CS 162 Operating Systems and Systems Programming
Professor: Anthony D. Joseph
Spring 2004

Lecture 16: Survey of I/O systems

16.0 Main Points

- Overhead, latency, bandwidth definitions
- How terminals and disks work (networks later).

16.1 Definitions

Overhead: CPU time to initiate operation (can't be overlapped)
Latency: time to initiate 1 byte operation
Bandwidth: rate of I/O transfer, once initiated

General rule of thumb: abstraction of byte transfers, but batch into block I/O for efficiency – pro-rates overhead, latency over larger unit.

16.2 Terminal: keyboard and display

Terminal connects to computer via serial line; same concept applies to modem connections – type characters, get characters back to display.

For example, RS-232 is bit serial: start bit, character code, stop bit.
Typical bandwidth: 9600 baud (transmission rate in bits/sec) = 900 bytes/sec
up to 53Kbit/s = 5.3Kbytes/sec

Even though keyboards/displays reflect a relatively small rate of transfer, can still swamp CPU due to overhead of handling each byte.

What if interrupt per byte?

10 users (vi/emacs) (or 10 people using a modem)

900 interrupts/sec per user
Overhead of handling interrupt = 100 µsec
Implies:
Devote whole computer to simply handling interrupts!

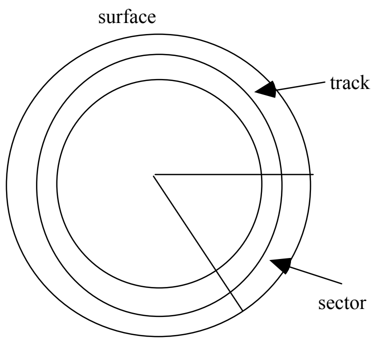
Alternative: **Direct memory access** (DMA). Instead of interrupt on every byte, perform block transfer and interrupt CPU when block transfer is done. Allows much higher transfer rates.

Another alternative: **bitmap display**. Can't type very fast, but need lots of display bandwidth. CPU writes to video memory what should be displayed, display reads video memory, illuminating pixels.

Can't see quantization on displays all that easily (except take 184!), but can see this on calculators!

Raster displays are how TV sets, laser printers, and monitors work.

16.3 Disk Organization



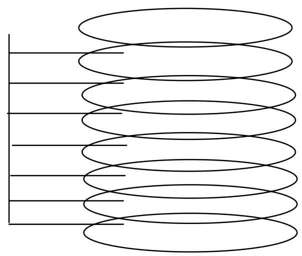
Disk **surface**: circular disk, coated with magnetic material
Quarter-sized to 12 – 14" platters

Tracks: concentric rings around disk surface, bits laid out serially along each track

Like a CD, disk is always spinning.

Each track is split up into **sectors**: arc of track, also minimum unit of transfer

CD's come individually, but magnetic disks come organized in a **disk pack**, by ganging a set of **platters** together in a stack (each platter is 2 surfaces)



Disk is read via a comb – 2 read/write "head assembly" at end of each arm (typically, only put disk heads to read inside surfaces, external surfaces are too exposed).

Cylinder: corresponding track on each surface

Disk operation is in terms of radial coordinates, not x, y, z: move arm to correct track, wait for disk to rotate under head, select head, then transfer as it's going by.

16.4 Disk Performance

To read or write disk block:

1. **Seek**: position heads over cylinder
   Time for seek depends on how fast you can move the arm. Typically, 10 msec to move all the way across disk

2. **Rotational delay**: wait for sector to rotate underneath head: typically, 120 cycles per second => 8 msec per rotation
3. Select head
4. **Transfer** bytes (typically, 4MB/s, 1KB/sector => 0.25ms)

Thus, overall time to do disk I/O:
Seek + Rotational delay + Transfer

Seek and rotational delays are latency, transfer rate is bandwidth.

If random place on disk, then seek (5 ms average) + rotational delay (4 ms average) + transfer (0.25 ms). Thus, roughly 10 msec to fetch/put data, mostly seek and rotational delay. In other words, 100 KB/s

If random place in same cylinder, then no seek needed, just rotational delay (4 ms average) + transfer (0.25 ms), or roughly 5 ms. to fetch/put 1KB of data. In other words, 200 KB/s.

If next sector on same track, then no rotational delay, so just transfer time – 4 MB/s.

Key to using disk effectively (and therefore to everything in file systems!) is to minimize seek and rotational delay.

16.5 Disk Tradeoffs

How do we choose disk sector size?

Need to synch head with rotation speed – synch bits together

Put 100-1000 bits in between each sector, to allow system to measure how fast disk is spinning (also allows tolerance to small changes in track length)

What if sector was 1 byte?

- Space efficiency – only 1% of disk has useful space
- Time efficiency – each seek takes 10 ms, means transfer at a rate of 50 – 100 char/sec

What if sector was 1KBytes?

- Space efficiency: 90% of disk has useful space
- Time efficiency: transfer at 100KB/sec

What if sector was 1M bytes?

- Space efficiency: almost all of disk has useful space
- Time efficiency: transfer at disk bandwidth (4MB/sec) – seek, rotation no longer matter.

But 1MB blocks are really wasteful if need only 1 byte.

16.6 Disk Technology Trends

Typical disk today:

Disk capacity	100 GB
# of surfaces per pack	18
# of tracks per surface	4096
# of sectors per track	32-64
# of bytes per sector	1KB
# of revolutions per min	10,000

Transfer rate: # of bytes rotating under head per second

1KB bytes/sector * (32-64) sectors/cycle * 10,000 cycles/sec

= 320 – 640 MB/sec (peak)

1. Disks getting smaller, for similar capacity
   - Smaller => spin disk faster (less rotational delay, higher bandwidth)
   - Smaller => less distance for head to travel (faster seeks)
   - Smaller => lighter weight (for portables)
2. Disk data getting denser (more bits/square inch, allows smaller disks without sacrificing capacity). Tracks closer together => faster seeks
3. Disks getting cheaper (factor of 2 per year since 1991)

Nevertheless, disk density ($ per byte) improving much faster than mechanical limitations (seek, rotational delay).

Key to improving density: get head close to surface. Heads are spring loaded, aerodynamically designed to fly as close to surface as possible (also, lightweight to allow for faster seeks)

What happens if head contacts surface?

Head crash – scrapes off magnetic information

Problem – space gets so small, no room for air molecules. Instead, fill with special inert gas!