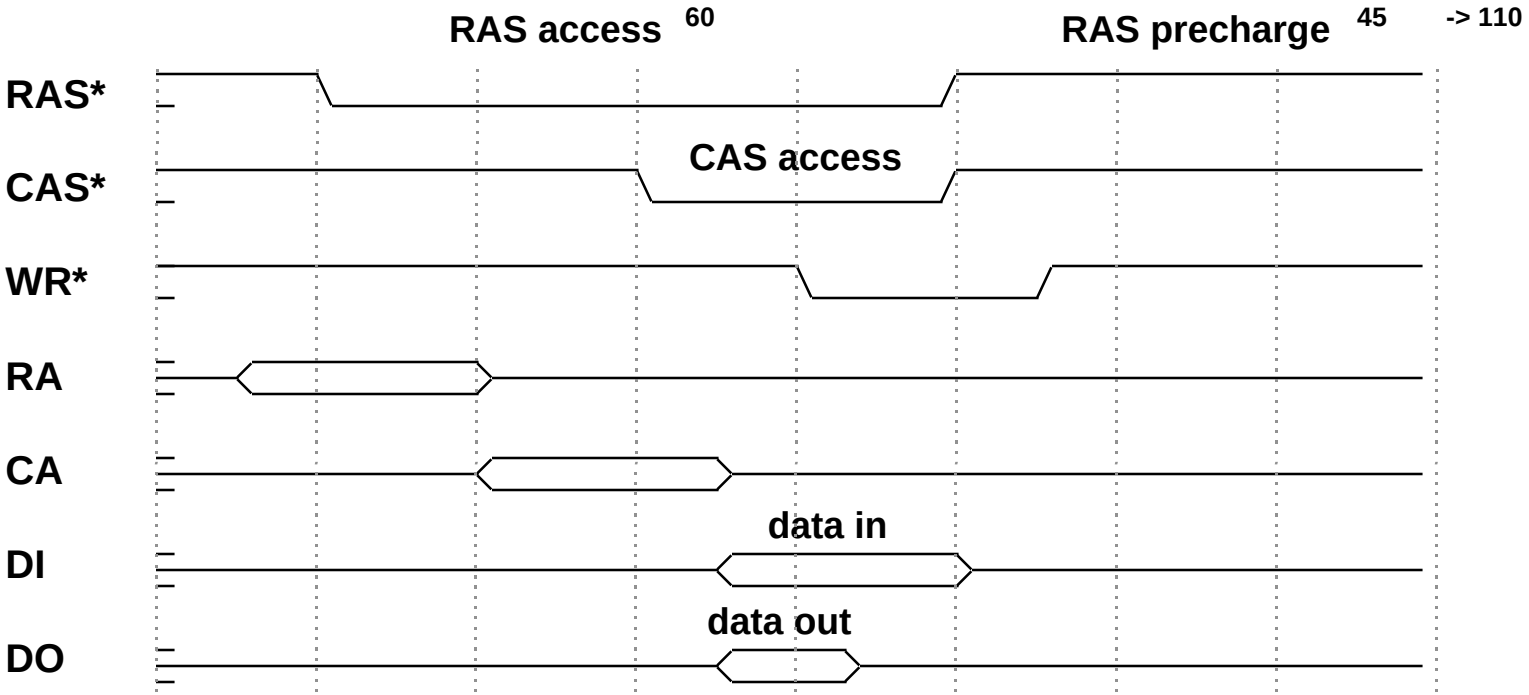
Basic Read


RAS access 60
RAS precharge 40-45 -> 110
RAS*
CAS*
CAS access
WR*
RA
CA
DI
data out
DO

Basic Write (Early Write)

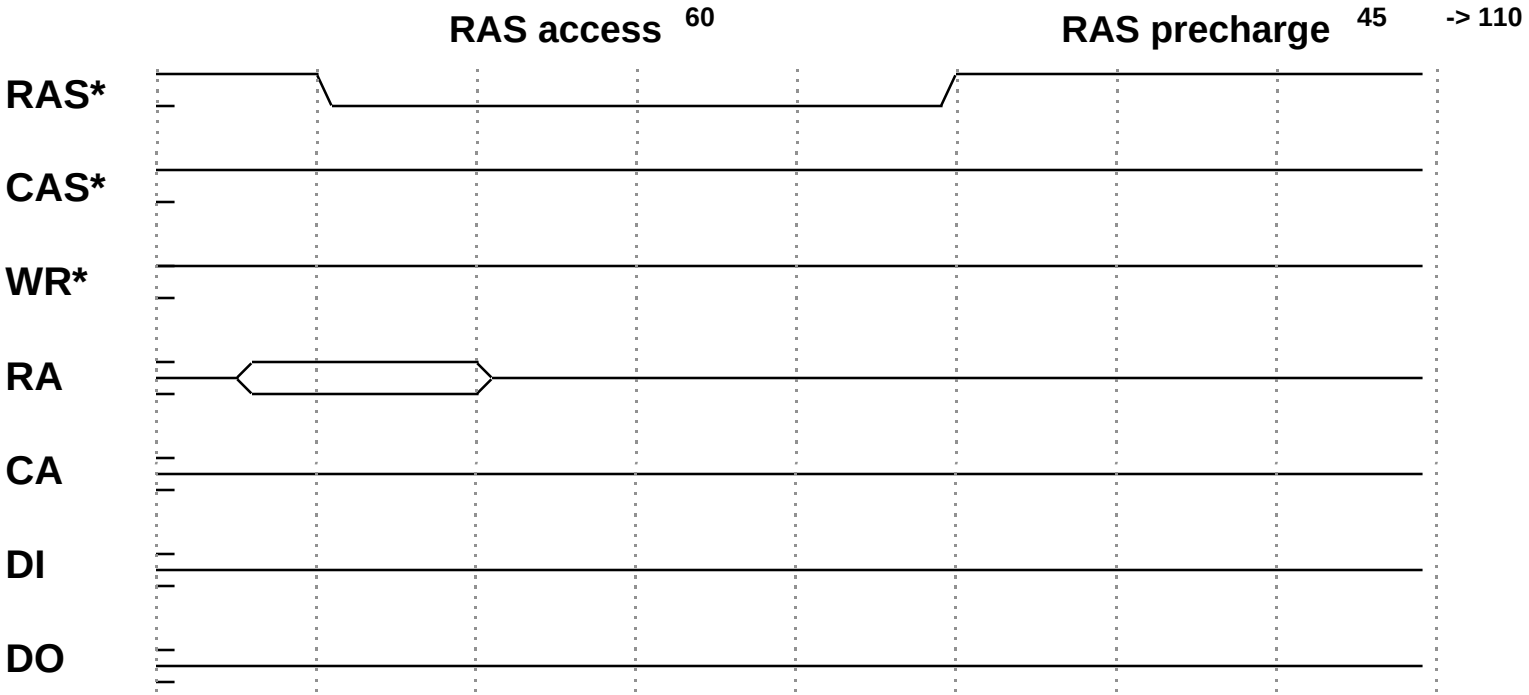


Basic Read-Write (Late Write)



- read-write operations for ECC memory
- implement a higher level interlock

Basic Refresh Timing

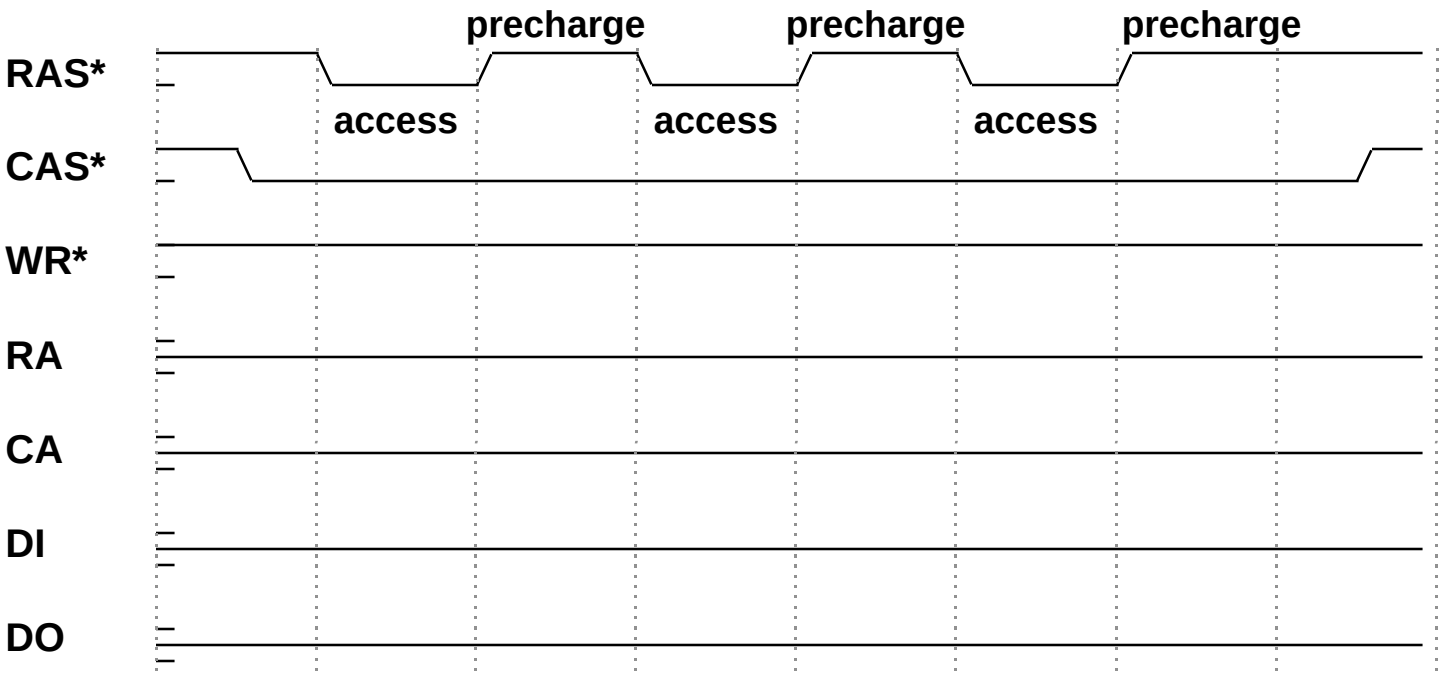


for memory size M, need approximately $(((\log_2 M) / 2) +/- 1)$ reads every refresh interval

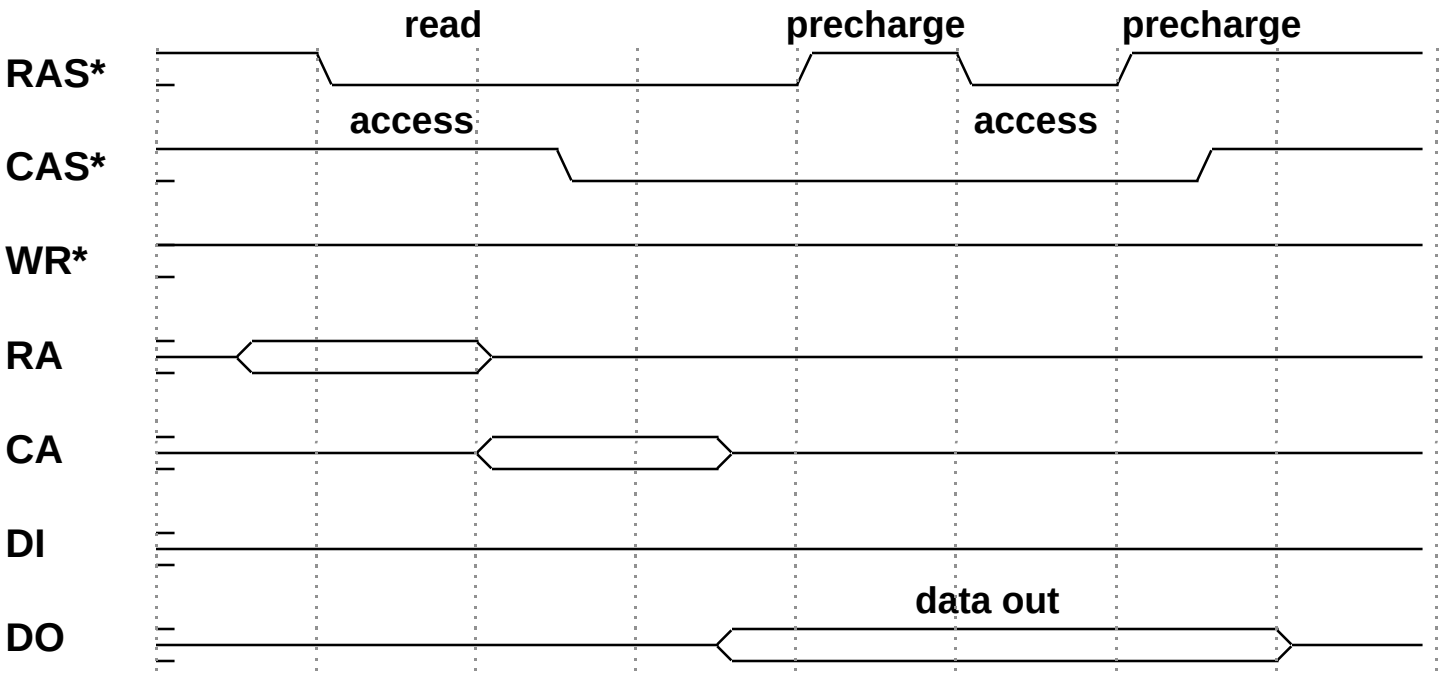
Basic Refresh Strategies

- refresh all rows in a single burst during each refresh period
- refresh one row at a time
- software does the refreshing
- refresh a few rows in a burst periodically
- try to hide the refreshes when the higher level cache doesn't need access
- let something like video hardware do it for you
- DRAM controller
- use pseudo-static RAMs

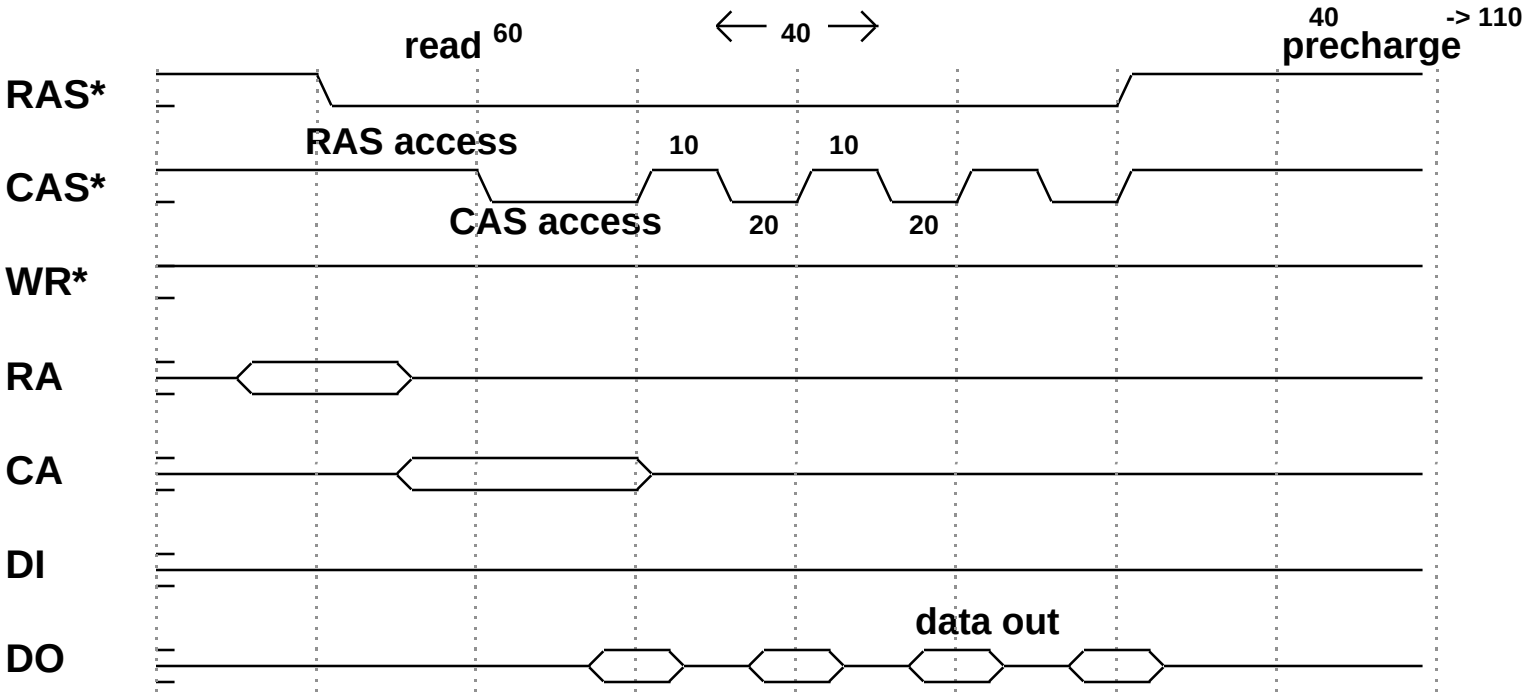
CAS Before RAS Refresh (internal refresh counters)



Hidden Refresh (From Read)

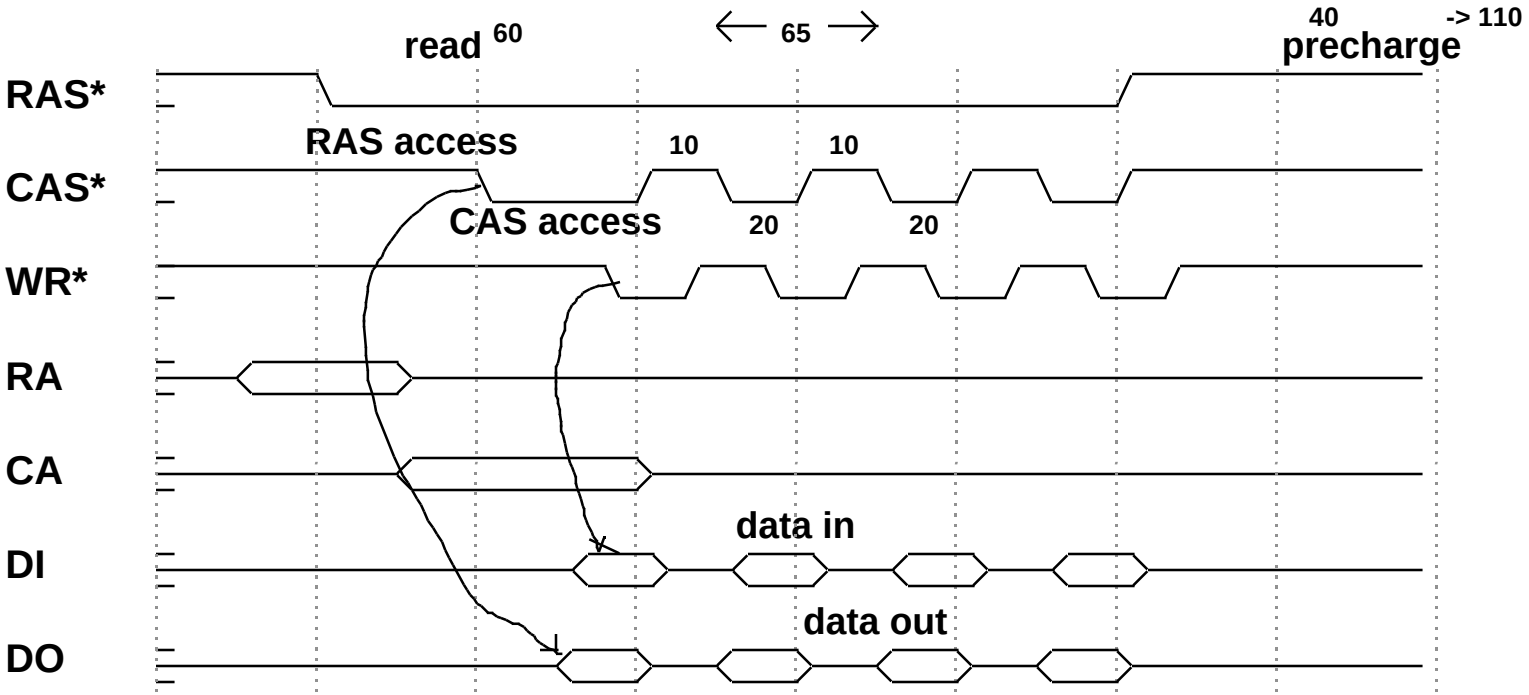


Nibble Mode Read



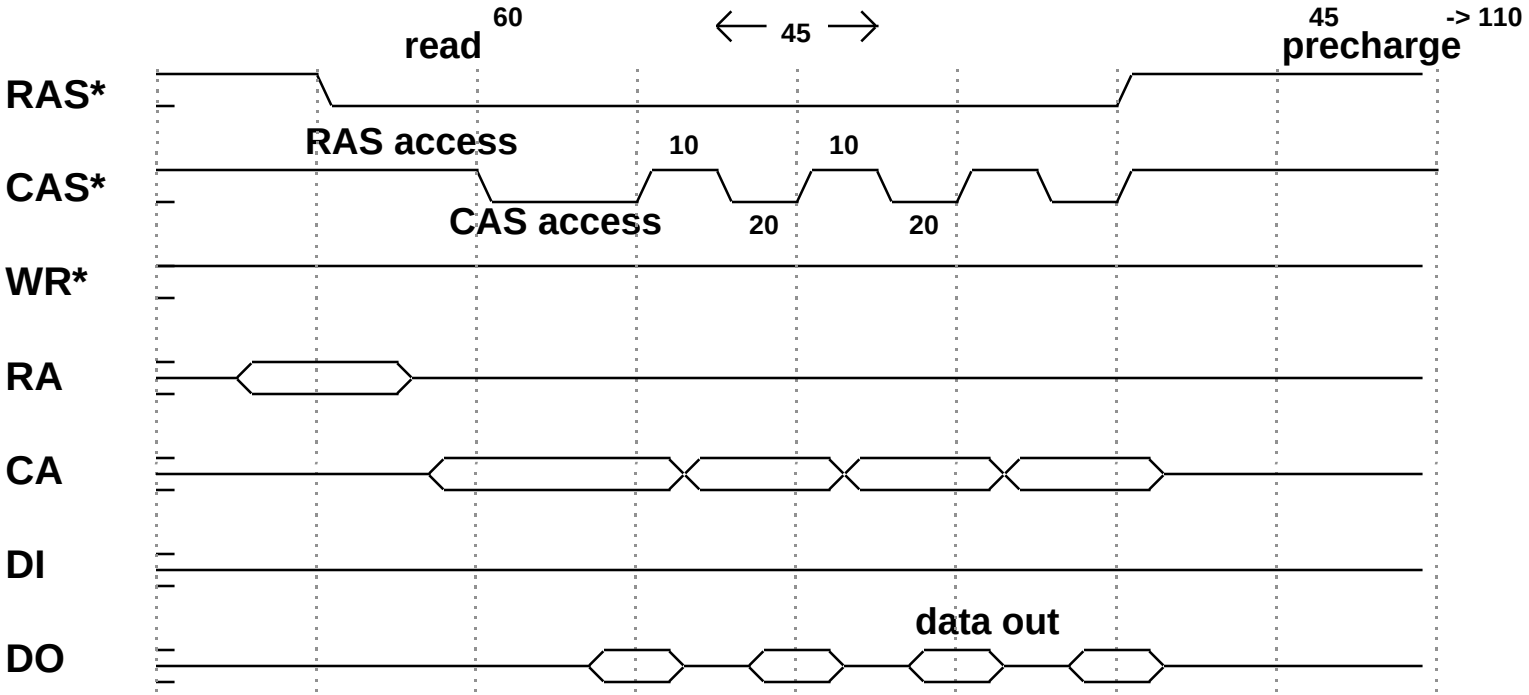
- sort of like four DRAM banks in one (internal address counter)
- great for filling cache lines
- sequential modulo access only
- intel (and other) processors have not always been cooperative !

Nibble Mode Read-Write

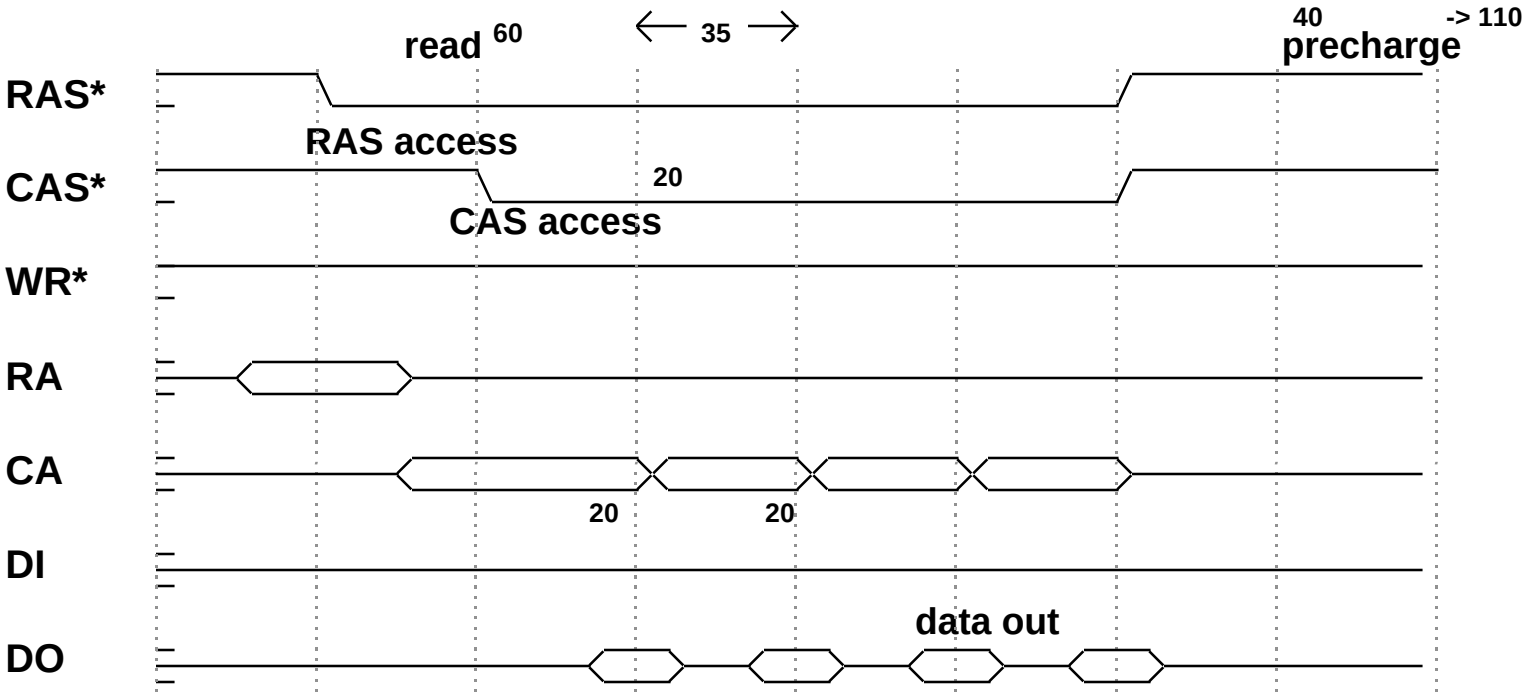


- **ECC protected memory must do read-writes for all partial writes and all reads that have correctable errors**

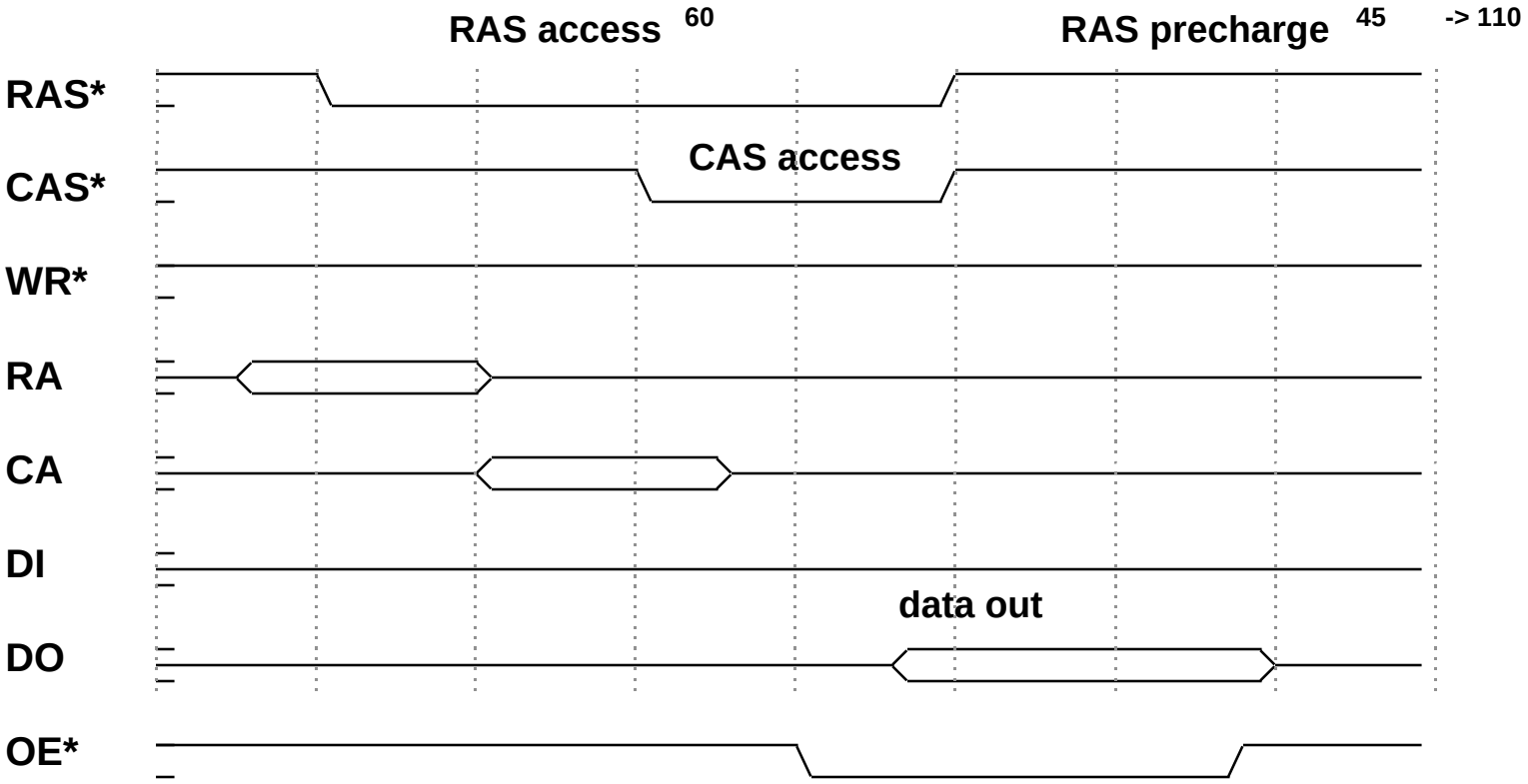
Page Mode Read



Static Column Read



Extended Data Out



+ employs a latch on the output so that data-out can overlap with precharge
+ precharge does not have to be held off while capturing output data
+ improves cycle time on accesses to different pages

overlap TLB lookup w/ RAS access

historical

- **put the page offset address bits on DRAM row-addresses**
- **put the physical page frame address bits on DRAM column-addresses**
- **unfortunately, we must start accesses on all DRAM banks (even though only one will be used) -- uses extra power**
- **this defeats the benefits of using an open DRAM "page" as a fast access cache since we are defeating the spatial locality of the lower address bits within the virtual memory page**
- **no longer very practical since there are now many more virtual address bits specifying a page frame than there are column address bits to the DRAMs**
- **no longer necessary since the TLB access is now very fast while the DRAM access is still very slow**

Delaying RAS Precharge

- use the current “open” page that is on the sense amplifiers !
- seen already with block transfers
- what about more random non-block transfers ?
  - use DRAM row as a page cache
  - put high processor addresses on DRAM row and low processor addresses on DRAM column
  - wait to see where the next access is addressed to
  - must time-out on maximum RAS “on” time
  - close the page after the last access to the page (if known)
    - » last address in the page (like if they have been sequential)
    - » last access to this page because another page is likely needed next (like if the processor is changing state of some kind)
  - time-out on waiting for the next access

Summary + Future

- bit density has increased about 1 million fold since introduction
- everything else about DRAMs hasn't changed much in 30 years !
- many variants are available (video)
- they are slow and all reports indicate they will remain slow
- there is no good replacement for them
- synchronous DRAMs
- special Burst Mode DRAMs
- special interface DRAMs (like RAMBUS)
- two banks in one chip (helps eliminate RAS precharge delays among other uses)
- mixed logic/DRAM on one chip
  - what is the logic that we should put in there ??
- what is the next thing that will replace DRAM ??

End

END