Review: Common Framework for Memory Hierarchy

° Question 1: Where can a Block (Page) be Placed (Block Placement)
  • Cache:
    - direct mapped, n-way set associative
  • VM:
    - fully associative
° Question 2: How is a block (Page) found (Block Identification)
  • index,
  • index the set and search among elements
  • search all cache entries or separate lookup table
° Question 3: Which block (Page) be replaced (Block Replacement)
  • Random, LRU, LFU, NRU (Not-Recently-Used)
° What happens on a write (Write Strategy)
  • In case hit: write through vs. write back
  • In case miss: write allocate vs. write no-allocate on a write miss
The Big Picture: Where are We Now?

Today's Topic: I/O Systems

- Network
- Control
- Datapath
- Memory
- Processor
- Input
- Output

I/O System Design Issues

- Dependability
- Expandability and Diversity
- Performance and Cost

- Processor
- Cache
- Memory - I/O Bus
- Main Memory
- I/O Controller
- Disk
- I/O Controller
- Graphics
- Network
Example: Pentium System Organization

Types and Characteristics of I/O Devices

- Behavior: how does an I/O device behave?
  - Input: read only
  - Output: write only, cannot read
  - Storage: can be reread and usually rewritten

- Partner:
  - Either a human or a machine is at the other end of the I/O device
  - Either feeding data on input or reading data on output

- Data rate:
  - The peak rate at which data can be transferred:
    - Between the I/O device and the main memory
    - Or between the I/O device and the CPU
I/O Device Examples

<table>
<thead>
<tr>
<th>Device</th>
<th>Behavior</th>
<th>Partner</th>
<th>Data Rate (Mbit/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keyboard</td>
<td>Input</td>
<td>Human</td>
<td>0.0001</td>
</tr>
<tr>
<td>Mouse</td>
<td>Input</td>
<td>Human</td>
<td>0.0038</td>
</tr>
<tr>
<td>Sound input</td>
<td>Input</td>
<td>Machine</td>
<td>3.00</td>
</tr>
<tr>
<td>Laser Printer</td>
<td>Output</td>
<td>Human</td>
<td>3.20</td>
</tr>
<tr>
<td>Graphics Display</td>
<td>Output</td>
<td>Human</td>
<td>800 - 8000</td>
</tr>
<tr>
<td>Network / LAN</td>
<td>Input or Output</td>
<td>Machine</td>
<td>100 - 1000</td>
</tr>
<tr>
<td>Network / Wi-LAN</td>
<td>Input or Output</td>
<td>Machine</td>
<td>11 – 54</td>
</tr>
<tr>
<td>Optical Disk</td>
<td>Storage</td>
<td>Machine</td>
<td>80.0</td>
</tr>
<tr>
<td>Magnetic Disk</td>
<td>Storage</td>
<td>Machine</td>
<td>240 - 2560</td>
</tr>
</tbody>
</table>

I/O System Performance

° I/O System performance depends on many aspects of the system:
  • The CPU
  • The memory system:
    - Internal and external caches
    - Main Memory
  • The underlying interconnection (buses)
  • The I/O controller
  • The I/O device
  • The speed of the I/O software
  • The efficiency of the software’s use of the I/O devices

° Two common performance metrics:
  • Throughput: I/O bandwidth
  • Response time: Latency

° Foundations of queueing theory
Producer-Server Model

Throughput:
- The number of tasks completed by the server in unit time
- In order to get the highest possible throughput:
  - The server should never be idle
  - The queue should never be empty

Response time:
- Begins when a task is placed in the queue
- Ends when it is completed by the server
- In order to minimize the response time:
  - The queue should be empty
  - The server will be idle

Throughput versus Response Time

- The knee of the curve is a little more throughput results in much longer response time; M/D/1, M/M/1, and M/G/1 queueing, etc.
Throughput Enhancement

Throughput can be improved by:
- Throwing more hardware at the problem

Response time is much harder to reduce:
- Ultimately it is limited by the speed of light
- You cannot bribe God!

I/O Benchmarks for Magnetic Disks

Supercomputer application:
- Examples: large-scale scientific problems
- I/O dominated by access to large files on magnetic disks
- The overriding supercomputer I/O measures is data throughput:
  - Date rate: bytes/second that can be transferred between disk and memory

Transaction processing:
- Examples: Airline reservations systems and bank ATMs
- A lot of small changes to a large body of shared data (Between 2 and 10 disk I/Os)
- concerned with I/O rate: the number of disk accesses per second

File system:
- Example: UNIX file system
- 80% of accesses are to files less than 10 KB
- 67% of the accesses are reads, 27% of the accesses are writes
- Concerned with creation of synthetic file system benchmarks
What Aspects Are Most Important

- Desktop, server, embedded computers:
  - I/O dependability and Cost

- Desktop and Embedded systems:
  - Response time and diversity of I/O devices

- Server systems:
  - Throughput and expandability of I/O devices

Magnetic Disks

- Purpose and characteristics:
  - Long term, nonvolatile storage
  - Large, inexpensive, and slow
  - Lowest level in the memory hierarchy

- Two major types:
  - Floppy disk
  - Hard disk

- Both types of disks:
  - Rely on a rotating platter coated with a magnetic surface
  - Use a moveable read/write head to access the disk

- Advantages of hard disks over floppy disks:
  - Platters are more rigid (metal or glass) so they can be larger
  - Higher density because it can be controlled more precisely
  - Higher data rate because it spins faster
  - Can incorporate more than one platter
Organization of a Hard Magnetic Disk

- A stack of platters, a surface with a magnetic coating

- Typical numbers (depending on the disk size):
  - 10,000 to 50,000 tracks per surface
  - 100 to 500 sectors per track
  - A sector (512+ B) is the smallest unit that can be read or written

- Originally, all tracks have the same number of sectors; today:
  - "Constant" bit density: ZBR records more sectors on the outer tracks

Device Controllers

- I/O devices have two components:
  - mechanical component
  - electronic component

- The electronic component is the device controller
  - may be able to handle multiple but identical devices

- Controller's tasks
  - convert serial bit stream to block of bytes
  - perform error correction as necessary
  - make blocks available to main memory

Standard interface: IDE, SCSI
Magnetic Disk Characteristic

- Disk head: each side of a platter has separate disk head
- Cylinder: all the tracks under the head at a given point on all surface
- Read/write data is a three-stage process:
  - Seek time: position the arm over the proper track
  - Rotational latency: wait for the desired sector to rotate under the read/write head
  - Transfer time: transfer a block of bits (sector) under the read-write head
- Average seek time as reported by the industry:
  - Typically in the range of 8 ms to 15 ms
  - (Sum of the time for all possible seek) / (total # of possible seeks)
- Due to locality of disk reference, actual average seek time may:
  - Only be 25% to 33% of the advertised number
  - Locality: successive access to the same file and OS disk scheduling
- Read ahead: read more than requested into disk buffer/cache (MBs), taking spatial locality, so as to amortize the long access
Typical Numbers of a Magnetic Disk

- **Rotational Latency:**
  - Most disks rotate at 5400 – 15,000 RPM
  - Approximately 16 ms per revolution
  - An average latency to the desired sector is halfway around the disk: 8 ms

- **Transfer Time** is a function of:
  - Transfer size (usually a sector): 1 KB / sector
  - Rotation speed: 5400 RPM to 7200 RPM to ……
  - Recording density: typical diameter ranges from 1.8 to 14 in
  - Typical data rate values: 30 to 80 MB/sec
    - Disk cache data rate can go to 320 MB/sec

Disk I/O Performance

- **Disk Access Time** = Seek time + Rotational Latency + Transfer time + Controller Time + Queueing Delay

- **Estimating Queue Length:**
  - Utilization = \( U = \frac{\text{Request Rate}}{\text{Service Rate}} = \frac{\lambda}{\mu} \)
  - Mean Queue Length = \( U / (1 - U) \)
  - As Request Rate (\( \lambda \)) \( \to \) Service Rate (\( \mu \))
    - Mean Queue Length \( \to \) Infinity
Example of Disk Performance

° Question:

512 byte sector, rotate at 10,000 RPM, advertised seeks is 6 ms, transfer rate is 50 MB/sec, controller overhead is 0.2 ms, queue idle so no service time, calculate the disk access time and data rate

° Disk Access Time = Seek time + Rotational Latency + Transfer time + Controller Time + Queueing Delay

° Disk Access Time = 6 ms + 0.5R / 10000 RPM + 0.5 KB / 50 MB/s + 0.2 ms
  = 6 ms + 3 ms + 0.01 ms + 0.2
  = 9.2 ms

° Data rate = 512 B / 9.2 ms = 55 KB/sec

° If real seeks are 1/4 advertised seeks, then access time is 4.7 ms and data rate 110 KB/sec, with rotational delay greater than 60% of the time!

° What is a good block size for reading?
  • A trade-off between space utilization and data rate, see OS!

Disk History

1989:
63 Mbit/sq. in
60,000 MBytes

1997:
1450 Mbit/sq. in
2300 MBytes

1997:
3090 Mbit/sq. in
8100 MBytes

“Makers of disk drives crowd even more data into even smaller spaces”
### Magnetic Disk Examples

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Seagate ST373453</th>
<th>Seagate ST3200822</th>
<th>Seagate ST 94811A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk diameter (inches)</td>
<td>3.5</td>
<td>3.50</td>
<td>2.50</td>
</tr>
<tr>
<td>Formatted data capacity (GB)</td>
<td>73.4</td>
<td>200</td>
<td>40</td>
</tr>
<tr>
<td>Internal cache size (MB)</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>MTTF (hours)</td>
<td>1,200,000</td>
<td>600,000</td>
<td>330,000</td>
</tr>
<tr>
<td>Number of disk surfaces (heads)</td>
<td>8</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Rotation speed (RPM)</td>
<td>15,000</td>
<td>7200</td>
<td>5400</td>
</tr>
<tr>
<td>Transfer rate (MB/sec)</td>
<td>57-86</td>
<td>32-58</td>
<td>34</td>
</tr>
<tr>
<td>Power/box (watts)</td>
<td>2,900</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>GB/watt</td>
<td>4</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>Price in 2004</td>
<td>$400, $5/GB</td>
<td>$100, $0.5/GB</td>
<td>$100, $2.5/GB</td>
</tr>
</tbody>
</table>

### The Future of Magnetic Disks

Price per GB in PC disks, dropping a factor of 10,000 1983-2001

© 2003 Elsevier Science (USA). All rights reserved.
Magnetic Tapes

° Traditionally, tapes enjoyed a 10-100 times advantage over disks in price per GB and were the technology of choice for disk backup
  • In 2001, the price of a 40 GB IDE disk is about the same as that of a 40 GB tape

° More organizations are using networks and remote disks to replicate the data geographically
  • SANs: Storage Attached Networks or Storage Area Networks

Flash Memory

° Purpose (for embedded systems):
  • Long term, nonvolatile storage
  • Small space, low power consumption
  • Read access time comparable to DRAMs
  • Writing much slower than DRAMs
    - Erasing first, then writing
  • Also used as a rewritable ROM in embedded systems
    - Allow software to be upgraded without replacing chips

° Two major types, depending on the building blocks:
  • NOR
  • NAND

*Flash memory is often faster than disk for reading, slower for writing*
RE: Dependability: Reliability and Availability

- Reliability: is a measure of the continuous service accomplishment (or, equivalently, the time to failure) from a reference initial instant
  - MTTF: mean time to failure
  - MTTR: mean time to repair (service interruption)
  - MTBF: mean time between failures
  - $\frac{1}{MTTF}$: the rate of failures (failure rate)
- If a collection of modules have exponentially distributed lifetimes (the age of the modules is not important in probability of failure) and failures of different modules are independent with each other
  - $\frac{1}{MTTF}$: the rate of failures (failure rate)
- Availability: a measure of the service accomplishment with respect to the alternation between the two states of accomplishment and interruption.
  - Module availability = $\frac{MTTF}{MTTF + MTTR}$

RE: Example of Reliability and Availability

- Assume a disk subsystem with the following components and MTTF
  - 10 disks, each rated at 1,000,000-hour MTTF
  - 1 SCSI controller, 500,000-hour MTTF
  - 1 power supply, 200-000-hour MTTF
  - 1 fan, 200-000-hour MTTF
  - 1 SCSI cable, 1,000,000-hour MTTF
- It is known that the (1) devices have exponentially distributed lifetimes and (2) failures of different modules are independent with each other
- Question: what is the MTTF of the system?
- Answer:
  - $\frac{1}{MTTF}$: the rate of failures (failure rate)
  - $\frac{1}{MTTF_{system}} = 10 \times \frac{1}{1000000} + \frac{1}{500000} + \frac{1}{200000} + \frac{1}{200000} + \frac{1}{100000} = \frac{23}{1000000}$ hours
  - The MTTF for the system is the inverse of the failure rate
  - $MTTF_{system} = \frac{1}{Failure-rate_{system}} = \frac{1000000 \text{ hours}}{23} = 43,500 \text{ hours}$
Disk Arrays (RAIDs)

- A new organization of disk storage:
  - Arrays of small and inexpensive disks
  - Increase potential throughput by having many disk drives:
    - Data is spread/striped over multiple disk
    - Multiple simultaneous accesses are made to several disks
  - RAID: Redundant Arrays of Inexpensive/Independent Disks

- Reliability is lower than a single disk:
  - But availability can be improved by adding redundant disks:
    Lost information can be reconstructed from redundant information

RAID 0 (No Redundancy)

- RAID0:
  - A disk array in which data are striped among disks but there is no redundancy to tolerate disk failure
### RAID 1 (Mirroring)

- **RAID1 (popular in some server systems):**
  - Use twice as many disks as does RAID0 for tolerating disk failure
  - Whenever data are written to one disk, they are also written to a redundant disk; always two copies of the information – most expensive RAID solution
  - Tolerate one disk failure at least, up to the number of Data/mirror disks
    - Worst case, both a data disk and its mirror disk fail

### Berkeley History: RAID-I

- **RAID-I (1989)**
  - Consisted of a Sun 4/280 workstation with 128 MB of DRAM, four dual-string SCSI controllers, 28 5.25-inch SCSI disks and specialized disk striping software

- **RAID I – RAID6**

- Today RAID is $20+ billion dollar industry, 80% nonPC disks sold in RAIDs
**RAID 3 (Bit-interleaved Parity)**

Original data disks

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Parity = \( b_0 \ XOR \ b_1 \ XOR \ b_2 \ XOR \ldots b_8 \) = \((b_0 + b_1 + b_2 + \ldots b_8) \mod 2\)

- RAID3:
  - Needs enough redundant information to restore the lost information upon a failure, instead of a complete copy of the original data for each disk as in RAID1
  - Reads or writes go to all disks in the group, with one extra disk to hold the check information in case of a failure
  - Tolerate 1 disk failure only

**RAID 4 and RAID 5 (Block-Interleaved Parity)**

- RAID5 (Distributed block-interleaved parity): popular
  - Avoids that the parity disk be the bottle-neck
RAID0 – RAID6

<table>
<thead>
<tr>
<th>RAID level</th>
<th>Minimum number of disk faults survived</th>
<th>Example data disks</th>
<th>Corresponding check disks</th>
<th>Corporations producing RAID products at this level</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Nonredundant striped</td>
<td>0</td>
<td>0</td>
<td>widely used</td>
</tr>
<tr>
<td>1</td>
<td>Mirrored</td>
<td>1</td>
<td>8</td>
<td>EMC, Compaq (Tandem), IBM</td>
</tr>
<tr>
<td>2</td>
<td>Memory-style ECC</td>
<td>1</td>
<td>8</td>
<td>Storage Concepts</td>
</tr>
<tr>
<td>3</td>
<td>Bit-interleaved parity</td>
<td>1</td>
<td>8</td>
<td>Network Appliance</td>
</tr>
<tr>
<td>4</td>
<td>Block-interleaved parity</td>
<td>1</td>
<td>8</td>
<td>Storage Concepts</td>
</tr>
<tr>
<td>5</td>
<td>Block-interleaved distributed parity</td>
<td>1</td>
<td>8</td>
<td>widely used</td>
</tr>
<tr>
<td>6</td>
<td>P + Q redundancy</td>
<td>2</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

RAID levels, their fault-tolerance, and overhead in redundant disks

The Big Picture: Where are We Now?

- How to connect I/O to the rest of the computer?
A bus is a shared communication link
It uses one set of wires (control & data) to connect multiple subsystems

Advantages of Buses

- **Versatility:**
  - New devices can be added easily
  - Peripherals can be moved between computer systems that use the same bus standard

- **Low Cost:**
  - A single set of wires is shared in multiple ways
Disadvantages of Buses

° It creates a communication bottleneck
  • The bandwidth of that bus can limit the maximum I/O throughput
  • Increasingly replaced by networks and switches; SANs

° The maximum bus speed is largely limited by physical factors:
  • The length of the bus
  • The number of devices on the bus
  • The need to support a range of devices with:
    - Widely varying latencies
    - Widely varying data transfer rates

The General Organization of a Bus

° Control lines:
  • Signal requests and acknowledgments (transaction protocol)
  • Indicate what type of information is on the data lines

° Data lines carry information between the source and the destination:
  • Data and Addresses
    - Processor-Memory often has separate address line
    - Pentium
  • Complex commands

° A bus transaction includes two parts:
  • Sending the address
  • Receiving or sending the data
**Master versus Slave**

A bus transaction includes two parts:
- Sending the address
- Receiving or sending the data

Master is the one who starts the bus transaction by:
- Sending the address
- If more than one Master, arbitration is required among masters to decide which gets the bus next; often fixed priority for each device
  - Higher bandwidth: split transactions, packet-switched bus, etc.

Slave is the one who responds to the address by:
- Sending data to the master if the master asks for data
- Receiving data from the master if the master wants to send data

**Output Operation**

Output: Processor sending memory data to the I/O device

Step 1: Request Memory
- Processor
- Control (Memory Read Request)
- Memory
- Data (Memory Address)
- I/O Device (Disk)

Step 2: Read Memory
- Processor
- Control
- Memory
- Data
- I/O Device (Disk)

Step 3: Send Data to I/O Device
- Processor
- Control (Device Write Request)
- Memory
- Data
- I/O Device (Disk)
- I/O Device Address and then Data
**Input Operation**

- **Input**: Processor makes memory receiving data from I/O device

**Step 1: Request Memory**

- **Processor** → **I/O Device (Disk)** → **Memory**
  - Control (Memory Write Request)
  - Data (Memory Address)

**Step 2: Receive Data**

- **Processor** → **I/O Device (Disk)** → **Memory**
  - Control (I/O Read Request)
  - Data (I/O Device Address and then Data)

**Types of Buses**

- **Processor-Memory Bus** (design specific)
  - Short and high speed
  - Only need to match the memory system
    - Maximize memory-to-processor bandwidth
  - Connects directly to the processor

- **I/O Bus** (industry standard)
  - Usually is lengthy and slower
  - Need to match a wide range of I/O devices
  - Connects to the processor-memory bus or backplane bus

- **Backplane Bus** (industry standard)
  - Backplane: an interconnection structure within the chassis
  - Allow processors, memory, and I/O devices to coexist
  - Cost advantage: one single bus for all components
A Computer System with One Bus: Backplane Bus

- A single bus (the backplane bus) is used for:
  - Processor to memory communication
  - Communication between I/O devices and memory
- Advantages: Simple and low cost
- Disadvantages: slow and the bus can become a major bottleneck
- Example: early IBM PC

A Two-Bus System

- I/O buses tap into the processor-memory bus via bus adaptors:
  - Processor-memory bus: mainly for processor-memory traffic
  - I/O buses: provide expansion slots for I/O devices
- Apple Macintosh-II
  - NuBus: Processor, memory, and a few selected I/O devices
  - SCSI Bus: the rest of the I/O devices
A Three-Bus System (Bus Hierarchy)

- A small number of backplane buses tap into the processor-memory bus
  - Processor-memory bus is used for processor memory traffic
  - I/O buses are connected to the backplane bus
- Advantage: loading on the processor bus is greatly reduced
- Example: IBM RS6000

Clocking: Synchronous and Asynchronous

- Synchronous Bus:
  - Example: Processor-memory bus
  - Includes a clock in the control lines for synchronization
  - A fixed protocol for communication that is relative to the clock
  - Advantage: involves very little logic (cheap) and can run very fast
  - Disadvantages:
    - Every device on the bus must run at the same clock rate
    - To avoid clock skew, they cannot be long if they are fast
- Asynchronous Bus:
  - Example I/O bus
  - It is not clocked
  - It can accommodate a wide range of devices
  - It can be lengthened without worrying about clock skew
  - It requires a handshaking protocol to coordinate/synchronize the transmission of data between sender and receiver: a series of steps, sender/receiver proceed to the next step when both agree.
A Handshaking Protocol

Example: an I/O device requests a word of data from memory

- Three control lines
  - ReadReq: indicate a read request for memory
    Address is put on the data lines at the same line
  - DataRdy: indicate the data word is now ready on the data lines
    Data is put on the data lines at the same time
  - Ack: acknowledge the ReadReq or the DataRdy of the other party

Example of Buses

- Peripheral Component Interconnect (PCI) and PCI Extended (PCI-X)
  - Connect main memory to peripheral devices

- IDE/ATA and SCSI interface to storage devices
  - Integrated Drive Electronics (IDE) connects 2 disks to PC
  - AT-bus Attachment (ATA) extends IDE; wider & faster
  - Small Computer System Interconnect (SCSI) connects up to 7 devices
Interfacing Storage Devices to the CPU

A typical interface of I/O devices and an I/O bus to the CPU-memory bus

Reading

- CO 4: Chapter 6