A Holistic Approach to DRAM (and Systems)

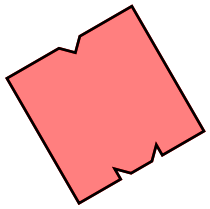
Prof. Bruce Jacob

Electrical & Computer Engineering

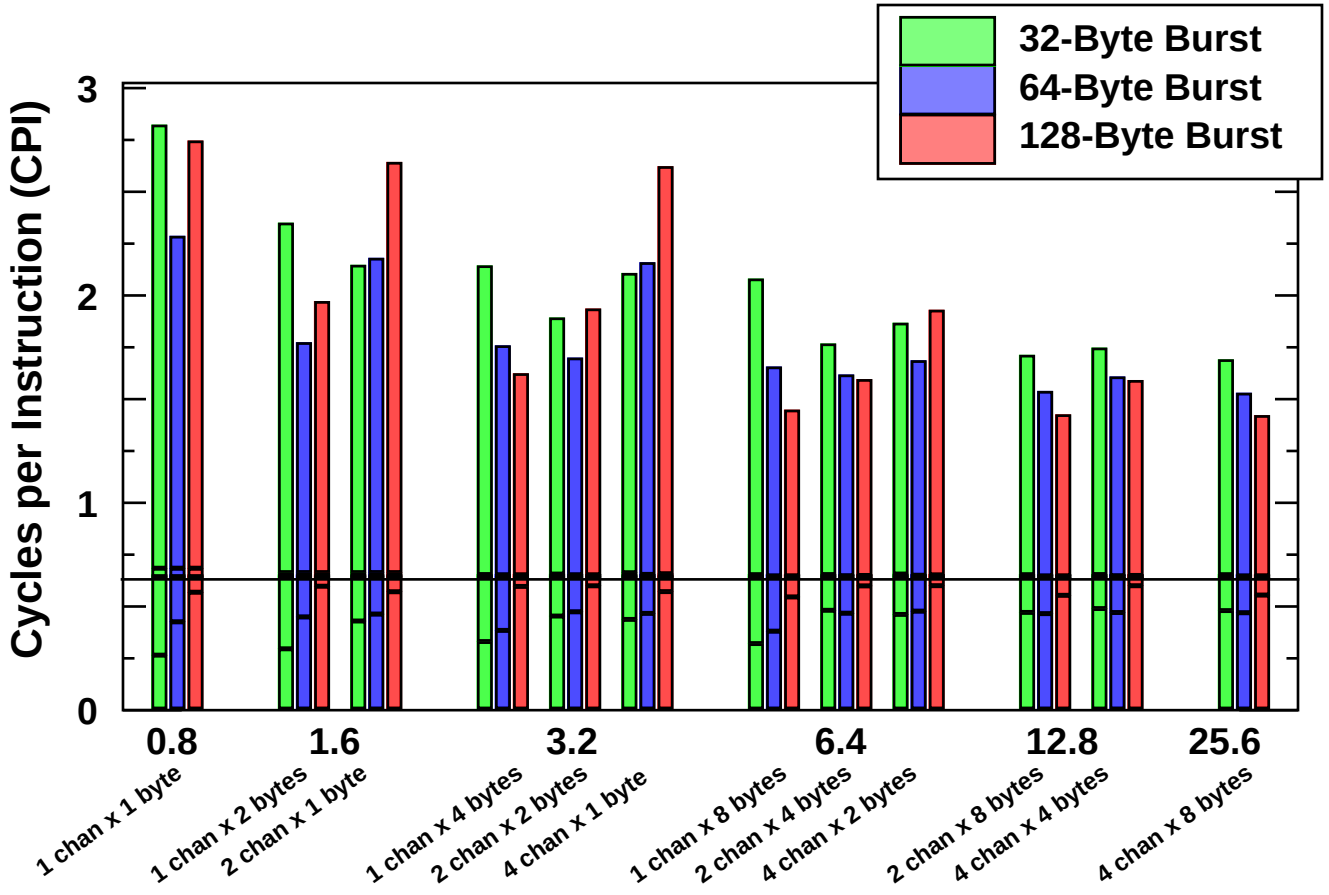
University of Maryland, College Park

OUTLINE

- **Anecdotes, Vision**
- **Our Past & Present Work**
- **Anecdotes Revisited**
- **Conclusions**



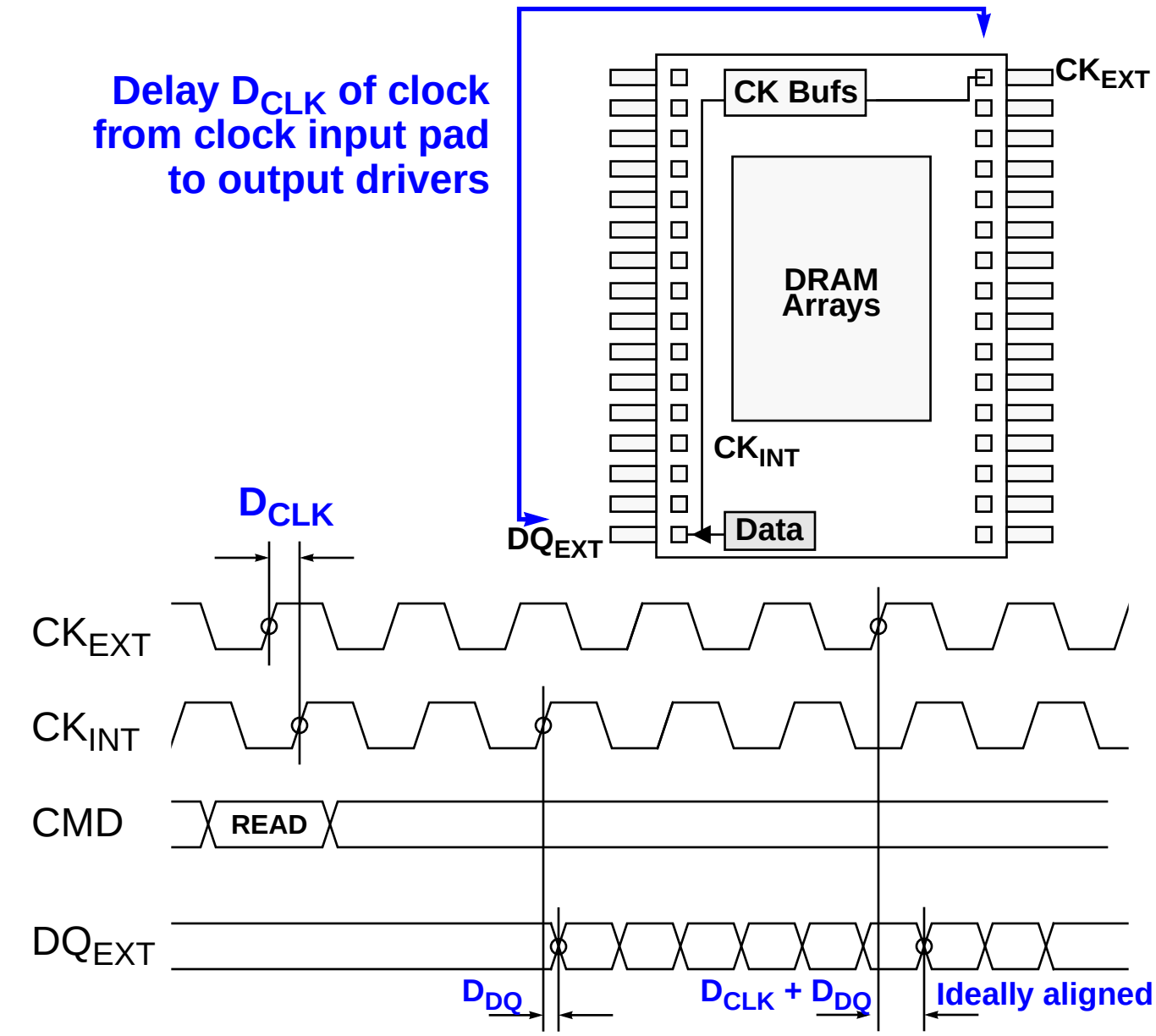
Anecdote I: System Issues



System Bandwidth
(GB/s = Channels * Width * 800MHz)

Benchmark = GCC (SPEC 2000), 2 banks/channel

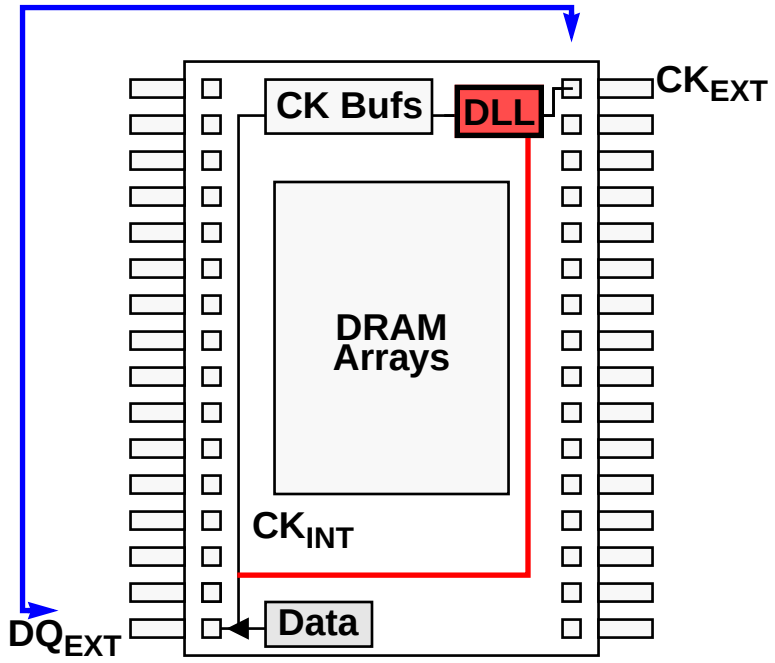
Anecdote II: DDR's DLL

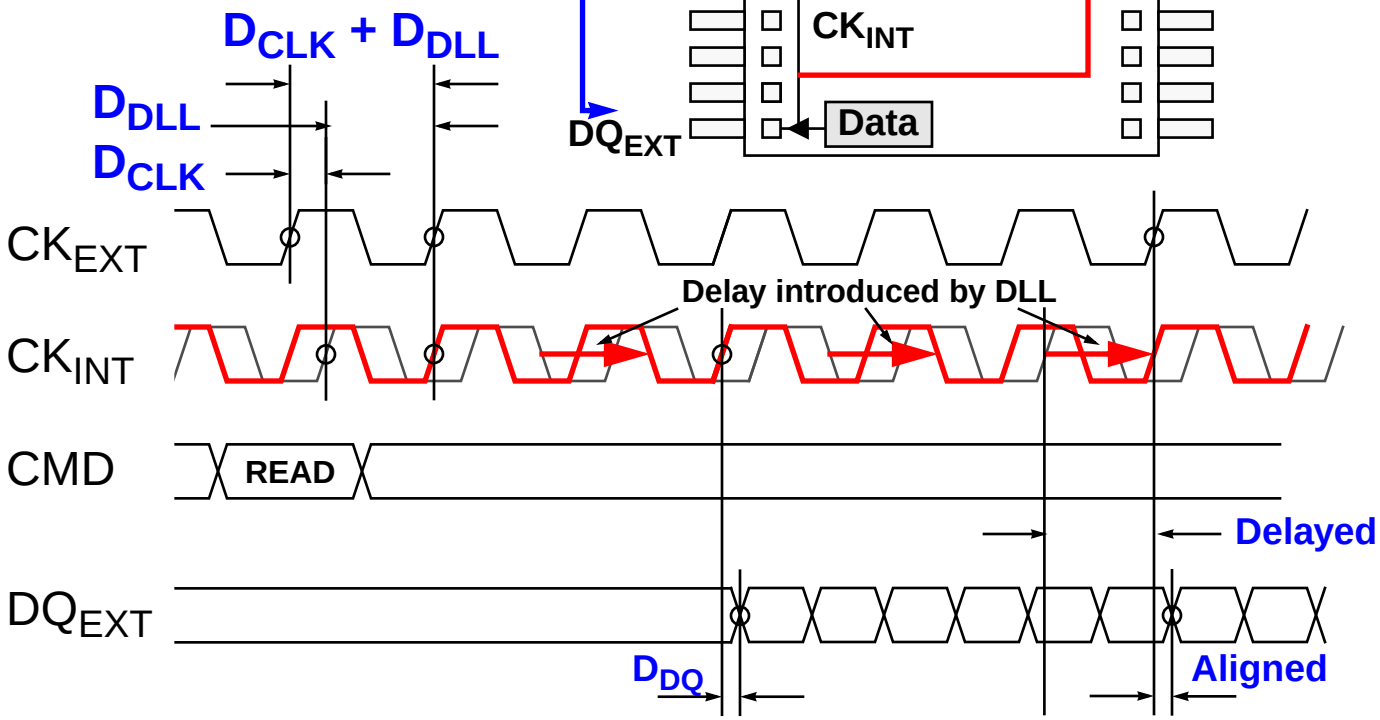


Anecdote II: DDR's DLL

Delay D_{CLK} of clock

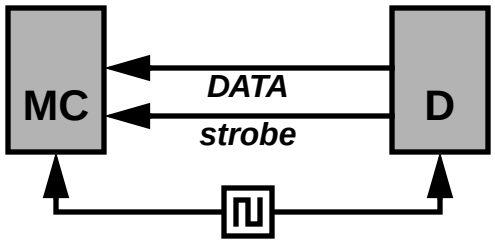
Point of DLL:
to align DQ output with system clock (minimize internal skew by eliminating D_{CLK})



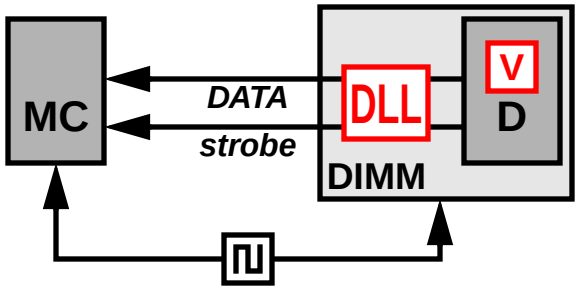


Anecdote II: DDR's DLL

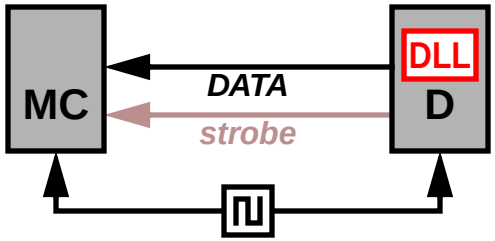
A handful of alternatives:



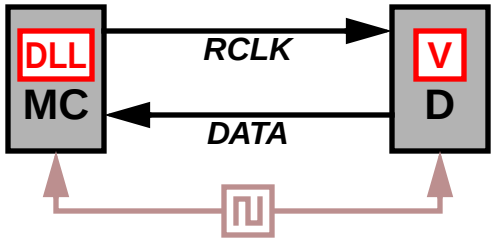
Unassisted



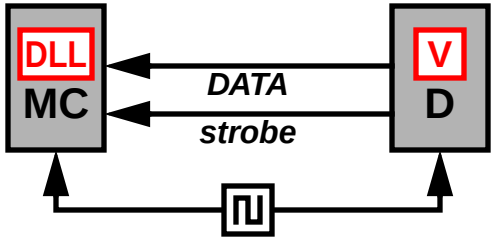
DLL on module



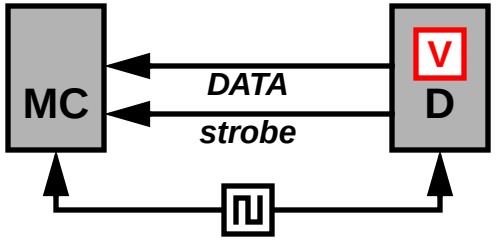
DLL on DRAM



Read clock



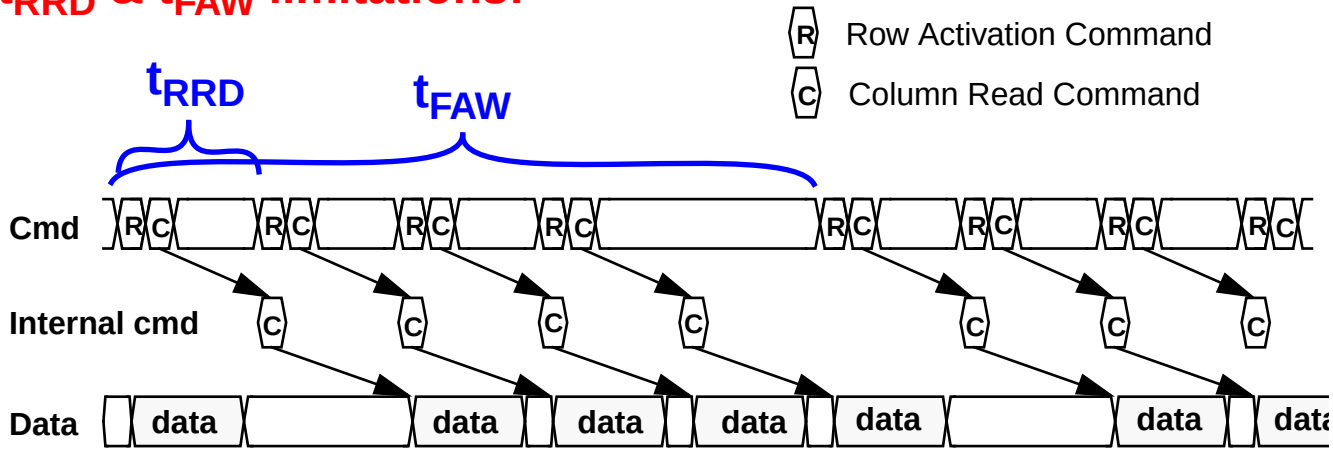
DLL on MC



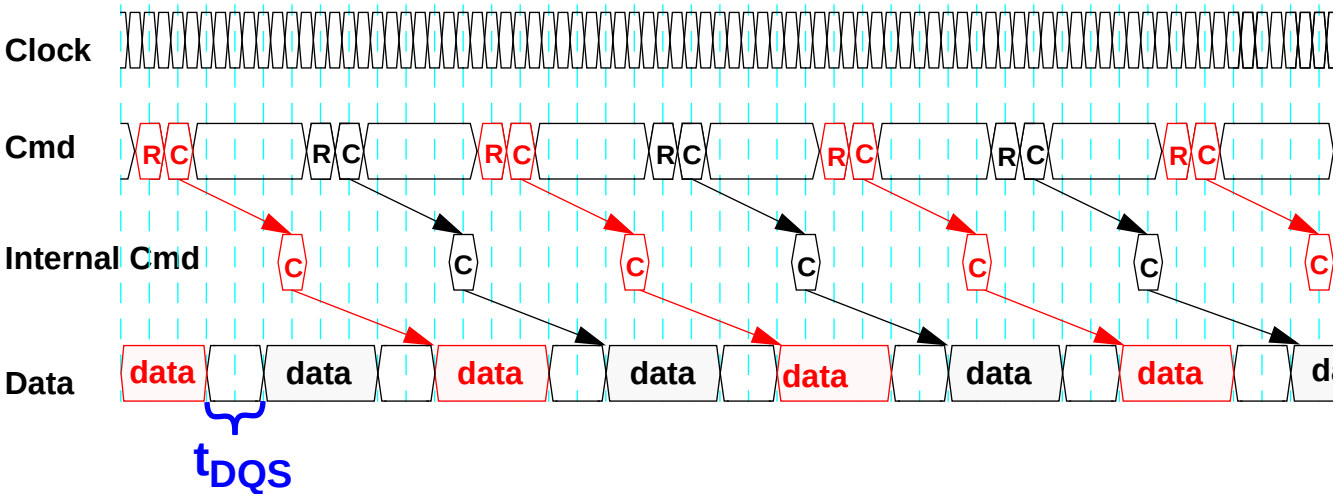
Static delay w/ recalibration

Anecdote III: Circuit v System

t_{RRD} & t_{FAW} limitations:



t_{DQS} limitations:



Vision

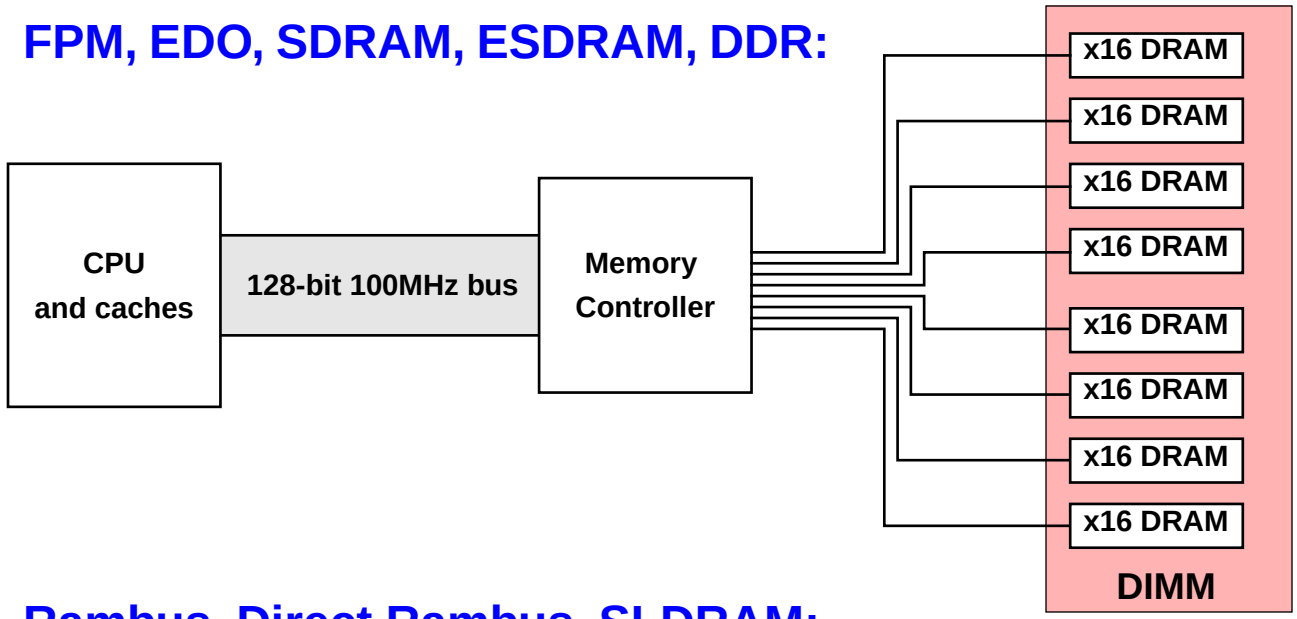
Must make circuit-level decisions considering system-level ramifications

Must make system-level decisions considering circuit-level ramifications

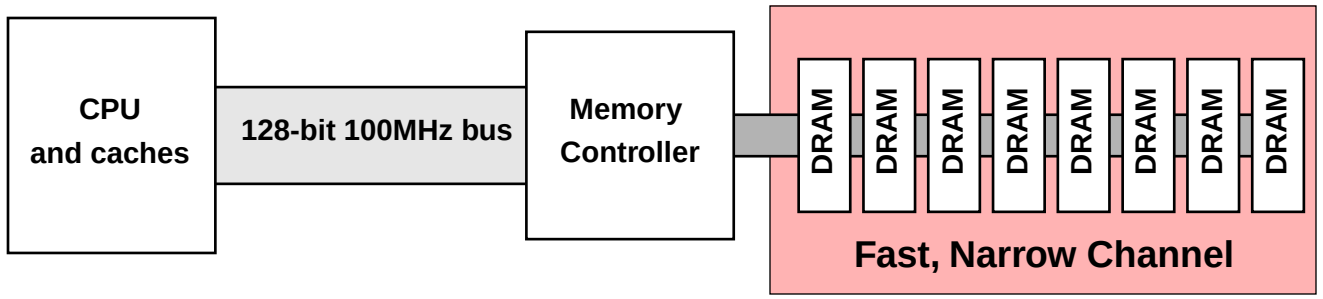
(holistic approach)

Past Work: Device-Level

FPM, EDO, SDRAM, ESDRAM, DDR:



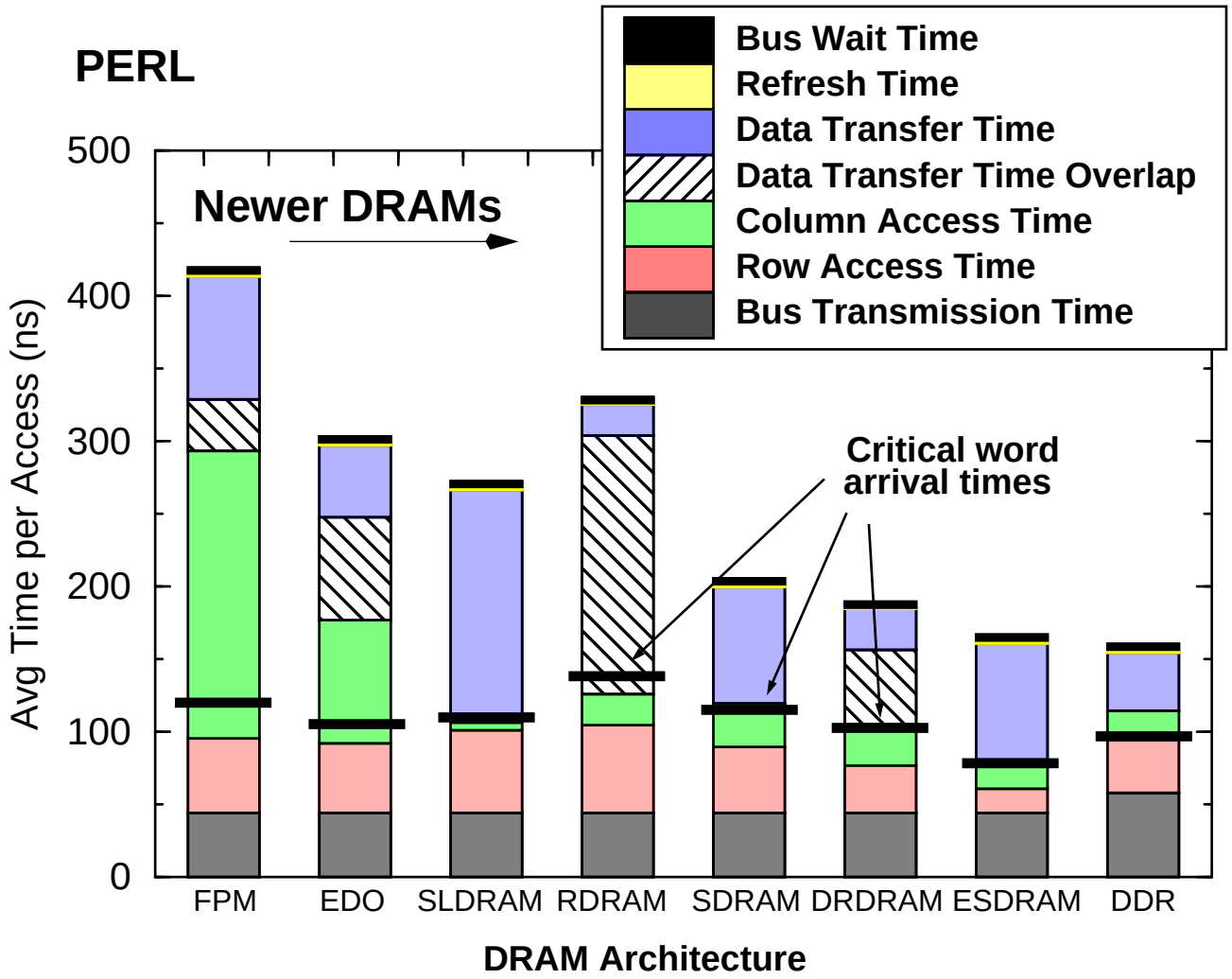
Rambus, Direct Rambus, SLDRAM:



[Cuppu et al. ISCA 1999]

Past Work: Device-Level

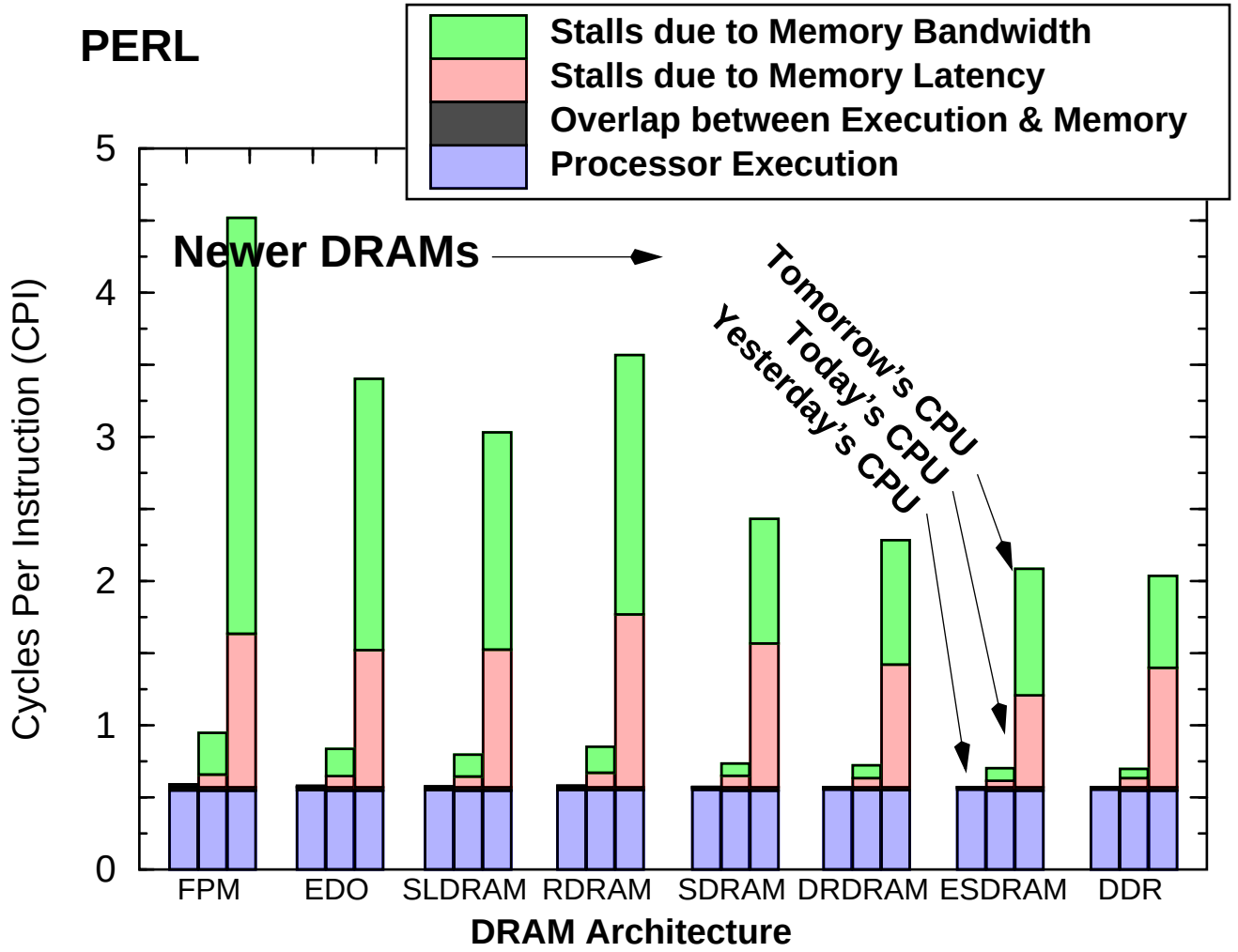
Average Latencies



[Cuppu et al. ISCA 1999]

Past Work: Device-Level

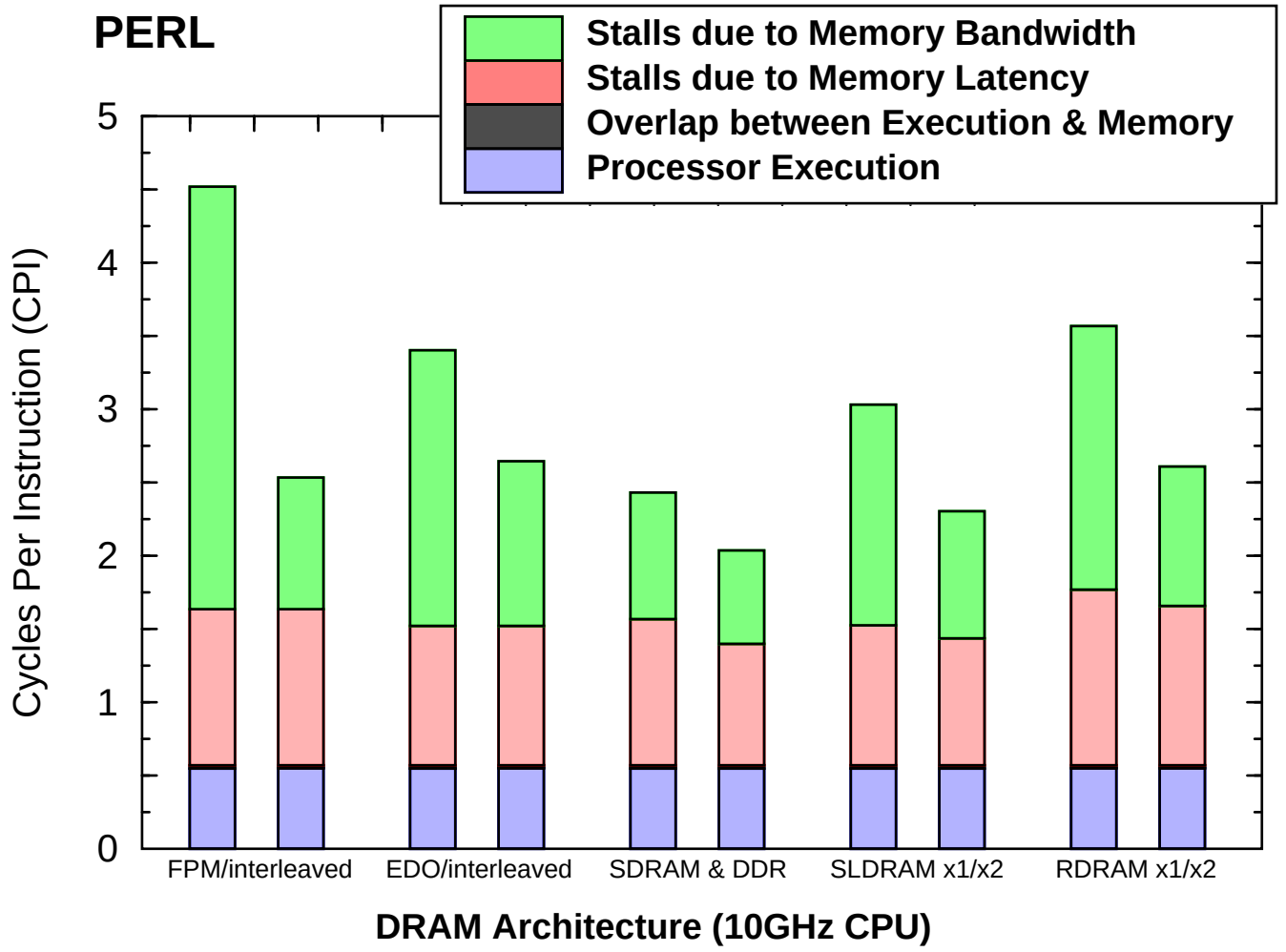
Bandwidth-Enhancing Techniques I:



[Cuppu et al. ISCA 1999]

Past Work: Device-Level

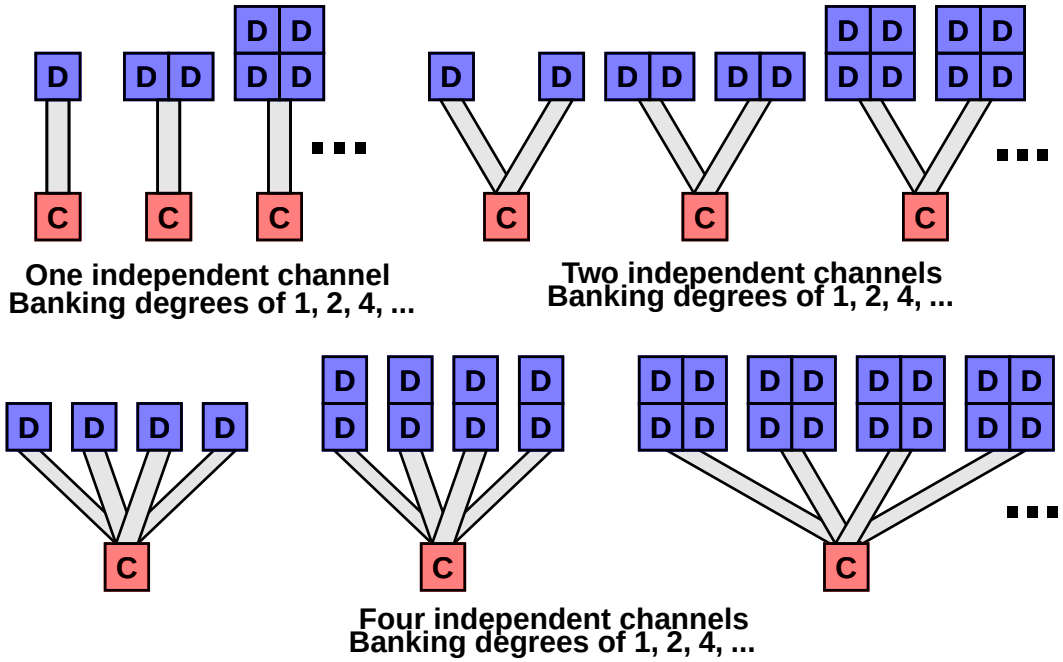
Bandwidth-Enhancing Techniques II:



[Cuppu et al. ISCA 1999]

Past Work: System-Level

Even when we restrict our focus …

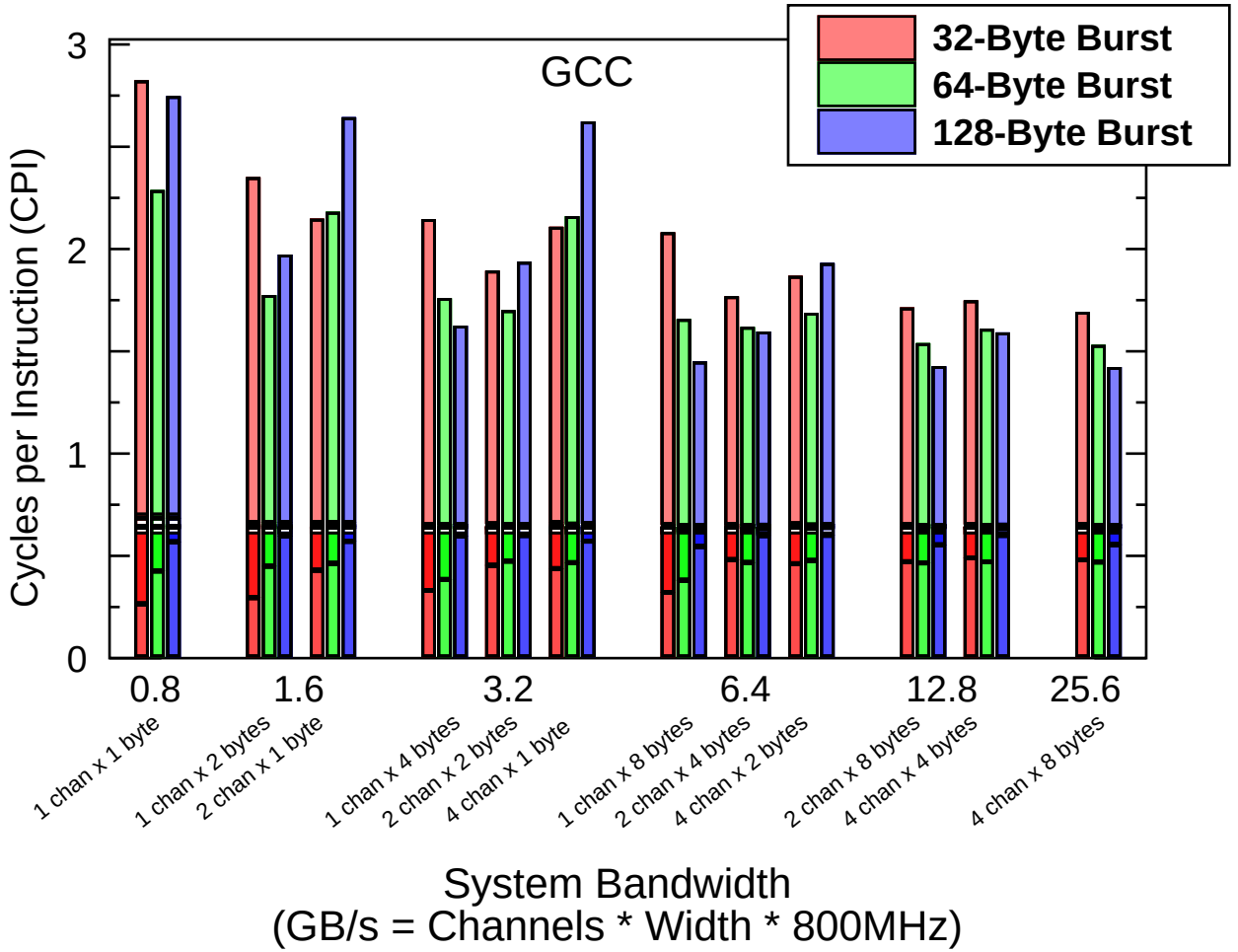


1, 2, 4	800 MHz Channels
8, 16, 32, 64	Data Bits per Channel
1, 2, 4, 8	Banks per Channel (Indep.)
32, 64, 128	Bytes per Burst

[Cuppu & Jacob ISCA 2001]

Past Work: System-Level

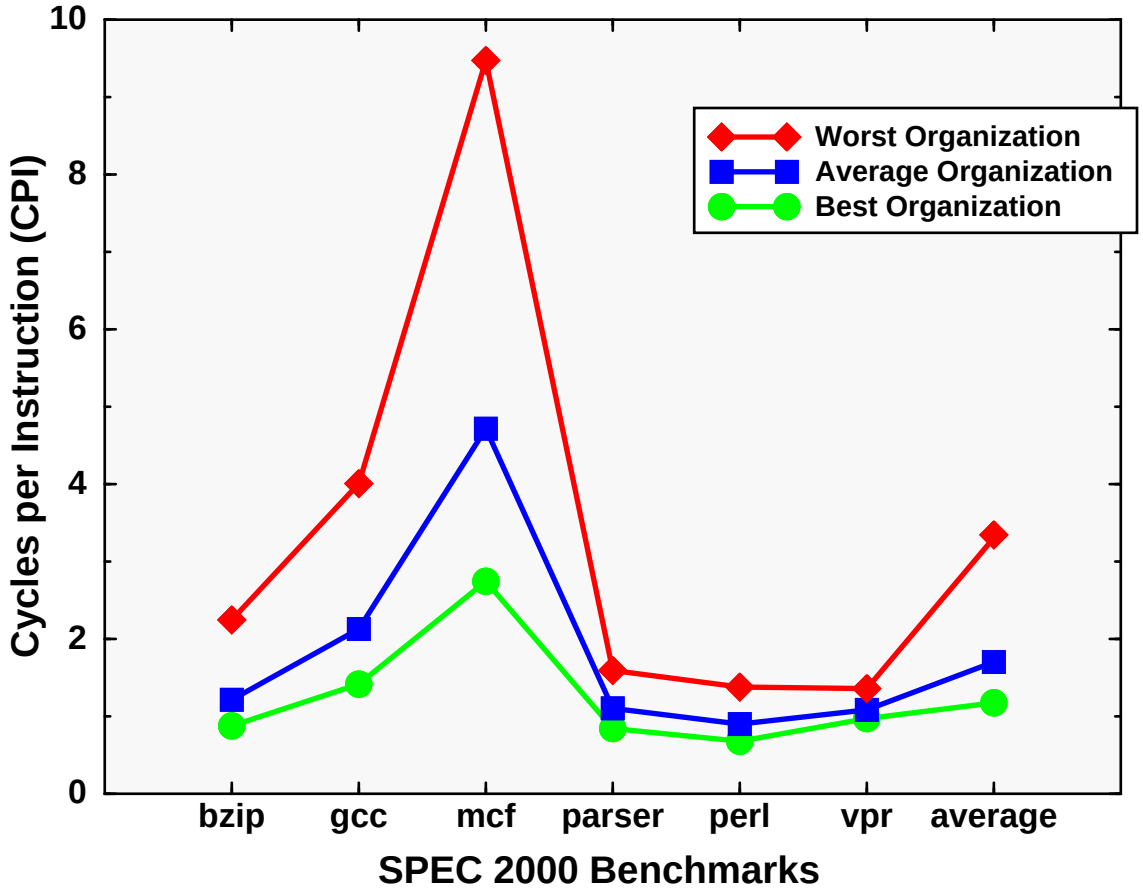
... the design space is FAR from regular …



[Cuppu & Jacob ISCA 2001]

Past Work: System-Level

... and the cost of poor judgment is high.



[Cuppu & Jacob ISCA 2001]

An Aside

Past work used first-order models.

Present work uses models accurate to *second & third order* effects …

[Definition: *Zero'th Order*]

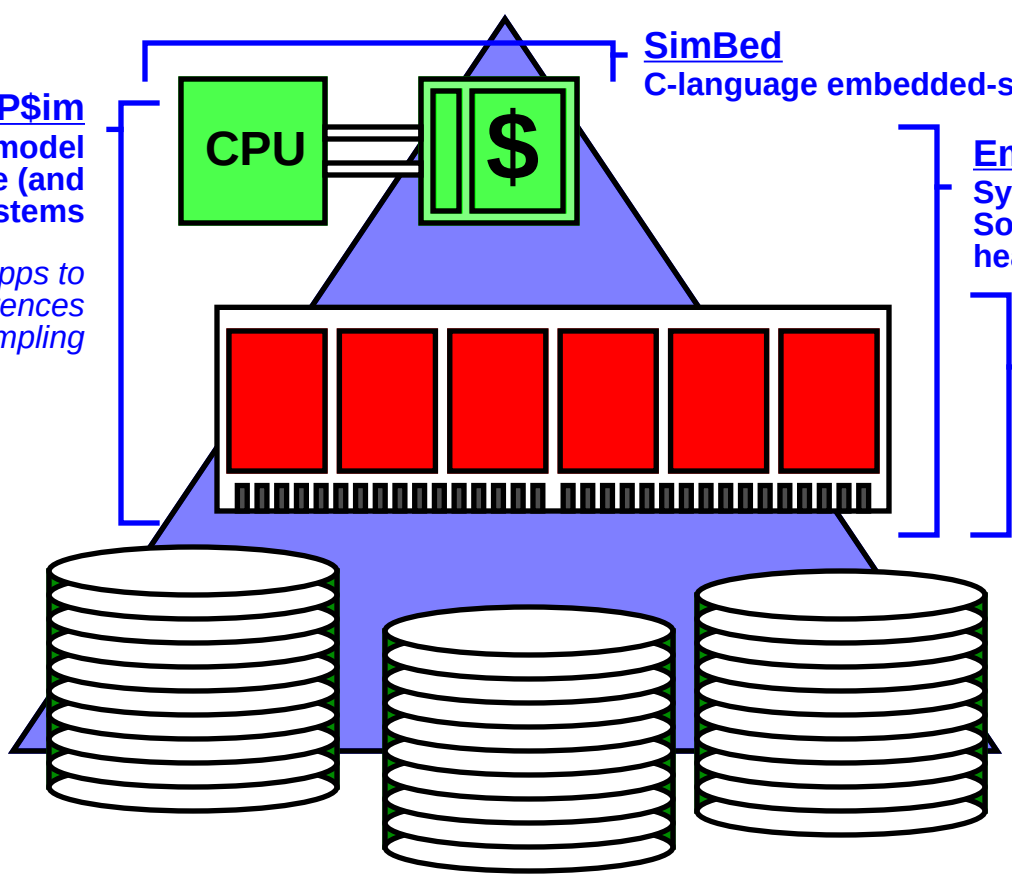
```
...

if ( INSTR.is_loadstore ) {
    if (L1_cache_miss( INSTR.daddr )) {
        if (L2_cache_miss( INSTR.daddr )) {

            cycles += DRAM_LATENCY;

                OR

            INSTR.ready = now() + DRAM_LATENCY;

        }
    }
}

...
```

An Aside

Past work used first-order models.

Present work uses models accurate to *second & third order* effects …

Past & Present Work



SimBed
C-language embedded-system model

CMP$im
Pin tool to model CMP cache (and DRAM) systems

Executes apps to 1 trillion references without sampling

EmPower
SystemC model of SoC, incorporating heat and power

DRAMsim
C-language DRAM model

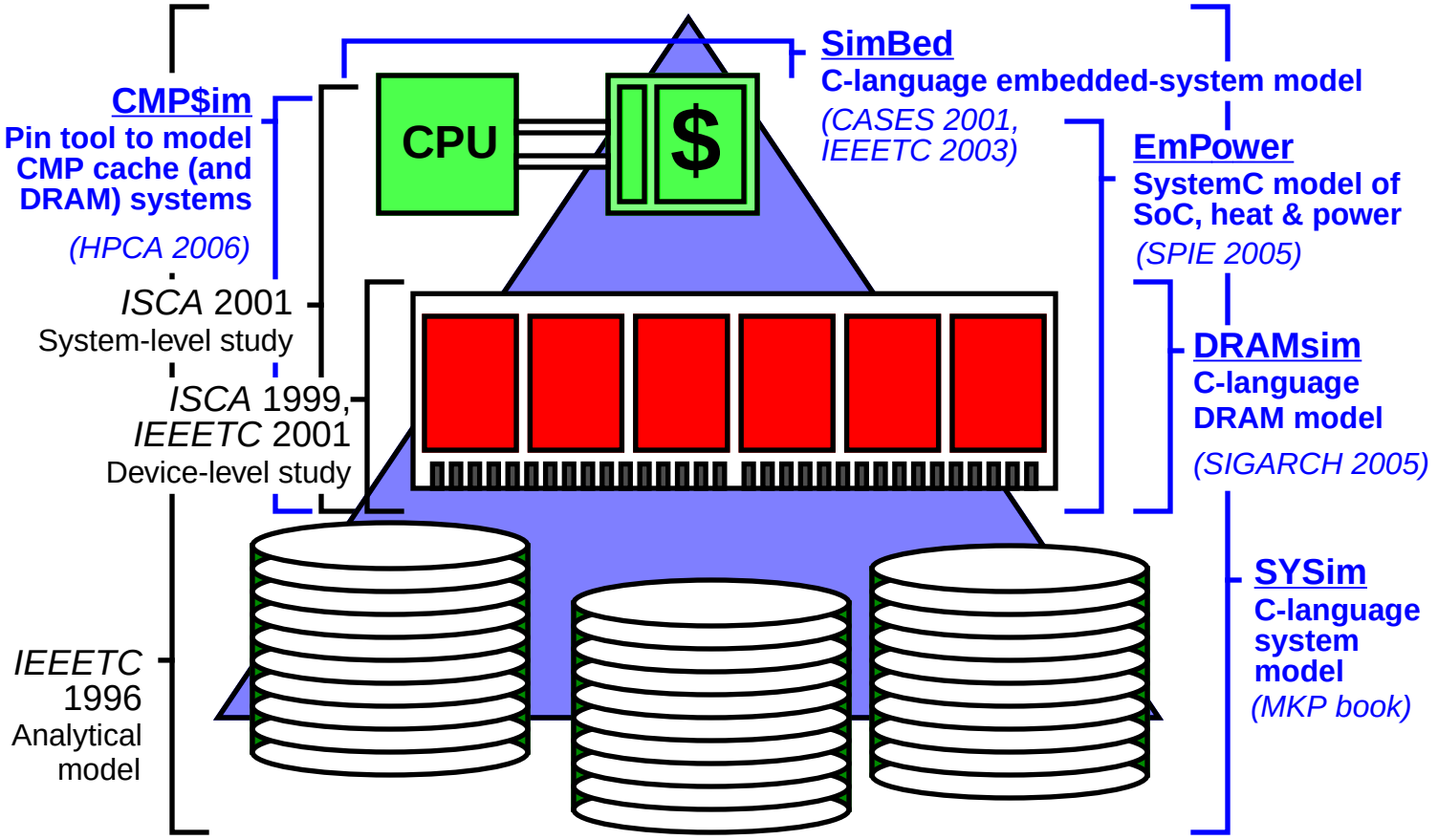
DRAM, BIU, MC GEMS, M5, etc.

SYSim
C-language system model

Models both performance & energy

Recent development work

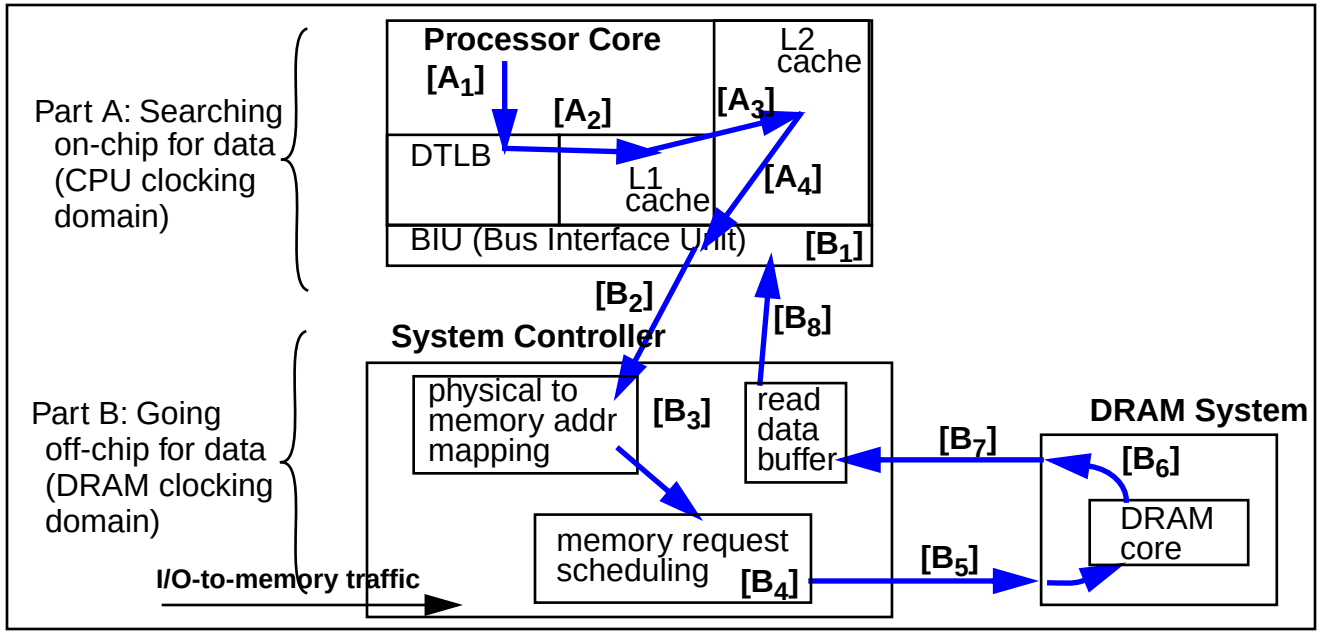
Past & Present Work



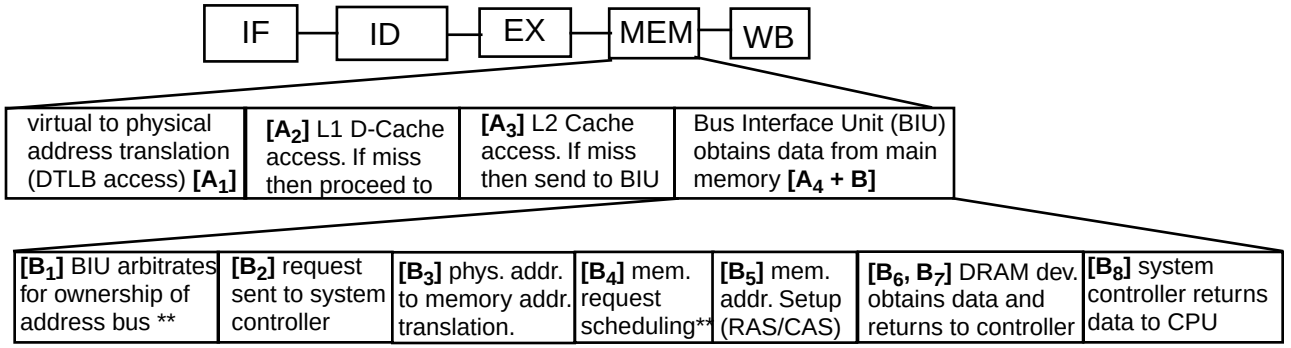
***IEEETC* 1996:**	**System-level analytical tool for cost/performance**
***ISCA* 1999, *IEEETC* 2001:**	**DRAM device-level characterization**
***CASES* 2001, *IEEETC* 2003:**	**Performance & energy modeling of RTOS, CPU, memory**
***ISCA* 2001, *IEEE Micro* 2003:**	**DRAM system-level characterization**
***SPIE* 2005:**	**SystemC modeling of energy in systems-on-chip**
***SIGARCH* 2005:**	**DRAMsim released to community**
***ISPASS* 2005, *HPCA* 2006:**	**Characterization of bioinformatics workloads**

DRAMsim

Execution of a Load Instruction



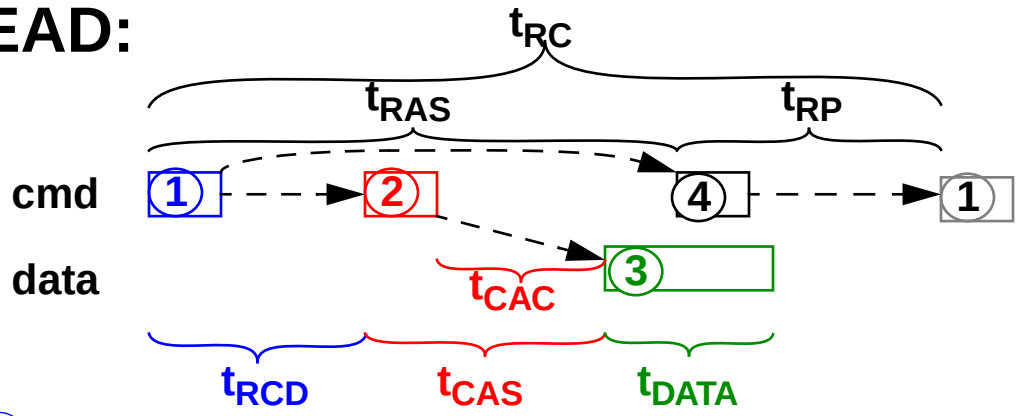
Stages of instruction execution:



** Steps not required for some processor/system controllers. protocol dependant.

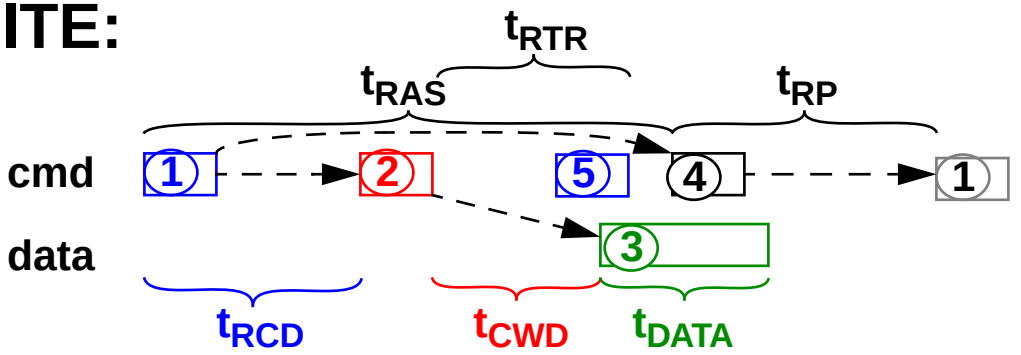
DRAMsim

READ:



1. Active: Open Row, t_{RCD} time later, a CAS command may be issued to the DRAM chip
2. CAS: Column Read command, t_{CAS} time later, data begins to be placed onto the Data bus. We use t_{CAC} to factor out command transmission time.
3. Data: The number of cycles that the data transmits over the Data bus
4. Precharge: Close the Row, this command may be issued t_{RAS} time after the Active command. After t_{RP} time, another active command may be issued.

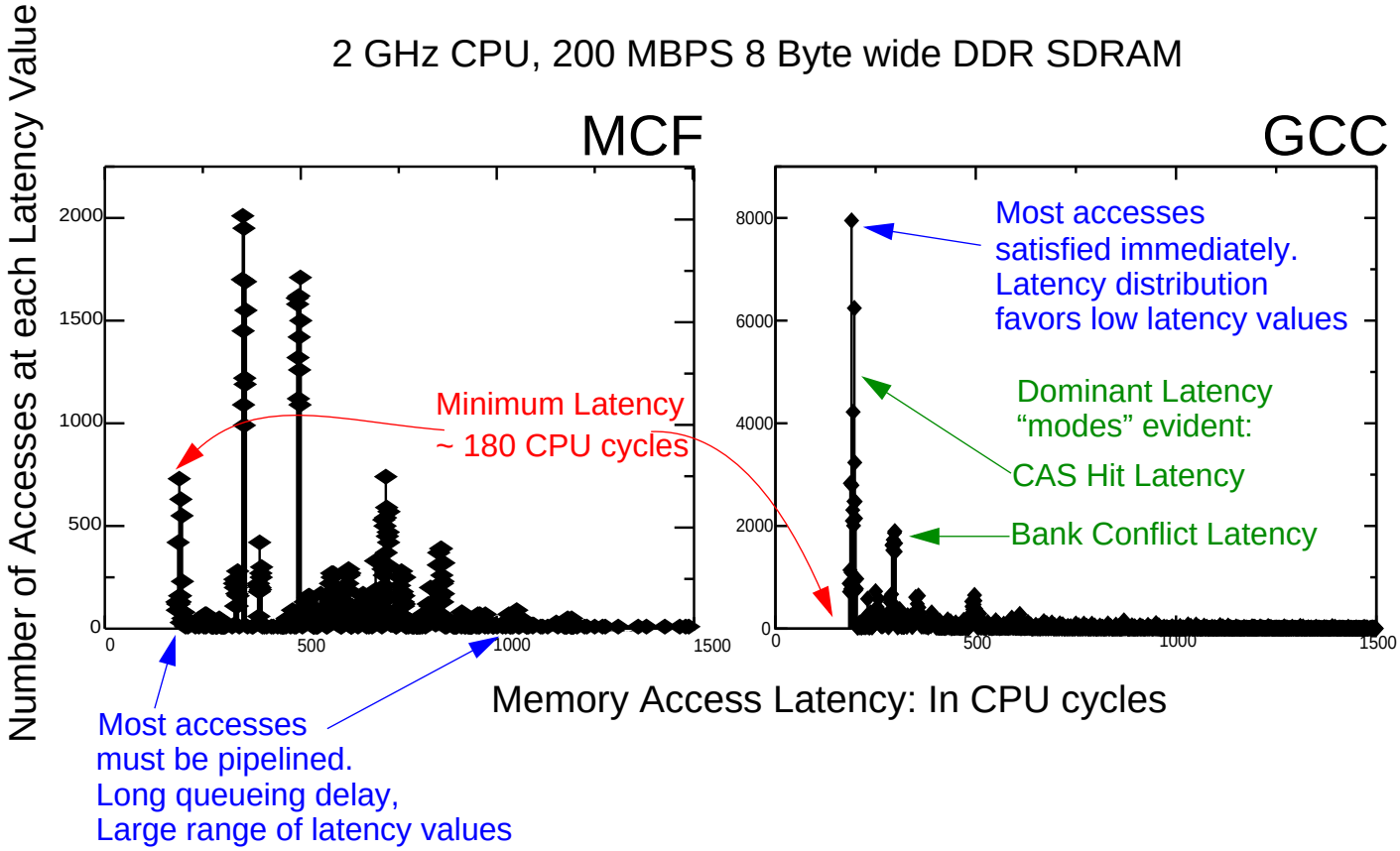
WRITE:



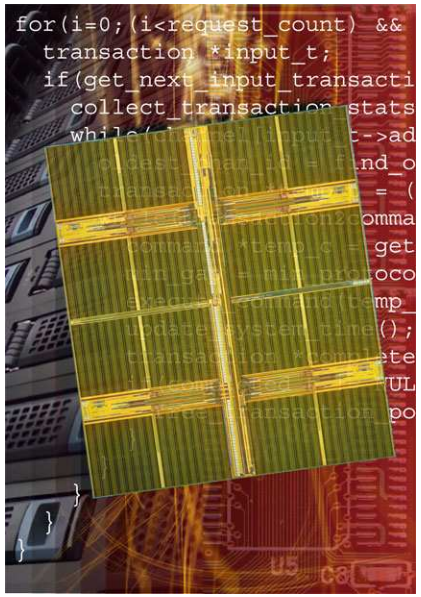
2. CWD: Column Write Delay, the number of cycles that the controllers must wait before placing the data onto the data bus.
5. RTR: Retirement delay, this is for systems with write delay buffers.(RDRAM)

DRAMsim

Memory Access Latency Distribution



DRAMsim



University of Maryland
DRAMsim: A Detailed Memory-System Simulation Framework

Bruce Jacob – *blj@ece.umd.edu* – http://www.ece.umd.edu/~blj/

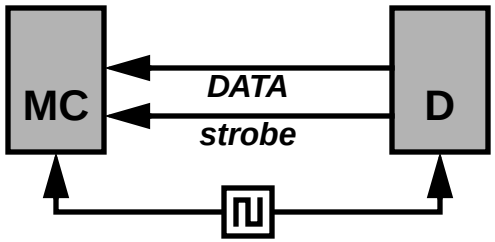
- Overview of DRAMsim
- Why do we need this level of accuracy?
- Screenshots of GUI trace viewers
- DRAMsim version 2
- Download DRAMsim
- Contact information

Brought to you by Maryland Memory-Systems Research

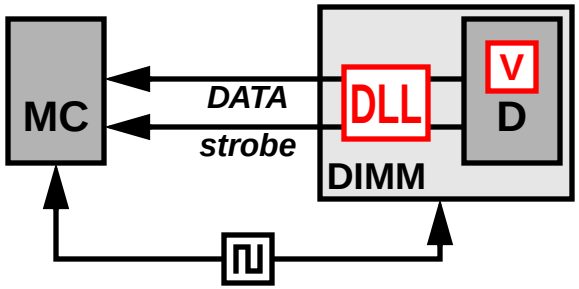
http://www.ece.umd.edu/dramsim

Accuracy: *Why?*

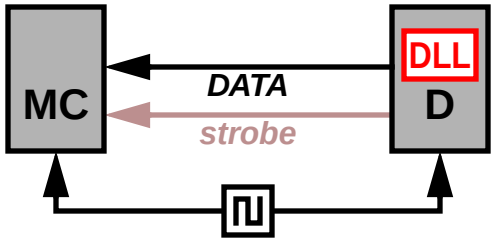
Benefit: *Insights (Anecdote II, revisited)*



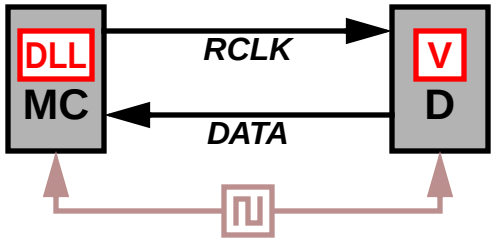
Unassisted



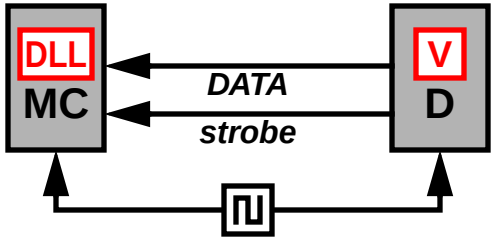
DLL on module



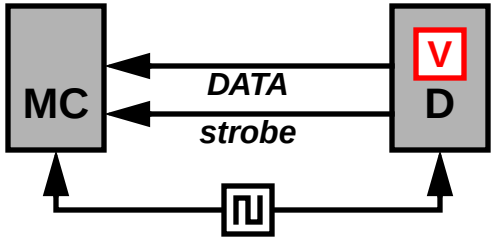
DLL on DRAM



Read clock



DLL on MC



Static delay w/ recalibration

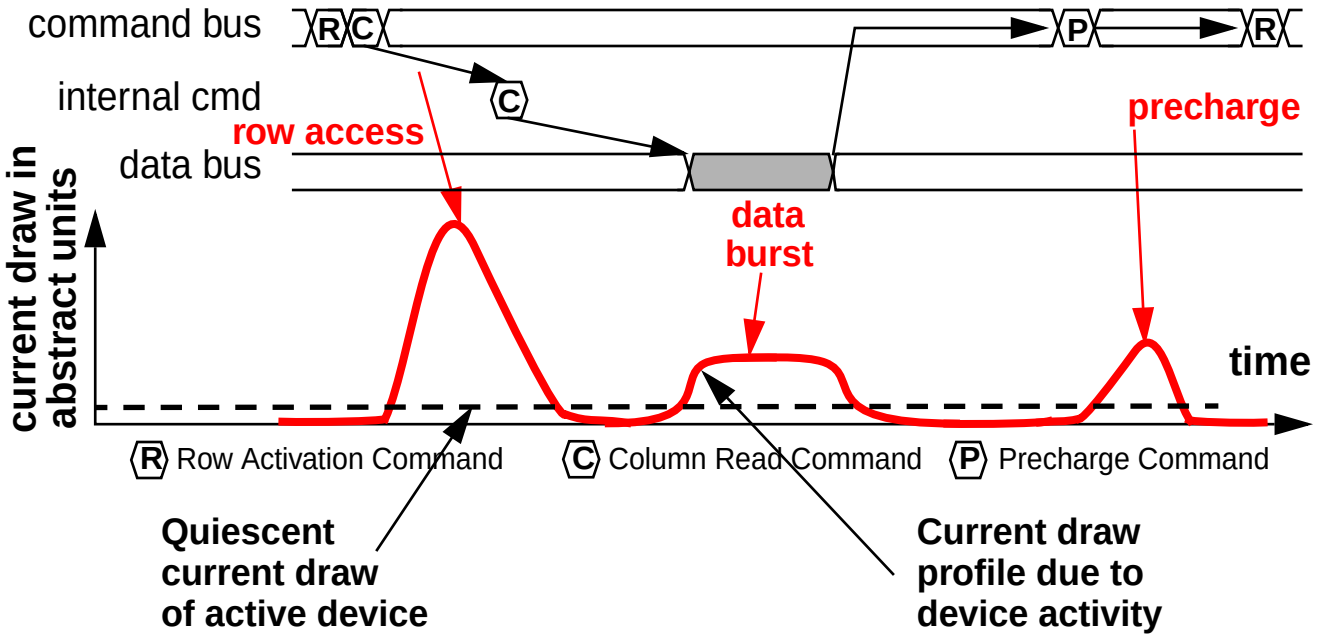
Accuracy: *Why?*

Benefit: *Insights (Anecdote II, revisited)*

SCHEME	*COST*	*EFFECTIVENESS (Uncertainty in read)*
No DLL	0	D_{CLK} + Xmit + wire + Recv + Clk skew
on DRAM	16xDLL	Xmit + wire + Recv + Clk skew
on MC	2xDLL 16xVern	wire + Recv
on DIMM	2xDLL 16xVern	wire + Recv + Clk skew
Read CLK	2xDLL 16xVern	wire + Recv
Static	16xVern	Xmit + wire + Recv

- **Cost = for 2-DIMM system, 8 DRAM parts per DIMM note: "cost" applies to both die area *and* power**
- **Uncertainty = *very rough, intuitive idea***

Anecdote III, revisited



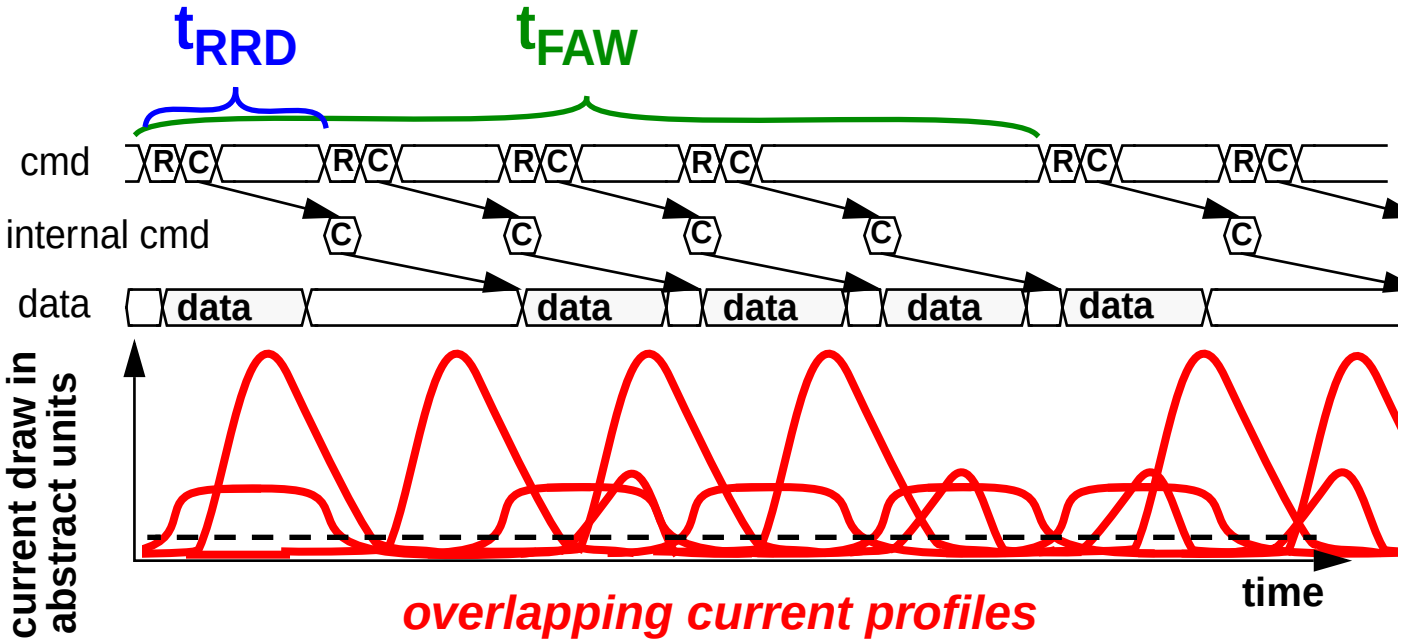
Power consumption in DRAM devices:

- **Row activation, data read-out, bank precharge: all are relatively expensive operations**
- **Current draw of operation additive to quiescent value**

… So what's the big deal?

Anecdote III, revisited

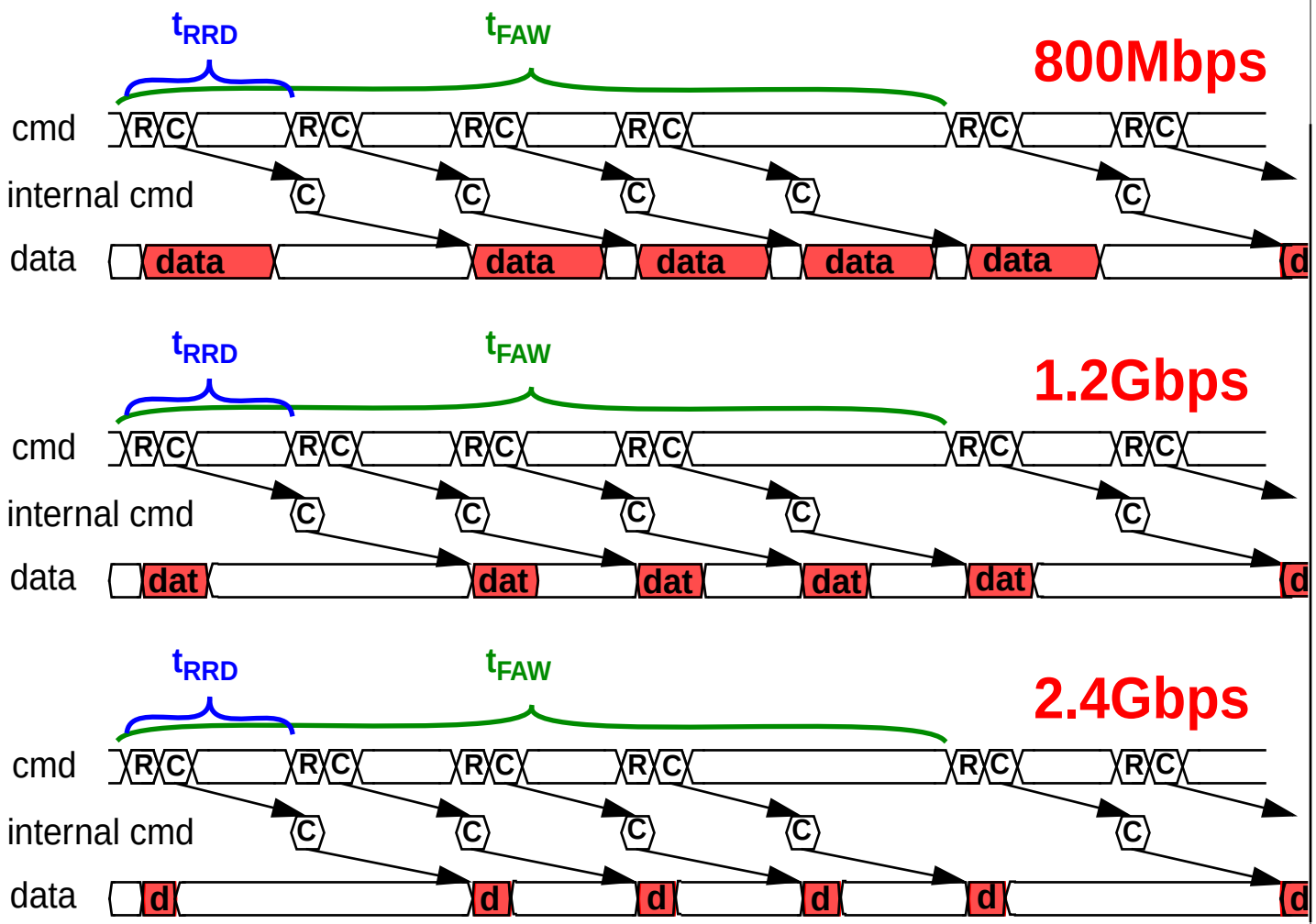
t_{RRD} & t_{FAW} ***protocol-level limitations placed upon device to limit maximum current draw***



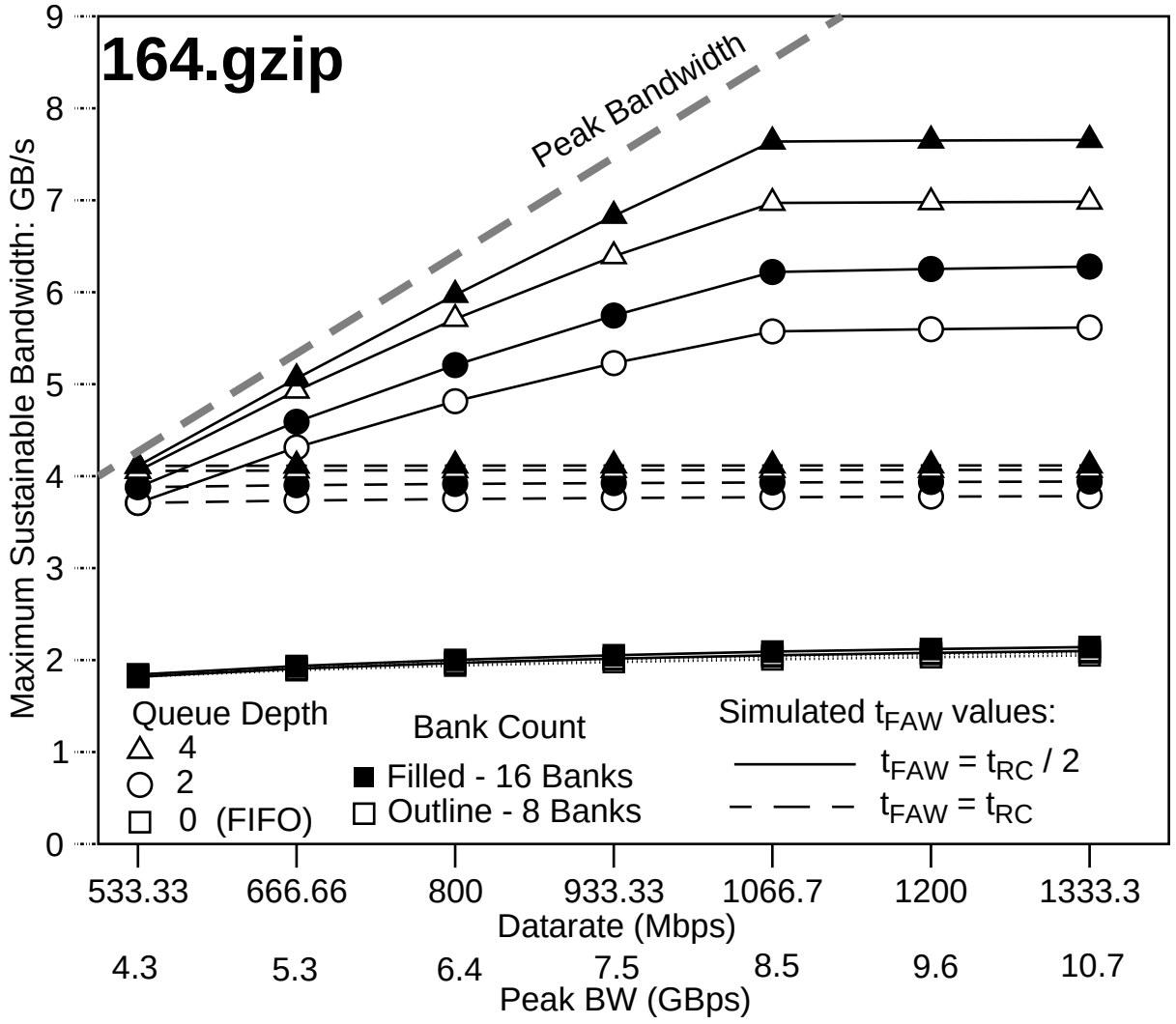
- **Severely limits bus efficiency from single rank**
- **Problem worsens in future: parameters defined in *nanoseconds*, not *cycles***

Anecdote III, revisited

- **t_{RRD} & t_{FAW} — Problem worsens in future: parameters defined in *nanoseconds*, not *cycles***



Max. Sustainable Bandwidth



t_{RC} = 60ns, burst of eight, 8B wide channel

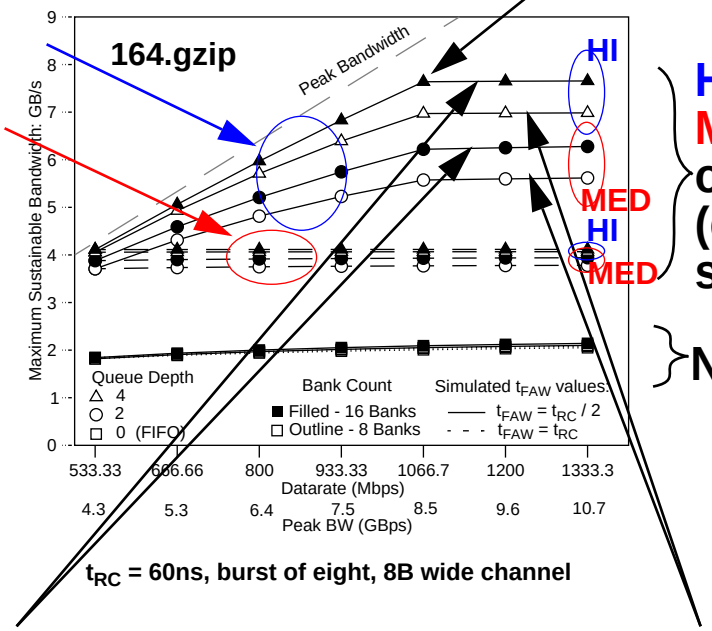
Max. Sustainable Bandwidth

t_{FAW} Impact for Two Different Simulated Values:

($t_{FAW} = t_{RC}/2$)

($t_{FAW} = t_{RC}$)

Inflection Point For $t_{FAW} = t_{RC} / 2$



High and Moderate Levels of Reordering (e.g. system sophistication)

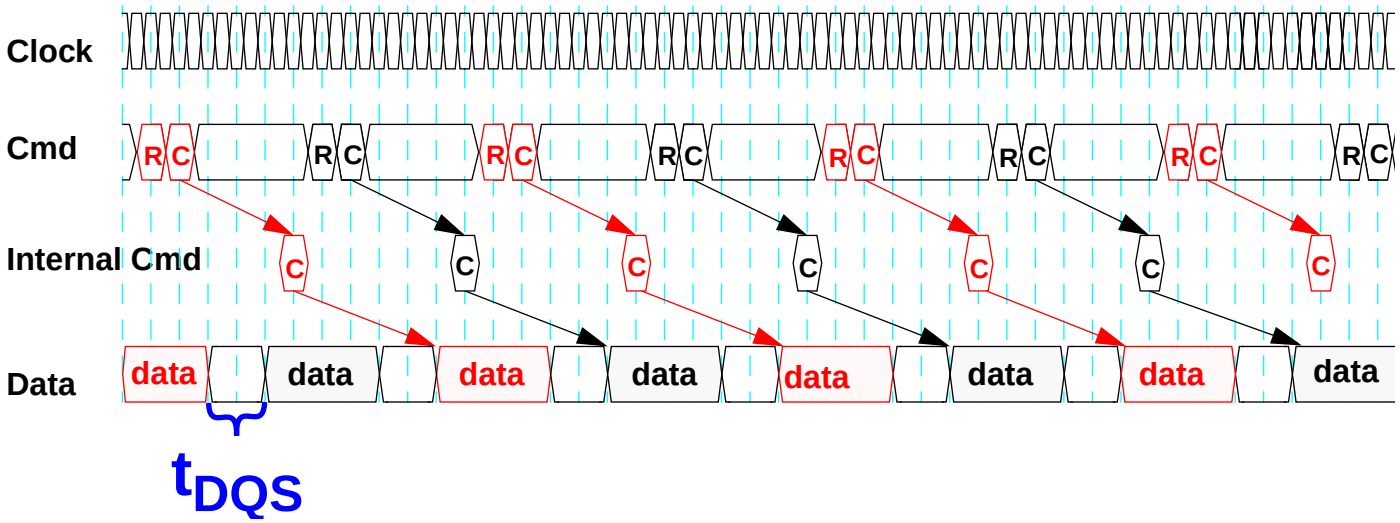
No Reordering

t_{RC} = 60ns, burst of eight, 8B wide channel

16 BANKS improves bandwidth over 8 BANKS by ~10%
(how does this compare with incremental cost?)

But Wait, There's More …

t_{DQS} ***protocol-level limitation placed upon ranks to prevent data-bus collisions on rank hand-off***



- **Severely limits bus efficiency from multiple ranks**
- **Luckily, it is defined in *cycles* and not *nanoseconds***

Solution I: Scheduling

Problems created by t_{FAW} + t_{RRD} + t_{DQS}

- t_{FAW} + t_{RRD} — Must spread out ACT commands
- t_{DQS} — Must switch ranks infrequently

Salient point: t_{FAW} does not place limit on *total* number of open banks

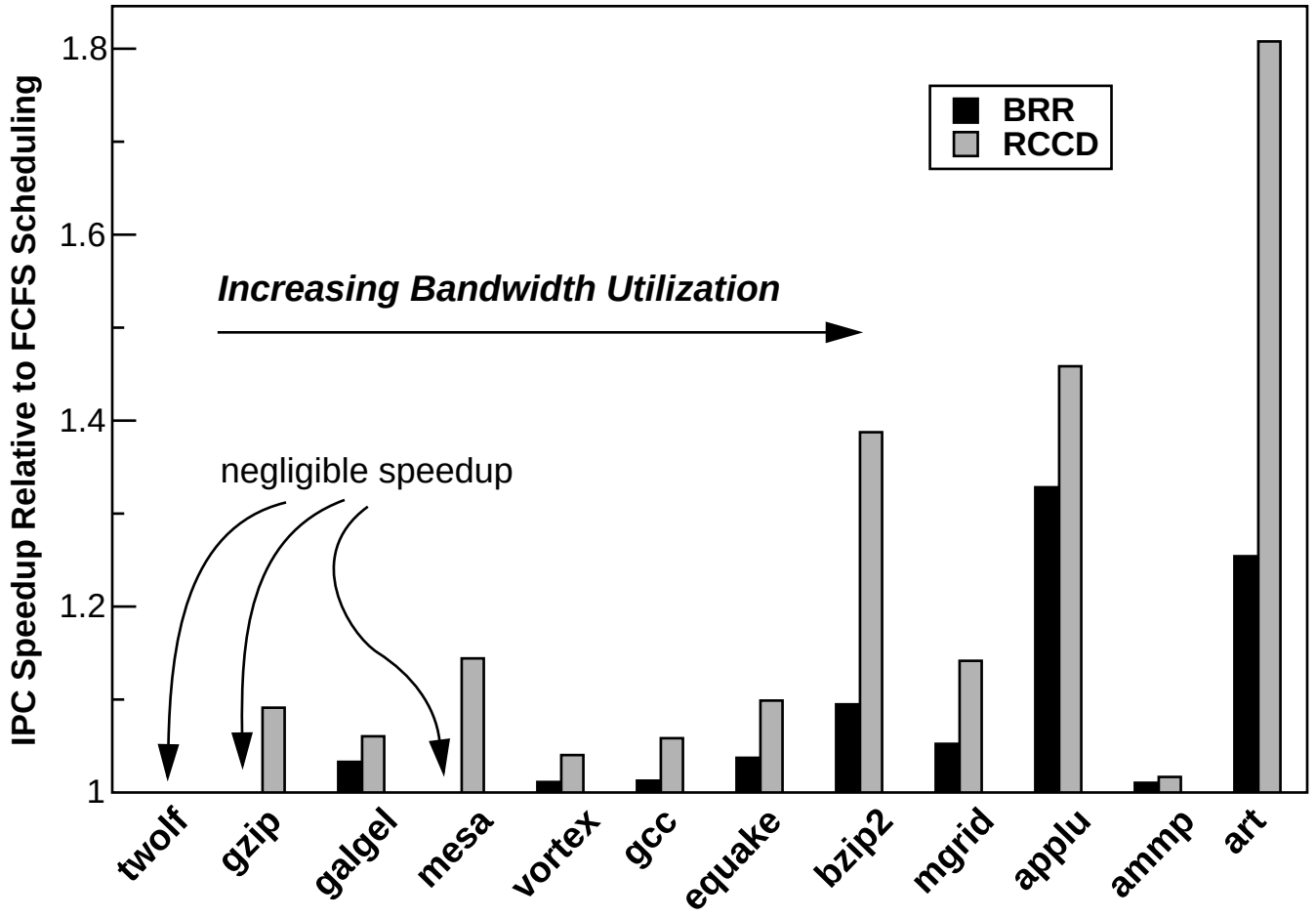
Problem can be solved with scheduling: *row-column command decoupling (RCCD)*

- Schedule ACT commands far before their corresponding READ commands
- Schedule large number of bank-reads before switching ranks

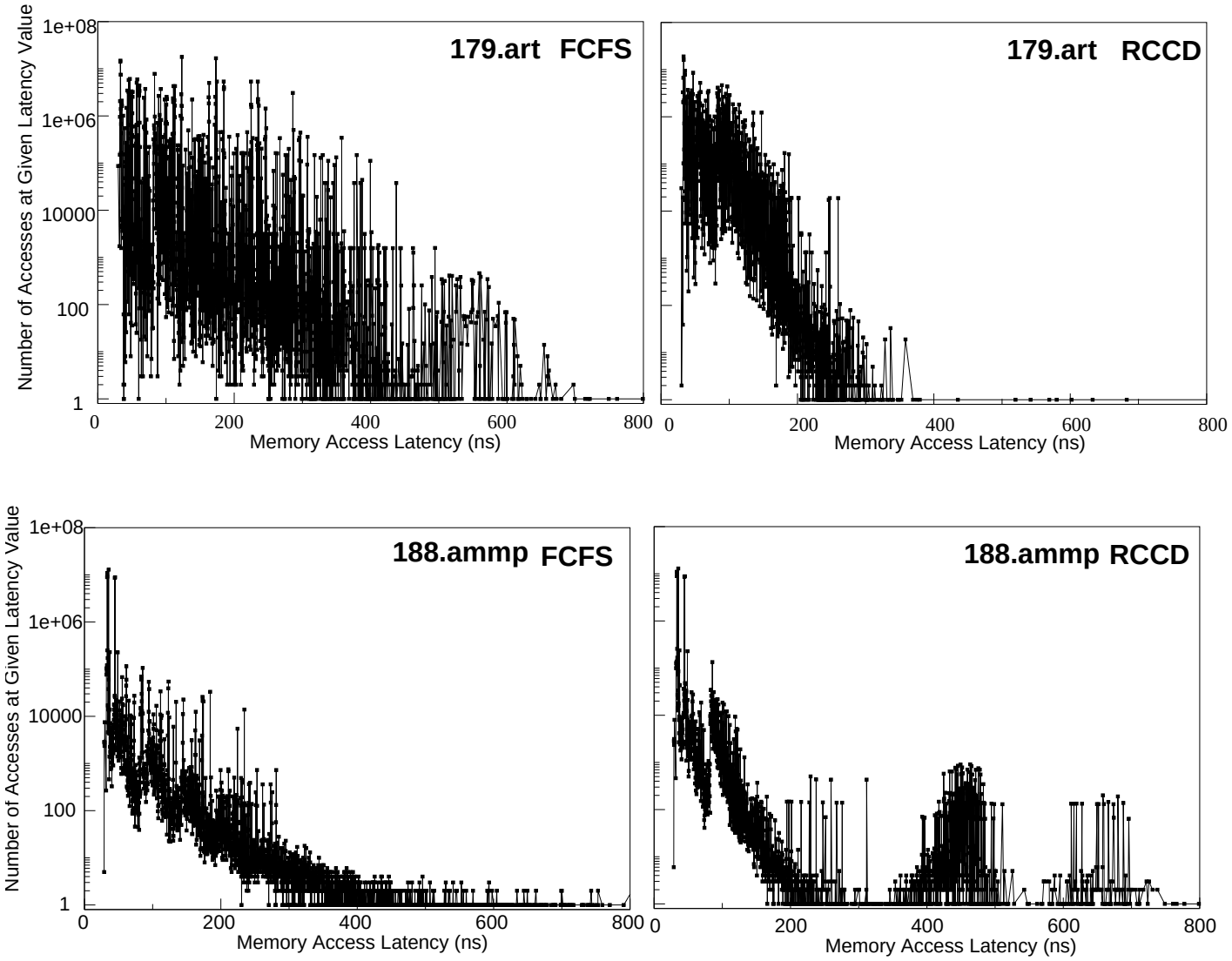
[patent pending]

Solution I: Scheduling

IPC Speedup Relative to FCFS Scheduling



Solution I: Scheduling

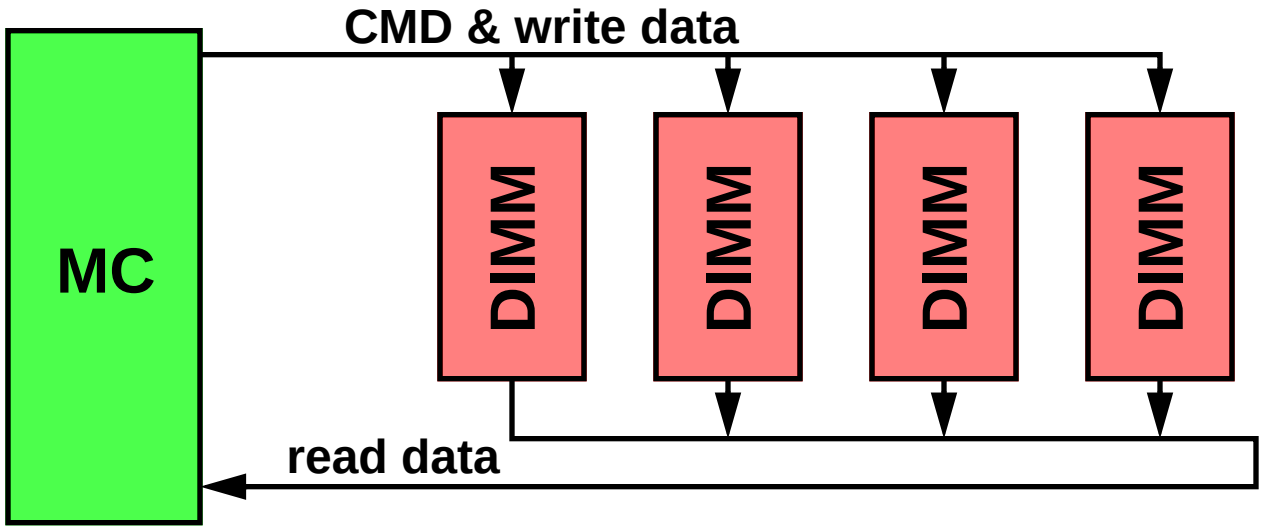


Solution II: Topology, etc.

Problems solved by t_{FAW} + t_{RRD} + t_{DQS}

- t_{FAW} + t_{RRD} — Instantaneous current draw in device
- t_{DQS} — Bus collisions on rank handoffs

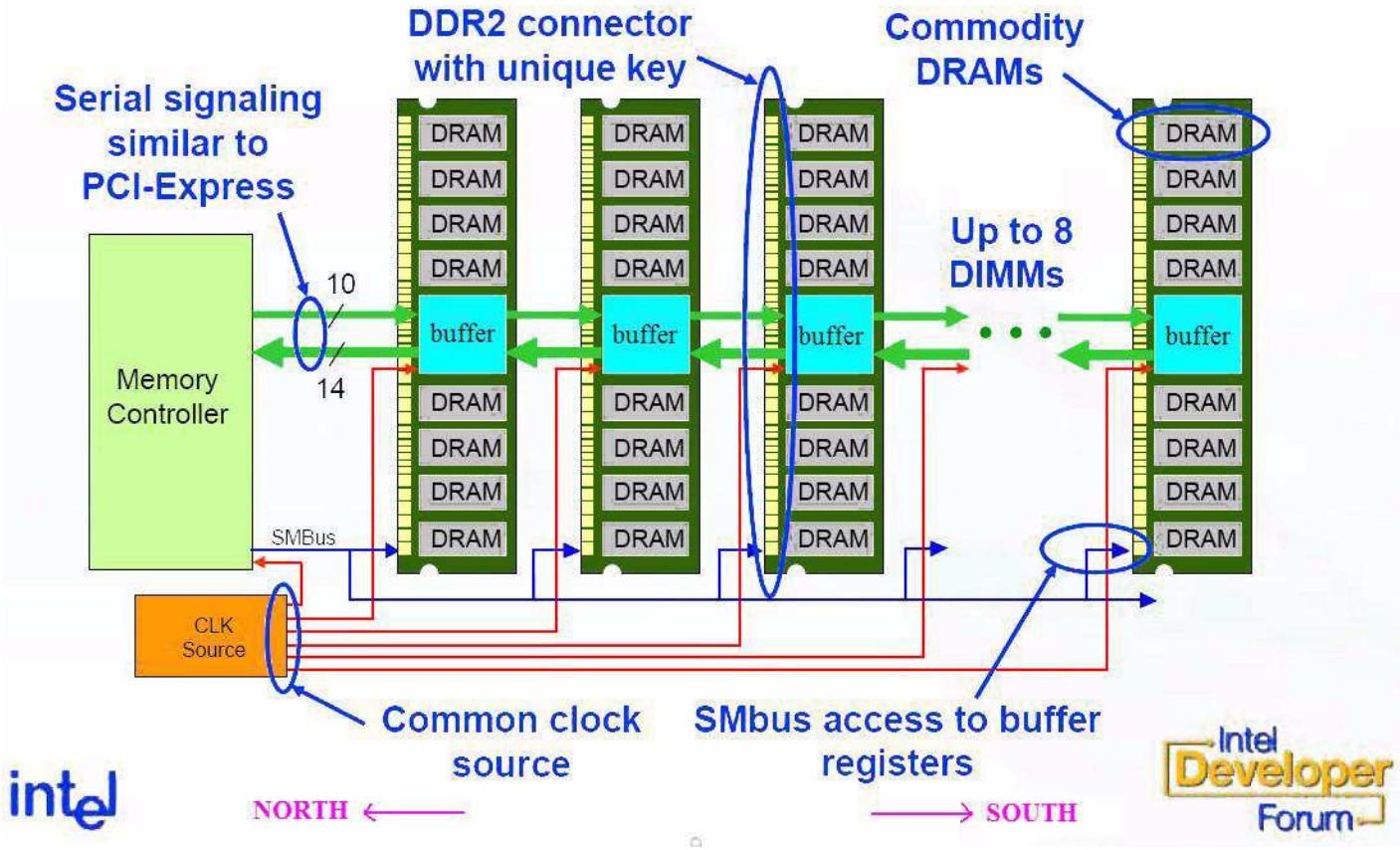
Any alternative solution will do …



- Topology eliminates collisions (can account for static DIMM-DIMM skew with Vernier-type solution)
- Note: solution requires source-synchronous clocking

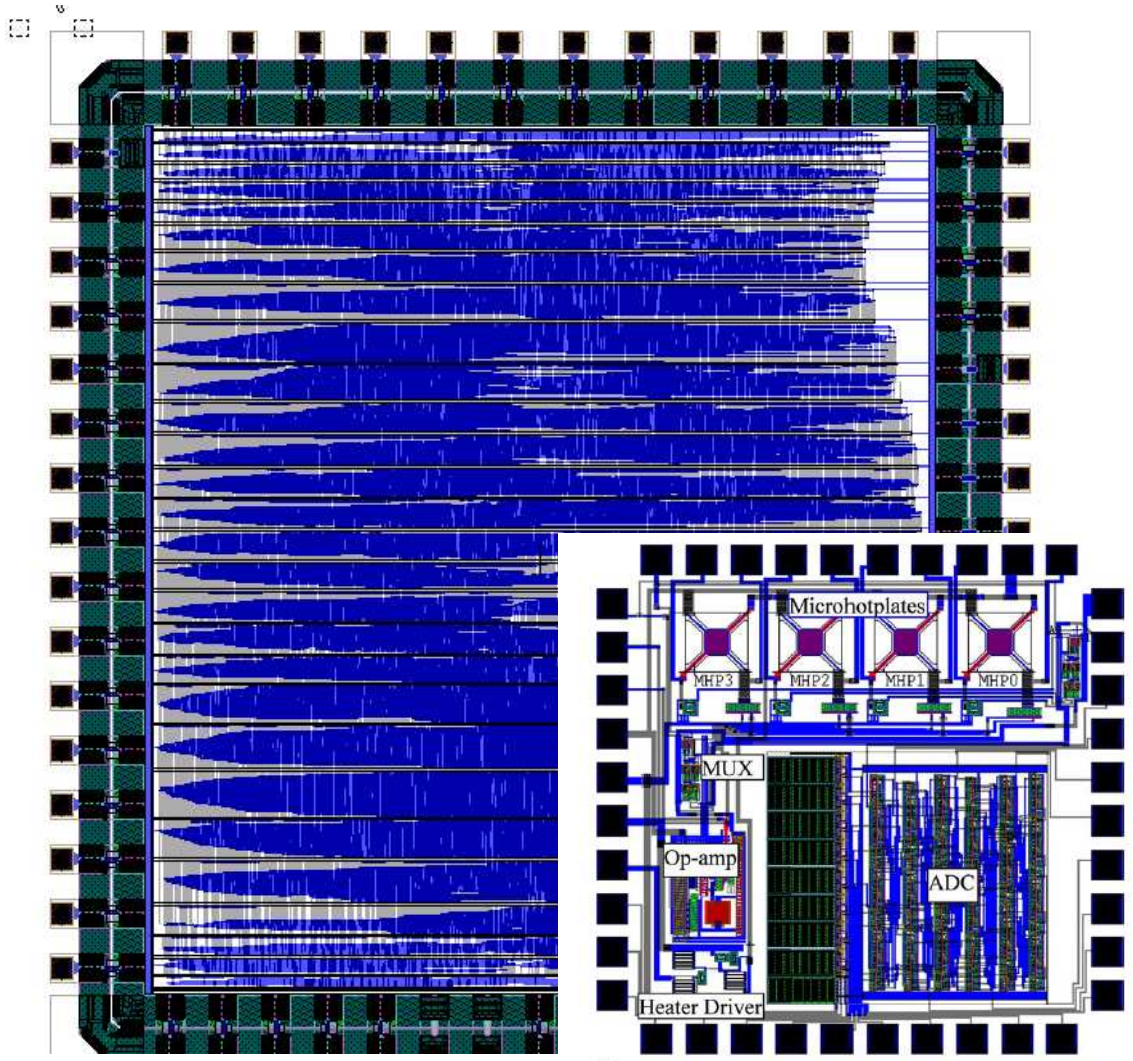
Interesting Side Note

Fully Buffered DIMM



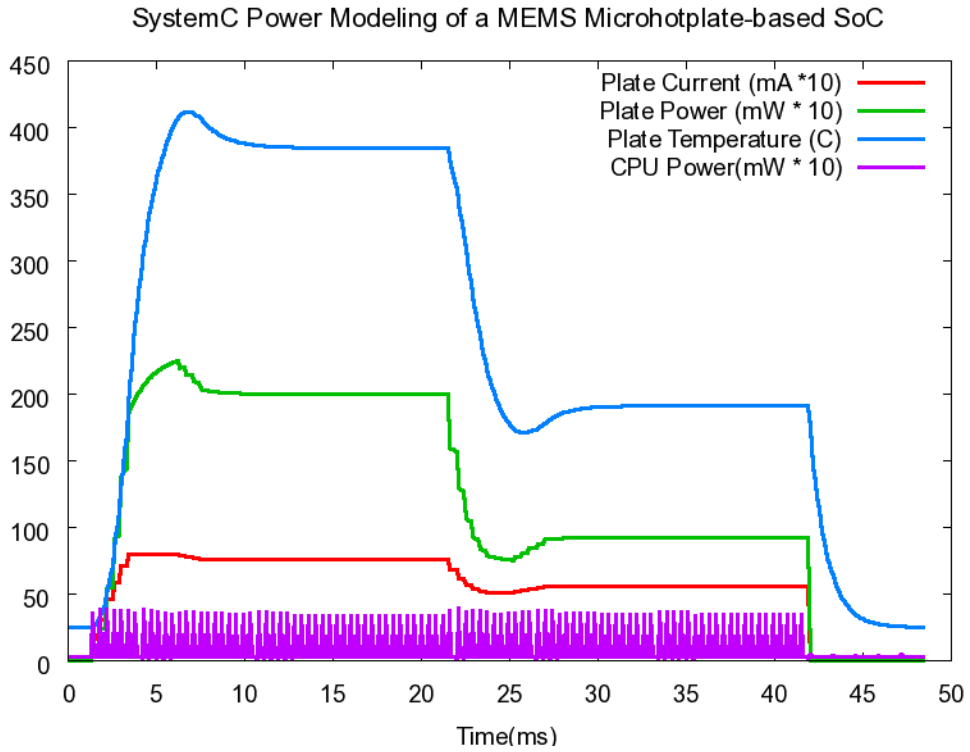
Nth Order Effects: Heat, EMI

EmPower: *First Target Application*



Nth Order Effects: Heat, EMI

EmPower: Initial Results



Summary

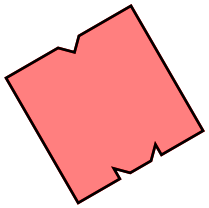
No longer appropriate to optimize subsystems in isolation: local optima do not yield globally optimal system

Systemic behaviors: unanticipated interactions yielding inefficiencies

Specific instances:

- **t_{FAW} + t_{RRD} + t_{DQS} severely limits BW**
- **Choice of DLL on DDR SDRAMs to de-skew parts**

Many problems can be addressed by system-level solutions; can be better than circuit-level solutions



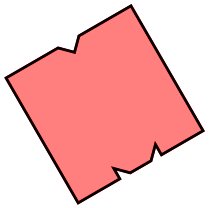
Et Cetera

(CURRENT) MEMSYS GRAD STUDENTS:

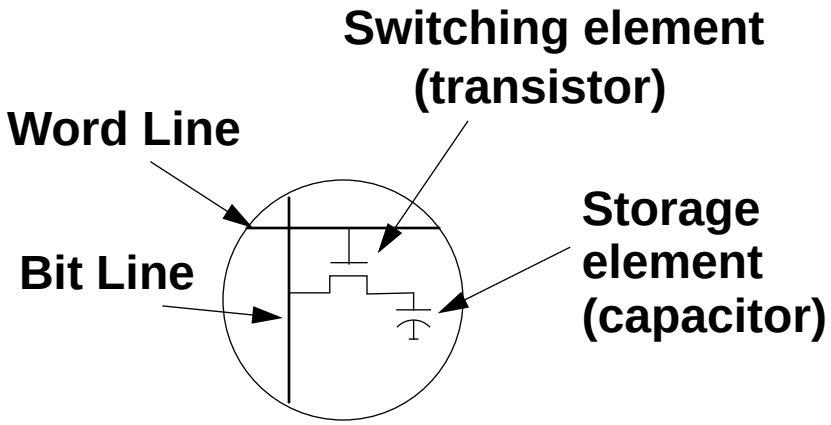
- **Dave Wang:** *DRAMsim*, t_{FAW} + t_{RRD} + t_{DQS} studies, etc.
- **Aamer Jaleel:** *CMP$im*, bioinformatics, etc.
- **Brinda Ganesh:** *DRAMsim*, FB-DIMM power mgmt
- **Samuel Rodriguez:** SRAM circuit-level details
- **Ankush Varma:** SystemC system-on-chip energy model
- **Sadagopan Srinivasan:** SoC memory system issues
- **Nuengwong Tuaycharoen:** *SYSim* development
- **Hongxia Wang:** SRAM circuit integrity issues
- **Joe Gross:** *DRAMsim II* development

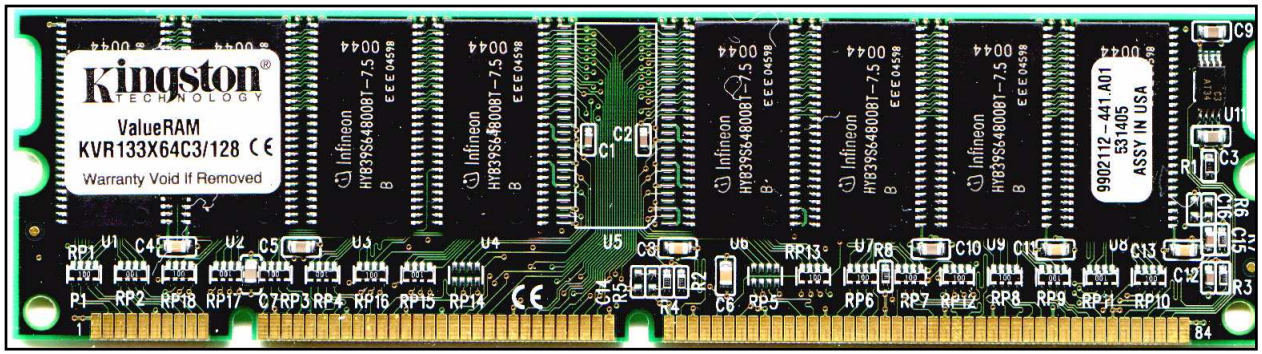
CONTACT INFO:

- **Prof. Bruce Jacob**
  ECE Dept., University of Maryland, College Park, MD
- **`www.ece.umd.edu/~blj/`**
  `blj@ece.umd.edu`



DRAM: Brief Primer

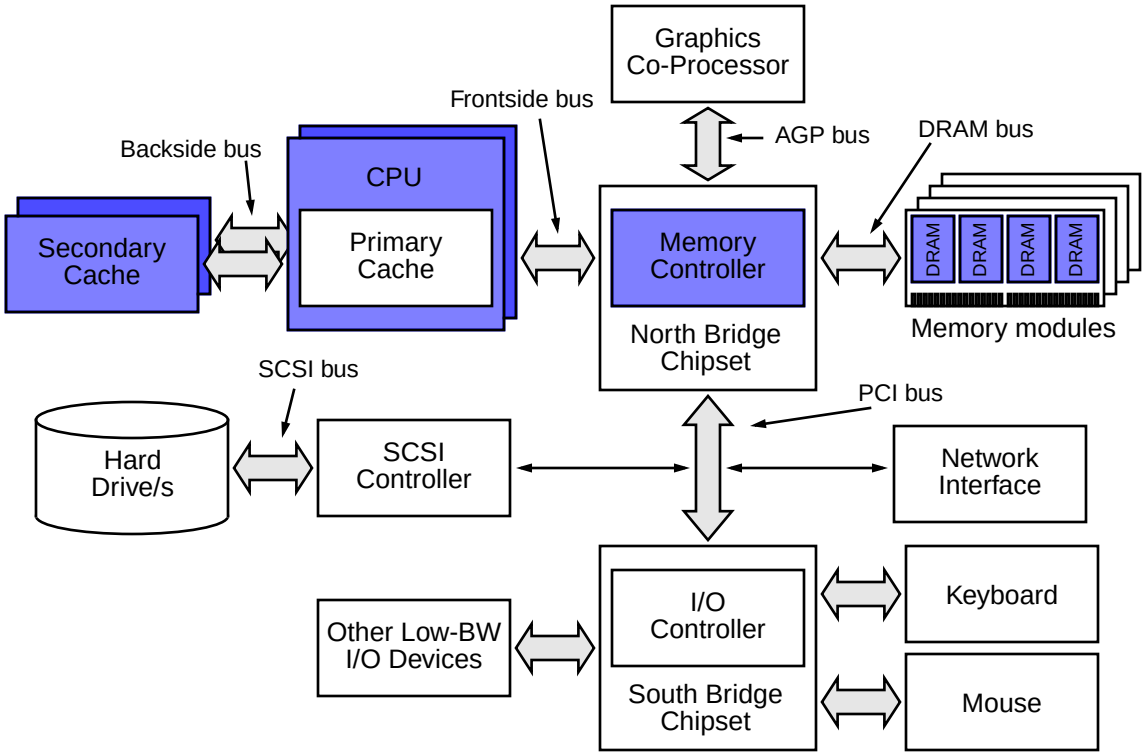




Dual In-line Memory Module (DIMM)

(printed circuit board w/ DRAM chips on it)

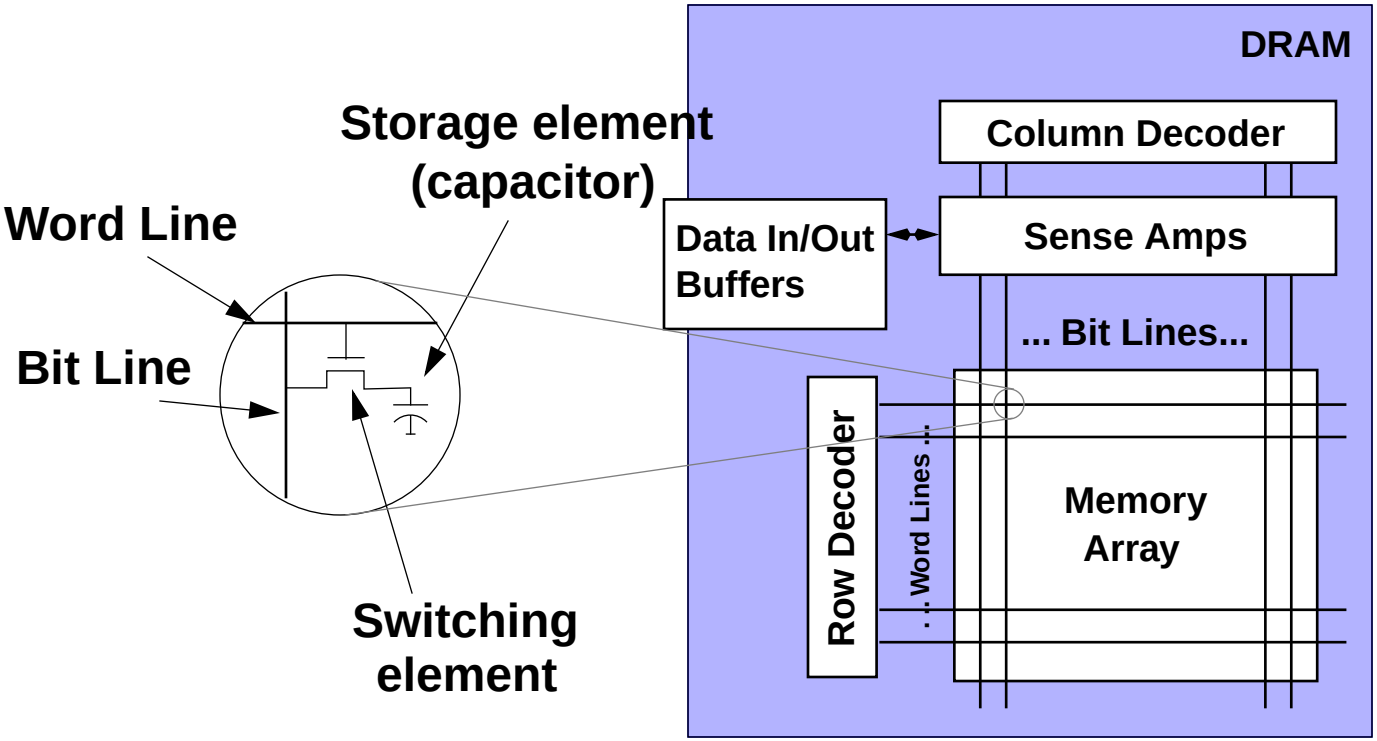
DRAM: Brief Primer



The memory system (in blue)
… and DRAM's typical place within it.

(typical PC-style desktop system)

DRAM: Brief Primer

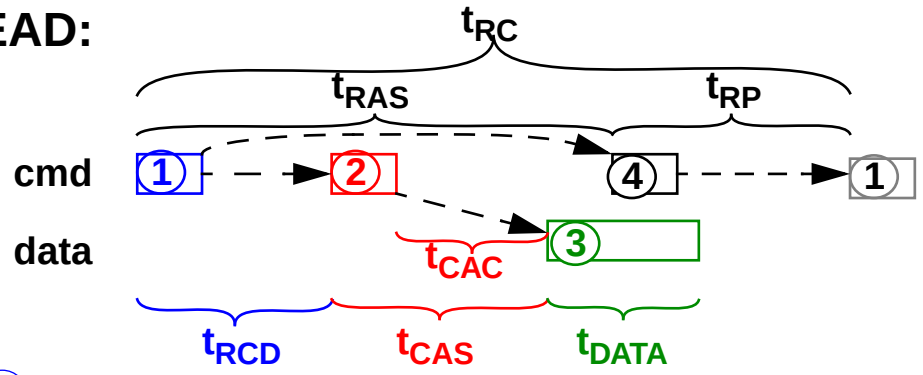


The DRAM device

DRAM: Brief Primer

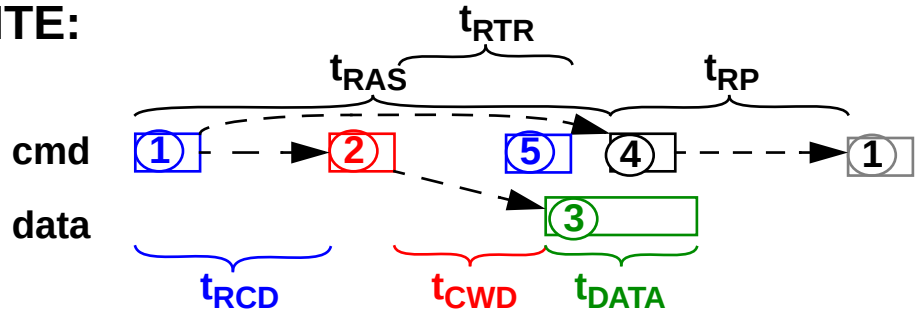
Access Protocol

READ:



1. Active: Open Row, t_{RCD} time later, a CAS command may be issued to the DRAM chip
2. CAS: Column Read command, t_{CAS} time later, data begins to be placed onto the Data bus. We use t_{CAC} to factor out command transmission time.
3. Data: The number of cycles that the data transmits over the Data bus
4. Precharge: Close the Row, this command may be issued t_{RAS} time after the Active command. After t_{RP} time, another active command may be issued.

WRITE:



2. CWD: Column Write Delay, the number of cycles that the controllers must wait before placing the data onto the data bus.
5. RTR: Retirement delay, this is for systems with write delay buffers.(RDRAM)