



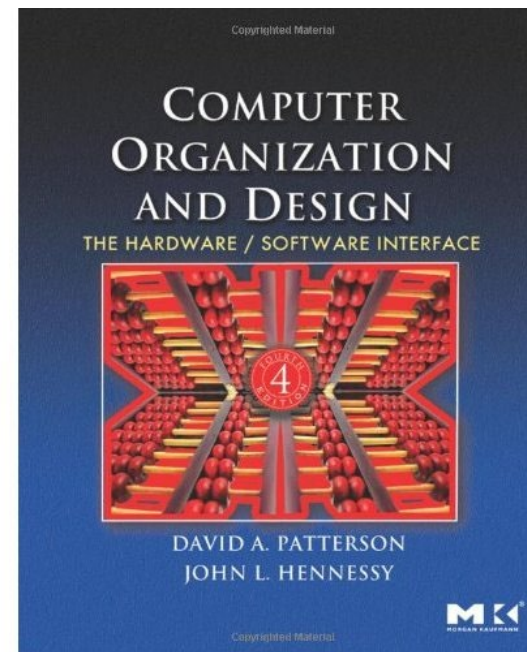
Informatik der Systeme – Chapter 8: Storage

Prof. Dr. Michael Menth

<https://kn.inf.uni-tuebingen.de>

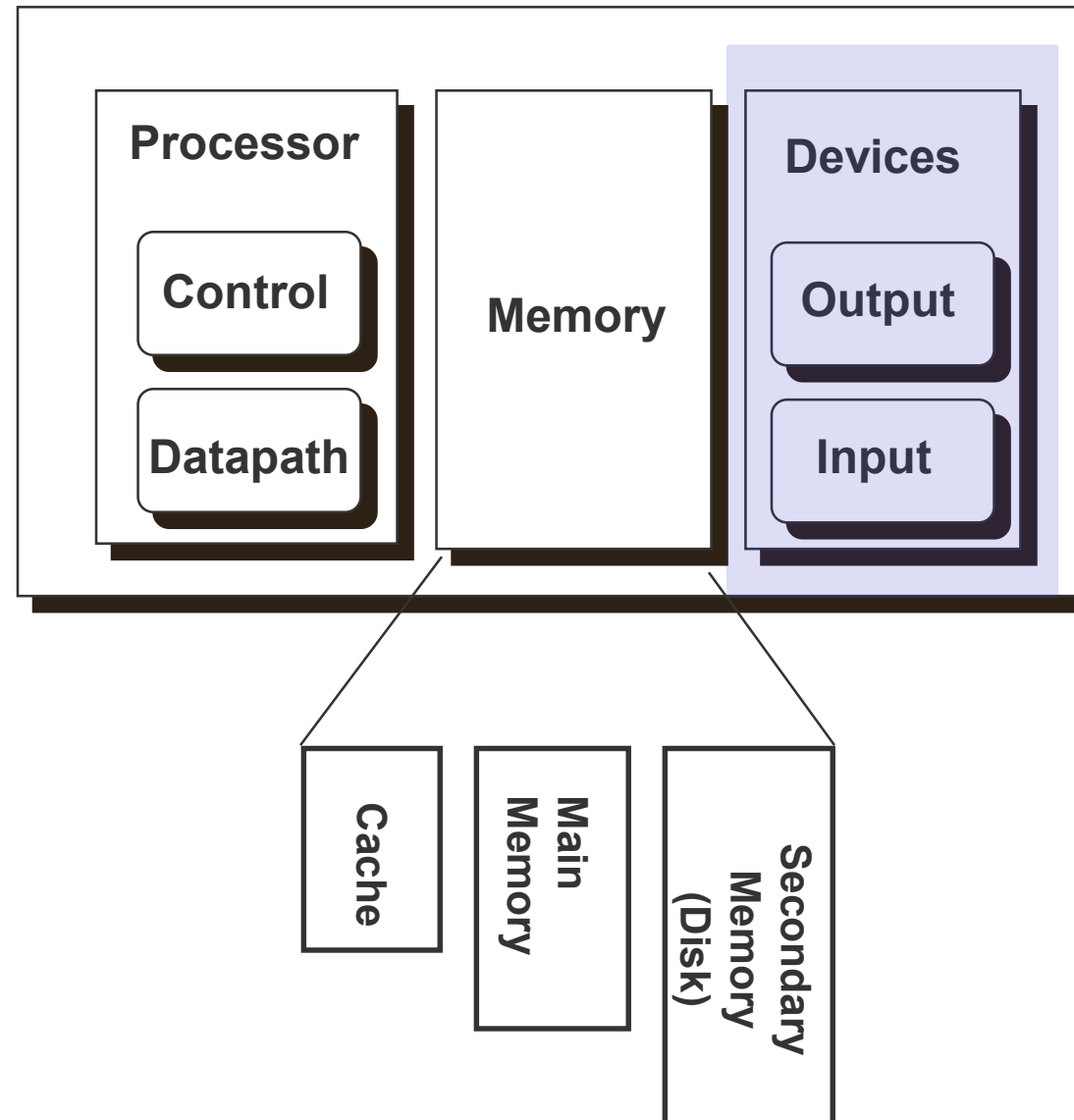


- ▶ This set of slides is an adaptation of Prof. Mary Jane Irwin's lecture notes, <http://www.cse.psu.edu/research/mdl/mji/>
- ▶ Adapted from Patterson & Hennessy: "Computer Organization and Design", 4th Edition, © 2008, MK
- ▶ German translation: Patterson & Hennessy: "Rechnerorganisation und Rechnerentwurf"
- ▶ Some content and figures taken from Andrew S. Tanenbaum: "Computerarchitektur", 5th edition, 2006





Major Components of a Computer





► Purpose

- Long term, **non-volatile** storage
- Lowest level in the memory hierarchy

► Slow, large, inexpensive

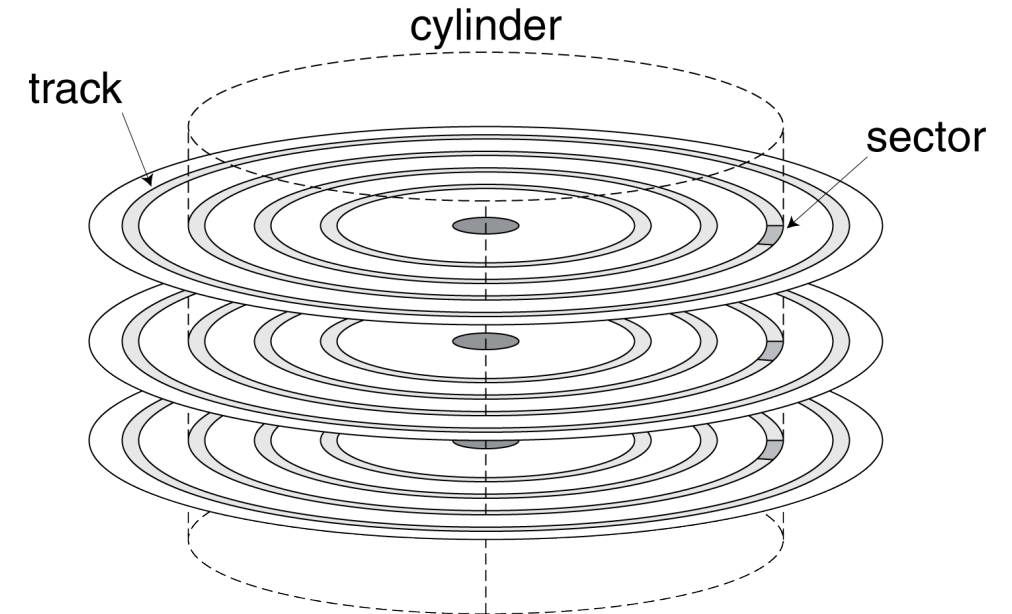
► General structure

- A rotating platter coated with a magnetic surface
- A moveable read/write head to access the information on the disk



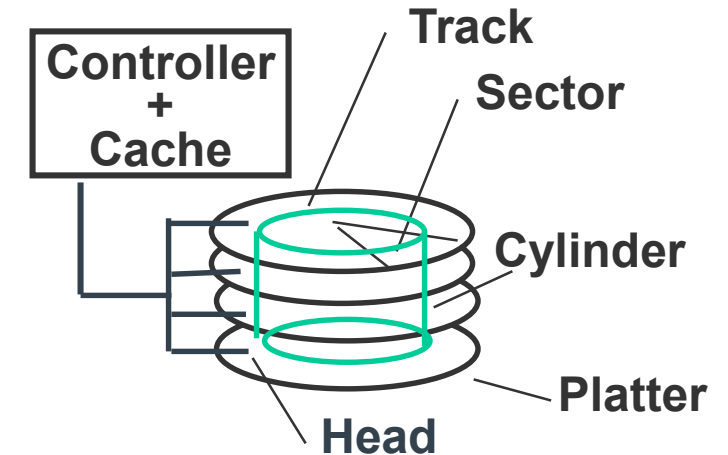


- ▶ 1 to 4 platters (each with 2 recordable surfaces) per disk of 1" to 3.5" in diameter
- ▶ Rotational speeds of 5,400 to 15,000 RPM
- ▶ 10,000 to 50,000 **tracks** per surface
- ▶ **Cylinder**
 - All the tracks under the head at a given point on all surfaces
- ▶ 100 to 500 **sectors** per track
 - The smallest unit that can be read/written (typically 512 B)





- ▶ **Seek time:** position the head over the proper track (3 to 13 ms avg)
 - Due to locality of disk references the actual average seek time may be only 25% to 33% of the advertised number
- ▶ **Rotational latency:** wait for the desired sector to rotate under the head ($\frac{1}{2}$ of $1/\text{RPM}$ converted to ms)
 - $0.5/5400 \text{ RPM} = 5.6 \text{ ms}$ to $0.5/15000 \text{ RPM} = 2.0 \text{ ms}$
- ▶ **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 in 2008)
 - The disk controller's "cache" takes advantage of spatial locality in disk accesses
 - Cache transfer rates are much faster (e.g., 375 MB/s)
- ▶ **Controller time:** the overhead the disk controller imposes in performing a disk I/O access (typically $< 0.2 \text{ ms}$)
- ▶ Access time possibly includes queuing delay if other accesses are pending





► Manufacturers quote average seek time

- Based on all possible seeks
- Locality and OS scheduling lead to smaller actual average seek times

► Example

- The average time to read or write a 512 B sector for a disk rotating at 15,000 RPM with average seek time of 4 ms, a 100 MB/s transfer rate, and a 0.2 ms controller overhead
 - Avg disk read/write = $4.0 \text{ ms} + 0.5 / (15,000 \text{ RPM} / (60 \text{ sec/min})) + 0.5 \text{ KB} / (100 \text{ MB/sec}) + 0.2 \text{ ms} = (4.0 + 2.0 + 0.005 + 0.2) \text{ ms} = 6.205 \text{ ms}$
- If the measured average seek time is 25% of the advertised average seek time, then
 - Avg disk read/write = $(1.0 + 2.0 + 0.005 + 0.2) \text{ ms} = 3.205 \text{ ms}$
- The rotational latency is usually the largest component of the access time



► Higher-level disk interfaces

- Present logical sector interface to host
- Have a **microprocessor disk controller** that can lead to performance optimizations
 - Has SRAM disk **cache** which supports fast access to data that was recently read and
 - Often also includes prefetch algorithms to try to anticipate demand

► Standards

- SCSI (Small Computer Systems Interface)
 - A set of standards (commands, protocols, and electrical and optical interfaces) for physically connecting and transferring data between computers and peripheral devices
 - Old, fast, but expensive
 - Was used for hard disks and tape drives
- ATA (Advanced Technology Attachment)
 - An interface standard for the connection of storage devices such as hard disks, solid-state drives (SSD, flash memory), and CD-ROM drives
 - Parallel ATA (PATA) largely replaced by serial ATA (SATA = faster!)



► Higher-level disk interfaces

- Present logical sector interface to host
- Have a **microprocessor disk controller** that can lead to performance optimizations
 - Has SRAM disk **cache** which supports fast access to data that was recently read and
 - Often also includes prefetch algorithms to try to anticipate demand

► Standards

- NVMe (Non-Volatile Memory Express)
 - SATA was not designed with fast flash storage in mind and was limited to 6 Gb/s
 - SATA supports only a single queue (HDDs have a single R/W head)
 - NVMe supports multiple IO queues
 - Optimized for systems with multiple CPU cores



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°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



Disk Latency & Bandwidth Milestones

	CDC Wren	SG ST41	SG ST15	SG ST39	SG ST37
RSpeed (RPM)	3600	5400	7200	10000	15000
Year	1983	1990	1994	1998	2003
Capacity (GB)	0.03	1.4	4.3	9.1	73.4
Diameter (inches)	5.25	5.25	3.5	3.0	2.5
Interface	ST-412	SCSI	SCSI	SCSI	SCSI
Bandwidth (MB/s)	0.6	4	9	24	86
Latency (ms)	48.3	17.1	12.7	8.8	5.7

Patterson, CACM Vol 47, #10, 2004

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



- ▶ Fault: failure of a component, may or may not lead to system failure
- ▶ 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 reliability
 - Either improve the quality of the components
 - Fault avoidance: preventing fault occurrence by construction
 - Or design the system to continue operation in the presence of faulty components
 - Fault tolerance: using redundancy to correct or bypass faulty components (hardware)
 - Fault detection versus fault correction
 - Permanent faults versus transient faults
- ▶ Improving availability
 - Increase MTTF: fault avoidance, fault tolerance, fault forecasting
 - Reduce MTTR: improved tools and processes for diagnosis and repair



► A disk manufacturer quotes MTTF as 1,200,000 hrs (137 yrs)

► Fallacy

- Wrong: a disk will work that long
- This is not the MTTF of a single disk!

► Correct interpretation

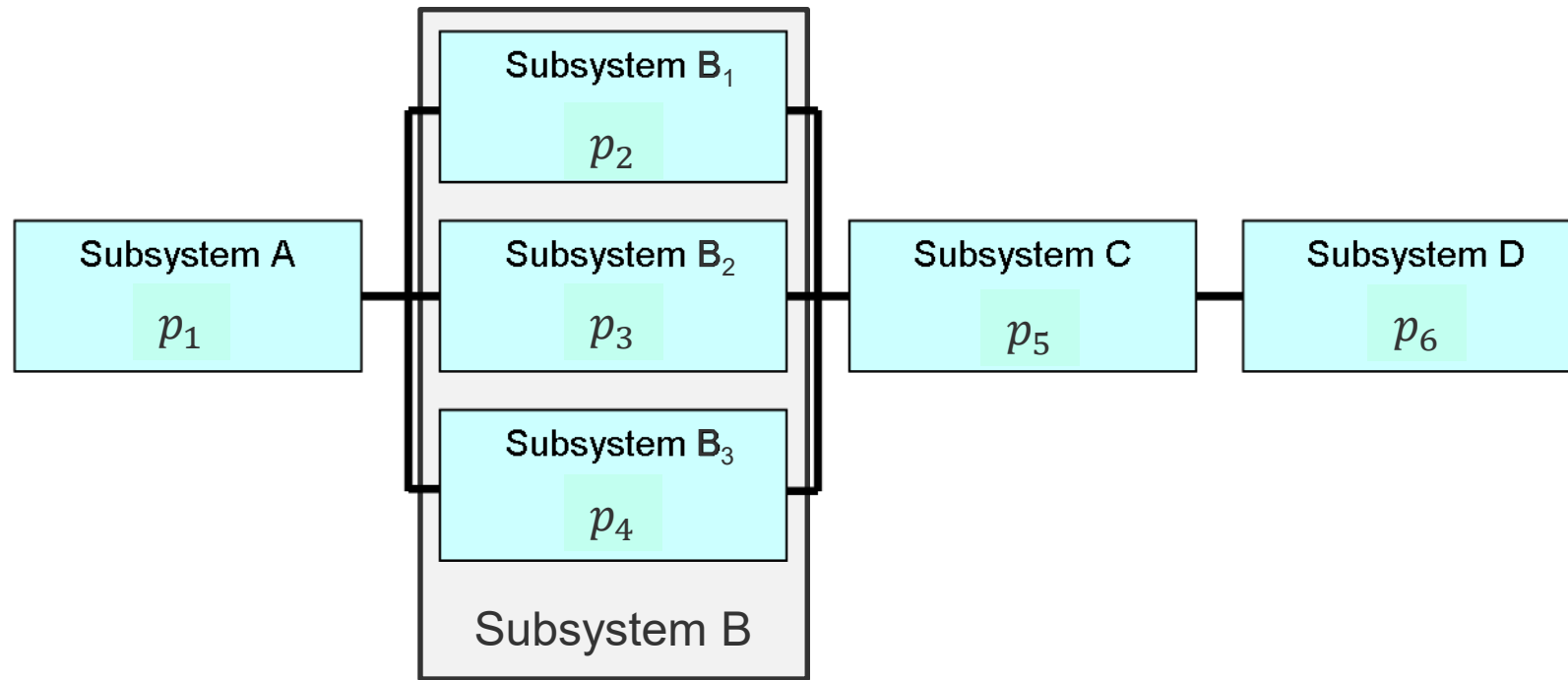
- Annual failure rate: $AFR = \frac{8760 \frac{\text{hrs}}{\text{disk}} / \text{year}}{1200000 \text{ hrs} / \text{failure}} = 0.0073 \frac{\text{failures}}{\text{year}}$
- Probability of a disk failure is 0.73%
- One out of 137 fails within a year

► Probability of disk failures

- 1st year high, then low, then increasing again



► Availability block diagram



- Availabilities $1 - p_i$ (failure probabilities p_i)
- Availability of subsystem B: $(1 - p_7) = 1 - (p_2 \cdot p_3 \cdot p_4)$
- Availability of overall system: $(1 - p_1) \cdot (1 - p_7) \cdot (1 - p_5) \cdot (1 - p_6)$



- ▶ Redundant Array of Independent (old: Inexpensive) Disks
 - Increase potential **throughput** by having many disk drives
 - Data is spread over multiple disk
 - Multiple accesses are made to several disks at a time $\Rightarrow$ parallelism improves performance
 - **Reliability** of all disks in the RAID is lower than a single disk

- ▶ But **availability can be improved** by adding redundant disks (RAID)
 - Lost information can be reconstructed from redundant information
 - MTTR: mean time to repair is in the order of hours
 - MTTF: mean time to failure of disks is tens of years
 - Provides fault-tolerant storage system if failed disks can be “hot swapped”
 - Assumes independent disk failures – too bad if the building burns down!



- ▶ RAID 0: No redundancy (“AID”?)
 - Just stripe data at bit level over multiple disks
 - Only improves performance
- ▶ RAID 1: Mirroring
 - Replicate data to backup disks, no striping
 - On disk failure, read from mirror
- ▶ Nested (hybrid) versions of different RAIDs exist
- ▶ RAID 0+1 and RAID 10 (small differences exist)
 - Data striped over N disks, each mirrored to another backup disk
 - <https://www.starline.de/magazin/technische-artikel/raid-0-1-vs-raid-10>
- ▶ RAID 2: Error correcting code (ECC) (Hamming code parity)
 - N + E disks (e.g., 10 + 4)
 - Split data at bit level across N disks and generate E-bits through ECC
 - On disk failure, reconstruct data from working and redundant disks
 - Too complex, not used in practice



- ▶ RAID 3: **byte-interleaved** parity on $N + 1$ 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 on $N + 1$ disks
 - Data **striped** across N disks **at block level**
 - Operation analogous to RAID 3 on block level
 - Not widely used



Advantage of RAID 4 over RAID 3

► Small reads

- RAID 3 accesses all data disks
- RAID 4 accesses only block on a single disk

► Small writes

- RAID 3 accesses all data disks and the parity disk
- RAID 4 accesses only a single data disk and the parity disk

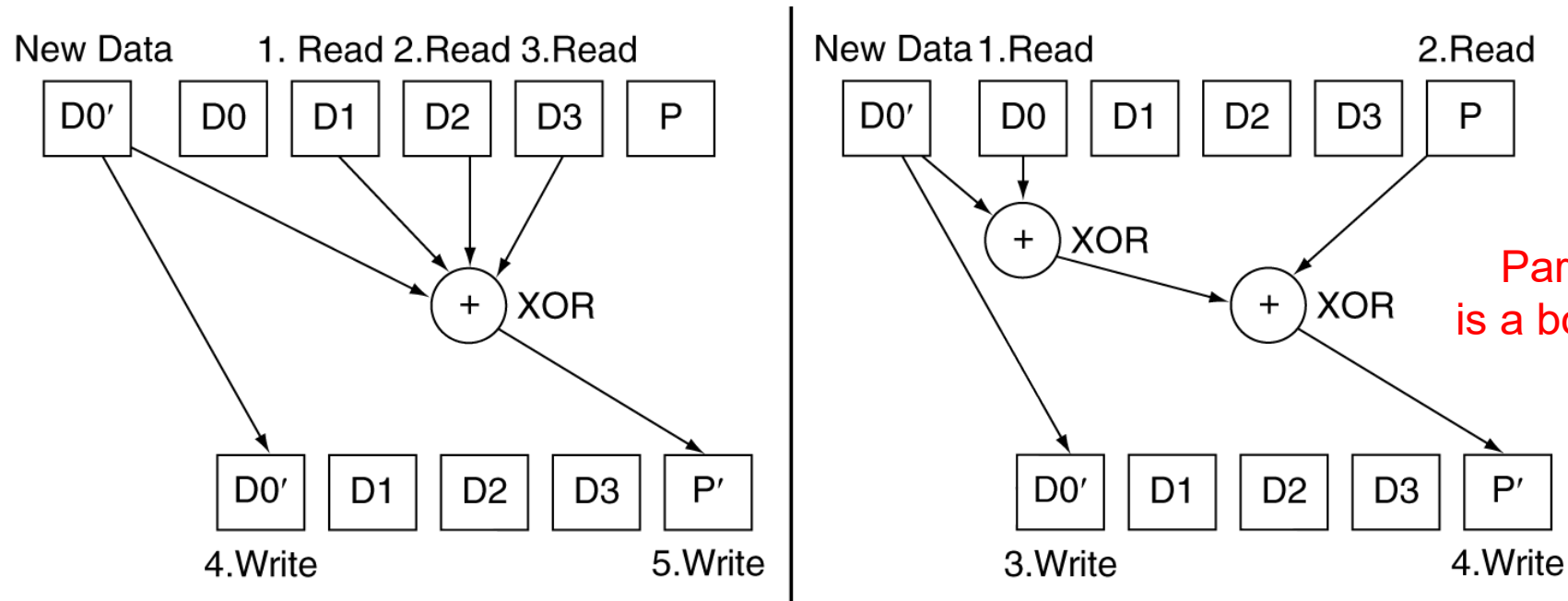
Aside:

$$P = D0 \oplus D1 \oplus D2 \oplus D3$$

$$\Rightarrow D1 \oplus D2 \oplus D3 = D0 \oplus P$$

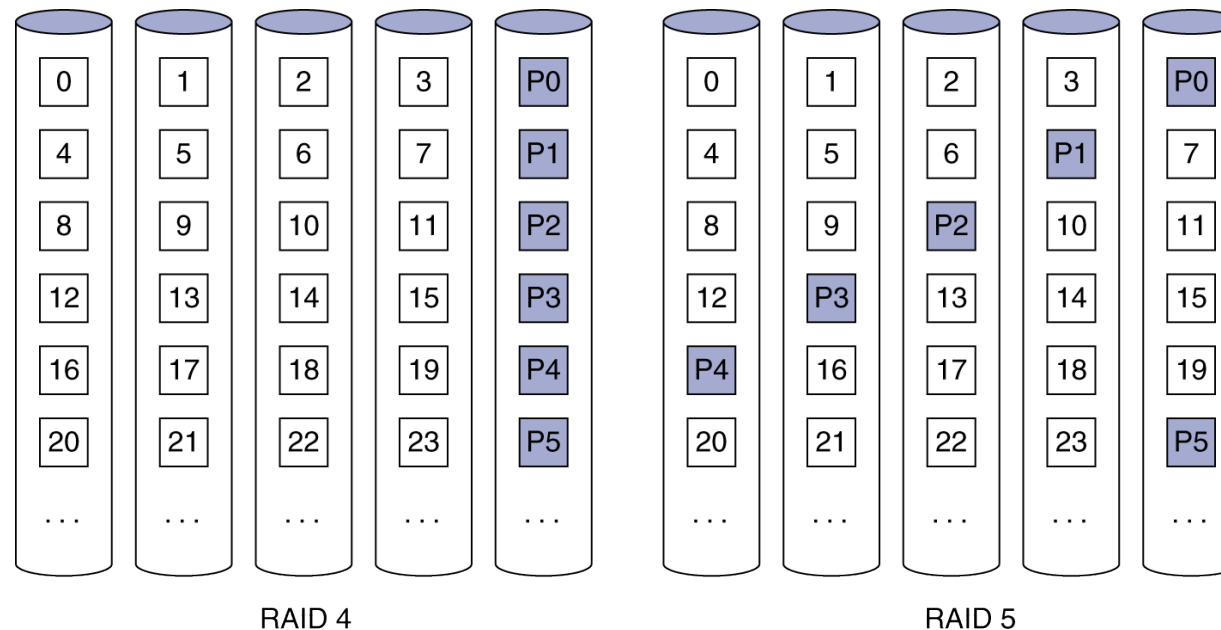
$$P' = D0' \oplus D1 \oplus D2 \oplus D3$$

$$= D0' \oplus D0 \oplus P$$





- ▶ RAID 5: distributed parity on $N + 1$ disks
 - Like RAID 4, but parity blocks distributed across disks
 - Avoids parity disk being a bottleneck
- ▶ Widely used





- ▶ RAID 6: P + Q redundancy on N + 2 disks
 - Like RAID 5, but two lots of parity
 - Greater fault tolerance through more redundancy

- ▶ RAID 1 and RAID 5 – widely used in servers
 - An estimate is that 80% of disks in servers are RAIDs

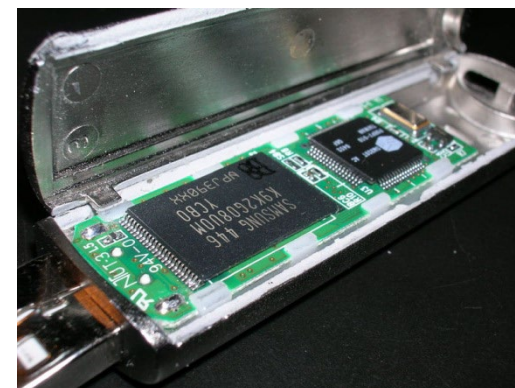


- ▶ High capacity
- ▶ Long-term storage without loss of information
- ▶ Other advantages eroded by improved disk technology
- ▶ <https://spectrum.ieee.org/computing/hardware/three-advances-make-magnetic-tape-more-than-a-memory>
- ▶ <https://spectrum.ieee.org/computing/hardware/why-the-future-of-data-storage-is-still-magnetic-tape>





- ▶ Non-volatile semiconductor storage
 - 100× – 1000× faster than disk
 - Smaller, lower power, more robust than disks
 - But more \$/GB (between disk and DRAM)
 - More resistant to physical shock
- ▶ 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 access only per block
 - Cheaper per GB
 - Used for USB keys, media storage, ...
- ▶ Flash bits wear out after 100K – 1M's of accesses
 - Wear leveling: remap data to less used blocks



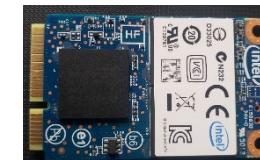


Solid State Drives

- ▶ Main storage component in modern computers
- ▶ Use of flash storage to store data permanently
- ▶ Various form factors exist. Most notable shown on the right
 - 2.5" SSD
 - Uses HDD form factor and older SATA interface
 - mSATA SSD (mini-SATA)
 - Introduced to fit in thin laptops
 - Still uses the SATA interface
 - M.2 SSD
 - Smaller form factor and different physical connection
 - Uses the newer NVMe interface
- ▶ There are more form factors out there (U.2, NF1, etc.)

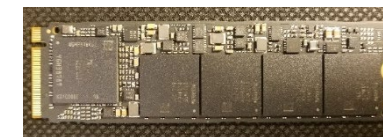


2.5" SSD



50,8 mm x
29,85 mm x
4,85 mm (2,00"
x 1,18" x 0,19")

mSATA SSD



22 mm x
{30, 42, 60, 80, 110} mm

M.2 SSD