

Computersystemen 2

Theorie

1. Herhaling

Inhoud

- Grootheden
- Harvard / Von Neumann architectuur
- Registers / RAM geheugen

Grootheden

- **Binair**

Macht	van	2	
1 Ki	2^{10}	1024	1,024 K
1 Mi	2^{20}	1024^2	$1,024^2$ M
1 Gi	2^{30}	1024^3	$1,024^3$ G
1 Ti	2^{40}	1024^4	$1,024^4$ T
1 Pi	2^{50}	1024^5	$1,024^5$ P

- **Decimaal**

Macht	van	10	
1 K	10^3	1000	$1,024^{-1}$ Ki
1 M	10^6	1000^2	$1,024^{-2}$ Mi
1 G	10^9	1000^3	$1,024^{-3}$ Gi
1 T	10^{12}	1000^4	$1,024^{-4}$ Ti
1 P	10^{15}	1000^5	$1,024^{-5}$ Pi

Zet om

- 68 608 = Ki
- 46 080 = Ki
- 5 242 880 = Mi
- 150 KiB = KB
- 300 MiB = MB
- 2 GB = GiB
- 5 MB = MiB
- Past een bestand van 95 GiB op een HD van 100 GB?
- 0x100000 = □i
- 0x4000 0000 = □i

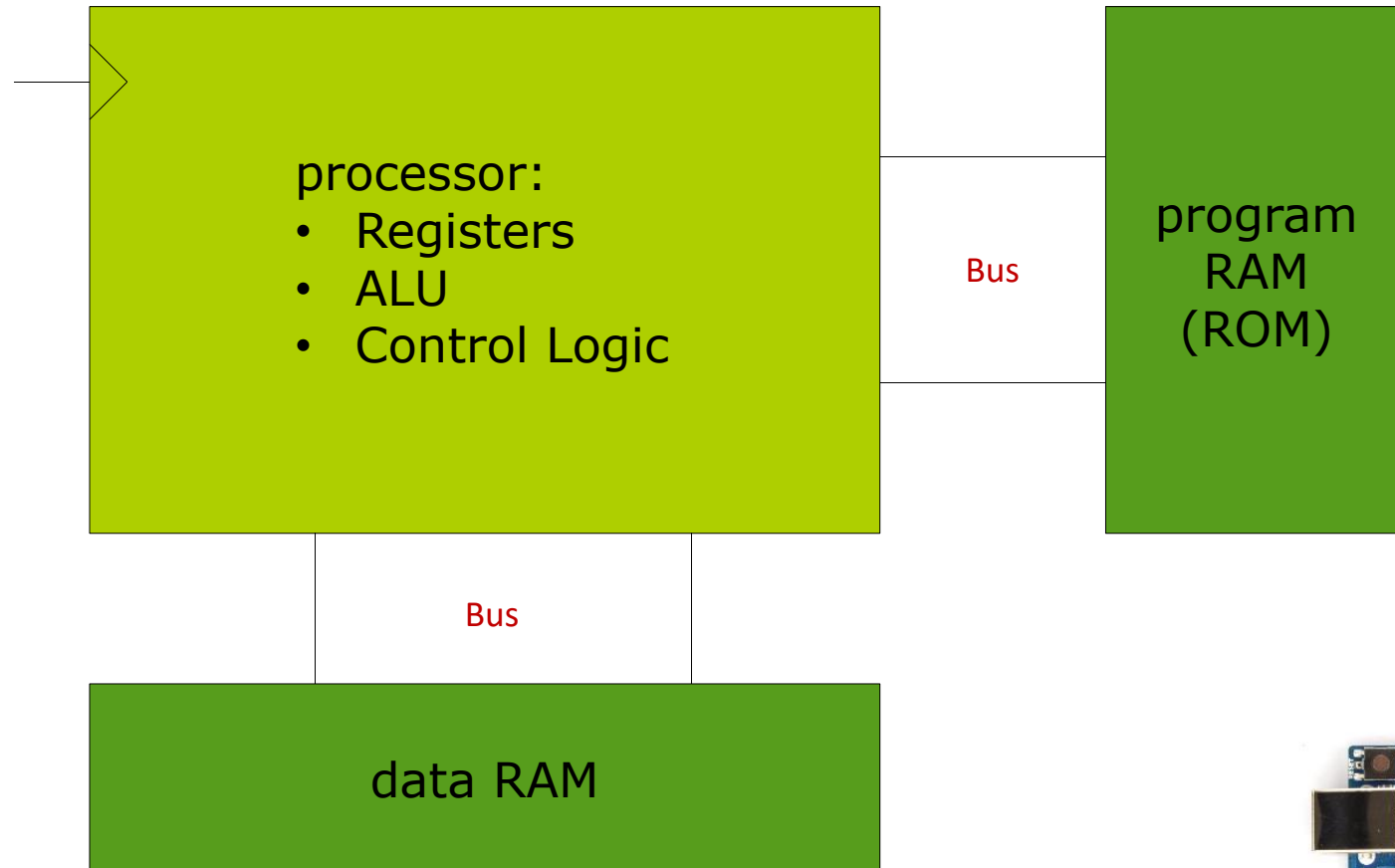
Zet om

1. $68\,608 = 68\,608 / 1024 \text{ Ki} = 67 \text{ Ki}$
2. $46\,080 = \dots\dots\dots \text{ Ki}$
3. $5\,242\,880 = \dots\dots\dots \text{ Mi}$
4. $150 \text{ KiB} = \dots\dots\dots \text{ KB}$
5. $300 \text{ MiB} = 300 * 1,024^2 \text{ MB} = 314,6 \text{ MB}$
6. $2 \text{ GB} = \dots\dots\dots \text{ GiB}$
7. $5 \text{ MB} = \dots\dots\dots \text{ MiB}$
8. Past een bestand van 95 GiB op een HD van 100 GB?
9. $0x100000 = 1 * 16^5 = (2^4)^5 = 2^{20} = 1 \text{ Mi}$
10. $0x4000\,0000 = \dots\dots\dots \text{ Ki}$

Inhoud

- Grootheden
- Harvard / Von Neumann architectuur
- Registers / RAM geheugen

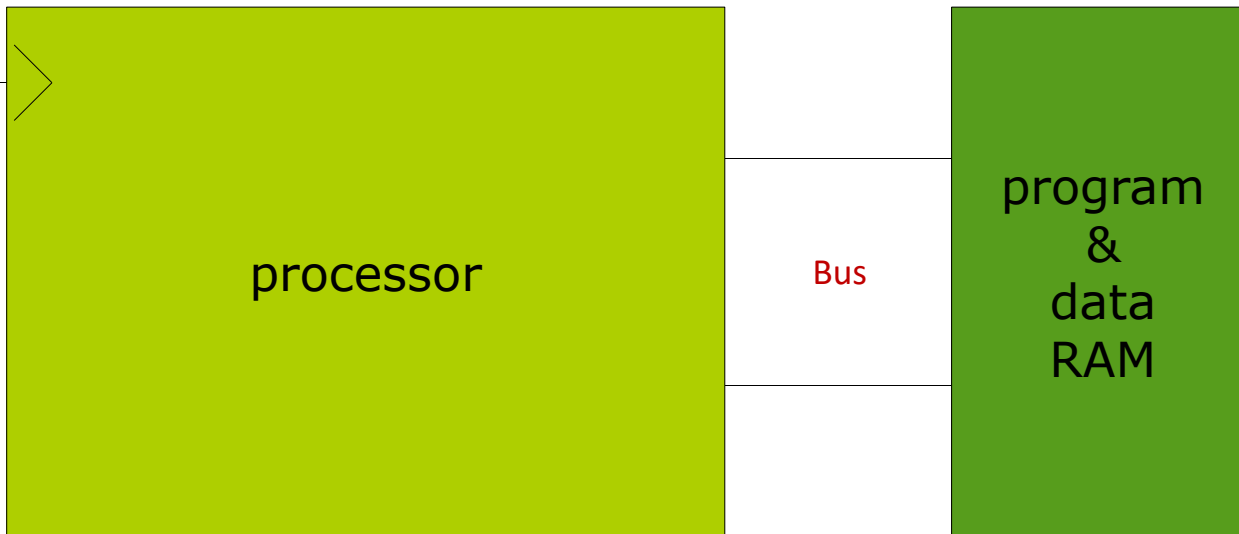
Harvard architectuur



Arduino



Von Neumann

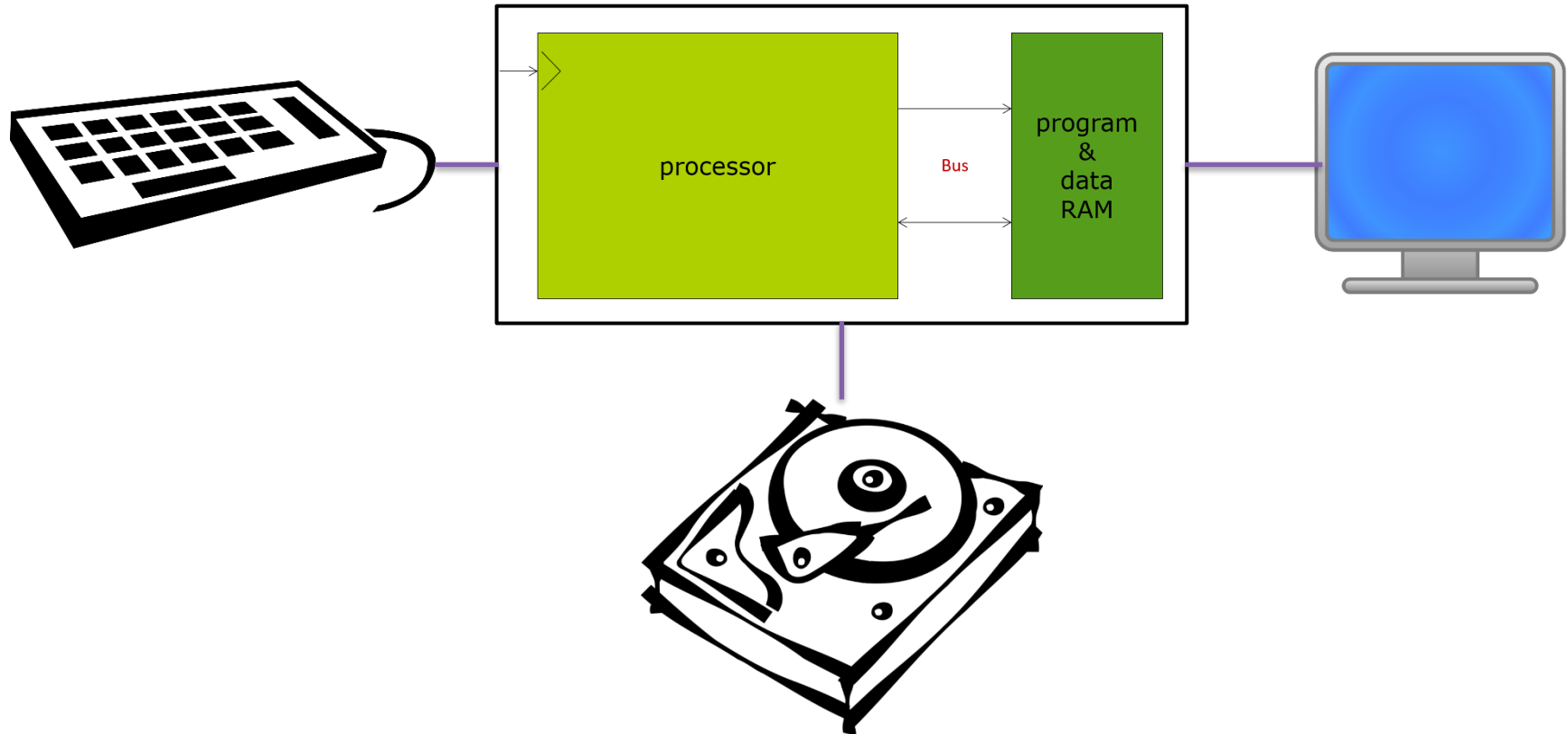


Raspberry Pi

Instructiecyclus: fetch, decode, execute



Von Neumann – with I/O devices

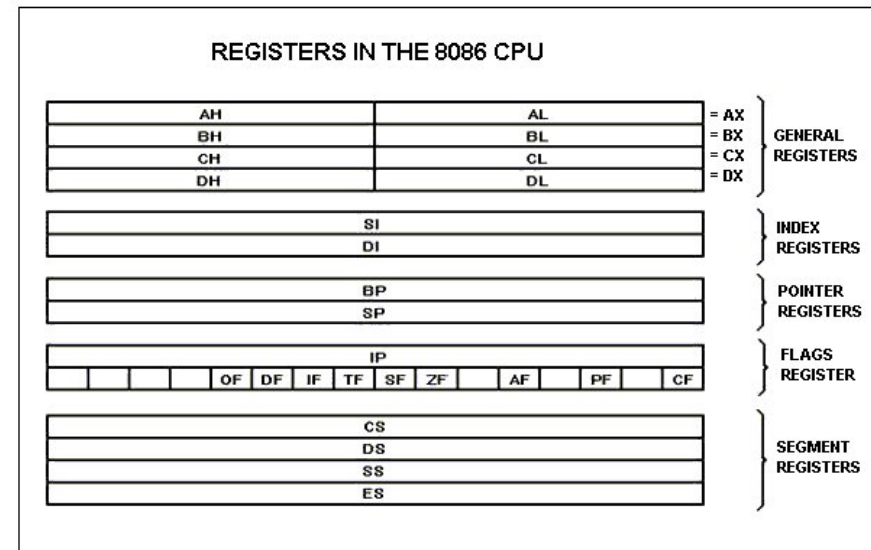


Inhoud

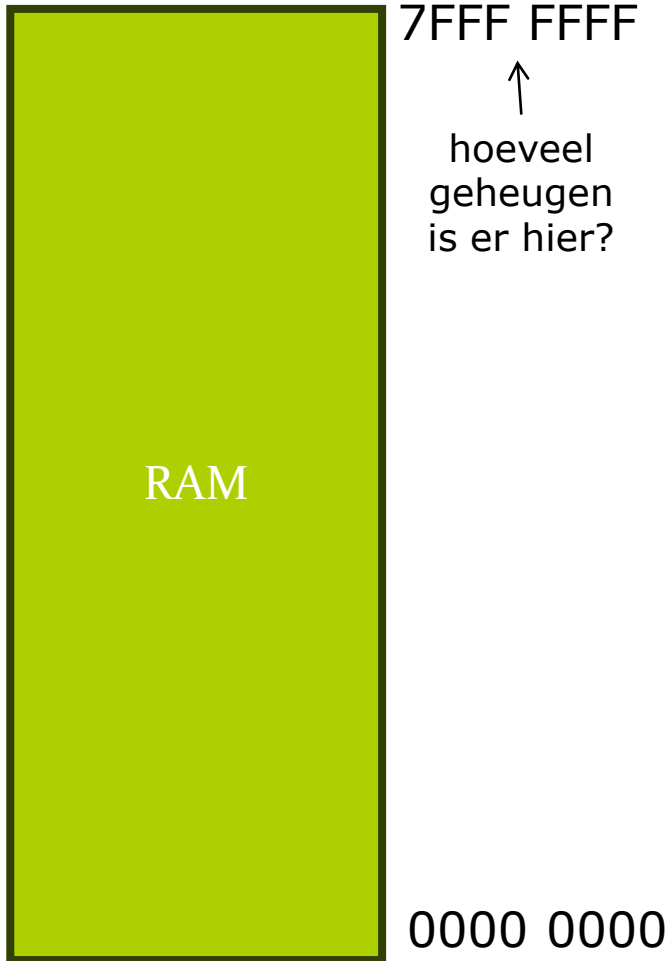
- Grootheden
- Harvard / Von Neumann architectuur
- Registers / RAM geheugen

Registers

- Gegevensregisters (= tijdelijke opslagplaatsen)
- Adresregisters
 - indexregisters
 - segmentregisters
 - stackpointer
- Stuur- en statusregisters
 - program counter
 - flags



RAM Geheugen



- code
 - machine-code: bytes
 - instructie: instructie code + argumenten
 - vb:
 - laad waarde uit geheugen nr register
 - stockeer waarde uit register in geheugen
 - laad register met waarde
 - tel 2 registers op
 - spring naar ander adres als...
- data
 - getallen (bytes, words, floating point, ...)
 - tekst (ASCII, unicode, EBCDIC, ...)
- zowel code als data zijn bytes
 - data kan als code uitgevoerd worden!
 - beveiliging...

Voorbeeld Intel 8086

- 16 bit processor
 - registers zijn 16 bit
 - grootste getal in register: $2^{16} = 2^6 \cdot 2^{10} = 64 \text{ Ki}$
- Adresbus 20 bit
 - maximale grootte RAM geheugen: $2^{20} = 1 \text{ Mi}$
- Hoe kan je 20 bit adressen maken met 16 bit registers?
 - door combinatie van een register met een segment register
 - adres = CS * 16 + IP
 - maximale grootte: $2^{16} * 16 + 2^{16}$
- PS:
 - Intel Pentium: 32 bit registers en 32 bit adresbus
 - i7: 64 bit en adresbus 40-→52 bit

