Hoofdstuk 2: Hardware

## Geef een beknopt overzicht van de geschiedenis van de programmeertalen (met bekendste voorbeelden indien van toepassing).

* Eerste generatie: machinetaal
* Tweede generatie: assembleertaal
  + Assembler
* Derde generatie: hogere programmeertalen
  + FORTRAN, COBOL, BASIC, PASCAL, C
* Vierde generatie: niet-procedurele en objectgeoriënteerde programmeertalen
  + C++, C#, Visual Basic, Java, XML
* Vijfde generatie: probleemoplossende talen
  + Prolog

## Wat voor soort programmeertaal is Java? Leg dit soort programmeertaal uit. Waarom wordt Java tegenwoordig veel gebruikt?

Een object-georiënteerde programmeertaal. Hierbij staan objecten (bv. gegevens, keuzemenu’s, invulvakken,…) centraal. Kan draaien op verschillende besturingssystemen.

## Benoemde zes belangrijkste onderdelen van een computer, en per onderdeel de functie.

* Moederbord: centrale printplaat, alle onderdelen worden hierop bevestigd
* CPU: de rekenkracht van de computer
* GPU: zorgt er voor dat grafische toepassingen correct verwerkt worden
* RAM: tijdelijke processen opslaan
* Harde schijf of SSD: opslaan van data
* Voeding: geeft stroom aan de onderdelen

## De sloten/sockets/ports/connectoren van een moederbord moet je kunnen benoemen zo een afbeelding is gegeven.

## Waarvan is SSD, SATA, EUFI, RAM en ROM en PCIe de afkorting? Definieer met 2 zinnen elke term.

* SSD: Solid State Drive
  + Medium om gegevens op te slaan. Opvolger van de harde schijf.
* SATA: Serial Advanced Technology Attachment
  + Hardware interface om gegevens tussen computer en SSD/HDD/CD-ROM/DVD/.. te versturen.
* UEFI: Unified Extensible Firmware Interface
  + Firmware op het moederbord waarop bepaalde systeeminstellingen gedaan kunnen worden. Opvolger van BIOS.
* RAM: Random Access Memory
  + Vluchtig geheugen waar processen op draaien. Wanneer de computer afgesloten wordt verdwijnt alle data uit het RAM-geheugen.
* ROM: Read Only Memory
  + Opslaggeheugen waarvan data enkel gelezen kan worden, niet geschreven. Geheugen is niet vluchtig. Bijvoorbeeld CD-ROM.
* PCI-E: Peripheral Component Interconnect Express
  + Slot op het moederbord dat gebruikt wordt om GPU’s, netwerkkaarten, geluidskaarten,… op aan te sluiten.

## Wat is het verschil tussen een DIMM en DDR4/DDR5?

DIMM betekent Dual Inline Memory Module en is niet verschillend van DDR4/DDR5. DIMM is de formfactor van een RAM stick. DDR5 is sneller dan DDR4 (nieuwere generatie RAM). DDR5 werkt ook altijd in dual channel mode en draait op 11V spanning ipv 12V (DDR4).

## USB-C is een speciale aansluiting. Leg de twee verschillende mogelijkheden uit.

* Externe opslag
* Thunderbolt: andere kabels nodig; aansluiten monitor, toetsenbord, muis, …

## Leg het verschil tussen simplex, half duplex en duplex uit.

* Simplex: data kan verstuurd worden maar niet ontvangen
* Half-duplex: data kan verstuurd en ontvangen worden, maar slechts één van de twee tegelijk
* Full duplex: data kan tegelijk verstuurd en ontvangen worden

## Leg een lane uit bij PCIe. Is het een parallelle of een seriële techniek? Waarom wordt tegenwoordig welke techniek gebruikt? Verklaar u nader.

Een lane zorgt voor een communicatiepad in twee richtingen. Serieel. Tegenwoordig wordt serieel gebruikt. Omdat bij parallel er sprake is van timing skew, waarbij aparte elektrische signalen over verschillende conductoren lopen. Hierdoor komen niet alle signalen op hetzelfde moment aan, wat ervoor kan zorgen dat de verstuurde data niet correct aankomt.

## Welke fysieke specificaties definieert het moederbord?

* Centrale printplaat
* Aansluitingen voor alle computeronderdelen (slots)
* Aansluitingen voor randapparatuur
* Wordt vastgeschroefd in de computerkast

## Welke drie functies heeft een chipset?

* System controller
  + Kloksnelheid van de processor
  + Onderbreken van taken (interrupts)
  + Gegevenstransport naar het intern geheugen
  + Energiebeheer
* Peripheral controller
  + Businterface voor uitbreidingskaarten
  + Interface voor harde schijven
  + Toetsenbordcontroller
  + I/O poort controller
* Memory controller
  + Besturing van werkgeheugen

## Waarvan is DMA de afkorting? Leg DMA beknopt uit.

Direct Memory Access. Zorgt voor snelle overdracht tussen randapparatuur en het werkgeheugen. Zonder tussenkomst van de CPU.

## Waarvan is IRQ de afkorting? Leg uit wat een IRQ is. Welke twee soorten onderscheidt men? Wat was vroeger het probleem met IRQ’s? Waarom komt dat tegenwoordig niet meer voor?

Interrupt requests. Een processor krijgt vaak vele verzoeken van devices. Door middel van IRQ’s bepaald de chipset welke verzoeken eerst uitgevoerd worden.

Vroeger waren er maar 16 IRQ adressen beschikbaar. Dit is verholpen met IRQ sharing en meerdere IRQ-adressen.

## Waarvan is BIOS de afkorting? Waarop wordt en werd de BIOS bewaard?

Basic Input Output System. Het flash geheugen.

## Leg het verband tussen BIOS en CMOS uit.

De BIOS-instellingen worden bewaard in het CMOS-geheugen.

## Waarvoor dient het CMOS-geheugen op het moederbord? Waarvan is het de afkorting?

BIOS-instellingen opslaan. Complementary Metal-Oxide Semiconductor

## Waarvan is UEFI de afkorting? Welke voordelen biedt UEFI t.o.v. de klassieke BIOS?

Unified Extensible Firmware Interface.

* Grafisch opgebouwd (muis mogelijk)
* Hogere resolutie mogelijk
* Preboot Execution Environment
  + Instellingen vanop afstand
* Makkelijker op pre-boot applicaties te draaien
  + Diagnosetools
  + Image tools
  + Partitioning tools

## Welke standaard van voeding wordt momenteel bijna altijd gebruikt voor thuisgebruik? Welke standaard wordt gebruikt in de industrie?

ATX. Flex ATX.

## Waarvoor dient de 80 plus certificering?

Geeft de garantie dat het rendement bij een voeding minimum 80% is.

## Leg de Harvard-architectuur uit. Wat is het verschil met de Von Neumann-architectuur? Welke architectuur is het snelst? Wat is het nadeel van de Harvard-architectuur? Waar wordt welke architectuur gebruikt?

Heeft gescheiden bussen en adresruimtes voor data en instructies. Bij Von Neumann worden instructies één voor één opgehaald en uitgevoerd, bij Harvard kan dit gelijktijdig. Harvard is het snelst. Sommige bewerkingen zijn onmogelijk of ingewikkelder bij Harvard.

Von Neumann wordt in de meeste hedendaagse computers gebruikt. Harvard bij microcontrollers en DSP’s.

## Wat is een instructieset? Wat is het verband met een processorfamilie? Geef de twee belangrijkste instructiesets.

Een verzameling van alle mogelijke machinecodes die een processor kan verwerken.

Elke processorfamilie heeft zijn eigen instructieset.

De x86 instructieset en de ARM instructieset.

## Leg in drie stappen de instructieafhandeling bij x86 uit.

* Prefetch: ophalen van de instructie uit het inwendig geheugen
* Decode: omzetten van de instructie naar een voor de processor verstaanbare opdracht
* Execute: uitvoeren van de instructie

## Bespreek beknopt pipelining a.d.h.v een zelfgemaakte tekening.

Zorgt ervoor dat een aantal instructiefases gelijktijdig kunnen worden uitgevoerd.

|  |  |  |
| --- | --- | --- |
| Prefetch A |  |  |
| Prefetch B | Decode A |  |
| Prefetch B | Decode B | Execute A |
|  | Decode C | Execute B |
|  |  | Execute C |

## Bespreek beknopt Dynamic Branch Prediction (met tekening).

De processor probeert te voorspellen welke instructie eerst uitgevoerd moet worden.



## Bespreek beknopt Out of Order Execution (met tekening).

Indien een instructie niet afhankelijk is van een andere instructie, wordt deze vroeger in de pipeline geplaatst.

Stel dat:

* A = 1
* B = 2
* C = A + B
* D = 4



## Wat is een thread en wat is hyper threading.? Wat is het verschil tussen hyper threading en multi core?

Een thread is de kleinste reeks geprogrammeerde instructies die onafhankelijk kan worden beheerd. Hyper threading betekent dat één enkele processorkern twee logische kernen bevat die de caches en externe interfaces deelt.

Het verschil is dat bij multi core de processor meerdere fysieke kernen bevat.

## Wat verstaat men met een Multi-core processor?

Dat de meerdere fysieke kernen in een CPU afzonderlijke cachegeheugens hebben en afzonderlijk werken.

## Welke (zes) factoren kunnen de snelheid van een processor bepalen?

* Kloksnelheid
* Aantal instructies per cyclus
* Cache
* Pipelining
* Aantal kernen
* Hyperthreading

## Waarom werkt een computer met twee kloksnelheden?

Geen idee

## Leg aan de hand van een voorbeeld het verschil uit tussen een RISC en een CISC-processor. Waarvan zijn het de afkortingen?

RISC-processoren zoals de ARM processor in een smartphone zijn eenvoudig te produceren en miniaturiseren. Dit is zuiniger en kleiner dan een CISC-processor.

* RISC: Reduced Instruction Set Computing
* CISC: Complex Instruction Set Computing

## Hoe tracht RISC even snel te zijn dan CISC?

Een RISC-processor heeft minder schakelingen, geen microcode, een kleinere instructieset en eenvoudiger gecodeerde instructies. Door de kleinere instructieset moet de processor meer instructies uitvoeren. Dit wordt gecompenseerd door hogere snelheden.

## Hoe komt het dat de x86 instructieset zo populair is? Tip: RISC en CISC.

De x86 instructieset bevat ook RISC-concepten. Alleen complexe code in microcode.

## Bij welke architectuur wordt microcode gebruikt? Waarom kan bij een bepaalde architectuur dit weggenomen worden?

Bij x86 architectuur (CISC). Dit kan weggenomen worden bij RISC omdat de instructies simpel genoeg zijn gecodeerd.

## Leg de “wet van Moore” uit.

Het aantal transistors in een microprocessor verdubbelt elke twee jaar.