A Performance Comparison of Contemporary DRAM Architectures

Vinodh Cuppu\textsuperscript{1}, Bruce Jacob\textsuperscript{1}, Brian Davis\textsuperscript{2}, and Trevor Mudge\textsuperscript{2}

\textsuperscript{1} ECE Dept., Univ. Maryland, College Park
\textsuperscript{2} EECS Dept., Univ. Michigan, Ann Arbor

OUTLINE:
\begin{itemize}
  \item Motivation & Background
  \item Experiments
  \item Results
\end{itemize}

Dilemma: THIS ...

STATUS QUO in MEMORY-SYSTEM RESEARCH:

\begin{verbatim}
... if (memory_instruction(INSTR)) {
  if (L1_cache_miss( data_addr(INSTR) ){
    if (L2_cache_miss( data_addr(INSTR) ){
        cycles += DRAM_LATENCY;
    }
  }
}
...
\end{verbatim}
Motivation

HERE'S WHAT YOU MISS:

DRAM LATENCY:

DATA TRANSFER

OVERLAP

COLUMN ACCESS

ROW ACCESS

BUS TRANSMISSION
Goal

PRELIMINARY DRAM STUDY:

- Bus Transmission
- Row Access
- Column Access
- Data Transfer
- Bus Wait/Synch Time
- Stalls Due to Refresh
- The OVERLAP of These Components (with each other) (with CPU execution)

MODEL EXISTING TECHNOLOGY

DRAM Primer

BUS TRANSMISSION
A COMPARISON OF DRAM ARCHITECTURES

Bruce Jacob
University of Maryland

DRAM Primer

ROW ACCESS

<table>
<thead>
<tr>
<th>CPU</th>
<th>MEMORY CONTROLLER</th>
<th>BUS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data In/Out Buffers</th>
<th>Column Decoder</th>
<th>Sense Amps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row Decoder</th>
<th>Memory Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CPU</th>
<th>MEMORY CONTROLLER</th>
<th>BUS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data In/Out Buffers</th>
<th>Column Decoder</th>
<th>Sense Amps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row Decoder</th>
<th>Memory Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

A COMPARISON OF DRAM ARCHITECTURES

Bruce Jacob
University of Maryland

DRAM Primer

COLUMN ACCESS

<table>
<thead>
<tr>
<th>CPU</th>
<th>MEMORY CONTROLLER</th>
<th>BUS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data In/Out Buffers</th>
<th>Column Decoder</th>
<th>Sense Amps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row Decoder</th>
<th>Memory Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CPU</th>
<th>MEMORY CONTROLLER</th>
<th>BUS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data In/Out Buffers</th>
<th>Column Decoder</th>
<th>Sense Amps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row Decoder</th>
<th>Memory Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>
A COMPARISON OF DRAM ARCHITECTURES
Bruce Jacob
University of Maryland

DRAM Primer

DATA TRANSFER

note: page mode enables overlap with COL

DRAM Primer

BUS TRANSMISSION

note: overlapped component not shown
DRAM Primer

Read Timing for Conventional DRAM

![Diagram of Read Timing for Conventional DRAM]

DRAM Primer

Read Timing for Fast Page Mode DRAM

![Diagram of Read Timing for Fast Page Mode DRAM]
DRAM Primer

Read Timing for Extended Data Out DRAM

[A diagram showing the timing for Extended Data Out DRAM]

DRAM Primer

Read Timing for Synchronous DRAM

[A diagram showing the timing for Synchronous DRAM]
**DRAM Primer**

**Read Timing for Rambus DRAM**

**Simulator Overview**

**CPU: SimpleScalar v3.0a**
- 8-way out-of-order
- L1 cache: split 64K/64K, lockup free x32
- L2 cache: unified 1MB, lockup free x1
- L2 blocksize: 128 bytes

**Main Memory: 8 64Mb DRAMs**
- 100MHz/128-bit memory bus
- Optimistic open-page policy
  (close-immediately can be calculated)

Represents a “typical” workstation
DRAM Configurations

**FPM, EDO, SDRAM, ESDRAM:**

- CPU and caches
- 128-bit 100MHz bus
- Memory Controller
- Multiple P arallel Channels

**Rambus, Direct Rambus, SLDRAM:**

- CPU and caches
- 128-bit 100MHz bus
- Memory Controller
- Fast, Narrow Channel

*Note: TRANSFER WIDTH of Direct Rambus Channel*

- equals that of ganged FPM, EDO, etc.
- is 2x that of Rambus & SLDRAM

**Strawman: Rambus, etc.**

- CPU and caches
- 128-bit 100MHz bus
- Memory Controller
- Multiple Parallel Channels
A comparison of DRAM architectures

Bruce Jacob
University of Maryland

Overhead: Memory vs. CPU

Variable: speed of processor & caches

Definitions (var. on Burger, et al)

- \( t_{\text{PROC}} \) — processor with perfect memory
- \( t_{\text{REAL}} \) — realistic configuration
- \( t_{\text{BW}} \) — CPU with wide memory paths
- \( t_{\text{DRAM}} \) — time seen by DRAM system
Memory & CPU — PERL

![Graph showing cycles per instruction (CPI) for different DRAM configurations]

- **FPM**
- **EDO**
- **SDRAM**
- **ESDRAM**
- **DRDRAM**

**Legend:**
- Stalls due to Memory Bandwidth
- Stalls due to Memory Latency
- Overlap between Execution & Memory
- Processor Execution

**Note:** SLDRAM & RDRAM 2x data transfers
Ganged Rambus Channels

Cost-Performance

FPM, EDO, SDRAM, ESDRAM:
- Lower Latency => Wide/Fast Bus
- Increase Capacity => Decrease Latency
- Low System Cost

Rambus, Direct Rambus, SLDRAM:
- Lower Latency => Multiple Channels
- Increase Capacity => Increase Capacity
- High System Cost

1 DRDRAM = Multiple SDRAM
Conclusions

100MHz/128-bit Bus is **Current Bottleneck**
- Solution: Fast Bus/es & MC on CPU
  (e.g. Compaq Alpha, Sony Emotion, ...)

Current DRAMs Solving **Bandwidth Problem**
(but **not Latency Problem**)  

There is **Locality** in DRAM Accesses
(but **how important** is this?)

**SPECint ’95 Fits in 1MB Cache**

Future Work

**Improve Model:**
- DDR, DDR-II, MoSys, VCDRAM, etc.
- More realistic bus
  (scheduling, turnaround, etc.)
- Memory controller overhead
- Dual-bus latency vs. single-bus
  (include memory controller on CPU)

**Exploit DRAM Concurrency**

Large Systems (bandwidth or latency?)

Small Systems: DSP + IRAM = D-IRAM