

# Formules Computerarchitectuur

Haroen Viaene

19 januari 2015

# 1 Computervoeding

$$P = U * I \text{ (vermogen = spanninging * stroom)}$$

$$Q = I * t \text{ (capaciteit = stroom * t)}$$

## 2 Geheugen

### Localiteitsprincipes

tijdsgebonden: je zal iets niet lang erna opnieuw nodig hebben

plaatsgebonden: je zal de volgende data ook nodig hebben

### 2.1 RAM

DIMM: twee kanten werken apart, SIMM: onderling verbonden (omkeerbaar)

statisch: flipflop (cache), dynamisch condensator (destructief, herladen) (RAM)

$$\frac{\frac{\text{bytes}}{\text{transfer}} \cdot \frac{\text{transfers}}{\text{burst}} \frac{\text{cycli}}{\text{sec}}}{\frac{\text{cycli}}{\text{burst}}} \text{ vb: } \frac{8(=64\text{bit}) \cdot \frac{\text{bytes}}{\text{transfer}} \cdot 4(=1+1+1+1) \cdot \frac{\text{transfers}}{\text{burst}} 60\text{MHz} \frac{\text{cycli}}{\text{sec}}}{14(=5+3+3+3) \frac{\text{cycli}}{\text{burst}}}$$

FP-DRAM (fast pace): schrijft de paar volgende lijnen ook

EDO-RAM (extended data out): data blijft staan op de uitgang terwijl nieuwe al klaar-gezet wordt

SD-RAM (Synchronous DRAM): volgende kolommen worden sowieso gelezen op de snelheid van de klok

DDR (Double Data Rate): hogere kloksnelheid, idem voor DDR2,3,4

#### optimalisatie

Interleaving: wissel af tussen twee banken om hersteltijd te vermijden

Dual channel: gebruik twee banken tegelijk

Buffered: slaat alle data op op een cache

is van toepassing op alle systemen

## 2.2 Cache

Vervangen cache

FIFO: first in first out

LRU: least recently used

LFU: least frequently used

ARC: adaptive replacement cache (LFU en LRU combinatie)

types cache

fully associative: elk deeltje vd RAM heeft een plaats

direct mapped: elk deel vd geheugenbank heeft 1 plaats in cache, als overschreven wordt is algoritme nodig

set-associative: RAM opgedeeld in sets, binnen die sets wordt overschreven

## 2.3 formule

$$lines = \frac{cache}{linesize}$$

$$sets = \frac{linesize}{wayset}$$

$$sets = 2^X \text{ en } linesize = 2^Y$$

$$taggrootte = (bussize - X - Y) \cdot lines$$

$$toegangstijd: t_{totaal} = HR_{L1} \cdot t_{L1} + (1HR_{L1}) \cdot HR_{L2} \cdot t_{L2} + (1HR_{L1}) \cdot (1HR_{L2}) \cdot t_{RAM}$$

## 3 Data

sector: kleinste hoeveelheid door kop leesbaar

track: lijn op schijf

schijf: ...

FCFS: first come first serve

SSTS: shortest seek time first

SCAN: van 0 → einde, doet in terugkeren waaraan er toch gepasseerd wordt

C-SCAN: slaat alles over in terugkeren

LOOK: SCAN, maar gaat niet tot einde

C-LOOK: LOOK, maar slaat alles over in terugkeren

### 3.1 bestandssystemen

HALP