Computer System Components

CPU Core
1 GHz - 3.2 GHz
4-way Superscaler
RISC or RISC-core (x86):
  - Deep Instruction Pipelines
  - Dynamic scheduling
  - Multiple FP, integer FUs
  - Dynamic branch prediction
  - Hardware speculation
CPU Core
1 GHz - 3.2 GHz
4-way Superscaler
RISC or RISC-core (x86):
  - Deep Instruction Pipelines
  - Dynamic scheduling
  - Multiple FP, integer FUs
  - Dynamic branch prediction
  - Hardware speculation

SDRAM
PC100/PC133
100-133MHz
64-128 bits wide
2-way interleaved
~ 900 MBYTES/SEC
64bit)
Current Standard
Double Date
Rate (DDR) SDRAM
PC3200
200 MHZ DDR
64-128 bits wide
4-way interleaved
~3.2 GBYTES/SEC
(one 64bit channel)
~6.4 GBYTES/SEC
(two 64bit channels)
RAMbus DRAM (RDRAM)
400MHz DDR
16 bits wide (32 banks)
~ 1.6 GBYTES/SEC

System Bus

Off or On-chip

L1
L2
L3

All Non-blocking caches
L1 16-128K 1-2 way set associative (on chip), separate or unified
L2 256K- 2M 4-32 way set associative (on chip) unified
L3 2-16M 8-32 way set associative (off or on chip) unified

Examples: Alpha, AMD K7: EV6, 200-400 MHz
Intel PII, PIII: GTL+ 133 MHz
Intel P4 800 MHz

Memory Controller

Memory Bus

Memory

Controllers

Disks Displays Keyboards

I/O Devices:

North Bridge

South Bridge

Chipset

I/O Buses

NICs

Example: PCI, 33-66MHz
32-64 bits wide
133-528 MBYTES/SEC
PCI-X 133MHz 64 bit
1024 MBYTES/SEC

Networks
The Memory Hierarchy

• The Motivation for The Memory Hierarchy:
  – CPU/Memory Performance Gap
  – The Principle Of Locality

→ Cache $$$$$$

• Cache Concepts:
  – Organization, Replacement, write strategies
  – Cache Performance Evaluation: Memory Access Tree
  – Multi-Level Caches

• Classification Steady-State Cache Misses: The Three C’s of cache Misses:

• Techniques To Improve Cache Performance:
  • Reduce Miss Rate
  • Reduce Cache Miss Penalty
  • Reduce Cache Hit Time

• Main Memory:
  – Performance Metrics: Latency & Bandwidth
    • Key DRAM Timing Parameters
  – DRAM System Memory Generations
  – Basic Memory Bandwidth Improvement Techniques

• Virtual Memory
  • Benefits, Issues/Strategies
  • Basic Virtual → Physical Address Translation: Page Tables
  • Speeding Up Address Translation: Translation Lookaside Buffer (TLB)
A Typical Memory Hierarchy

- **Processor**
  - Control
  - Datapath
    - Registers
    - On-Chip Level
      - One Cache
      - $L_1$
    - Second Level Cache
      - (SRAM)
      - $L_2$
  - Main Memory
    - (DRAM)
  - Virtual Memory, Secondary Storage
    - (Disk)
  - Tertiary Storage
    - (Tape)

**Speed (ns):**
- 1s
- 10s
- 100s
- 10,000,000s (10s ms)
- 10,000,000,000s (10s sec)

**Size (bytes):**
- 100s
- Ks
- Ms
- Gs
- Ts
Main Memory

- Main memory generally utilizes Dynamic RAM (DRAM), which use a single transistor to store a bit, but require a periodic data refresh by reading every row increasing cycle time.

- Static RAM may be used for main memory if the added expense, low density, high power consumption, and complexity is feasible (e.g. Cray Vector Supercomputers).

- Main memory performance is affected by:
  - **Memory latency**: Affects cache miss penalty, M. Measured by:
    - **Access time**: The time it takes between a memory access request is issued to main memory and the time the requested information is available to cache/CPU.
    - **Cycle time**: The minimum time between requests to memory (greater than access time in DRAM to allow address lines to be stable)
  - **Memory bandwidth**: The maximum sustained data transfer rate between main memory and cache/CPU.

(In Chapter 5.8 - 5.10)
Four Key DRAM Timing Parameters

• **$t_{\text{RAC}}$:** Minimum time from RAS (Row Access Strobe) line falling to the valid data output.
  - Sometimes quoted as the nominal speed of a DRAM chip
  - For a typical 64Mb DRAM $t_{\text{RAC}} = 60$ ns

• **$t_{\text{RC}}$:** Minimum time from the start of one row access to the start of the next (memory cycle time).
  - $t_{\text{RC}} = t_{\text{RAC}} + \text{RAS Precharge Time}$
  - $t_{\text{RC}} = 110$ ns for a 64Mbit DRAM with a $t_{\text{RAC}}$ of 60 ns

• **$t_{\text{CAC}}$:** Minimum time from CAS (Column Access Strobe) line falling to valid data output.
  - 12 ns for a 64Mbit DRAM with a $t_{\text{RAC}}$ of 60 ns

• **$t_{\text{PC}}$:** Minimum time from the start of one column access to the start of the next.
  - $t_{\text{PC}} = t_{\text{CAC}} + \text{CAS Precharge Time}$
  - About 25 ns for a 64Mbit DRAM with a $t_{\text{RAC}}$ of 60 ns
Simplified DRAM Speed Parameters

- **Row Access Strobe (RAS) Time**: (similar to $t_{RAC}$):
  - Minimum time from RAS (Row Access Strobe) line falling to the first valid data output.
  - A major component of memory latency.
  - Only improves ~ 5% every year.

- **Column Access Strobe (CAS) Time/data transfer time**: (similar to $t_{CAC}$)
  - The minimum time required to read additional data by changing column address while keeping the same row address.
  - Along with memory bus width, determines peak memory bandwidth.
# DRAM Generations

<table>
<thead>
<tr>
<th>Year</th>
<th>Size</th>
<th>RAS (ns)</th>
<th>CAS (ns)</th>
<th>Cycle Time</th>
<th>Memory Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1980</td>
<td>64 Kb</td>
<td>150-180</td>
<td>75</td>
<td>250 ns</td>
<td>Page Mode</td>
</tr>
<tr>
<td>1983</td>
<td>256 Kb</td>
<td>120-150</td>
<td>50</td>
<td>220 ns</td>
<td>Page Mode</td>
</tr>
<tr>
<td>1986</td>
<td>1 Mb</td>
<td>100-120</td>
<td>25</td>
<td>190 ns</td>
<td></td>
</tr>
<tr>
<td>1989</td>
<td>4 Mb</td>
<td>80-100</td>
<td>20</td>
<td>165 ns</td>
<td>Fast Page Mode</td>
</tr>
<tr>
<td>1992</td>
<td>16 Mb</td>
<td>60-80</td>
<td>15</td>
<td>120 ns</td>
<td>EDO</td>
</tr>
<tr>
<td>1996</td>
<td>64 Mb</td>
<td>50-70</td>
<td>12</td>
<td>110 ns</td>
<td>PC66 SDRAM</td>
</tr>
<tr>
<td>1998</td>
<td>128 Mb</td>
<td>50-70</td>
<td>10</td>
<td>100 ns</td>
<td>PC100 SDRAM</td>
</tr>
<tr>
<td>2000</td>
<td>256 Mb</td>
<td>45-65</td>
<td>7</td>
<td>90 ns</td>
<td>PC133 SDRAM</td>
</tr>
<tr>
<td>2002</td>
<td>512 Mb</td>
<td>40-65</td>
<td>5</td>
<td>80 ns</td>
<td>PC2700 DDR SDRAM</td>
</tr>
</tbody>
</table>

8000:1 (Capacity) 15:1 (~bandwidth) 3:1 (Latency)
Simplified Asynchronous DRAM Read Timing

Memory Cycle Time = \( t_{RC} = t_{RAC} + \) RAS Precharge Time

- \( t_{RC} \): Minimum time from RAS (Row Access Strobe) line falling to the valid data output.
- \( t_{RAC} \): Minimum time from the start of one row access to the start of the next (memory cycle time).
- \( t_{CAC} \): Minimum time from CAS (Column Access Strobe) line falling to valid data output.
- \( t_{PC} \): Minimum time from the start of one column access to the start of the next.

Source: http://arstechnica.com/eaedia/r/ram_guide/ram_guide.part2-1.html
Page Mode DRAM

**Regular DRAM Organization:**
- N rows x N column x M-bit
- Read & Write M-bit at a time
- Each M-bit access requires a RAS / CAS cycle

**Fast Page Mode DRAM**
- N x M “register" to save a row

---

Diagram illustrating the memory access sequence with RAS_L, CAS_L, Row Address, Col Address, and Junk.
Fast Page Mode DRAM

- Fast Page Mode DRAM
  - \(N \times M\) “SRAM” to save a row

- After a row is read into the register
  - Only CAS is needed to access other M-bit blocks on that row
  - RAS_L remains asserted while CAS_L is toggled

A read burst of length 4 shown
Simplified Asynchronous Fast Page Mode (FPM) DRAM Read Timing

Typical timing at 66 MHz : 5-3-3-3 (burst of length 4)
For bus width = 64 bits = 8 bytes  cache block size = 32 bytes
It takes = 5+3+3+3 = 14 memory cycles or 15 ns x 14 = 210 ns to read 32 byte block.
Read Miss penalty for CPU running at 1 GHz = M = 15 x 14 = 210 CPU cycles
Simplified Asynchronous Extended Data Out (EDO) DRAM Read Timing

- Extended Data Out DRAM operates in a similar fashion to Fast Page Mode DRAM except putting data from one read on the output pins at the same time the column address for the next read is being latched in.

**EDO Read**

**EDO DRAM speed rated using tRAC ~ 40-60ns**

Typical timing at 66 MHz: 5-2-2-2 (burst of length 4)

For bus width = 64 bits = 8 bytes  
Max. Bandwidth = 8 x 66 / 2 = 264 Mbytes/sec

It takes = 5+2+2+2 = 11 memory cycles  or  
15 ns x 11 = 165 ns to read 32 byte cache block

Minimum Read Miss penalty for CPU running at 1 GHz = M = 11 x 15 = 165 CPU cycles

Source: http://arstechnica.com/paedia/r/ram_guide/ram_guide.part2-1.html
# Synchronous DRAM Interface Characteristics Summary

<table>
<thead>
<tr>
<th></th>
<th>SDRAM</th>
<th>DDR (Double Data Rate) SDRAM</th>
<th>RAMbus</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Potential Bandwidth</strong></td>
<td>0.8 GB/s</td>
<td>2.133 GB/s (similar to PC3200)</td>
<td>1.6 GB/s</td>
</tr>
<tr>
<td><strong>Interface Signals</strong></td>
<td>64(72) data, 168 pins</td>
<td>64(72) data, 168 pins</td>
<td>16(18) data, 184 pins</td>
</tr>
<tr>
<td><strong>Interface Frequency</strong></td>
<td>100 MHz</td>
<td>133 MHz</td>
<td>400 MHz</td>
</tr>
<tr>
<td><strong>Latency Range</strong></td>
<td>30-90 nS</td>
<td>18.8-64 nS</td>
<td>17.5-42.6 nS</td>
</tr>
</tbody>
</table>

| **# of Banks per DRAM Chip** | 2 | 4 | 4? | 32 |

The latencies given only account for memory module latency and do not include memory controller latency.
Synchronous Dynamic RAM, SDRAM (mid 90s)

Organization

SDRAM speed is rated at max.
clock speed supported:
100MHZ = PC100
133MHZ = PC133

DDR SDRAM (late 90s - current)

organization is similar but four banks are used in each DDR SDRAM chip instead of two.

Data transfer on both rising and falling edges of the clock

DDR SDRAM rated by maximum memory bandwidth
PC3200 = 8 bytes x 200 MHz x 2
= 3200 Mbytes/sec
### Simplified SDRAM/DDR SDRAM Read Timing

#### SDRAM

**Typical timing at 133 MHz (PC133 SDRAM):** 5-1-1-1  
For bus width = 64 bits = 8 bytes  
Max. Bandwidth = 133 x 8 = 1064 Mbytes/sec  
It takes = 5+1+1+1 = 8 memory cycles or 7.5 ns x 8 = 60 ns to read 32 byte cache block  
Minimum Read Miss penalty for CPU running at 1 GHz =  
\[ M = 7.5 \times 6 = 45 \text{ CPU cycles} \]

#### DDR SDRAM:

**Possible timing at 133 MHz (DDR x2):** 5 - .5 - .5 - .5  
(PC2100 DDR SDRAM)  
For bus width = 64 bits = 8 bytes  
Max. Bandwidth = 133 x 2 x 8 = 2128 Mbytes/sec  
It takes = 5 + .5 + .5 + .5 = 6.5 memory cycles or 7.5 ns x 8 = 45 ns to read 32 byte cache block  
Minimum Read Miss penalty for CPU running at 1 GHz =  
\[ M = 7.5 \times 6 = 49 \text{ CPU cycles} \]
Basic Memory Bandwidth Improvement Techniques

• **Wider Main Memory:**
  Memory width is increased to a number of words (usually up to the size of a cache block).
  ⇒ Memory bandwidth is proportional to memory width.
  e.g. Doubling the width of cache and memory doubles potential memory bandwidth available to the CPU.

• **Interleaved (Multi-Bank) Memory:**
  Memory is organized as a number of independent banks.
  – Multiple interleaved memory reads or writes are accomplished by sending memory addresses to several memory banks at once.
  – **Interleaving factor:** Refers to the mapping of memory addressees to memory banks. Goal reduce bank conflicts.
  e.g. using 4 banks (width one word), bank 0 has all words whose address is:
    
    \[(\text{word address mod} \ 4) = 0\]
Three examples of bus width, memory width, and memory interleaving to achieve higher memory bandwidth

Simplest design:
Everything is the width of one word (lowest performance)

Wider memory, bus and cache (highest performance)

Narrow bus and cache with interleaved memory banks
Memory Bank Interleaving

Access Pattern without Interleaving: (One Bank)

- D1 available
- Start Access for D1
- Start Access for D2

Access Pattern with 4-way Interleaving:

- Bank Cycle Time
- Access Bank 0
- Access Bank 1
- Access Bank 2
- Access Bank 3

We can Access Bank 0 again

Number of banks ≥ Number of cycles to access word in a bank
Memory Width, Interleaving: Performance Example

Given the following system parameters with single unified cache level L1 (ignoring write policy):

- Block size = 1 word
- Memory bus width = 1 word
- Miss rate = 3%
- Miss penalty = 32 cycles
  
  (4 cycles to send address, 24 cycles access time, 4 cycles to send a word)

Memory access/instruction = 1.2
CPI_{execution} (ignoring cache misses) = 2

Miss rate (block size = 2 word = 8 bytes) = 2%
Miss rate (block size = 4 words = 16 bytes) = 1%

- The CPI of the base machine with 1-word blocks = 2 + (1.2 x 0.03 x 32) = 3.15

Increasing the block size to two words gives the following CPI:

- 32-bit bus and memory, no interleaving, M = 2 x 32 = 64 cycles
  CPI = 2 + (1.2 x .02 x 64) = 3.54
- 32-bit bus and memory, interleaved, M = 4 + 24 + 8 = 36 cycles
  CPI = 2 + (1.2 x .02 x 36) = 2.86
- 64-bit bus and memory, no interleaving, M = 32 cycles
  CPI = 2 + (1.2 x 0.02 x 32) = 2.77

Increasing the block size to four words; resulting CPI:

- 32-bit bus and memory, no interleaving, M = 4 x 32 = 128 cycles
  CPI = 2 + (1.2 x 0.01 x 128) = 3.54
- 32-bit bus and memory, interleaved, M = 4 + 24 + 16 = 44 cycles
  CPI = 2 + (1.2 x 0.01 x 44) = 2.53
- 64-bit bus and memory, no interleaving, M = 2 x 32 = 64 cycles
  CPI = 2 + (1.2 x 0.01 x 64) = 2.77
- 64-bit bus and memory, interleaved, M = 4 + 24 + 8 = 36 cycles
  CPI = 2 + (1.2 x 0.01 x 36) = 2.43
- 128-bit bus and memory, no interleaving, M = 32 cycles
  CPI = 2 + (1.2 x 0.01 x 32) = 2.38
Three-Level Cache Example

- CPU with $CPI_{\text{execution}} = 1.1$ running at clock rate = 500 MHz
- 1.3 memory accesses per instruction.
- $L_1$ cache operates at 500 MHz with a miss rate of 5%
- $L_2$ cache operates at 250 MHz with a local miss rate 40%, $(T_2 = 2$ cycles$)$
- $L_3$ cache operates at 100 MHz with a local miss rate 50%, $(T_3 = 5$ cycles$)$
- Memory access penalty, $M= 100$ cycles. Find CPI.

With No Cache, $CPI = 1.1 + 1.3 \times 100 = 131.1$

With single $L_1$, $CPI = 1.1 + 1.3 \times 0.05 \times 100 = 7.6$

With $L_1$, $L_2$ $CPI = 1.1 + 1.3 \times (0.05 \times 0.6 \times 2 + 0.05 \times 0.4 \times 100) = 3.778$

$$CPI = CPI_{\text{execution}} + \text{Mem Stall cycles per instruction}$$

Mem Stall cycles per instruction = Mem accesses per instruction $\times$ Stall cycles per access

Stall cycles per memory access $= (1-H_1) \times H_2 \times T_2 + (1-H_1) \times (1-H_2) \times H_3 \times T_3 + (1-H_1)(1-H_2) (1-H_3)x M$

$= 0.05 \times 0.6 \times 2 + 0.05 \times 0.4 \times 0.5 \times 5 + 0.05 \times 0.4 \times 0.5 \times 100$

$= 0.097 + 0.0075 + 0.00225 = 1.11$

$$CPI = 1.1 + 1.3 \times 1.11 = 2.54$$

Speedup compared to $L_1$ only $= 7.6/2.54 = 3$

Speedup compared to $L_1$, $L_2$ $= 3.778/2.54 = 1.49$

Repeated here from lecture 8
3-Level (All Unified) Cache Performance
Memory Access Tree (Ignoring Write Policy)
CPU Stall Cycles Per Memory Access

CPU Memory Access

L1 Hit:
Stalls = H1 x 0 = 0
(No Stall)

L1 Miss:
% = (1-H1)

L2 Hit:
(1-H1) x H2 x T2

L2 Miss:
% = (1-H1)(1-H2)

L3 Hit:
(1-H1) x (1-H2) x H3 x T3

L3 Miss:
(1-H1)(1-H2)(1-H3) x M

Stall cycles per memory access
= (1-H1) x H2 x T2 + (1-H1) x (1-H2) x H3 x T3 + (1-H1)(1-H2)(1-H3)x M

AMAT = 1 + Stall cycles per memory access

Repeated here from lecture 8
Program Steady-State Main Memory Bandwidth-Usage Example

- In the previous example with three levels of cache (all unified, ignore write policy)
- CPU with \( \text{CPI}_{\text{execution}} = 1.1 \) running at clock rate = 500 MHz
- 1.3 memory accesses per instruction.
- \( L_1 \) cache operates at 500 MHz with a miss rate of 5%
- \( L_2 \) cache operates at 250 MHz with a local miss rate 40\%, \( (T_2 = 2 \text{ cycles}) \)
- \( L_3 \) cache operates at 100 MHz with a local miss rate 50\%, \( (T_3 = 5 \text{ cycles}) \)
- Memory access penalty, \( M = 100 \text{ cycles} \) (to deliver 32 bytes to CPU)

- We found the CPI:
  - With No Cache, \( \text{CPI} = 1.1 + 1.3 \times 100 = 131.1 \)
  - With single \( L_1 \), \( \text{CPI} = 1.1 + 1.3 \times .05 \times 100 = 7.6 \)
  - With \( L_1, L_2 \) \( \text{CPI} = 1.1 + 1.3 \times (.05 \times .6 \times 2 + .05 \times .4 \times 100) = 3.778 \)
  - With \( L_1, L_2, L_3 \) \( \text{CPI} = 1.1 + 1.3 \times 1.11 = 2.54 \)

Assuming:
- instruction size = data size = 4 bytes, all cache blocks are 32 bytes

For each of the three cases with cache:
  - What is the total number of memory accesses generated by the CPU per second?
  - What is the percentage of these memory accesses satisfied by main memory?
  - Percentage of main memory bandwidth used by the CPU?
Program Steady-State Main Memory Bandwidth-Usage Example

- Memory requires 100 CPU cycles = 200 ns to deliver 32 bytes, thus total main memory bandwidth = 32 bytes / (200 ns) = 160 x 10^6 bytes/sec

- The total number of memory accesses generated by the CPU per second = (memory access/instruction) x clock rate / CPI = 1.3 x 500 x 10^6 / CPI = 650 x 10^6 / CPI
  - With single L1 = 650 x 10^6 / 7.6 = 85 x 10^6 accesses/sec
  - With L1, L2 = 650 x 10^6 / 3.778 = 172 x 10^6 accesses/sec
  - With L1, L2, L3 = 650 x 10^6 / 2.54 = 255 x 10^6 accesses/sec

- The percentage of these memory accesses satisfied by main memory:
  - With single L1 = L1 miss rate = 5%
  - With L1, L2 = L1 miss rate x L2 miss rate = .05 x .4 = 2%
  - with L1, L2, L3 = L1 miss rate x L2 miss rate x L3 miss rate = .05 x .4 x .5 = 1%

- Memory Bandwidth used
  - With single L1 = 32 bytes x 85x10^6 accesses/sec x .05 = 136 x10^6 bytes/sec
    or 136/160 = 85% of total memory bandwidth
  - With L1, L2 = 32 bytes x 172x10^6 accesses/sec x .02 = 110 x10^6 bytes/sec
    or 110/160 = 69% of total memory bandwidth
  - With L1, L2, L3 = 32 bytes x 255x10^6 accesses/sec x .01 = 82 x10^6 bytes/sec
    or 82/160 = 51% of total memory bandwidth
Dual (64-bit) Channel PC3200 DDR SDRAM has a theoretical peak bandwidth of

400 Mhz x 8 bytes x 2 = 6400 MB/s

Is memory bandwidth still an issue?

Source: The Tech Report 1-21-2004
PC3200 DDR SDRAM has a theoretical latency range of 18-40 ns (not accounting for memory controller latency or other address/data line delays).

Is memory latency still an issue?

X86 CPU Cache/Memory Performance Example:
AMD Athlon XP/64/FX Vs. Intel P4/Extreme Edition

Main Memory: Dual (64-bit) Channel PC3200 DDR SDRAM
peak bandwidth of 6400 MB/s

Source: The Tech Report 1-21-2004

Source: The Tech Report 1-21-2004
AMDAthlon T-Bird Vs. Intel PIII (Just for historic purposes:)

AMD Athlon T-Bird 1GHz
L1: 64K INST, 64K DATA (3 cycle latency), both 2-way
L2: 256K 16-way 64 bit bus
  Latency: 7 cycles
L1,L2 on-chip

Intel PIII GHz
L1: 16K INST, 16K DATA (3 cycle latency), both 2-way
L2: 256K 8-way 256 bit, Latency: 7 cycles
L1,L2 on-chip (32 byte blocks)

Main Memory:

PC2100
133MHZ DDR SDRAM 64bit
Peak bandwidth: 2100 MB/s
Latency Range: 19ns - 64ns

PC133
133MHZ SDRAM 64bit
Peak bandwidth: 1000 MB/s
Latency Range: 25ns - 80ns

PC800
Rambus DRDRAM
400 MHZ DDR 16-bit
Peak bandwidth: 1600 MB/s
(1 channel)
Latency Range: 35ns - 80ns


Intel 840 uses two PC800 channels
X86 CPU Cache/Memory Performance Example:
AMD Athlon T-Bird Vs. Intel PIII, Vs. P4

AMD Athlon T-Bird 1GHZ
L1: 64K INST, 64K DATA (3 cycle latency),
    both 2-way
L2: 256K 16-way 64 bit bus
    Latency: 7 cycles
    L1,L2 on-chip

Intel P 4, 1.5 GHZ
L1: 8K DATA (2 cycle latency)
    4-way 64 byte blocks
    96KB Execution Trace Cache
L2: 256K 8-way 256 bit bus, 128 byte blocks
    Latency: 7 cycles
    L1,L2 on-chip

Intel PIII 1 GHZ
L1: 16K INST, 16K DATA (3 cycle latency)
    both 2-way 32 byte blocks
L2: 256K 8-way 256 bit bus,
    Latency: 7 cycles
    L1,L2 on-chip