Disk Storage & Dependability

Computer Organization

Monday, 17 October 2022

Many slides adapted from:
Computer Organization and Design,
Patterson & Hennessy
5th Edition, © 2014, MK
and from Prof. Mary Jane Irwin, PSU

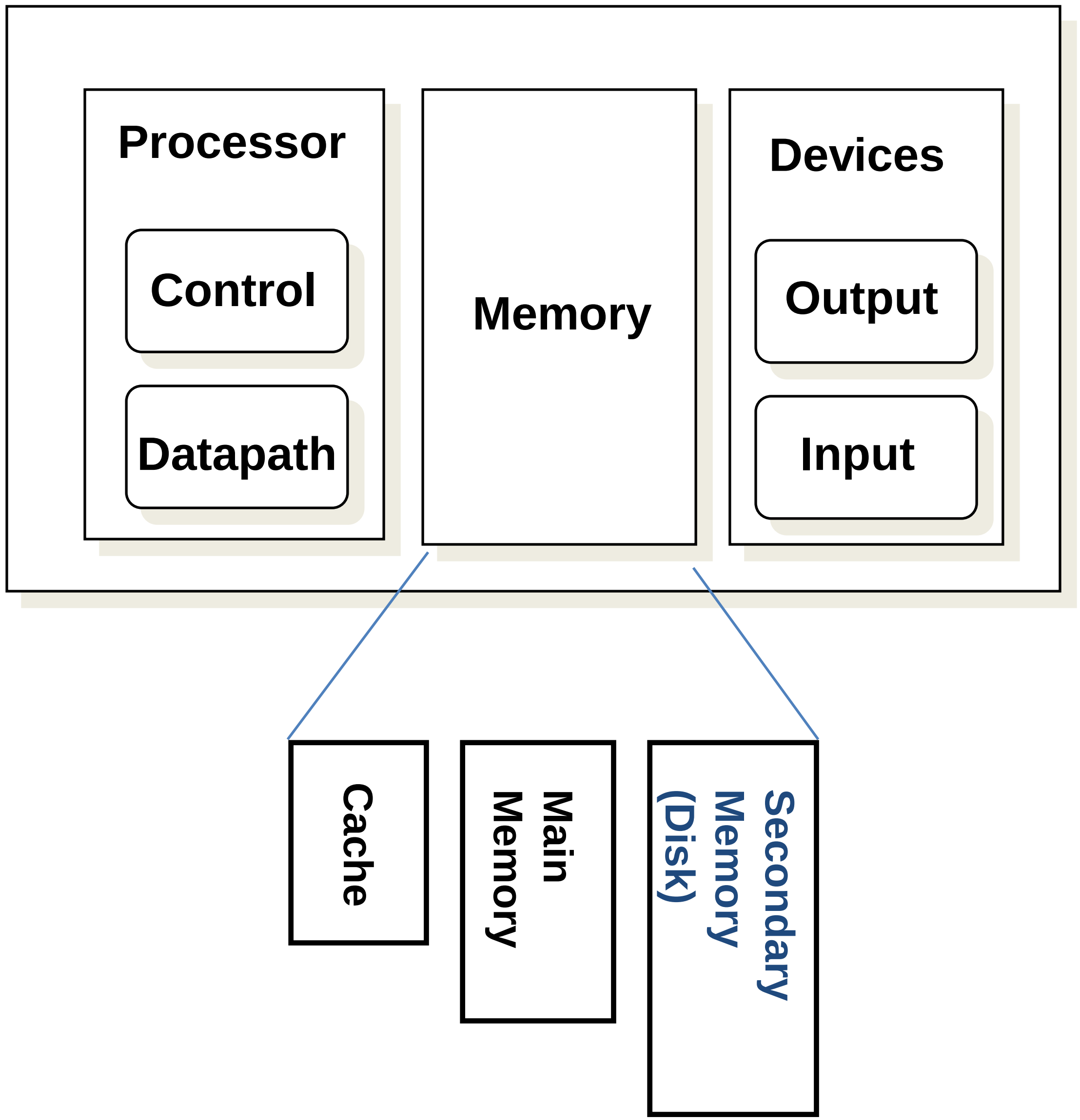
TÉCNICO LISBOA

Summary

- Previous Class
  - IO System

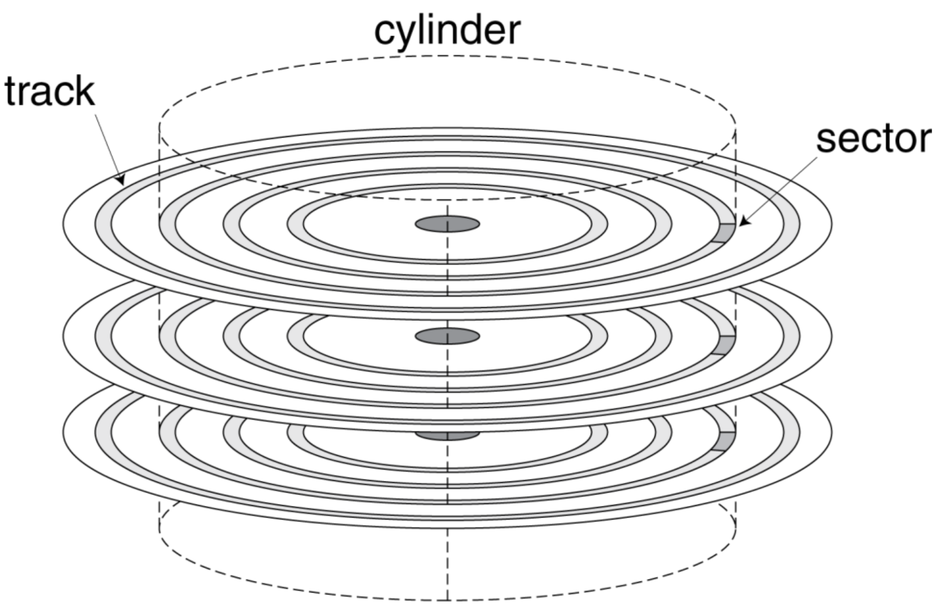
- Today:
  - Disk Storage
  - Dependability

Review: Major Components of a Computer

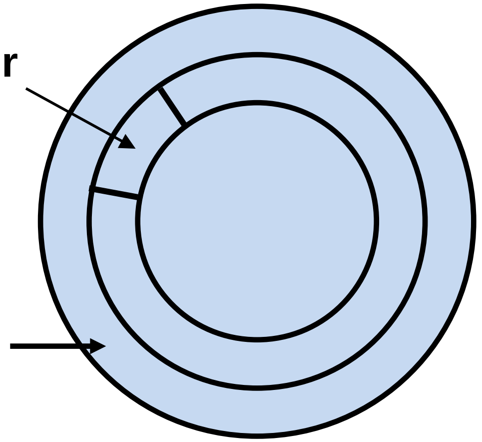


Disk Storage

- Nonvolatile, rotating magnetic storage

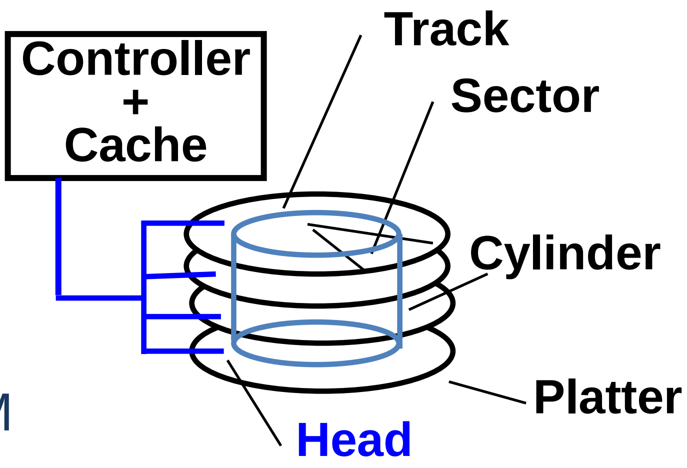


Magnetic Disk



- Purpose
  - Long term, nonvolatile storage
  - Lowest level in the memory hierarchy
    - slow, large, inexpensive
- General structure
  - A rotating platter coated with a magnetic surface
  - A moveable read/write head to access the information on the disk
- Typical numbers
  - 1 to 4 platters (each with 2 recordable surfaces) per disk of 2.5cm to 9.5cm in diameter
  - Rotational speeds of 5,400 to 15,000 RPM
  - 10,000 to 50,000 tracks per surface
    - cylinder - all the tracks under the head at a given point on all surfaces
  - 100 to 500 sectors per track
    - the smallest unit that can be read/written (typically 512B)

Magnetic Disk Characteristics



Disk read/write components

1. Seek time: position the head over the proper track (3 to 13 ms avg)

2. Rotational latency: wait for the desired sector to rotate under the head (½ of 1/RPM converted to ms)

   0.5/5400RPM = 5.6ms to 0.5/15000RPM = 2.0ms

3. Transfer time: transfer a block of bits (one or more sectors) under the head to the disk controller's cache (70 to 125 MB/s are typical disk transfer rates)

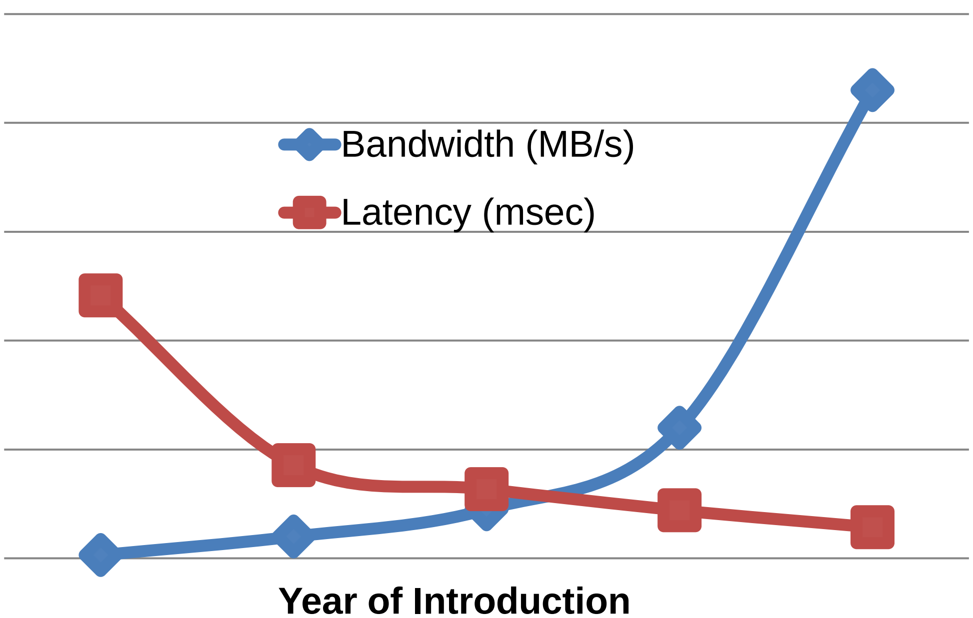
4. Controller time: the overhead the disk controller imposes in performing a disk I/O access (typically < .2 ms)
   - the disk controller's "cache" takes advantage of spatial locality in disk accesses

Disk Performance Issues

- Manufacturers quote average seek time
  - Based on all possible seeks
  - Locality and OS scheduling lead to smaller actual average seek times
- Smart disk controller allocate physical sectors on disk
  - Present logical sector interface to host
  - SCSI, ATA, SATA
- Disk drives include caches
  - Prefetch sectors in anticipation of access
  - Avoid seek and rotational delay

Latency & Bandwidth Improvements

- In the time that the disk bandwidth doubles the latency improves by a factor of only 1.2 to 1.4



- Disk latency is one average seek time plus the rotational latency.
- Disk bandwidth is the peak transfer time of formatted data from the media (not from the cache).

Magnetic Disk Examples (www.seagate.com)

Feature	Seagate ST31000340NS	Seagate ST973451SS	Seagate ST9160821AS
Disk diameter (inches)	3.5	2.5	2.5
Capacity (GB)	1000	73	160
# of surfaces (heads)	4	2	2
Rotation speed (RPM)	7,200	15,000	5,400
Transfer rate (MB/sec)	105	79-112	44
Minimum seek (ms)	0.8r-1.0w	0.2r-0.4w	1.5r-2.0w
Average seek (ms)	8.5r-9.5w	2.9r-3.3w	12.5r-13.0w
MTTF (hours@25^{o}C)	1,200,000	1,600,000	??
Dim (inches); Weight (lbs)	1x4x5.8; 1.4	0.6x2.8x3.9;0.5	0.4x2.8x3.9; 0.2
GB/cu.inch, GB/watt	43, 91	11, 9	37, 84
Power: op/idle/sb (watts)	11/8/1	8/5.8/-	1.9/0.6/0.2
Price in 2008, $/GB	~$0.3/GB	~$5/GB	~$0.6/GB

Flash Storage in Hard Drives

- Solid State Disc (SSD)
  - Up to 250 GB (25 €), 4TB (400 €)
  - Up to 540MB/s for reading and 520MB/s for writing
  - Low energy consumtion in idle (2mW) and active mode
    - Lower than traditional hard drives (HDD)
  - Near 1.000.000 writes for each cell
  - Data lasts up to 10 years
    - Not suitable for long term storage

- Hybrid Disc
  - Nonvolatile buffer for write accesses
  - Or used as permanent cache controlled by the OS

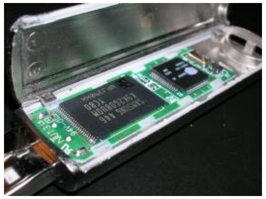
Flash Storage

- Nonvolatile semiconductor storage
  - 100x to 1000x faster than disk
  - Smaller, lower power, more robust
  - But more $/GB (between disk and DRAM)

Feature	Kingston	Transcend	RiDATA
Capacity (GB)	480	240	480
Bytes/sector	512	512	512
Transfer rates (MB/sec)	550r-500w	570r-460w	560r-510w
MTTF (hours)	>1,000,000	>1,000,000	>4,000,000
Price (2016)	$0.3/GB	~ $1.2/GB	~ $0.3/GB

Flash Types

- NOR flash: bit cell like a NOR gate
  - Random read/write access
  - Used for instruction memory in embedded systems
- NAND flash: bit cell like a NAND gate
  - Denser (bits/area), but block-at-a-time access
  - Cheaper per GB
  - Used for USB keys, media storage, …
- Traditional flash wears out after 1000’s of accesses
  - Not suitable for direct RAM or disk replacement
  - Wear levelling: remap data to less used blocks



Fallacy: Disk Dependability

- If a disk manufacturer quotes MTTF as 1,200,000hr (140yr)
  - A disk will work that long
- Wrong: this is the mean time to failure
  - What is the distribution of failures?
  - What if you have 1000 disks
    - How many will fail per year?

$$\text{Annual Failure Rate (AFR)} = \frac{8760\ \text{hrs / disk}}{1200000\ \text{hrs / failure}} = 0.73\%$$

$$\text{Failed Disks} = \frac{1000\ \text{disks} \times 8760\ \text{hrs / disk}}{1200000\ \text{hrs / failure}} = 7.3$$

Fallacies

- Disk failure rates are as specified
  – Studies of failure rates in the field
    - Schroeder and Gibson: 2% to 4% vs. 0.6% to 0.8%
    - Pinheiro, *et al.*: 1.7% (first year) to 8.6% (third year) vs. 1.5%
  – Why?

- A 1GB/s interconnect transfers 1GB in one sec
  – But what's a GB?
  – For bandwidth, use 1GB = 10^9 B
  – For storage, use 1GiB = 2^{30} B = 1.075×10^9 B
  – So 1GB/sec is 0.93GB in one second
    - About 7% error

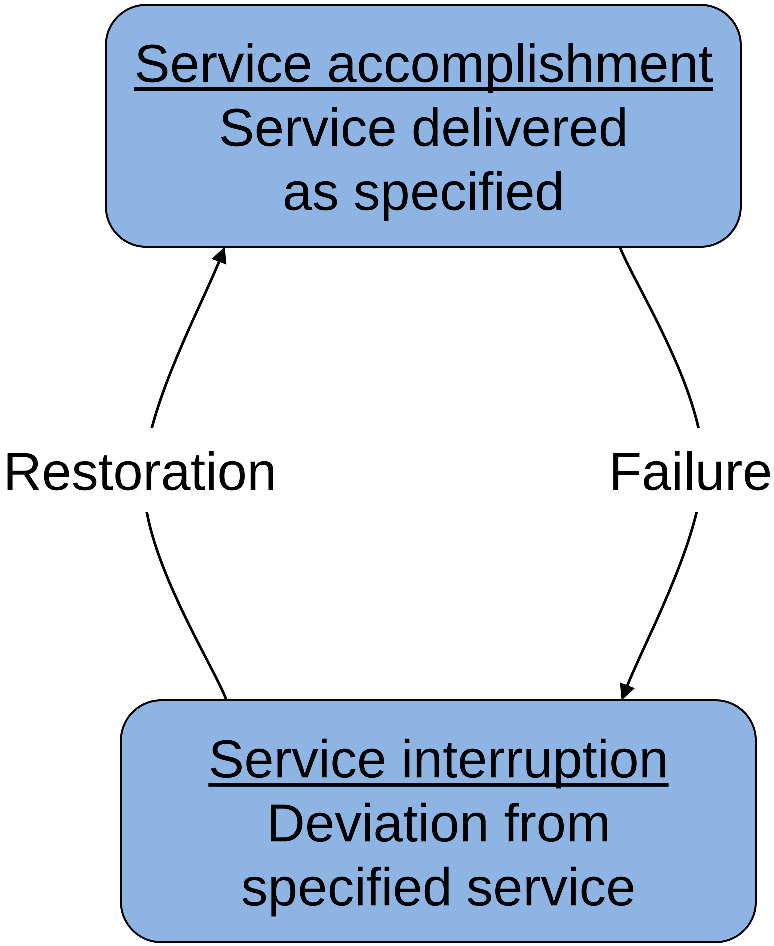
Fallacy: Disk Scheduling

- Best to let the OS schedule disk accesses
  - But modern drives deal with logical block addresses
    - Map to physical track, cylinder, sector locations
    - Also, blocks are cached by the drive
  - OS is unaware of physical locations
    - Reordering can reduce performance
    - Depending on placement and caching

Pitfall: Backing Up to Tape

- Magnetic tape used to have advantages
  - Removable, high capacity
- Advantages eroded by disk technology developments
- Makes better sense to replicate data
  - E.g, RAID, remote mirroring

Dependability



- Fault: failure of a component
  - May or may not lead to system failure

Dependability Measures

- Reliability: mean time to failure (MTTF)
- Service interruption: mean time to repair (MTTR)
- Mean time between failures
  - MTBF = MTTF + MTTR
- Availability = MTTF / (MTTF + MTTR)
- Improving Availability
  - Increase MTTF: fault avoidance, fault tolerance, fault forecasting
  - Reduce MTTR: improved tools and processes for diagnosis and repair
- To increase MTTF, either improve the quality of the components or design the system to continue operating in the presence of faulty components
  1. Fault avoidance: preventing fault occurrence by construction
  2. Fault tolerance: using redundancy to correct or bypass faulty components (hardware)
     - Fault detection versus fault correction
     - Permanent faults versus transient faults

RAID 0 & 1 & 2

- RAID 0: Parallelism
  - No data replication or redundancy
- RAID 1: Mirroring
  - N + N disks, replicate data
    - Write data to both data disk and mirror disk
    - On disk failure, read from mirror
- RAID 2: Error correcting code (ECC)
  - N + E disks (e.g., 10 + 4)
  - Split data at bit level across N disks
  - Generate E-bit ECC
  - Can tolerate *limited* disk failure, since the data can be reconstructed
  - Too complex, not used in practice

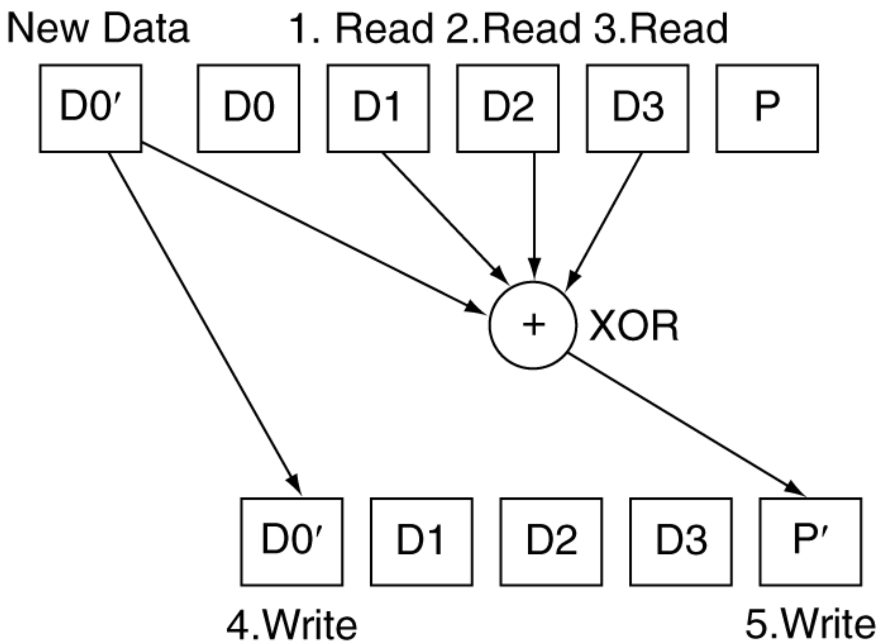
RAID 3: Bit-Interleaved Parity

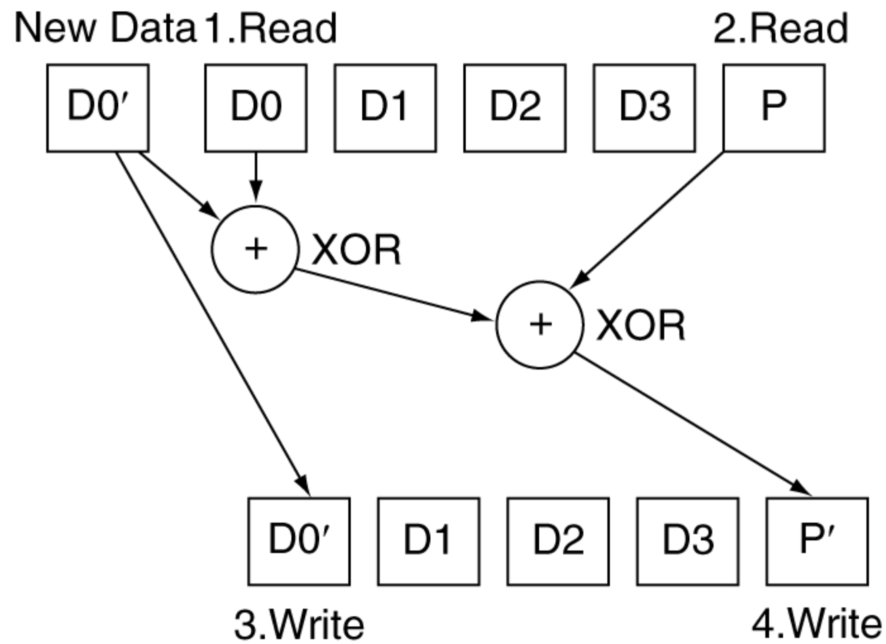
- N + 1 disks
  - Data striped across N disks at byte level
  - Redundant disk stores parity
  - Read access
    - Read all disks
  - Write access
    - Generate new parity and update all disks
  - On failure
    - Use parity to reconstruct missing data
- Not widely used

RAID 4: Block-Interleaved Parity

- N + 1 disks
  - Data striped across N disks at block level
  - Redundant disk stores parity for a group of blocks
  - Read access
    - Read only the disk holding the required block
  - Write access
    - Just read disk containing modified block, and parity disk
    - Calculate new parity, update data disk and parity disk
  - On failure
    - Use parity to reconstruct missing data
- Not widely used

RAID 3 vs RAID 4



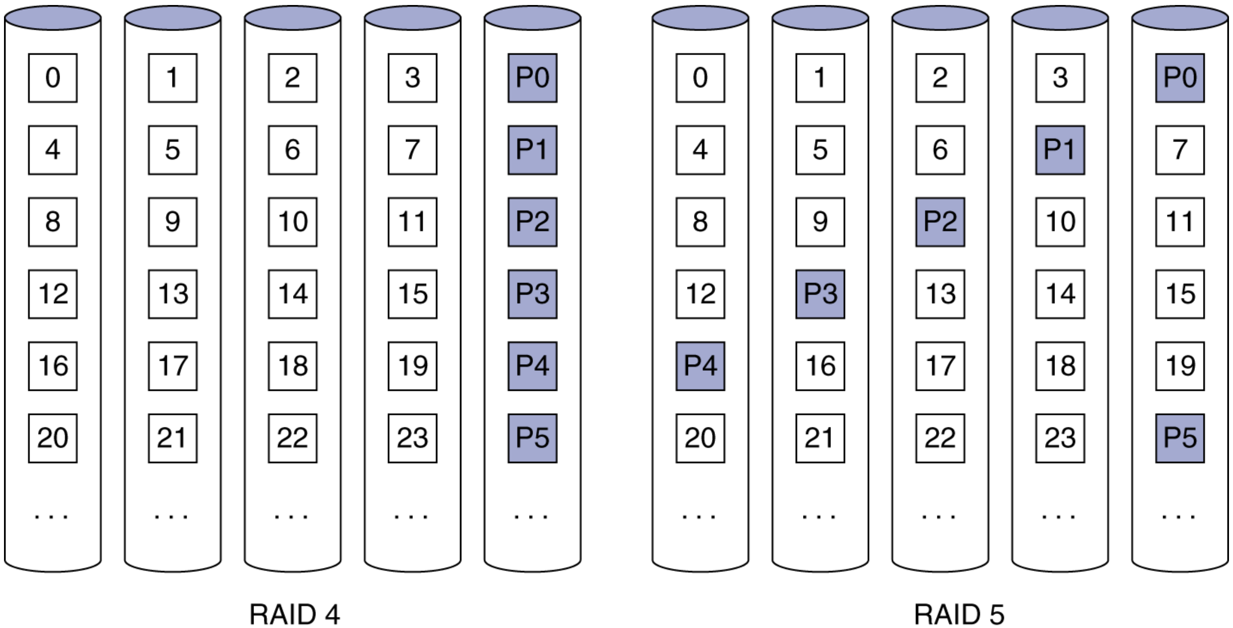


3 reads and 2 writes
involving *all* the disks

2 reads and 2 writes
involving just *two* disks

RAID 5: Distributed Parity

- N + 1 disks
  - Like RAID 4, but parity blocks distributed across disks
    - Avoids parity disk being a bottleneck
    - Writes can be performed in parallel
- Widely used



RAID 6: P + Q Redundancy

- N + 2 disks
  - Like RAID 5, but two lots of parity
  - Greater fault tolerance through more redundancy

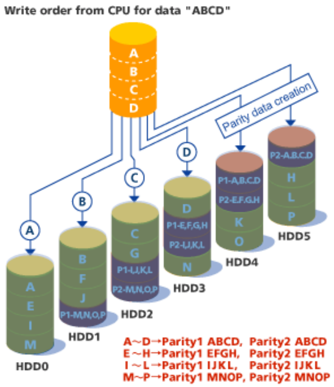
- Multiple RAID or Nested RAID
  - More advanced systems give similar fault tolerance with better performance
    - RAID 01, RAID 10, …

RAID Summary

- RAID can improve performance and availability
  - High availability requires hot swapping
- Assumes independent disk failures
  - Too bad if the building burns down!

Error Detection / Correction Codes

- Data Storage
  - CDs and DVDs
  - RAID
  - ECC memory



- Paper bar codes
  - UPS (MaxiCode)
  - QR Code

- Communications
  - Cellphones
  - Satellites / Space

Codes are all around us

Error Detection with Parity Bit

Encoding:

$$m_1 m_2 \ldots m_k \Rightarrow m_1 m_2 \ldots m_k p_{k+1}$$

where $p_{k+1} = m_1 \oplus m_2 \oplus \ldots \oplus m_k$

- Detects one-bit error since this gives odd parity

- Cannot be used to correct 1-bit error since any odd-parity word is equal distance Δ to k+1 valid codewords.

Hamming Distance

The Hamming distance between two words is the number of differences between corresponding bits.

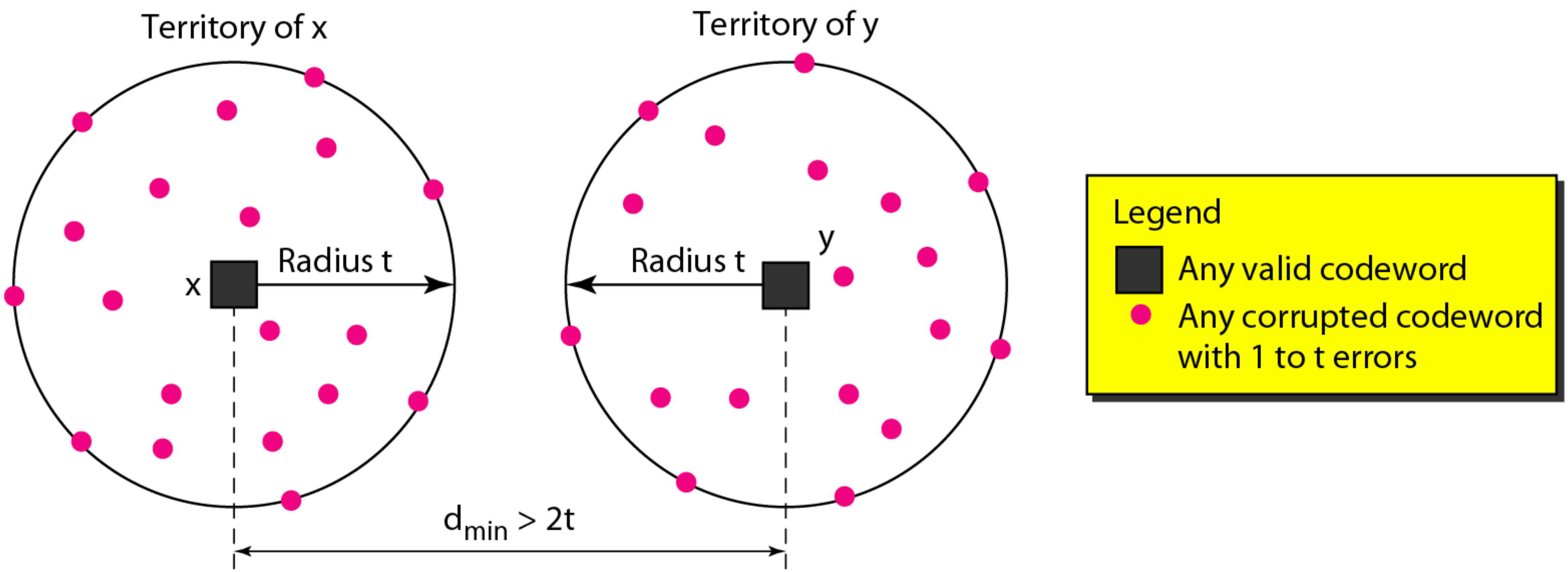
The Hamming distance d(000, 011) is 2:

$$000 \oplus 011 = 011 \quad \text{(two 1s)}$$

The Hamming distance d(10101, 11110) is 3:

$$10101 \oplus 11110 = 01011 \quad \text{(three 1s)}$$

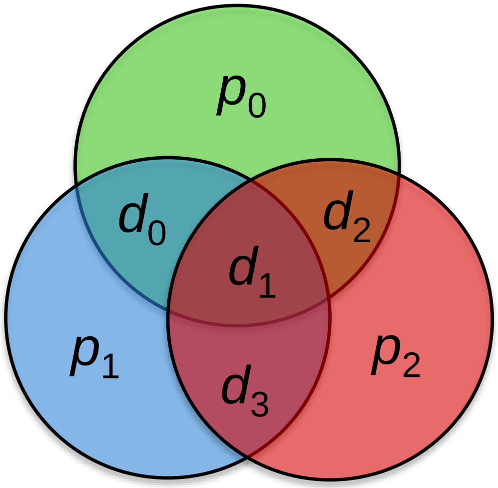
Error Correction

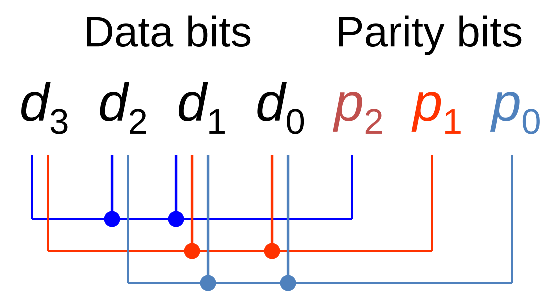


To guarantee correction of up to t errors in all cases, the minimum Hamming distance in a block code must be

$$d_{min} = 2t + 1$$

Hamming Codes





$s_2\ s_1\ s_0$	Error
0 0 0	None
0 0 1	p_0
0 1 0	p_1
0 1 1	d_0
1 0 0	p_2
1 0 1	d_2
1 1 0	d_3
1 1 1	d_1

Redundancy: 3 check bits for 4 data bits
Unimpressive, but gets better with more data bits
(7, 4); (15, 11); (31, 26); (63, 57); (127, 120)

Capability: Corrects any single-bit error

$$s_2 = d_3 \oplus d_2 \oplus d_1 \oplus p_2$$
$$s_1 = d_3 \oplus d_1 \oplus d_0 \oplus p_1$$
$$s_0 = d_2 \oplus d_1 \oplus d_0 \oplus p_0$$

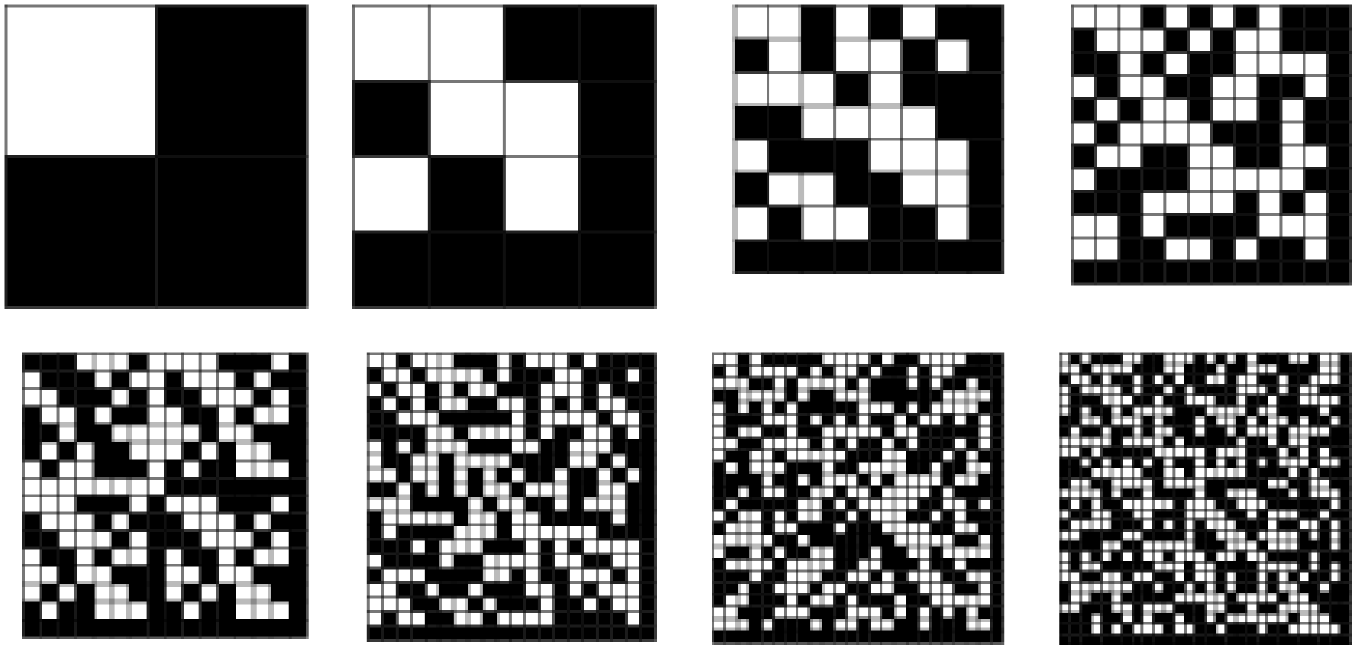
$s_2\ s_1\ s_0$
Syndrome

Reed-Muller Code

- Encoding contains more redundant information to increase the number of errors that can be corrected if needed

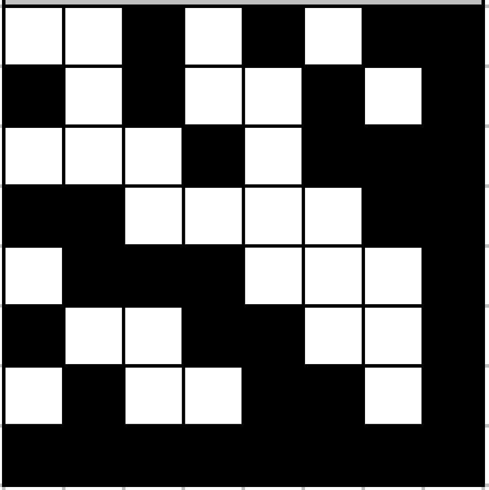
- Uses Hadamard matrices for encoding and decoding stronger error-correcting codes.

Hadamard Matrices



- Each row is a possible code
- Each row in the matrix has a Hamming distance d
- Can fix $\lfloor (d-1) / 2 \rfloor$ errors

Encoding Example



Input		Hadamard Matrix		Output
000				00101011
001				10100101
010				00010111
011	+		=	11000011
100				01110001
101				10011001
110				01001101
111				11111111
3 bits				8 bits

Hamming Distance = 4

(4 – 1) / 2 = 1 (fixable error)

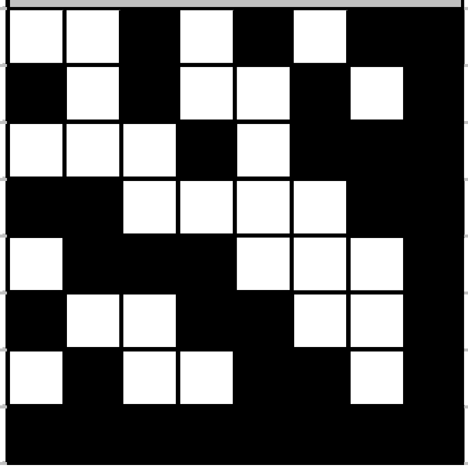
Example:

110 → 01001101

Decoding Example

Example:

0101 1101 → ?



Mapped	Hadamard Matrix		Possible Values		Compare		Differences
000			0010 1011		0101 1101		4
001			1010 0101		0101 1101		5
010			0001 0111		0101 1101		3
011		:	1100 0011	-	0101 1101	=	5
100			0111 0001		0101 1101		3
101			1001 1001		0101 1101		3
110			0100 1101		0101 1101		**1**
111			1111 1111		0101 1101		3

Result:

0101 1101 → 0100 1101 → 110

Disk Storage & Dependability

Computer Organization

Monday, 17 October 2022

Many slides adapted from:
Computer Organization and Design,
Patterson & Hennessy
5th Edition, © 2014, MK
and from Prof. Mary Jane Irwin, PSU

TÉCNICO LISBOA