

H1: Talstelsels en codesystemen

Inleiding

talstelsel: het stelsel waarin met een beperkt aantal cijfers elk getal kan worden uitgedrukt

- decimaal talstelsel: bevat 10 cijfers (0,1,2,3,4,5,6,7,8,9), grondtal = 10
- binair talstelsel: bevat 2 cijfers (0 en 1), grondtal = 2
- hexadecimaal talstelsel: bevat 16 cijfers (0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F), grondtal = 16

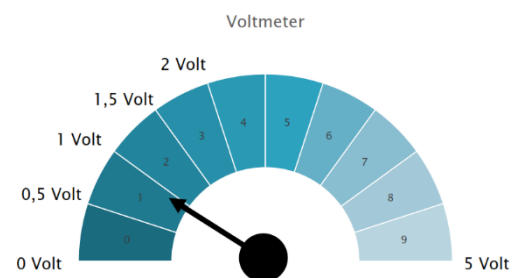
grondtal: de basis van een talstelsel

- Elk cijfer in een getal stelt een veelvoud van een macht van het grondtal voor
bv. decimaal talstelsel: $524 = (5 \times 10^2) + (2 \times 10^1) + (4 \times 10^0)$
 $= (5 \times 100) + (2 \times 10) + (4 \times 1)$
- De cijfers in een talstelsel zijn altijd kleiner dan het grondtal zelf
bv. decimaal talstelsel: $0,1,2,3,4,5,6,7,8,9 < 10$

naam	grondtal	cijfers	voordelen	nadelen	vb
decimaal talstelsel	10	0,1,2,3,4,5,6,7,8,9	Gebruikt in het dagelijkse leven.	Slecht bruikbaar om computers op elektronicaniveau te laten werken.	15 _{dec}
binair talstelsel	2	0,1	Goedkopere en meer betrouwbare computers.	Slecht leesbaar en verwerkbaar voor mensen, vooral bij grotere getallen.	1111 _{bin}
hexadecimaal talstelsel	16	0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F	Binaire getallen verkort voorstellen (o.a. MAC-adressen), makkelijke conversie	-	F _{hex}

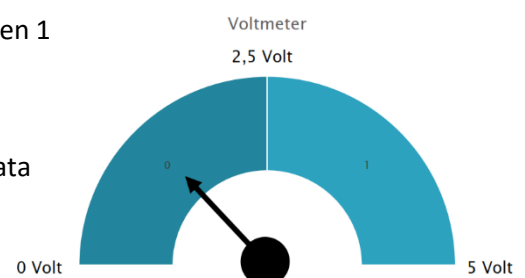
decimale computer: elk cijfer van het decimale talstelsel komt overeen met een bep. voltage in de 0/5 volt-range

- Technisch gezien mogelijk om te bouwen
- Zou duur zijn
- Zou foutgevoeliger zijn: wordt herkend als waarde 1, maar kan evengoed hevig vervormd waarde 2 zijn geweest
 - ↳ voltages verder uit elkaar leggen is geen oplossing, computers zouden minder snel kunnen werken en meer warmte ontwikkelen



binare computer: de 0/5 volt-range is opgedeeld in twee delen, nl. 0 en 1

- Goedkopere componenten
- Minder foutgevoelig
 - ↳ zelfs bij hevige vervorming van het signaal wordt de data nog steeds correct herkend als 0



Omzettingen

Binair naar decimaal

Neem elk cijfer van het binaire getal en vermenigvuldig dit met de overeenkomstige macht van 2 (Vergeet de 0-en ook niet te vermenigvuldigen met hun overeenkomstige macht)

$$\begin{aligned}
 11011_{\text{bin}} &= 1 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4 \\
 &= 1 \times 1 + 1 \times 2 + 0 \times 4 + 1 \times 8 + 1 \times 16 \\
 &= 1 + 2 + 0 + 8 + 16 \\
 &= 27_{\text{dec}}
 \end{aligned}$$

Hexadecimaal naar decimaal

Neem elk cijfer van het hexadecimale getal en vermenigvuldig dit met de overeenkomstige macht van 16

$$\begin{aligned}
 1A5B_{\text{hex}} &= 1 \times 16^0 + 5 \times 16^1 + 10 \times 16^2 + 1 \times 16^3 \\
 &= 11 \times 0 + 5 \times 16 + 10 \times 256 + 1 \times 4096 \\
 &= 11 + 80 + 2560 + 4096 \\
 &= 6747_{\text{dec}}
 \end{aligned}$$

Decimaal naar binair

2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
65536	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

1. Zoek in de tabel de hoogste macht dat in het te converteren getal past en schrijf dit bij je uitkomst.
2. Trek dit 'grootste getal' af van het te converteren getal. Dit geeft een restwaarde.
3. Loop in volgorde alle kleinere machten van 2 af.
 - ↳ Indien de macht in het restgetal past: trek het af van het restgetal en schrijf een 1 in de binaire schrijfwijze
 - ↳ Indien de macht niet in het restgetal past: schrijf een 0 in de binaire schrijfwijze

Decimaal naar hexadecimaal (en omgekeerd)

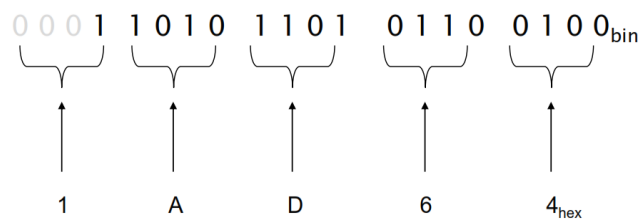
8	7	6	5	4	3	2	1
0	0	0	0	0	0	0	0
1	268435456	1	16777216	1	1048576	1	65536
2	536870912	2	33554432	2	2097152	2	131072
3	805306368	3	50331648	3	3145728	3	196608
4	1073741824	4	67108864	4	4194304	4	262144
5	1342177280	5	83886080	5	5242880	5	327680
6	1610612736	6	100663296	6	6291456	6	393216
7	1879048192	7	117440512	7	7340032	7	458752
8	2147483648	8	134217728	8	8388608	8	524288
9	2415919104	9	150994944	9	9437184	9	589824
A	2684354560	A	167772160	A	10485760	A	655360
B	2952790016	B	184549376	B	11534336	B	720896
C	3221225472	C	201326592	C	12582912	C	786432
D	3489660928	D	218103808	D	13631488	D	851968
E	3758096384	E	234881024	E	14680064	E	917504
F	4026531840	F	251658240	F	15728640	F	983040

1. Zoek het grootste getal in de tabel dat in het te converteren getal past.
2. Schrijf het getal dat links van dit 'grootste getal' staat op als deel van je uitkomst (ook 0!)
3. Trek dit 'grootste getal' af van het te converteren getal. Dit geeft een restwaarde.
4. Ga naar de volgende kolom, en herhaal tot er geen rest meer is.

Hexadecimaal naar binair

1. Zet elk cijfer van het hexadecimaal getal individueel om naar het overeenkomstige binaire getal.
2. Schrijf de gevonden binaire getallen op in dezelfde volgorde als de hexadecimale cijfers.

Decimaal	Binair	Hexadecimaal
0 _{dec}	0000 _{bin}	0 _{hex}
1 _{dec}	0001 _{bin}	1 _{hex}
2 _{dec}	0010 _{bin}	2 _{hex}
3 _{dec}	0011 _{bin}	3 _{hex}
4 _{dec}	0100 _{bin}	4 _{hex}
5 _{dec}	0101 _{bin}	5 _{hex}
6 _{dec}	0110 _{bin}	6 _{hex}
7 _{dec}	0111 _{bin}	7 _{hex}
8 _{dec}	1000 _{bin}	8 _{hex}
9 _{dec}	1001 _{bin}	9 _{hex}
10 _{dec}	1010 _{bin}	A _{hex}
11 _{dec}	1011 _{bin}	B _{hex}
12 _{dec}	1100 _{bin}	C _{hex}
13 _{dec}	1101 _{bin}	D _{hex}
14 _{dec}	1110 _{bin}	E _{hex}
15 _{dec}	1111 _{bin}	F _{hex}



Voorstelling in de computer

bit: een cijfer van een binair getal in een computer, dus met als waarde 0 of 1

byte: meeste bekende grootte van gegroepeerde bits

Naam	Aantal bits
nibble	4 bits
byte/cel	8 bits
word	16 bits
double word	32 bits

Positieve/negatieve getallen

Two's complement

De eerste bit van een byte doet dienst als tekenbit:

- 0 = positief getal (gewoon omrekenen naar decimaal talstelsel om de waarde te kennen)
- 1 = negatief getal (alle bits inverteren, laatste bit wordt 1 en omrekenen naar decimaal talstelsel)
 - ↳ bv. opgave: 1000 0101_{bin} → 0111 1010_{bin} → 0111 1011_{bin} → -123_{dec}

Waarom? Het aftrekken van 2 getallen komt overeen met het optellen van diezelfde getallen, waarvan er één getal negatief werd gemaakt. (= makkelijk voor computers)

Signed/unsigned

- signed number: zowel positieve als negatieve getallen
 - ↳ kan grotere waarden bewaren, omdat een tekenbit ontbreekt
 - ↳ kan even veel verschillende waarden opslaan als een unsigned number
- unsigned number: enkel positieve getallen

Kommagetallen

IEEE-notatie (Institute of Electrical and Electronic Engineers)

= *Gestandaardiseerde manier van (komma)getalrepresentatie (geen conversies bij uitwisseling)*

- Wetenschappelijke notatie: mantisse x grondtal^{exponent} = $m \times g^e$
 - bv. $125,21 = 1,2521 \times 10^2$
 - $101_{\text{bin}} = 1,01000_{\text{bin}} \times 2^2$
- Binaire getallen omzetten:

$$\begin{aligned}
 110,11_{\text{bin}} &= 1 \times 2^{-2} + 1 \times 2^{-1} + 0 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 \\
 &= 1 \times \frac{1}{4} + 1 \times \frac{1}{2} + 0 \times 1 + 1 \times 2 + 1 \times 4 \\
 &= 0.25 + 0.5 + 0 + 2 + 4 \\
 &= 6.25_{\text{dec}}
 \end{aligned}$$
- IEEE-notatie: gestandaardiseerde notatie die vastlegt:
 - Hoeveel bits gebruikt wordt voor de *mantisse*
 - Hoeveel bits gebruikt wordt voor de *exponent*
 - De manier waarop de *tekenbit* wordt opgeslagen
- Single Precision: 32 bits in totaal, bestaande uit 1 tekenbit, 8 exponentbits en 23 mantissebits
- Double Precision: 64 bits in totaal, bestaande uit 1 tekenbit, 11 exponentbits en 52 mantissebits
- Sommige getallen uit het decimale talstelsel kunnen niet exact worden voorgesteld in het binaire talstelsel: bv. $0.1_{\text{dec}} = 0.001100110011001100\ldots_{\text{bin}}$

Floating point naar decimaal

1. Indien in hexadecimale notatie → omzetten naar binaire notatie
2. Verdeel de bits in teken-, exponent- en mantissebits
 - Tekenbit (0 = positief, 1 = negatief)
 - Exponent (zet om naar decimaal en verminder met 127 bij single en 1023 bij double precision)
 - Mantisse (voeg extra 1 toe links aan mantissebits en reken uit)
3. Reken uit: mantisse $\times 2^{\text{exponent}}$

Decimaal naar floating point

1. Getal omzetten naar binair
2. Getal omzetten naar wetenschappelijke notatie (normaliseren): mantisse $\times 2^{\text{exponent}}$
 - tot 1 getal voor de komma
3. Uiteindelijke bits samenstellen:
 - a. Getal na de komma = mantisse (+ verder aanvullen met nullen tot er 23 mantissebits zijn)
 - b. Exponent = macht + 127 (bij single precision) of 1023 (bij double precision)
 - c. Tekenbit zetten (0 = positief, 1 = negatief)

Bereik

bereik: de mogelijke waarden die kunnen worden bijgehouden in een aantal bits, rekening houdend met: signed/unsigned, al dan niet kommagetal en de grootte van de groep van bits

Tekens/symbolen

ASCII (American Standard Code for Information Interchange)

- Eerste standaard voor het voorstellen van tekens (symbolen/letters/cijfers)
- 7bits per teken
- Mapping door middel van tabel
- Aan sommige tekens is geen weergave verbonden: linefeed, carriage return, tab, end of file, ...
- Bij cijfers komen de ASCII-waarde en de binaire waarde niet overeen
- Extended ASCII tabel: 8 bits per teken i.p.v. 7 bits (256 tekens i.p.v. 128)

Unicode

- Uitgebreidere versie van ASCII (ook Russische, Griekse, Chinese e.a. karakters)
 - biedt een uniek getal voor elk teken, ongeacht het platform, het programma of de taal
 - bespaart kosten (geen aanpassingen nodig)
- Zeer brede ondersteuning (IBM, Apple, HP, IBM, Microsoft, Oracle, SAP, Sun, ... + interne representatie van symbolen in .NET, Java)
- Nu onderhouden door Unicode Consortium
- UTF-8
 - Variabele lengte (1 tot 4 bytes)
 - Maximale compatibiliteit met ASCII (eerste 128 karakters komen overeen)
 - Kan alle mogelijke tekens die in Unicode werden opgenomen weergeven
 - Onder andere gebruikt op Internet
 - In vele gevallen compacter dan UTF-16
- UTF-16
 - Variabele lengte (1 tot 2 keer 16 bits)
 - Kan alle mogelijke tekens die in Unicode werden opgenomen weergeven
 - Niet zo compatibel als UTF-8

EBCDIC (Extended Binary-Coded Decimal Interchange Code)

- Op IBM mainframes gebruikt
- Nu in onbruik geraakt (tegenwoordig meestal ASCII of UNICODE)

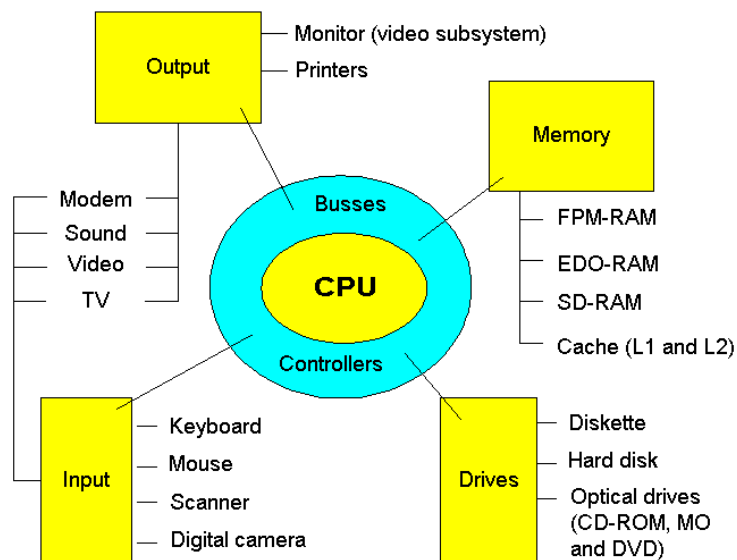
H2: Computerarchitectuur

Von Neumann-architectuur

John von Neumann: Hongaarse wiskundige die als eerste een bruikbaar model opstelde van hoe een computer in elkaar zit (1945)

↳ 5 hoofdonderdelen:

- CPU/CVE (central processing unit/centrale verwerkingseenheid)
= *meest centrale component van een computersysteem, doet uiteindelijk al het werk*
- Invoer (o.a. muis, toetsenbord, scanner, digitale camera)
- Uitvoer (o.a. monitor, printers)
- Werkgeheugen (o.a. RAM, caches)
= *vluchtig geheugen voor het inladen van programma's voor uitvoering*
- Permanent geheugen (o.a. diskettes, harde schijven, CD-ROM's, DVD's)
= *geheugen dat niet verloren geraakt bij stroomuitval*
- (Chips en bussen)
= *bussen verbinden alle componenten, chips leiden de onderlinge werking van de onderdelen in goede banen (niet te onderschatten componenten in een computersysteem!)*



© Michael B. Karbo 1997

H3: Het werkgeheugen

Geheugenkarakteristieken

geheugencapaciteit: de hoeveelheid informatie dat men kan bewaren in een geheugen

→ eenheden: bits, nibble, bytes, woorden (32/64/... bits)

Afkorting	Eenheid	
1 Kb	1 kilobit	1024 bits
1 Mb	1 megabit	1024^2 bits = 1 048 576 bits
1 Gb	1 gigabit	1024^3 bits = 1 073 741 824 bits
1 KB	1 kilobyte	1024 bytes
1 MB	1 megabyte	1024^2 bytes = 1 048 576 bytes
1 GB	1 gigabyte	1024^3 bytes = 1 073 741 824 bytes

informatiedichtheid: aantal bits per oppervlakte-eenheid

→ bv. aantal bits per mm²

toegankelijkheid: op welke manier de bits uit het geheugen kunnen opgevraagd worden

- onmiddellijke toegang: tijd nodig om een woord uit het geheugen te roepen is praktisch onafhankelijk van de plaats waar het woord zich bevindt.
 - ↳ statisch (geen bewegende onderdelen)
 - ↳ bv. RAM = Random Access Memories
- sequentiële toegang: informatie zit in een welbepaalde volgorde in het geheugen
 - ↳ bv. magneetband
- cyclische toegang: Niet zo sequentieel, quasi-onmiddellijke toegang
 - ↳ bevatten meestal bewegende onderdelen, soms statisch
 - ↳ bv. magneetschijf (harde schijf, diskette, ...)

adreseerbaarheid van geheugens

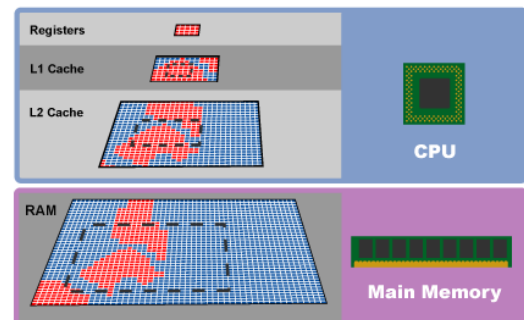
- kleinste adresseerbare eenheid: geheugencel (meestal 1 byte groot)
- volgens toegankelijkheid
 - *onmiddellijk*: individueel adresseerbaar (per byte)
 - ↳ een geheugenregister bestaat uit meerdere geheugencellen
 - ↳ adres van een geheugenregister = adres van de eerste geheugencel van het register
 - *sequentieel*: niet adresseerbaar
 - ↳ grote blokken van willekeurige lengte met spaties tussen
 - *cyclisch*: blokadresseerbaar (blokken van bv. 64 of 512 bytes)
 - ↳ de blokken gegevens die worden uitgewisseld tussen geheugens en het werkgeheugen zijn meestal van voorafbepaalde lengte
 - ↳ elk blok heeft een adres

bestendige/vluchtige gegevens

- bestendige gegevens: inhoud blijft behouden als de computer uitgeschakeld wordt, of wanneer elektrische stroom uitvalt
- vluchtige gegevens: inhoud gaat verloren wanneer de stroom uitvalt, maar kunnen snel informatie opnemen/afgeven

Cachegeheugens

- **Waarom?**
 - werkgeheugens kunnen niet volgen met de alsmaar grotere snelheden van nieuwe CPU's
 - het werkgeheugen is te ver verwijderd van de CPU, waardoor de CPU meer en meer werkloos is tijdens cycli
 - oplossing: cachegeheugens slaan gegevens/instructies op die veel gebruikt zijn/zullen worden dicht bij de CPU
- **Locality of Reference**: als de processor recent een bepaalde locatie in het geheugen heeft gebruikt, is de kans groot dat in de nabije toekomst de processor een nabije locatie zal nodig hebben.
 - wanneer we het gedeelte van het werkgeheugen dat momenteel in uitvoering is naar het cache-geheugen brengen, is de kans groot dat de processor de nodige informatie direct daartussen zal vinden
- **Cache-levels**:
 - Level 1-cache: zeer snel, klein, heel dicht bij CPU
 - tijdelijke opslag van veelgebruikte instructies en data
 - meestal gebouwd d.m.v. SRAM (Static Random Access Memory)
 - *zero wait-state interface* (wachtijd is verwaarloosbaar klein)
 - beperkt in grootte door directe toegang (vroeger 16KB, nu tot 128KB)
 - vroeger: *write-through* (alles in het cachegeheugen wordt volledig in het werkgeheugen bewaard)
 - nu: *write-back* (enkel in het cachegeheugen, naar werkgeheugen indien cache entry wordt vervangen)
 - Level 2-cache: iets trager en groter dan L1-cache, minder dicht bij CPU
 - brug tussen L1-cache en geheugen
 - gebouwd d.m.v. SRAM (Static Random Access Memory)
 - speciale bus om hoge snelheden te halen
 - evolutie in grootte (vroeger 512KB, nu meestal 1MB tot 12MB)



Werkgeheugens en soorten geheugens

- werkgeheugen: de brug tussen de processor (L2-cache) en de harde schijf
- communicatie met de processor gebeurt d.m.v.:
 - adresbussen (worden gebruikt om informatie door te geven i.v.m. adressering)
 - breedte bepaalt hoeveel geheugen er adresseerbaar is
 - tegenwoordig 36 bits/64 G geheugenplaatsen
 - databussen (vervoerbanden voor de gegevens zelf)
 - breedte bepaalt hoeveel data op 1 moment kan worden verplaatst
 - tegenwoordig 64-bits/8 bytes per keer
- bus-cyclus: elke transactie tussen processor en werkgeheugen
- aantal bits dat de CPU kan verwerken bepaalt het type geheugen
- 2 belangrijke soorten
 - Static RAM (SRAM)
 - gemaakt uit bipolaire transistors (4-6 transistors per opgeslagen bit)
 - gegevens worden behouden zolang er stroom wordt geleverd
 - zeer snel/duur/fysiek groot (in vergelijking met DRAM)
 - gebruikt meer energie dan DRAM
 - vooral voor cachegeheugens/registers
 - elk jaar 40% sneller (in rechte lijn met CPU-snelheden)
 - bv. SRAM-cel met D flip-flop
 - Dynamic RAM (DRAM)
 - bestaat uit een al dan niet geladen condensator (verliest lading snel)
 - ↳ gegevens worden constant herschreven om behouden te worden
 - ↳ gebruik van refresh-circuit (100+ keer/seconde herschreven)
 - ↳ per opgeslagen bit wordt 1 transistor en 1 condensator gebruikt
 - trager/goedkoper/fysiek kleiner dan SRAM
 - gebruikt minder energie dan SRAM
 - vooral voor werkgeheugens
 - elk jaar 7% sneller

DDR RAM

- snelheid wordt uitgedrukt in **Hertz** (aantal clock cycles per seconde)
 - 100 MHz = 100 miljoen clock cycles per seconde
 - nieuwste processoren: 3,8 GHZ = 3,8 miljard clock cycles per seconde
- **DDR RAM** (Double Data Rate RAM)
 - soort SDRAM (Synchronous Dynamic RAM)
 - werkt synchroon (aan dezelfde snelheid als de bus)
 - vandaag de dag gebruikt
 - snelheid is dubbel zo groot dan normaal, zonder de kloksnelheid aan te passen
 - ↳ Traditioneel: data geleverd wanneer klok omhoog gaat (klok 0 → 1)
 - ↳ DDR: data geleverd zowel als klok 0 → 1 en 1 → 0
 - introductie in 1999, door *NVIDIA GeForce* grafische kaarten
 - ↳ *AMD*: overgeschakeld in 2000, zorgde voor doorbraak
 - ↳ *Intel*: niet direct overgeschakeld (meer winst op SDRAM, pas begin 2002 veranderd)

- chips worden meestal benoemd door snelheid:
 - DDR266 werkt op 133 MHz bus
 - 266 MT/s (266 miljoen transfers per seconde) (verdubbeling snelheid)
- soms wordt ook de schrijfsnelheid van het geheugen vermeld
 - PC1600 is DDR-versie van PC100 SDRAM
 - gebruikt DDR200 chips en kan 1600 MBps aan ($100\text{MHz} \times 2 \text{ (DDR)} \times 64 \text{ bits} = 1600$)
- snelheid kan berekend worden met een formule:
$$\text{snelheid} = \text{kloksnelheid} * 2 \text{ (DDR)} * 64 \text{ (aantal overgebrachte bits)} / 8 \text{ (aantal bits/byte)}$$
- **Dual channel**: twee kanalen op moederbord die tegelijk kunnen aangesproken worden
 - ↳ DIMMs = Dual in-line memory modules
 - ↳ verbetering van ongeveer 10%
 - ↳ Werkt alleen als DIMMs:
 - in paren geïnstalleerd zijn (twee aparte DIMMs)
 - zelfde grootte/snelheid hebben
 - allebei single of double sided zijn
 - ↳ bijna geen verschil in praktijk → marketingfactor?
 - ↳ DDR RAM = meestal triple/quadruple channel
- **DDR2**
 - ↳ werkt nog steeds op basis van externe (bus)klok, zowel bij op- en neergaan
 - ↳ voordeel: meer gegevens doorstuurbaar (kloksnelheid van bus = 2 x kloksnelheid geheugen)
 - ↳ nadeel: grotere latency (3-9 bus cycles latency i.p.v. 2-3 bus cycles latency bij DDR)
 - ↳ door latency is DDR beter dan DDR2 op zelfde bussnelheid
- **DDR3**
 - ↳ nogmaals zelfde principe toegepast als bij overgang DDR naar DDR2
 - ↳ meest gebruikt (de meeste moederborden gebruiken enkel DDR2 of DDR3, grafische kaarten GDDR2 of GDDR3))

De geheugenpiramide



Level	Toegangstijd	Typische grootte	Technologie
Registers	1-3 ns	1 KB	CMOS
Level 1-cache	2-8 ns	8KB – 128KB	SRAM
Level 2-cache	5-12 ns	512 KB – 8 MB	SRAM
Werkgeheugen	10-60 ns	4 GB – 16 GB	DRAM
Harde schijf	3 000 000 – 10 000 000 ns	1 TB – 4 TB	Magnetisch

H4: Processoren

Inleiding

processor: het brein van een computer; voert instructies van een computerprogramma uit door een aantal (basis)operaties uit te voeren in machinetaal:

- Wiskundige operaties (plus, maal, ...)
- Logische operaties (and, or, not, ...)
- Input/outputoperaties aansturen

microprocessor: de volledige CPU zit vervat in één chip

- alle moderne CPU's

register: het 'werkblad' van de processor, waar de bewerkingen rechtstreeks op worden uitgevoerd

- ingebakken in de processor
- aantal/grootte is afhankelijk van de processor

Instructies

instructieset: de verzameling van bevelen die een processor verstaat en kan uitvoeren

- basisinstructieset: basisinstructies voor 'gewoon' gebruik
- instructies voor optimalisatie op bepaalde gebieden:
 - MMX (*MultiMedia eXtension Technology*)
 - ↳ optimaliseert multimedia-toepassingen
 - SSE (*Streaming SIMD extensions*)
 - ↳ uitgebreidere versie van MMX
 - ↳ vandaag SSE4 in i7 processor

low-level programmeertaal:

- complexe functionaliteiten zijn uitgesplitst naar atomaire opdrachten (basis-wiskundige berekeningen)
- moeilijk te begrijpen voor mensen
- makkelijk te begrijpen voor een CPU (moeten niet of nauwelijks geconverteerd worden)
- bv. assembler, machinetaal

machinetaal: low-level programmeertaal die specifiek is aan een bepaalde CPU(familie)

- een programma dat naar machinetaal werd omgezet zal dus niet op eender welke CPU werken
- gelukkig bestaan er grote families van CPU's die compatibel zijn met elkaar, zodat programma's in machinetaal toch kunnen uitgewisseld worden.

assembler: een voorstelling van machinetaal die voor een mens leesbaar is (maar nog steeds atomair is)

high-level programmeertaal:

- gemakkelijker te begrijpen voor mensen
- laten toe om op eenvoudigere manier complexe functionaliteiten te maken
- moeten omgezet worden naar machinetaal vooraleer ze kunnen worden uitgevoerd
- bv. Java (Processing), C++, C#, ...

Basiseigenschappen

Snelheid

kloksnelheid: het aantal pulsen dat de in de processor ingebouwde elektronische klok per seconde uitstuurt

- per puls van deze klok zal de processor een bepaalde bewerking uitvoeren
- in een computer zijn er ≠ klokken aanwezig, vaak werken deze allemaal op een ≠ snelheid
- uitgedrukt in:
 - Hertz (eenheid van frequentie)
 - MIPS (Millions of Instructions Per Second)
 - geen goede maatstaf voor snelheid
 - bepaalde instructies zijn krachtiger dan andere
 - door pipelining/branching moeten sommige instructies hernomen worden
- moderne consumentenprocessoren hebben kloksnelheden in het 1-4 GHz bereik (1 tot 4 miljard keren per seconde voeren zij dus een bewerking uit)

Wet van Moore: Gordon Moore (medeoprichter van Intel) stelde in 1965 dat het aantal transistors per vierkante centimeter zou verdubbelen per 2 jaar

- dit neemt ingrijpende snelheidsverhogingen met zich mee
- is tot op heden min of meer juist gebleken, maar zou niet lang meer juist blijven

32/64 bit

= grootte van de interne registers die een processor gebruikt

- vroeger 32 bit-processoren, sinds enkele jaren alleen nog 64 bit-processoren
- theoretisch: een 64 bits-processor kan 2x zoveel gegevens verwerken in dezelfde tijd
- in praktijk: andere zaken belangrijker voor verwerkingssnelheid:
 - hoeveelheid cache
 - multithreading/multi-core (meerdere gegevens tegelijk verwerken)
- voordelen:
 - (theoretische) snelheidswinst
 - mogelijkheid om meer geheugen te adresseren (32 bit = max 4GB RAM)

Hyperthreading

Klassiek gezien heeft elke applicatie één proces

threads: taken in een applicatie die simultaan kunnen worden uitgevoerd (bv. geluid, beelden in een game)

moeten expliciet worden geprogrammeerd in een applicatie

- een CPU met 1 core en zonder hyperthreading zal zeer snel afwisselen tussen:
 - meerdere processen
 - meerdere threads binnen een proces
- een hyperthreading CPU kan:
- in bepaalde gevallen meerdere threads tegelijk uitvoeren met één core
- de meeste besturingssystemen interpreteren dit als een extra CPU

Multi-core

multi-core processor: microprocessorchip met meerdere cores (microprocessors)

- Dual Core = 2 cores/chip, Quad Core = 4 cores/chip
- processen worden toegewezen aan aparte processoren (meer threads tegelijk draaien)
- cores delen meestal het cache-geheugen

system on a chip (SoC): Een chip die zowel een CPU als andere randapparatuur (audio, video, ...) en componenten van een computersysteem bevat (o.a. in smartphones)

multiprocessing: een computer met meer dan één processor

Functionele onderdelen van een CPU

- Arithmetic Logic Unit (ALU): voert wiskundige en logische operaties uit
- Floating Point Unit (FPU): verzorgt alle bewerkingen op floating point-getallen (veel ingewikkelder dan two's complement getallen → veel complexer dan de ALU)
- Registers: tijdelijke opslagplaatsen voor de bitreeksen die de CPU aan het bewerken is
- Bus Unit: regelt het dataverkeer van en naar de processor
- Cache: een soort buffer van data die uit het werkgeheugen komt (om sneller te kunnen werken)
- Decision-making en instruction prediction: probeert te voorspellen welke tak van een programma gekozen wordt, om op die manier bepaalde gedeelten van een programma in te laden (pipelining)
- Control Unit: controleert/coördineert de werking van andere componenten (haalt instructies uit het geheugen, decodeert deze gegevens en voert ze uit gebruik makende van de ALU)

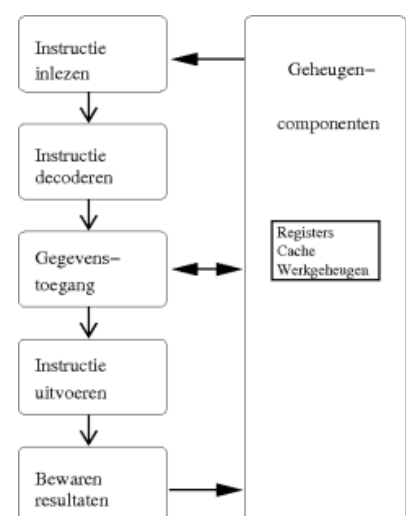
Processorcyclus

fetch-decode-execute cyclus:

1. Inlezen instructie: uit te voeren instructie inlezen uit geheugen (cache, RAM, permanent geheugen)
2. Instructie decoderen: analyse van wat de instructie doet
3. Gegevenstoegang: ophalen gegevens uit geheugen indien nodig
4. Instructie uitvoeren: de gedecodeerde instructie is klaar voor uitvoering
5. Bewaren resultaten: indien nodig naar geheugen (cache, RAM, permanent)

Elke processorarchitectuur implementeert deze stappen op zijn manier

- de stappen komen wel altijd terug
- soms worden stappen gecombineerd (performantiewinst)



Pipeline

= de *prefetch queue* en het systeem om instructies op een rij klaar te zetten voor uitvoering

- pas wanneer een instructie alle stappen van de pipeline heeft doorlopen, is de instructie uitgevoerd.
- per tik van de klok (kloksnelheid) van de processor, zal een instructie een stap van de pipeline kunnen doorlopen.

Hoe meer stappen de pipeline bevat, hoe "langer" of hoe "dieper" de pipeline is, en:

- de taken die per stadium moeten worden uitgevoerd zijn kleiner
- de elektronische schakeling per stap is minder complex
- er is minder tijd nodig om één stap van de pipeline uit te voeren
- er kan een hogere kloksnelheid gekozen worden voor de processor

Hoe minder diep de pipeline is:

- de taken die per stadium moeten worden uitgevoerd zijn groter
- de elektronische schakeling per stap is complexer
- er is meer tijd nodig om één stap van de pipeline uit te voeren
- men is gedwongen om voor een lagere kloksnelheid te kiezen

branch prediction: toekomstvoorspelling om de *prefetch queue* gevuld te houden met instructies als deze op voorhand niet zeker zijn (bv. bij een if/else-structuur)

- voorspelling is correct: de processor kan sneller verder werken, de pipeline was correct gevuld.
- voorspelling is niet correct:
 - de juiste instructies moeten aan de pipeline worden aangeleverd
 - heel de pipeline moet opnieuw doorlopen worden.
 - er worden x aantal klokslagen verloren, waarbij x = de lengte van de pipeline

Fysieke eigenschappen van een processor

Plaatsing

= fysiek op het moederbord van een computer d.m.v. een socket

- zorgt ervoor dat:
 - de processor fysisch op zijn plaats blijft zitten
 - de processor elektrische signalen kan uitwisselen met het moederbord en alle andere onderdelen van de computer, zoals het RAM-geheugen (zijn ook aan het moederbord verbonden)
- verschillende soorten sockets, op basis van aantal/layout/vorm van de contactpunten
- een CPU die voor een bepaalde socket werd gemaakt, kan enkel op een moederbord kan geïnstalleerd worden dat die bepaalde socket ondersteunt.

Opbouw

Alles op 1 chip => geïntegreerde schakeling

Bestaat uit miljarden transistoren

- grondstof is silicium (hoofdbestanddeel van zand)
- dit wordt gezuiverd en specifieke onzuiverheden worden toegevoegd om er halfgeleiders van te maken

Processoren worden met een optisch proces met meerderen tegelijk "gefotografeerd" op **wafers** (grote ronde schijven)

- deze worden dan verzaagd om de individuele processoren te verkrijgen
- niet alle "fotografisch" gevormde processoren kunnen worden gebruikt, er is altijd afval
- yield: het percentage van opbrengst uit 1 wafer

Processoren worden in principe steeds

- complexer → meer transistors
 - sneller → meer warmteontwikkeling (zelfs onder normale omstandigheden)
 - energiegulziger
- Om deze problemen voor te blijven, poogt men de processoren **steeds kleiner** te maken, met een grotere **transistor density**:
- 10 micrometer (jaren '70) → 22 nanometer (nu) → 11 nanometer (in de toekomst)

koeling houdt processoren binnen hun *werkingstemperatuur* (temperatuurrange waarin zij geacht worden om in te werken)

- passieve koeling:
 - metalen raster dat de warmte kan afvoeren (meestal aluminium/koper)
 - geen bewegende onderdelen (stil)
- actieve koeling:
 1. koperen plaat op de processor die de warmte kan afvoeren
 2. ventilator, aangedreven door motor (niet stil)
 3. warmtegeleidende pasta tussen processor en koeling (noodzakelijk)
 - kleiner dan passieve koeling bij zelfde koelcapaciteit.

behuizingen van computers worden gemaakt volgens bep. standaarden (afmeting/plaats van componenten)

- bv. ATX (verbetering van de koelingsmogelijkheden t.o.v. de AT) en ITX

Geschiedenis

- oorspronkelijk: op basis van electronicabuizen (zeer groot)
- in de loop der jaren: meer mogelijkheden en sneller (hogere kloksnelheid)
- nu: microprocessoren (transistoren die op één enkele chip passen)
- vroeger gebruikte Apple-processoren een andere CPU/instructieset, maar tegenwoordig zelfde CPU
- grote processorfabrikanten:
 - PC: Intel/AMD
 - mobiele apparaten (ARM): Qualcomm, NVidia, Samsung, Freescale, Texas Instruments
- CPU's gericht op nichemarkten:
 - Intel M (energiebesparend → laptops)
 - Intel Atom (energiebesparend → netbooks)

H5: Chipsets en bussen

Inleiding

moederbord: een grote printplaat die alle andere componenten fysiek met elkaar verbindt d.m.v. bussen

bussen: zorgen voor de overdracht van gegevens tussen verschillende componenten in de computer

- kloksnelheid van de bussen is aangepast aan de toepassing (CPU, keyboard, grafische kaart, ...)
- een computer bevat meestal 2 belangrijke bussen:
 - *systeembus (local bus/front-side bus/FSB/processorbus)*
 - ↳ verbindt de microprocessor met het werkgeheugen, I/O-apparaten, ...
 - ↳ hoge snelheid door de hoge bandbreedte (processor heeft veel gegevens nodig)
 - *shared bus*
 - ↳ verbonden met systeembus via bridge
 - ↳ gedeeld door ≠ I/O-apparaten (o.a. geluidskaart, modem, harde schijf)
 - ↳ bv. ISA, EISA, PCI, PCI-Express

Bussen

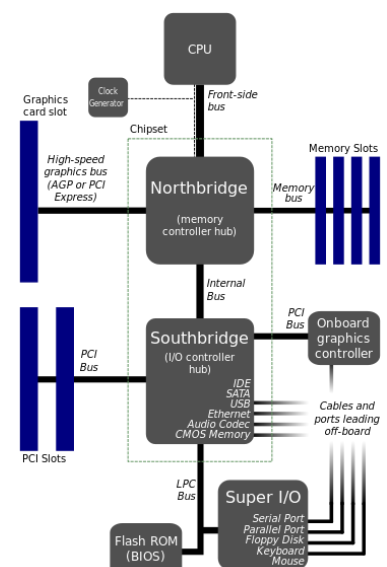
bus: aantal bij elkaar horende 'lijnen' die gedeeld worden door meerdere apparaten

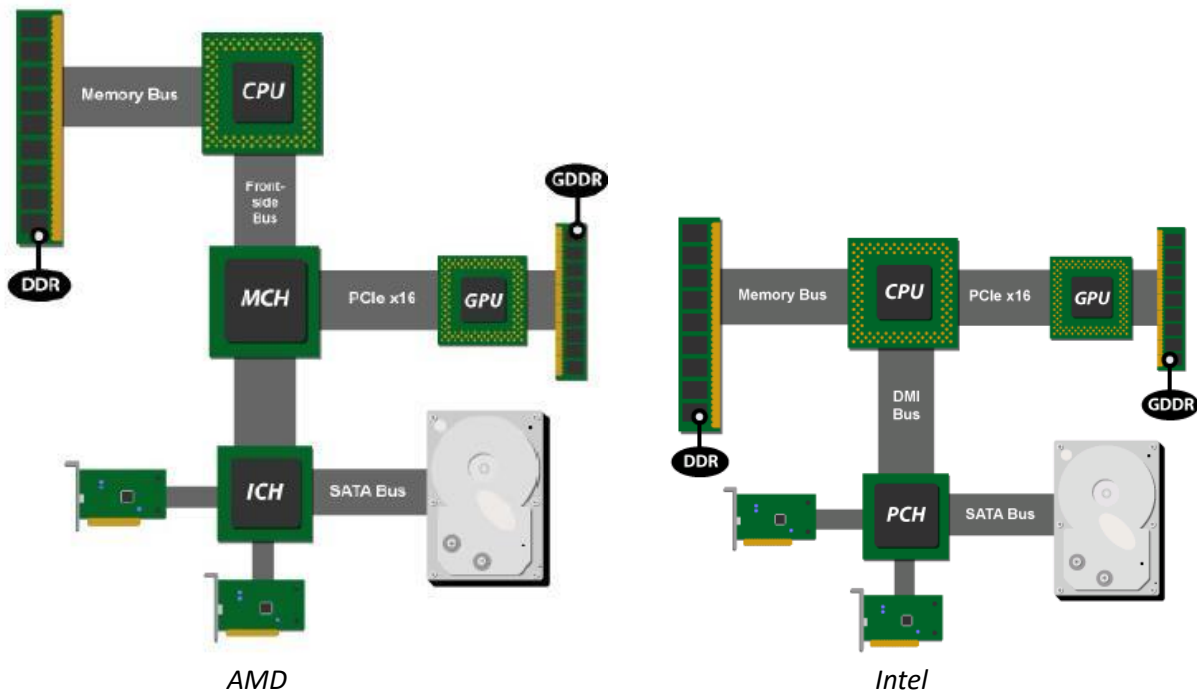
- databus: gegevens transporteren tussen processor en werkgeheugen of I/O apparaten
 - ↳ tweerichtingsverkeer
 - ↳ moet vrij zijn voor gegevenstransport kan plaatsvinden
 - ↳ breedte bepaalt hoeveel gegevens tijdens 1 klokcyclus kunnen worden verplaatst
 - vroeger: 8 bits (dus 1 byte tegelijkertijd)
 - nu: 64 bits (dus 8 bytes tegelijkertijd)
- adresbus: bepaalt de richting van het verkeer op de databus
 - ↳ eenrichtingsverkeer van processor naar werkgeheugen of I/O apparaat
 - ↳ breedte bepaalt o.a. hoeveel geheugen maximaal gebruikt kan worden
 - vroeger: 20 bits
 - nu: 36/40/44 bits
- controlebus: bevat het kloksignaal en een signaal dat bepaalt of het over een lees- of schrijfoperatie gaat

Geschiedenis

vroeger: een computer had een **north- en southbridge** op het moederbord geïnstalleerd

- northbridge: de verbinding tussen CPU/RAM/AGP-poort (grafische kaart)/southbridge)
 - de front-side bus verbindt de northbridge met de processor
 - tegenwoordig zit de memory controller (RAM) ingebouwd in de CPU (snel ler)
- southbridge: verbinding met tragere apparaten (harde schijven, toetsenbord, USB, PCI, ...)
 - verschillende bussystemen: PCI, USB, EIDE
- ↳ nog steeds aanwezig bij Intel Atom, Celeron, Pentium en Core 2 processoren





(MCH = Northbridge; PCH/ICH = Southbridge)

Chipsets

= verzameling van chips die op een moederbord voorkomen en die bepaalde taken op zich nemen

3 belangrijke functies:

- **system controller** (op processor/northbridge)
 - timers en oscillators: geven de kloksnelheid voor de verbonden onderdelen aan
 - interrupt controller: zorgt voor onderbrekingen in programma-uitvoering wanneer er iets belangrijks gebeurt
 - DMA controller (zie volgende pagina)
 - energiebeheer: bep. apparaten uitschakelen (interessant voor draagbare computers)
- **memory controller** (op processor/northbridge)
 - spreekt het geheugen aan
 - bepaalt snelheid/manier waarop het geheugen wordt gebruikt
 - zorgt voor foutencorrectie in het geheugen
 - *integrated memory controller*: memory controller zit direct op de processor
- **peripheral controller** (southbridge)
 - regelt verbindingen tussen northbridge/processor en randapparaatinterfaces:
 - *businterface* (PCI-bus + systeembus)
 - *floppy-drive interface*
 - *hard disk interface*
 - *toetsenbordcontroller* (scancodes van toetsenbord → code voor OS)
 - *in/uitvoerpoortcontroller* (in- en uitvoer van o.a. seriële en parallelle poort)

Bussystemen

= bussen die ontwikkeld zijn voor uitbreidingskaarten (typisch veel trager, bv. netwerk- en geluidskaarten)

Verre geschiedenis

- PC-bus: eerste belangrijke bus (4.77 MHz; 8 bits breed)
- ISA-bus (Industry Standard Architecture): verbreding van PC-bus (8MHz, 16 bits breed)
- EISA-bus (Extended Industry Standard Architecture): (8MHz, 32 bits breed)
- VESA-bus

Recentere geschiedenis

- PCI-bus (Peripheral Component Interconnect): eerste bus die ontkoppeld is van de processor door een (south)bridge; standaard voor producenten van uitbreidingskaarten (33MHz, 32 bits breed)
 - gemaakt door Intel
 - data- en adresbus zijn gemultiplexed (geen aparte adresbus: eerst adres doorgeven, dan data)
 - bursttransfers mogelijk (eerst adres opgeven van door te geven data, vervolgens achterliggende datablokken, zonder dat het adres opnieuw moet worden doorgegeven)
- PCI-X bus (PCI-extended): opvolger van PCI, nooit echt commercieel succes (high performance-bus)

Heden

- PCI Express/3GIO: volledig nieuwe busarchitectuur → niet compatibel met PCI
 - seriële communicatie (databits worden achter elkaar verstuurd i.p.v. parallel)
 - hoge bandbreedte (5 tot 80 Gbps)
 - kleine connectoren (grootte is evenredig met de snelheid van de aansluiting)
 - QoS (Quality of Service): prioriteit geven aan het versturen van belangrijke data
 - vervangt AGP (Accelerated Graphics Port) als aansluiting voor grafische kaart
 - zorgt voor snelheidswinst en nieuwe mogelijkheden (bv. verbinden van meerdere grafische kaarten)
- USB (Universal Serial Bus): seriële bus met zeer grote ondersteuning
 - Plug&Play (apparaten worden automatisch gedetecteerd en geconfigureerd)
 - Hot-plugging (aansluiten/afkoppelen zonder computer uit te zetten)
- SATA: aansluiting van harde schijven en CD-ROM/DVD spelers en schrijvers
 - seriële variant van vroegere ATA, met hogere snelheden

DMA

= een techniek die toelaat dat bepaalde apparaten rechtstreeks RAM geheugen kunnen aanspreken om te lezen en/of te schrijven, zonder dat de processor daarvoor gestoord moet worden

- interessant voor grote hoeveelheden data naar geheugen of van geheugen (bv. laden van een filmpje)
- bv. Geluidskaart die op achtergrond muziek afspeelt:
 1. CPU stelt soundcard in
 2. CPU stelt DMA in
 3. DMA kan beginnen: data direct van RAM naar geluidskaart

H6: Permanente magnetische gegevensopslag

Inleiding

permanente geheugens: geheugens die ook zonder stroom hun informatie kunnen behouden

Harde schijven

Opbouw

Platters

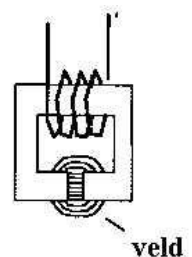
= *harde, onbuigbare schijven waarop de data door middel van magnetisme opgeslagen wordt*

- bekleed met ferromagnetisch materiaal
- het magnetisch veld zal aanwezig blijven op een bep. plaats op de platter eens het geactiveerd werd door de schrijfkop
- hoe meer platters, hoe groter de opslagruimte
- kunnen aan beide zijden beschreven worden
- moeten ronddraaien om beschreven/gelezen te worden (aan een snelheid van 3600 tot 15000tpm)

Koppen

= *bewegen zich samen over de platters d.m.v. de actuator en kunnen de magnetische data lezen/schrijven*

- een stukje ijzer in U-vorm, omwonden met een elektrische spoel
- voor elke platterzijde is er een kop
- lezen en schrijven wordt door dezelfde kop gedaan



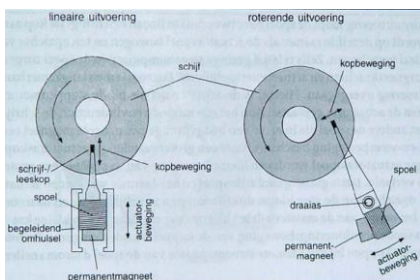
Actuator

= *de 'motor' die ervoor zorgt dat de koppen zich over de platters kunnen bewegen*

- beweging is beperkt van de binnenste tot de buitenste track
- beweegt uiterst precies (veel tracks naast elkaar)

Besturingselektronica:

- bevat elektronica die het geheel coördineert en aanstuurt
- bevat de elektronica om gegevens uit te wisselen met het moederbord.

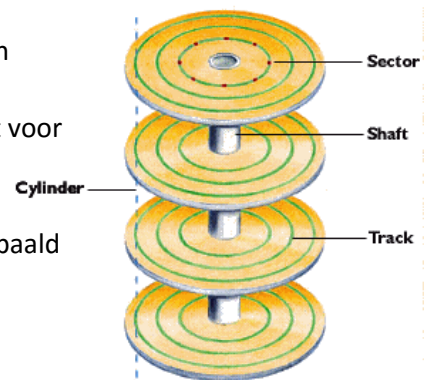


Opslag

data wordt op een gestructureerde manier bijgehouden op **platters**:

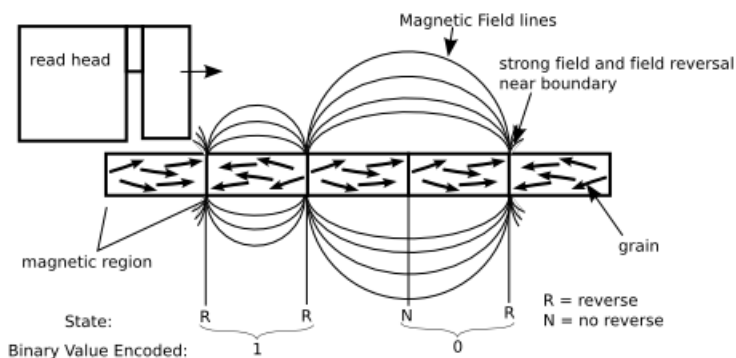
- een platter is opgedeeld in **sporen**, genummerd van binnen naar buiten
- elk spoor is opgedeeld in **sectoren**:
 - harde sectorindeling (het begin/grootte van de sectoren wordt voor altijd door de constructie vastgelegd)
 - ↳ bv. floppy disks
 - zachte sectorindeling (de onderverdeling in sectoren wordt bepaald door een stukje informatie die op de schijf zelf is opgeslagen)
 - ↳ je verliest een beetje schijfruimte hierdoor
 - ↳ vroeger: zelf doen (*low level formatting*)
 - ↳ nu: vanuit de fabriek
- wanneer er meerdere platters boven elkaar worden geplaatst, noemt men boven elkaar liggende sporen op die verschillende platters **cilinders**

Tracks, Cylinders, and Sectors



Niet de polariteit van het magnetisch veld stelt data voor, maar de **verandering van polariteit**

- gelijkblijvend magnetisch veld stelt een bit met binaire waarde 0 voor
- wisselend magnetisch veld stelt een bit met binaire waarde 1 voor



Lezen/schrijven

schrijven = magnetisch veld opwekken

1. Er wordt een elektrische stroom doorheen de schrijfkop gestuurd
2. Deze genereert een magnetisch veld met een via de stroomrichting gekozen oriëntatie
3. Het magnetisch veld wordt opgeslagen op het magnetisch materiaal waaruit de harde schijf bestaat (stelt één bit voor).
4. Het gegenereerde magnetisch veld is zo zwak, dat slechts een kleine oppervlakte van de onderliggende platter zal worden gemagnetiseerd.
zo kunnen zeer veel bits op de platter worden opgeslagen.

lezen = magnetisch veld detecteren

1. Er wordt in eerste instantie geen elektrische stroom op de leeskop gezet
2. De leeskop beweegt zich over de ronddraaiende platter
3. Wanneer de polariteit van het magnetisch veld veranderd, zal dit een elektrisch stroompje produceren in de leeskop, wat wordt gedetecteerd door de leeskop
4. Aangezien het opgeslagen magnetisch veld op de platter één bit voorstelde, kunnen we de waarde van deze bit afleiden uit de spanning/stroom.

lezen/schrijven **wordt beïnvloed door**

- de hoeveelheid elektriciteit
- het aantal windingen van de spoel
- het materiaal waaruit de spoel gemaakt is
- de gevoeligheid van het magnetisch materiaal op de platter
- de afstand van de kop tot de platter (hoe dichter, hoe sterker het magnetisch veld op de schijf en hoe kleiner de beschreven oppervlakte)

door de luchtverplaatsing als gevolg van de draaiing van de platters **zweeft de kop boven de platters**

- constante afstand = 0.000003mm
- om te vermijden dat de kop over de platters sleept (warmteontwikkeling + slijtage)
- wanneer de schijf tot stilstand komt laat de besturingselektronica de kop landen op een vooraf bepaalde positie waar geen data opgeslagen is
- wanneer de schijf start blijf de kop op zijn positie totdat hij begint te zweven

Eigenschappen

Belangrijke omgevingsfactoren

Warmte

vooral de actuator genereert veel warmte

- ↳ Binnenkant van behuizing ventileren met buitenlucht
 - ook om lucht te hebben om de koppen te laten vliegen
- ↳ Koellichamen die al dan niet actief gekoeld zijn.

Onzuiverheden binnen de behuizing

gebruik van luchtfilters om buitenlucht binnen te laten en *head crashes* door bv. stofdeeltjes te vermijden

Fysische schokken

head crash: de kop raakt de platter onverwacht door een externe oorzaak (bv. stofdeeltje, fysische schok)

- ↳ kop/platter kan beschadigd zijn → schijf wordt onbruikbaar
- ↳ oplossing: fysische schokken voorspellen en de kop naar een veilige zone verschuiven

Oppervlakedichtheid

= hoeveelheid informatie die op een bepaalde oppervlakte past (meestal bits/inch²)

Oppervlakedichtheid = TPI x BPI

- TPI: Tracks per inch (*track density*)
- BPI: Aantal bits per inch per track (*recording density/lineaire densiteit*)

Snelheid

access time: hoe snel de harde schijf klaar is om gegevens te lezen/schrijven

= command overhead + seek time + settle time + latency

command overhead: de tijd die nodig is om het R/W-commando door te geven aan de harde schijf (+0,5ms)

seek time: verplaatsen van de koppen naar het juiste spoor (duurtijd is afhankelijk van schijf, gem. 8-10ms)

settle time: tijd die nodig is om koppen te laten stabiliseren na verplaatsing

latency: de tijd om de schijf te laten draaien tot de gewenste sector (8,3ms bij 360° met 7200tpm)

ook harde schijven hebben een **cachegeheugen** om datatoegang te versnellen

- ook gebaseerd op "locality of reference"
- overhyped (maakt niet zo veel verschil uit, +-4% verschil)

Evolutie

- capaciteit verdubbelt om de 2 jaar
- kleiner en kleiner
- betere performantie (zoektijden/verplaatsing van koppen)

RAID

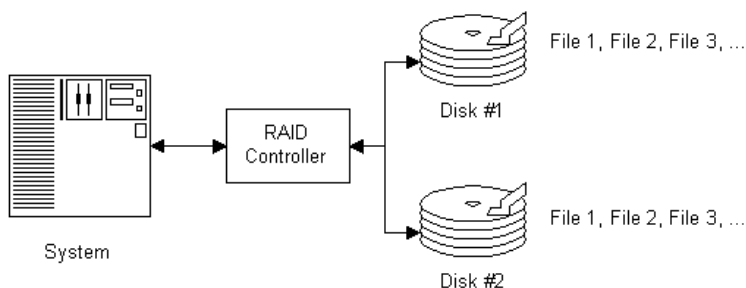
= Redundant Array of Inexpensive/Independant Disks

- meerdere harde schijven combineren, met als doel:
 - hogere performantie (snelle datatransfers)
 - hogere capaciteit
 - hogere foutentolerantie (*redundantie*: het meer dan benodigd voorkomen van iets)
- in realiteit moeten we afwegen tussen snelheid/kostprijs/kwaliteit

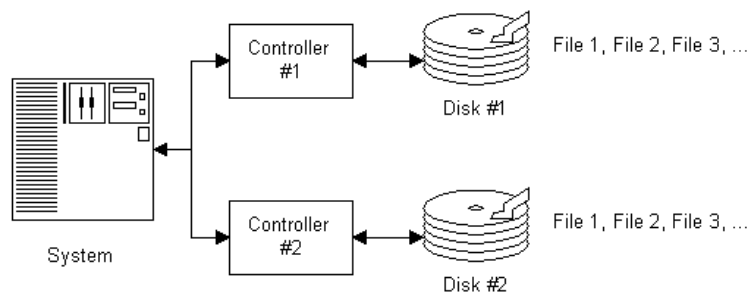
Concepten

disk mirroring/schijfspiegeling: alle data wordt op twee verschillende schijven bijgehouden

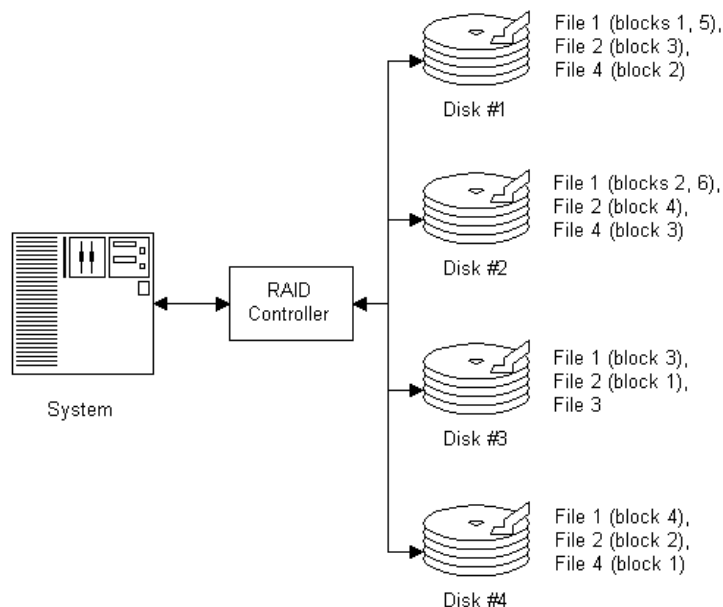
- op beide schijven staat op elk moment dezelfde data
- lezen is sneller, schrijven iets trager omwille van gezamenlijke controller
- andere schijf neemt over bij panne (systeem om de downtime te minimaliseren)
- nadeel: verdubbeling nodige ruimte → kosten ↑



duplexing: alle data wordt op twee verschillende schijven bijgehouden, maar met 2 schijfcontrollers
 → sneller lezen/schrijven



disk striping/schijvenspreiding: bestanden verdelen over meerdere schijven (sector-niveau of byte-niveau)
 → voordeel: sneller



parity: (N+1)-concept

- N-schijvenhouders bevatten de echte informatiebits
- (N+1)-schijvenhouders bevatten de testbits
 - ↳ ofwel gegevens herstellen bij fout
 - ↳ ofwel fout signaleren

→ hoge betrouwbaarheid, maar duurdere controllers dan bij disk mirroring
 → informatiecontrole werkt meestal met XOR-berekening (Exclusive OR)

Ingangen		Uitgang
0	0	0
1	0	1
0	1	1
1	1	0

RAID-levels

= combinatie van mirroring, striping, duplexing & parity (hoe complexer, hoe duurder de controller)

JBOD

= Just a Bunch Of Disks (een aantal schijven aan elkaar “plakken”; *spanned volumes*)

RAID level 0

= Data striping without parity

- bestanden in stukken over meerdere schijven wegschrijven
- hoge performantie, geen redundantie
- minimum 2 harde schijven, liefst zelfde grootte/snelheid/...

RAID level 1

= duplexing (mirroring met aparte controllers)

- hoge performantie, hoge redundantie (maar ook hoge kosten)
- minimum 2 identieke harde schijven (anders: kleinste schijf wordt gespiegeld)
- vooral voor lees-intensieve toepassingen (bv. webserver)

RAID level 2

= Bit level striping met Hammingcode foutdetectie en correctie

- bijna niet gebruikt (duur, veel schijven, complexe controller, performantie is niet fantastisch)
- Hammingcode: fouten van 1 bit verbeteren, 2 bit detecteren

RAID level 3

= Byte level striping met pariteit, met stripinggrootte < 1024 bytes

- minstens 3 harde schijven (2 voor striping, 1 voor pariteitbits)

RAID level 4

= Byte level striping met pariteit, met stripinggrootte > 1024 bytes

RAID level 5

= Block-level striping met gedistribueerde pariteit

- zelfde als RAID-4, alleen worden pariteitbits verspreid over de schijven (minimum 3)
- hoge performantie, hoge redundantie
- vooral voor lees-intensieve toepassingen (bv. webserver)

RAID level 6

= Block-level striping met duale gedistribueerde pariteit

- zelfde als RAID-6, alleen worden er 2 ≠ pariteitscontroles opgeslagen (2 schijven mogen uitvallen)
- minder gebruikt (vrij duur)
- minstens 4 harde schijven nodig

RAID-controllers

	Voordelen	Nadelen
Hardware	<ul style="list-style-type: none">- performanter- belast de andere resources minder- booten is mogelijk- geavanceerde mogelijkheden- transparant voor OS	duurder
Software	<ul style="list-style-type: none">- goedkoop- eenvoudig- duplexing m.b.v. software	<ul style="list-style-type: none">- performantie- booten is niet mogelijk van de array- geen geavanceerde mogelijkheden- afhankelijk van OS

H7: Optische geheugens en leesgeheugens

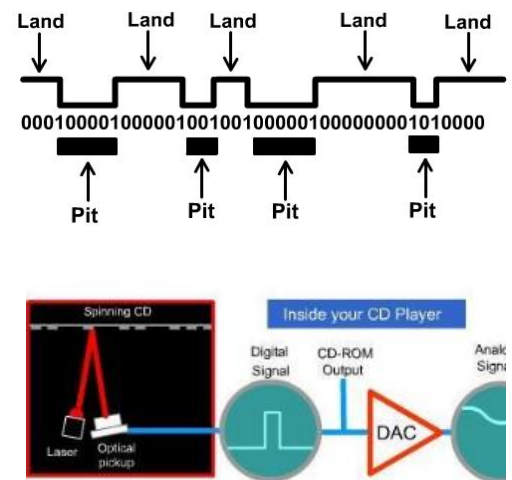
CD-ROM

CD-R

= Compact Disk – Read-Only Memory

Benodigdheden om CD-ROM's te lezen

- **disk**: ronde schijf van 12 of 8cm diameter, met aluminium laag die bestaat uit *pits* en *lands*
 - ↳ *pits* worden in het aluminium geperst bij productie
 - ↳ bevat bescherm laag + labellaag (veel minder beschermd)
 - ↳ normale CD-ROM: 700MB
 - ↳ spiraalvormig spoor (binnen → buiten)
 - ↳ redundantie (10% foutenverbeterende code)
- **laser**: scant het oppervlak van de disk af
 - ↳ laserlicht zal bij *lands* reflecteren op het optica
- **spin motor**: zorgt ervoor dat de CD-ROM ronddraait
 - ↳ laser kan enkel van spoor veranderen



Draaisnelheid

= uitgedrukt in verhouding tot de snelheid die nodig is om een klassieke audio-CD uit te lezen (350 tpm = single speed = 1x)

- niet altijd constant
- indien constante bitstroom gewenst:
 - buitenste gedeelte trager
 - binnenste gedeelte sneller

CD-R

= WORM-geheugen (*Write Once Read Many*)

- elke bitlocatie kan één maal beschreven worden en kan meermaals uitgelezen worden
- geen persing, maar branden: de disk is gecoat met een *dye* (speciaal materiaal dat bij plaatselijke verwarming al dan niet reflecterend wordt)
 - ↳ licht-absorberend → pits
 - ↳ niet licht-absorberend → lands
- kleur van CD komt overeen met dye (beschreven deel van CD is met het blote oog zichtbaar)
- laser kan op 2 vermogens werken:
 - hoog vermogen (schrijven)
 - laag vermogen (lezen)

CD-RW

= elke bitlocatie kan één meermaals beschreven en uitgelezen worden.

- geen persing, maar branden
- verschil tussen 0 en 1 is minder duidelijk "zichtbaar" → slechte compatibiliteit bij oudere lezers
- laser kan op 3 vermogens werken:

- write power (hoogste laservermogen/brandtemperatuur, maakt licht-absorberend)
- erase power (middelste laservermogen, laagste brandtemperatuur, maakt opnieuw reflectief)
- read power (laagste laservermogen, beïnvloedt staat v.h. materiaal niet)

DVD

= Digital Versatile Disc

- zelfde afmeting als CD-ROM, grotere capaciteit (minimum 4.7GB)

Werking

- Pits en lands kleiner dan bij CD-ROM (hogere datadichtheid)
- Fijnere laserfocussing dan bij CD-ROM
- Andere mechanismen voor adressering en foutcodering

Draaisnelheid

= uitgedrukt in verhouding tot de leessnelheid van de eerste DVD-spelers (1350 KB/s = single speed = 1x)

Mogelijke capaciteiten

- eventueel 2 lagen gescheiden door half doorlatende laag (*dual layer*)
- eventueel tweezijdig
- DVD-5 (4,7GB) tot DVD-18 (17,1GB)

Regiocodes

= beperking van afspelen van bepaalde content tot bepaalde regio's in de wereld d.m.v. encryptie

- zowel in DVD als in toestel
- zowel hardware als software
- reden: regio-naar-regio releases van films

DVD-R/DVD-RW

- herschrijfbaar, gelijkaardige werking aan CD-R/CD-RW
- ontwikkeld door het DVD Forum (hoogste compatibiliteit)

DVD+R/DVD+RW

- herschrijfbaar, gelijkaardige werking aan CD-R/CD-RW
- ontwikkeld door de DVD Alliance

DVD-RAM

- niet compatibel met andere DVD varianten
- hoge prijs
- bestaat in verschillende varianten (met/zonder harde behuizing voor bescherming)

Blu-ray

- zelfde afmeting als DVD en CD-ROM, grotere capaciteit (minimum 27GB)
- fragiel (zeer dunne beschermingslaag van slechts 100 micrometer)

Werking

- pits en lands nog kleiner (hogere informatiedichtheid)
- blauwe laser met kortere golflengte

Mogelijke capaciteiten

- Single-layer: 27 GB (2 uur HDTV @ 36-72Mbps)
- Double-layer: 54 GB

HD DVD

= oorspronkelijk een concurrent van de Blu-ray (stopgezet)

Leesgeheugens

= dient om snel uit te lezen, af en toe (trager) schrijven

ROM

= Read-Only Memory

- bestendige geheugens
- vroeger gebruikt voor bv. BIOS van computers

PROM

= Programmable ROM

- mogelijkheid om 1x te schrijven d.m.v. PROM-programmeur, vanaf dan read-only
- werkt met soort zekeringen (*fusable link*) die men kan opblazen door een spanningsstoot
- schrijven gaat zeer traag (1000x trager dan SRAM)
- bv. RFID-tags, auto-industrie

EPROM

= Erasable Programmable ROM

- volledig uitwisbaar door blootstelling aan ultravioletstralen (d.m.v. speciaal toestel)
- (her)programmeerbaar (na 10 jaar → informatieverlies)

EEPROM

= Electrically Erasable Programmable ROM

- elektrisch uitwisbaar en herschrijfbaar
- meer elektronica → sneller uitwissen + duurder

Flashgeheugens

Voordelen

- Hoger aantal mogelijke herbeschrijvingen
- Hogere informatiedichtheid
- Lagere fabricagekosten
- Laag energieverbruik (goed voor draagbare computers)
- Hoge snelheid t.o.v. andere bestendige geheugens
- Hoge betrouwbaarheid
- Compact t.o.v. een harde schijf

Gebruik

- Camera's
- GSM's
- Als gegevensdrager zelf: SD-kaart (*Secure Digital*), USB-memorystick, CF-kaart (*Compact Flash*), MMC-kaart (*Multimedia Card*), Smartmedia, Sony Memory Stick, ...

Eigenschappen

- Random write
- Enkel block erase (wissen kan alleen per block van data)
- Beperkt aantal *erase cycli*
 - ↳ graceful degradation: zeer geleidelijke en aanvaardbare slijtage van het geheugen door algoritmes die de data gelijk verspreiden over alle geheugencellen

SSD

= Solid State Drives (NAND-flash geheugen als harde schijf)

Voordelen

- Geen draaiende onderdelen:
 - geruisloos
 - laag energieverbruik
 - lage warmteontwikkeling
- Snelle start-up van systeem
- Betrouwbaar (beter bestand tegen schokken/temperatuur)

Nadelen

- Duurder dan harde schijven

Opslagmethode

= digitale waarden toekennen aan cellen met analoge waarde (lading/spanning/weerstand)

- SLC (*Single Level Cell*)
 - ↳ analoge waarde wordt verdeeld in 2 bereiken: hoog/laag level voor 1-bit en 0-bit
 - ↳ betrouwbaarder, sneller, duurzamer
- MLC (*Multi-Level Cell*)
 - ↳ analoge waarde wordt verdeeld in meer bereiken, meestal 4
 - ↳ mogelijkheid tot opslagen van 2 bits in 1 eenheid (veel compacter)
 - ↳ goedkoper, trager, minder betrouwbaar

H8: Poorten

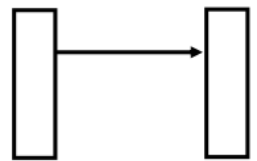
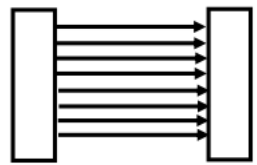
Inleiding

- poorten dienen als communicatiepoort (interface) tussen de computer en andere computers/randapparatuur
- gebruikt voor allerlei toepassingen: video uitwisselen (DVI, VGA, HDMI), data doorsturen (eSata, Firewire, USB), ...
- informatie wordt door poorten gestuurd d.m.v. elektrische signalen die over metalen geleiders worden gestuurd

Eigenschappen

Parallel/serieel

- **Parallele communicatie:** meerdere bits worden op een bepaald moment doorgestuurd over meerdere geleiders (draden)
 - meestal veelvouden van 8 bits (bv. 1/2/4 bytes tegelijk)
 - signal skew: gelijktijdig verzonden bits komen niet tegelijk aan
 - ↳ door externe invloeden (bv. \neq lengte van geleiders in een kabel)
 - ↳ onduidelijk bij welke set van bits een bit hoort
 - ↳ hoe hoger de snelheid, hoe erger
 - ↳ weinig toekomst
- **Seriële communicatie:** meerdere bits aan data worden gelijktijdig verzonden
 - 4 basisdraden:
 - één draad in de ene richting
 - één draad in de andere richting
 - twee nuldraden
 - trager dan parallel, behalve bij hogere snelheden/afstanden (geen last van signal skew)
 - bij meeste nieuwe poorten
 - minder geleiders → dunner en meer betrouwbare kabels
 - aangestuurd door de UART-controller (Universal Asynchronous Receiver/Transmitter)
 - ↳ zet de parallelle uitvoer van de systeembus om in seriële informatie



Aantal geleiders

bijna alle poorten hebben een aantal extra geleiders:

- synchronisatie (kloksignaal)
- aarding
- voeding

Types van duplex

- **Full duplex:** beide partijen kunnen op eenzelfde moment gegevens verzenden naar elkaar
 - bv. telefoongesprek
- **Half duplex:** slechts één van beide partijen kan op een bepaald moment gegevens verzenden naar de andere partij (het kanaal moet vrij zijn)
 - bv. walkie talkie

- **Simplex:** er is slechts één zender, de ontvanger heeft geen zendmogelijkheden
→ bv. broadcasting

Connectoren

- wordt gebruikt om kabels met een poort te verbinden
- 2 types:
 - male: 1 of meerdere pinnetjes die passen in de female connectoren
 - female: kunnen male connectoren vasthouden

Synchronisatie

bij het verzenden van data wordt er een bepaalde **kloksnelheid** afgesproken

- de ontvanger gebruikt dit ritme om te weten waar elke bit die wordt verzonden eindigt en begint.
- enkel op die manier kunnen de bits/bitreeksen correct worden geïnterpreteerd door de ontvanger

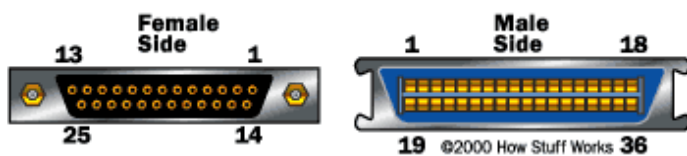
2 manieren:

- Synchrone transmissies: gesynchroniseerd door een externe klok
 - ↳ bv. een pulse op 1 geleider wilt zeggen dat het tijd is om informatie uit te lezen op een andere geleider
 - ↳ wordt gebruikt bij parallelle communicatiesystemen
- Asynchrone transmissies: gesynchroniseerd d.m.v. speciale signalen die mee verstuurd worden
 - ↳ data wordt in pakketjes (messages) gestuurd:
 - 1 startbit
 - 8 databits
 - 1 pariteitsbit
 - 1 stopbit
 - ↳ eenvoudiger/goedkoper (minder hardware(geleiders) nodig)
 - ↳ minder plaats voor echte data door synchronisatieinformatie
 - ↳ transmissie kan sneller opgezet worden: goed voor korte boodschappen door te sturen
 - ↳ wordt gebruikt bij seriële communicatiesystemen

Voorbeelden

Printerpoort (parallele poort)

= 3 meter lange kabel met parallelle communicatie (meestal gebruikt om te printen)



- meestal andere connector aan printerzijde: centronics poort
- oorspronkelijk *simplex*, later *half duplex* per geleider
- op computer zichtbaar als *LPT*

RS-232-standaard (Seriële poort)

- Seriële communicatie → langere afstanden mogelijk
- Full duplex
- UART controller zet parallelle signaal dat wordt verkregen van moederbord om naar serieel signaal
- op computer zichtbaar als *COM*

- Soms nog eens gebruikt voor netwerkapparatuur te configureren.
- Wordt ook door hobbyisten gebruikt om apparatuur aan te sluiten.
 - ↳ Gemakkelijk manipuleerbaar vanuit besturingssysteem of zelfgeschreven programma's
 - ↳ Zelfs op goedkope externe hardware beschikbaar

USB

= Universal Serial Bus

- gestandaardiseerd door USBIF, een forum v. belangrijke bedrijven (Intel, HP, NEC, Apple, Microsoft)
- brede waaier aan toepassingen (geen netwerkverbindingen/videotransmissie)
- stroomvoorziening is standaard 5V, minimaal 500mA (anders: externe stroomtoevoer)

Design

- lage productiekost
- geen schroeven
- onmogelijk verkeerd in te steken
- goede aarding

Standaarden (achterwaarts compatibel)

- USB 1.1 (12 Mbps, max. kabellengte van 3m)
- USB 2.0 (480 Mbps, max. kabellengte van 5m)
- USB 3.0 (4.8 Gbps, max. kabellengte van 5m)

Snelheden

- Low Speed (1.5Mbps, vooral voor Human Input Devices)
- Full Speed (12Mbps)
- High Speed (480Mbps, vanaf USB 2.0)
- Superspeed (4.8Gbps, vanaf USB 3.0)

Eigenschappen

- Plug and play: automatische herkenning en configuratie van nieuwe apparaten
- Hot swappable: apparaten kunnen insteken en uittrekken terwijl de computer aanstaat

Connectoren (met elk type A en B)

- USB A/B
- standaardconnector (USB 1.1/2.0)
- USB 3.0 (achterwaarts compatibel)
- Mini USB
- Micro USB



USB host controller

verzorgt communicatie tussen een USB poort en de rest van het computersysteem (organisatie v.h. verkeer)

- ↳ HCD (Host Controller Device): de programmeerinterface van de host controller
- ↳ 2 implementaties van HCD bij USB 1.0 en 1.1
 - OHCI (Open Host Controller Interface → Compaq)
 - UHCI (Universal Host Controller Interface → Intel)
- ↳ 1 implementatie bij USB 2.0: EHCI (Extended Host Controller Interface)
 - ondersteunt high-speed transfers

USB-hub

een soort "dominostekker" waarmee je een USB poort in een computer kan uitsplitsten

- ↳ verschillende hubs op mekaar aansluiten in boomstructuur
 - maximaal 5 levels diep
 - maximaal 127 apparaten (hubs tellen mee)
- ↳ verhoogt de kwaliteit van het signaal opnieuw
- ↳ een powered hub kan gekoppelde apparaten van meer stroom voorzien

Apparaatklassen

= indeling van USB-randapparaten, bepaald door USB-IF

- HID (Human interface device)
 - Toetsenbord
 - Muis
 - Joystick
 - Simulatieomgevingen
 - ...
- Mass storage
 - Externe harde schijven
 - USB memorysticks
 - Geheugenkaartlezers
 - ...
- Communications device
 - Modems
 - Netwerkkarten
 - Faxmachines
- Printer device (printers)
- Audio device (geluidskarten)
- Video device (webcams, capturing apparaten)

Firewire (IEEE 1394)

- Seriële communicatie
- Opvolger SCSI
- Ontwikkeld door Apple Computer in jaren '90
- **Hogere snelheid** (bij introductie) en **hogere stroomvoorziening** dan USB
- Komt **minder frequent** voor dan USB (initieel moesten royalties worden betaald)
- Max 63 randapparaten in boomstructuur of "daisy chain"
- **Peer-to-peer communicatie** mogelijk
 - ↳ rechtstreeks tussen printer en scanner zonder tussenkomst computer
 - ↳ eventueel als netwerk tussen computers

Toepassingen

- externe harde schijven
- video/audio-apparaten
- vooral gebruikt voor video-overdracht.

Standaarden

- FireWire 400 (100/200/400 Mbps, max 4,5 meter kabel, maar tot 16 keer doorkoppelen)
- FireWire 800 (800 Mbps)

Eigenschappen

- Plug and play
- Hot swapping
- 45 watt per poort

eSata

= externe variant op SATA

- wordt intern in computer gebruikt voor aansluiten van magnetische/optische media-apparaten
- vooral gericht op het aansluiten van externe schijven

Thunderbolt

- Op de markt gebracht door Intel (2011)
- Vooral commercieel succes na intrede op MacBook Pro
- Combinatie van PCI Express en DisplayPort in 1 serieel signaal
- Wisselstroomvoorziening
- Tot maximum 6 apparaten op 1 connector

VGA port

Om (analoog) videosignaal door te sturen naar beeldschermen (maximaal 1080p)

DVI connector

Om (digitaal) videosignaal door te sturen naar beeldschermen (flatscreens/projectoren, hogere resoluties dan VGA)

H9: Randapparatuur

Toetsenbord

Geschiedenis

- belangrijkste invoerapparaat
- alfabetische/numerieke tekens, controlettoetsen en functietoetsen
- verschillende soorten layouts:
 - azerty, qwerty → stammen af v.d. typmachines (afstand tussen hamertjes maximaliseren)
 - dvorak → speelt meer in op nieuwe eisen

Opbouw

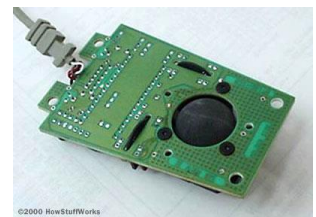
- **Scanmatrix/key matrix** (zegt welke toets men indrukt of loslaat)
 - zo ontstaat een *scancode*, een code die weergeeft
 - welke van de toetsen werd ingedrukt (plaats in de *key matrix*)
 - of het indrukken of loslaten was
- **Typemantics** (bepaalt de snelheid waarmee een lang ingedrukte toets moet worden doorgegeven)
- **Toetsenbordcontroller** (geeft scancodes door naar de computer, meestal via USB of PS/2)

Muis

Opbouw

Oorspronkelijk

- bal die tegen twee staafjes rolt (staafje voor X en Y-richting)
 - ↳ uitvinder *x-y position indicator* = Doug Engelbart (1968)
- lichtstraal voor detecteren van beweging (infrarood)
- on-board chip om signalen door te sturen naar de computer



Tegenwoordig (optische muis)

- geen roterende onderdelen meer
- **Werking**
 - LED-licht (*licht-uitstralende diode*) botst op oppervlakte en wordt weerkaatst naar een CMOS sensor
 - CMOS sensor zendt weerkaatste afbeelding naar een DSP (*digital signal processor*) voor analyse
 - DSP herkent patronen en hoe een patroon is verplaatst tegenover het vorige patroon
- **voordelen**: nauwkeuriger en beter bestand tegen vuil
- **nadeel**: werkt niet op alle oppervlakken (bv. glazen vlak)



Repetitive Strain Injury (RSI)

- **repetitive strain injury (RSI)**: aandoening ter gevolg van te veel of verkeerd gebruik van een muis
 - voorzorgen: regelmatig breaks nemen, typen (niet rammen), ...

Stroomvoorziening

- wegvallen stroom → dataverlies bij RAM-geheugen (tenzij recovery)
- **UPS (Uninterruptable Power Supply)**: tijdelijke backup-stroomvoorziening
 - zowel grote serverparken (dieselmotor) als thuiscomputers (batterij)

H11: Besturingssystemen

Doelen en functies

computersysteem: biedt eenvoudige/uniforme interface naar de complexe hardware

Geschiedenis

- Batchsystemen
 - verschillende applicaties 1 na 1 draaien
 - dure systemen → processor 100% bezighouden
 - batch-operator sloeg ≠ ponskaarten (programma's) op op tapes, die vervolgens naar het eigenlijke computersysteem werden overgebracht
- Batchsystemen met multiprogramming
 - 2 of meer progr. tegelijk in geheugen, om werkloze momenten v.d. processor op te vangen
 - *spooling*: in- en uitvoeropdrachten in wachtrij zetten
- Timesharing systemen
 - verschillende gebruikers kunnen tegelijk werken
 - verschillende terminals verbinden met het mainframe
 - het mainframe verdeelt zijn procestijd tussen de terminals
- Personal computer
 - introductie van LSI-schakelingen → prijzen voor computersystemen dalen
 - elke werknemer een aparte computer
 - meer aandacht voor gebruiksvriendelijkheid
 - bv. Unix, Microsoft DOS en Windows, Apple Macintosh

Andere varianten

- Real-time systemen (timing is enorm van belang)
 - *soft real-time systeem* (deadline kan gemist worden)
 - *hard real-time systeem* (deadline mag niet gemist worden, bv. monitoren van kerncentrale)
- Embedded systemen
 - besturingssysteem op specifieke hardware, met speciale eisen (bv. laag energieverbruik)
 - bv. GSM's, MP3-spelers, besturingssystemen in vliegtuigen
- Parallele systemen (meerdere processoren om de gelijktijdige uitvoering v. taken te optimaliseren)
- Gedistribueerde systemen (systemen die verbonden zijn via een netwerkverbinding)

Functies

- uitvoering van applicaties in goede banen leiden
- **drie belangrijke doelen:**
 - gemak (computer eenvoudig kunnen gebruiken)
 - efficiëntie (aangesloten hardware zo efficiënt mogelijk gebruiken)
 - mogelijkheid tot ontwikkeling (wijzigen/aanbrengen van systeemfuncties, zonder dat bestaande functionaliteit wordt aangetast)

Lagenmodel

1. **Gebruikersprogramma's** (tekstverwerker, browser, e-mailclient, ...)
2. **Systeemprogramma's** (shell, compilers, debug-tools, ...)
3. **Besturingssysteem** (Linux, Unix, Windows, Mac, Solaris, ...)
 - a. Process management (beheren van de draaiende processen)
 - b. Memory management (beheren van het werkgeheugen)
 - c. File management (beheren van bestanden op opslagmedia)
4. **Hardware** (processor, harde schijf, werkgeheugen, ...)

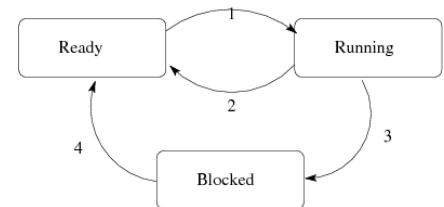
→ nadruk ligt op onafhankelijkheid

→ men zou een laag moeten kunnen vervangen, zonder dat de andere lagen daar last van ondervinden

Process management

- Uitvoering programma

- ↳ de processor voert instructies uit die in het werkgeheugen staan
- ↳ volgorde wordt bepaald door de programmateller (*EIP, Extended Instruction Pointer*)
- ↳ de EIP kan verwijzen naar code in verschillende programma's
- ↳ voor 1 programma zijn er 3 mogelijke toestanden
 - *running* (momenteel in uitvoering op de CPU)
 - *ready* (klaar om uit te voeren, maar CPU is bezig met ander proces)
 - *blocked* (wachten op externe gebeurtenis, zoals het voltooien van een I/O-operatie)
- ↳ overgangen 1 en 2 worden door de *process scheduler* uitgevoerd



- Procestabel (informatie om processen te kunnen herstellen na de *blocked* of *ready queue*)

- program counter (waarde van de programmateller/EIP, om op de juiste positie te hervatten)
- naam v.h. ouderproces (indien dit proces werd opgeroepen door een ander proces)
- starttijd
- gebruikte CPU-tijd (sinds het opstarten)
- eigenaar (wie heeft het proces opgestart)
- scheduling-informatie
 - *scheduler*: bepaalt welk proces op welk moment gebruik mag maken van de CPU
- lijst met bestanden die door het programma worden gebruikt

- Multiprogramming (meerdere processen tegelijk in het geheugen)

- ↳ CPU afwisselend toewijzen aan processen (doel: CPU-gebruik verhogen)
- ↳ is een grote optimalisatie indien het proces moet wachten op I/O

- Interproces communicatie

- ↳ dikwijls communicatie tussen processen (berichten/postbussen)
- ↳ race condities: het resultaat van een reeks processen hangt af van de volgorde waarin de opdrachten werden ingegeven
- ↳ kritische sectie: gedeelte van de code dat onafgebroken moet kunnen worden uitgevoerd om een juiste werking te garanderen
 - voorkomen dat meerdere processen dezelfde gegevens veranderen

Scheduling

- **Typische uitvoering proces:**
 - ↳ afwisseling I/O-bursts (invoer/uitvoer) en CPU-bursts (CPU-intensieve berekeningen)
 - ↳ bij I/O-burst proces 1, CPU-burst van proces 2
- **process scheduling-algoritmen:** beslissen welk proces op de processor mag uitvoeren
 - ↳ ≠ algoritmen worden vergeleken op de volgende factoren:
 - rechtvaardigheid (eerlijke verdeling van CPU-tijd)
 - efficiëntie (constant 100% processorgebruik)
 - responstijd (gebruikers mogen geen vertragingen merken)
 - turnaround (minimaliseer wachttijden voor gebruikers van batch-processen)
 - throughput (zo hoog mogelijk aantal processen/uur)
- **Preëemptieve vs. niet-preëemptieve scheduling**
 - niet-preëemptief: huidige proces op de processor kan pas naar *waiting queue* indien het dit zelf vraagt (tot Windows 3.11)
 - preëemptief: huidige proces op de processor kan op een willekeurig moment plaats ruimen voor een ander proces (vanaf Windows 95/Unix)

Memory management

Swapping

= een stuk van het RAM-geheugen tijdelijk wegschrijven naar de harde schijf

- onmogelijk om alle processen tegelijk in werkgeheugen te houden
- implementatie als swap-file of swap-partitie
- allocatie van het geheugen bij het inswappen van een proces:
 - ↳ beschikbare geheugen verdelen in blokken van n bytes
 - ↳ een blok is ofwel vrij (*gap*), ofwel toegewezen aan een proces (*procesruimte*)
 - ↳ het besturingssysteem houdt hiervan een *gelinkte lijst* bij
- ≠ werkwijzen om ongebruikte geheugenruimte te vinden in gelinkte lijsten:
 - *first fit* (eerste gap dat groot genoeg is → beste algoritme)
 - *next fit* (zoeken wordt gestart waar men vorige keer is gestopt)
 - *best fit* (volledige lijst wordt onderzocht, meest gepaste blok wordt toegewezen)
 - *worst fit* (grootst beschikbare gap wordt toegewezen)
- paging: een proces moet geen aaneensluitend blok in het geheugen innemen

Virtueel geheugen

- een programma gebruikt meer geheugen dan eigenlijk voorhanden is
- principe: het programma hoeft niet volledig in het geheugen te zitten (locality of reference...)

Geheugenruimte van een programma

- Vast gedeelte
 - text segment (werkelijke code en globale variabelen)
- Variabel gedeelte
 - stack segment (stapel v.h. programma, belangrijk voor het oproepen van functies)
 - heap segment (aanmaken v. dynamisch toegewezen variabelen)

File management

Blokgrootte

- Hoe bestanden opslaan?
 - aaneensluitende reeks van bytes (wat indien het bestand groter wordt?)
 - opdelen in blokken van gelijke grootte (overhead = meer verbruik van bronnen dan nodig)
- grootte blokken = afweging tussen datasnelheid en gebruiksgraad van de schijf
 - te groot: te veel vrije ruimte (*gaps*)
 - te klein: veel overhead voor het opzoeken en combineren van alle kleine blokken
 - meestal compromis (tussen 512 B en 4 KB)

Allocatie

= vrije ruimte op een schijf toekennen aan een bestand

3 belangrijke manieren:

- aaneensluitende allocatie: alle blokken van één bestand worden aaneensluitend op de schijf bewaard
 - ↳ eenvoudigste methode
 - ↳ wanneer het bestand te groot wordt moet het verplaatst worden naar een andere locatie
- linked list: elke blok wordt opgeslagen met een verwijzing naar het volgende blok (*pointer*)
 - ↳ verspilling van schijfruimte (oplossing: groepering in clusters)
 - ↳ als een *pointer* verloren gaat kan het bestand nog moeilijk gevonden worden
 - ↳ random toegang is niet mogelijk (het bestand moet sequentieel gevolgd worden)
- geïndexeerde allocatie: een *indexblok* houdt bij welke blokken bij een bepaald bestand horen
 - ↳ random toegang is mogelijk
 - ↳ meest interessante methode
 - ↳ nadelen:
 - een *indexblok* neemt meer plaats in dan een *linked list*
 - als het indexblok te klein is, kan met niet alle pointers van een bestand opslaan

Beheer van vrije ruimte

= ongebruikte blokken bijhouden

2 methodes:

- bitmap waarin voor elk blok een bit is voorzien
 - ↳ 1 = blok is in gebruik, 0 = blok is vrij
 - ↳ nadeel: kan enkel voor kleinere schijven gebruikt worden (bitmap past niet in het geheugen)
- gelinkte lijst
 - = een vrij blok op de schijf wordt gebruikt om pointers naar andere vrije schijfruimtes op te slaan

Veiligheid van files

Meerdere gebruikers afschermen t.o.v. elkaar d.m.v. Read, Write en Execute-rechten (NTFS, Ext3)