

Hoofdstuk 1

Computers en gegevens



UC Leuven
Limburg
MOVING MINDS

Computersystemen
(MBI26A)



H1: Computers en gegevens

- Verwijzingen naar de hardware
- Binaire voorstelling van integers
- Binaire voorstelling van tekst
- Binaire voorstelling van reals



UC Leuven
Limburg

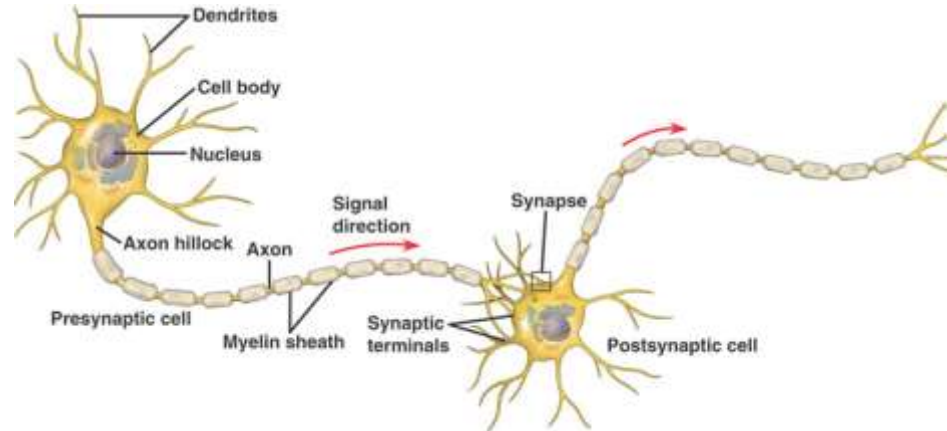
Gegevens





Gegevensvoorstelling

- **Mensen** slaan complexe data op als neuronverbindingen in de hersenen



- **Computers** gebruiken **bits** om data op te slaan
 - 0 of 1



- Binary Digit (bit)
- Bijvoorbeeld:
 - Het licht is uit (0) of aan (1)
 - Een auto is rood (0) of geel (1)
 - Een boek in de bib is aanwezig (0) of uitgeleend (1)
 - ...



- Meer mogelijkheden?

000 O+

001 O-

010 A+

011 A-

100 B+

101 B-

110 AB+

111 AB-

- Context van het bitpatroon is belangrijk!



Bits opslaan

- Processor en geheugen: transistors
- Harde schijf: magnetische lading
- CD: een putje of geen putje
- ...





Instructies

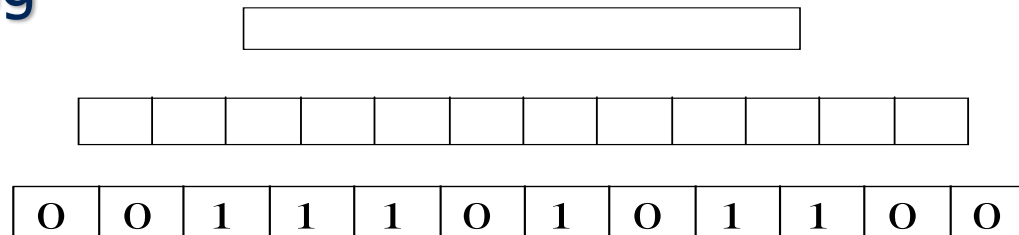


- Commando's die een processor kan uitvoeren
 - Tel dit getal op bij dat getal
 - Kopieer dit getal naar daar
 - Vergelijk deze getallen
 - ...



Registers

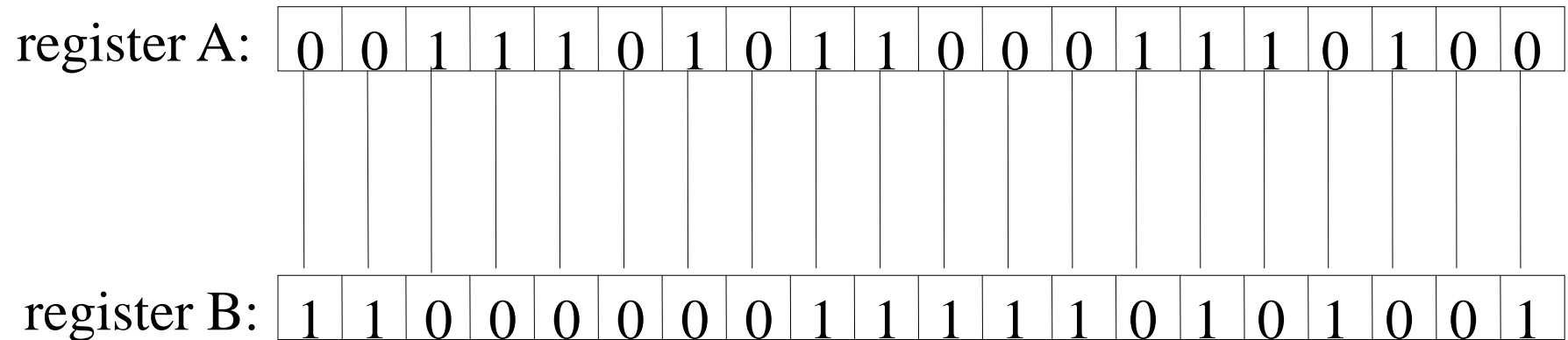
- Stukjes geheugen in een processor
- Bepaalde lengte (8, 16, 32, 64, 128 bits...)
- Inhoud kan als invoer/uitvoer dienen voor een instructie
- Intel x86-processoren hebben registers (EAX, EBX, ECX, ...) van 32 bits.
 - Register EAX bevat 00011110011011101011101101001101
 - Voorstelling





Verbindingen

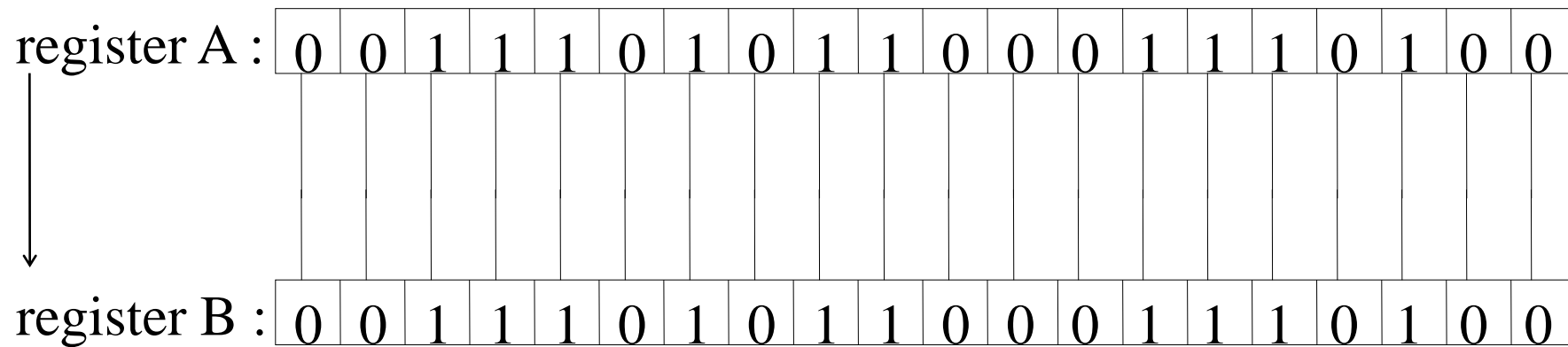
- Nodig om te kunnen kopiëren.
- Sommige registers zijn direct met mekaar verbonden





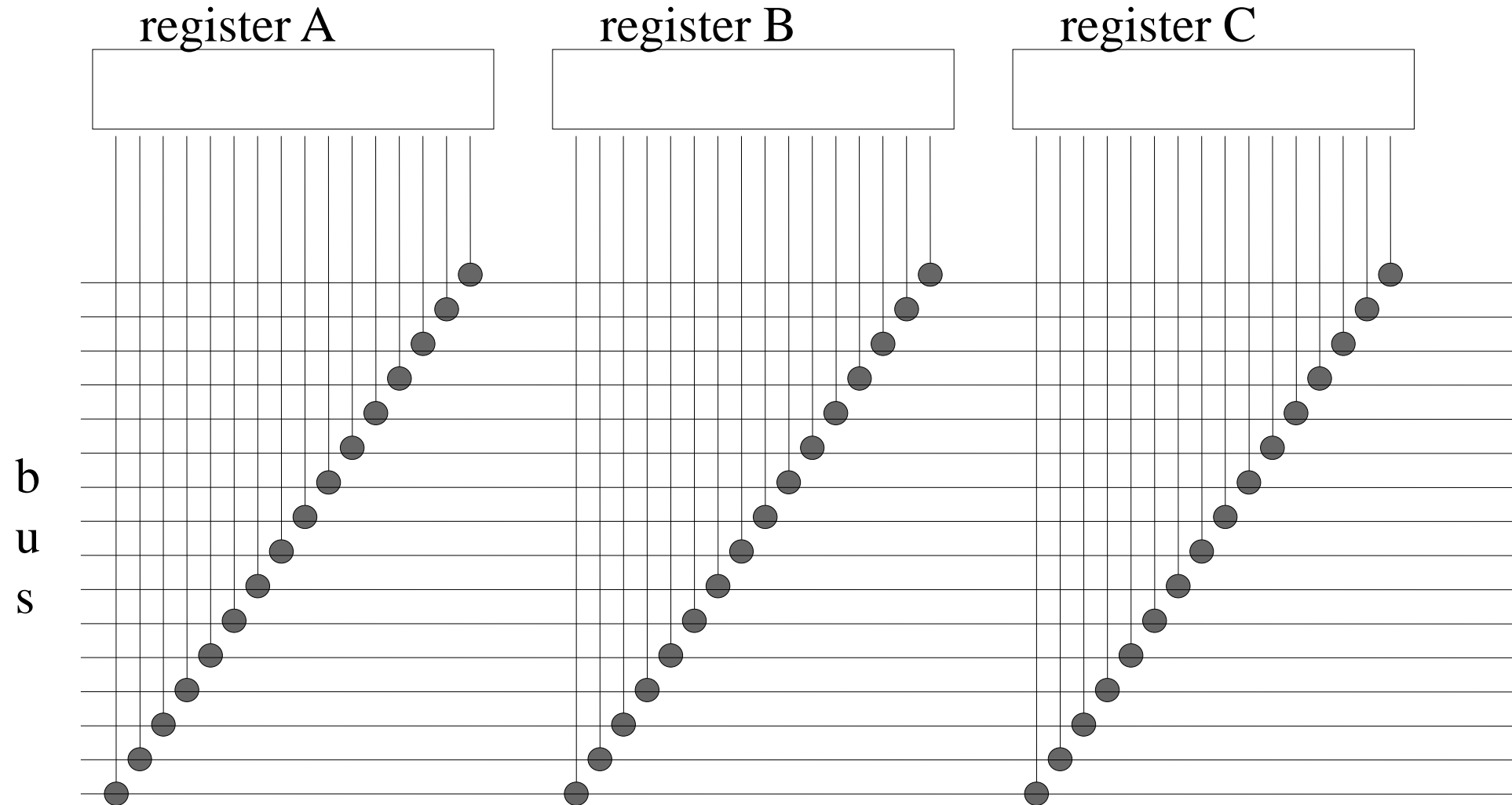
Kopieren via Directe Verbinding

- Processor krijgt een kopieerbevel





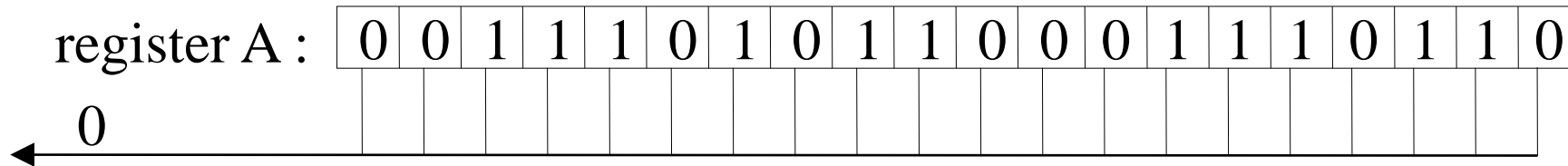
Busverbindingen



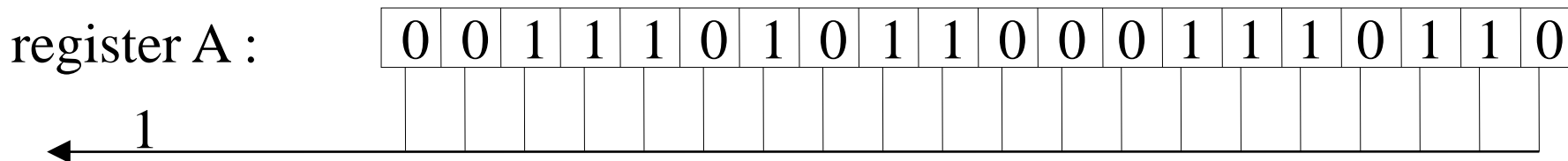


Parallel t.o.v. Serieel

- Parallel: meerdere bits tegelijkertijd
- Serieel: bits na mekaar over dezelfde geleider



Laatste bit;



Voorlaatste bit;



H1: Computers en gegevens

- Verwijzingen naar de hardware
- **Binaire voorstelling van integers**
- Binaire voorstelling van tekst
- Binaire voorstelling van reals



Natuurlijke Getallen

- Decimaal: 10 cijfers
 - 0, 1, 2, ... 9
 - Getallen groter dan 9, meerdere cijfers achter mekaar:
 - $4506 = 4 \times 10^3 + 5 \times 10^2 + 0 \times 10^1 + 6 \times 10^0$
- Binair: 2 cijfers
 - 0, 1
 - Getallen groter dan 1, meerdere bits achter mekaar:
 - $10011 = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (19)_d$



Leidende Nullen

- $10011 = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
 $= 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
 $= 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
- Dus: $10011 = 010011 = 0010011 = \dots$
 - Net zoals in decimaal!



- Van binair naar decimaal: OK
 - $10011 = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (19)_d$
- Van decimaal naar binair?
 - Omgekeerde aanpak!



128 64 32 16 8 4 2 1

1 0 0 0 0 0 1 1

→



Binaire Optelling

- Tafel:

0	0	1	1
<u>0</u>	<u>1</u>	<u>0</u>	<u>1</u>
0	1	1	10

- Met overdracht

(1)	(1)	(1)	(1)
0	0	1	1
<u>0</u>	<u>1</u>	<u>0</u>	<u>1</u>
1	10	10	11



Binaire Optelling

- Vb.

111111
00101101
00111011
01101000



- Wat is het dubbel van een getal?
- $10111 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
- $2 \times 10111 = 2 \times (1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0) = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 101110$
- $\times 2$: achteraan een 0 bijvoegen



Binair Rekenen

- Binair naar decimaal ✓
- Decimaal naar binair ✓
- Binaire optelling ✓
- Binaire vermenigvuldiging met 2 ✓
- Ook binaire aftrekking, vermenigvuldiging, deling, etc. bestaat
- Nadeel binair: lang!
 - $(9999)_d = (10011100001111)_b$



Hexadecimaal

- 16 cijfers:
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
- Vb. : $(A5B)_h = A \times 16^2 + 5 \times 16^1 + B \times 16^0$
 $= (2651)_d$



Hexadecimaal

- Verband met binair stelsel.

d.	h.	b.	d.	h.	b.
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	3	0011	11	B	1011
4	4	0100	12	C	1100
5	5	0101	13	D	1101
6	6	0110	14	E	1110
7	7	0111	15	F	1111



Hexadecimaal

- Wat is de binaire voorstelling van A5B?

De b.v. v. $B \times 16^0$ is : **1011**

De b.v. v. 5×16^1 is : **0101 0000**

De b.v. v. $A \times 16^2$ is : **1010 0000 0000**

De b.v. v. A5B is : **1010 0101 1011**

- Hexadecimaal : afkorting voor binair.



Hexadecimale Optelling

- Vb.

$$\begin{array}{r} 1 \quad 1 \\ \text{C3A} \\ + \text{DB9} \\ \hline 19\text{F3} \end{array}$$

$$\begin{aligned} A + 9 &= (19)_{\text{d}} = (13)_{\text{h}} \\ 1 + 3 + B &= (15)_{\text{d}} = \text{F} \\ C + D &= (25)_{\text{d}} = (19)_{\text{h}} \end{aligned}$$



Negatieve Getallen

- In decimaal: + en –
- In binair: enkel 0 en 1!
 - - voorstellen door 1
 - + voorstellen door 0
 - Waar? 1^e bit

Eis: *we moeten werken met een vast aantal bits (bijv. een register)*

Poging 1 (4 bits)

Decimaal	Binair
-1	1 001
6	+ 0 110
-7	1 111

Besluit: poging 1 werkt niet voor negatieve getallen



Poging 2 (4 bits)

- Welke getallen stellen we voor?
 - ..., -2, -1, 0, 1, 2, ...
 - Waar stoppen we? **Hangt af van het aantal bits!**
- Voor 4 bits: 2^4 unieke binaire combinaties
 - 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111
 - Plaats voor 8 negatieve en 8 niet-negatieve getallen (-8 tot 7 dus)



Poging 2 (4 bits)

- Niet-negatieve getallen?
- Doe gewoon:

0=0000	1=0001	2=0010	3=0011
4=0100	5=0101	6=0110	7=0111

Merk op: *elk niet-negatief getal begint met een 0-bit!*



Poging 2 (4 bits)

- Negatieve getallen?
- Hoe stellen we -1 voor? Door 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111?
- Lumineus idee: bij welk getal moeten we 1 optellen om 0 te bekomen?

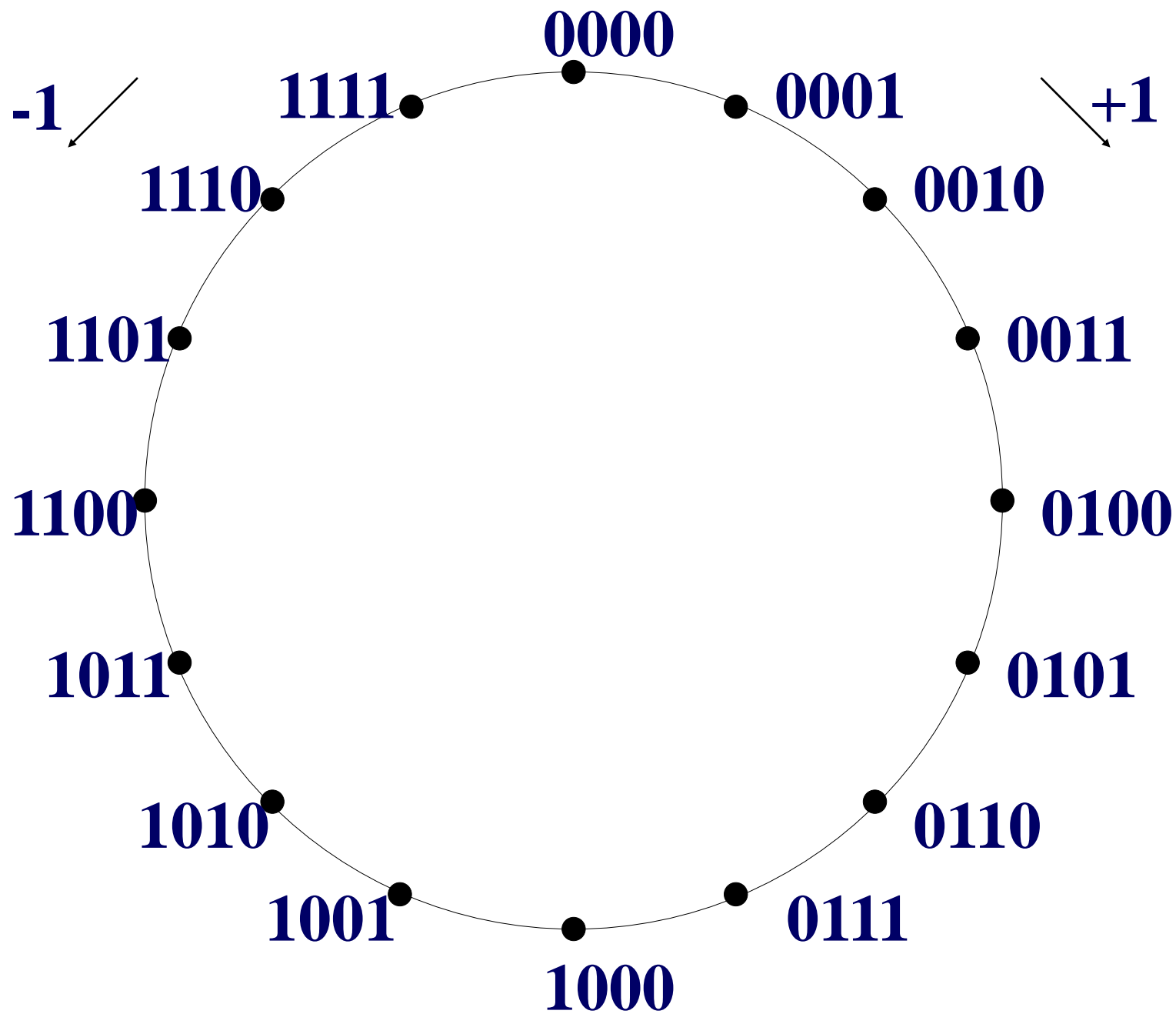
$$\begin{array}{r} \text{????} \\ +0001 \\ \hline 0000 \end{array}$$



Poging 2 (4 bits)

- $(-1)_d = (1111)_b$
- $(-2)_d = (????)_b$
 - Zelfde truc:

$$\begin{array}{r} 1110 \\ +0001 \\ \hline 1111 \end{array}$$



-1
6
?

Poging 2 (4 bits)

Decimaal	Binair
-1	1 111
6	+ 0 110
5	0 101

Besluit: poging 2 werkt voor alle gehele getallen



Negatieve Getallen

- Hoe vinden we de voorstelling van een negatief getal?

- Experiment:

- a) Neem een willekeurige bitrij 010110
- b) Neem het complement ervan 101001
- c) Tel beide op 111111
- d) Wat is het resultaat? (-1)_d

- $010110 + 101001 = 111111$

- $010110 + (101001 + 000001) = 000000$

Positief
getal

Overeenkomstig
negatief getal

Negatieve Getallen

- Vb.: wat is de binaire voorstelling van -9 (in 6 bits)
 1. Wat is de binaire voorstelling van +9? **001001**
 2. Wat is het complement hiervan? **110110**
 3. Wat is het complement+1? **110111** = **$(-9)_d$**
- Vb. 2: welk getal is 101110?
 - Negatief getal, dus eerst de complementsregel toepassen!
 1. Wat is het complement hiervan? **010001**
 2. Wat is het complement+1? **010010**
 3. Welk getal is dit? **$(18)_d$**
 - Dus: originele getal was **$(-18)_d$**



- Welk getal is $(1100\ 0011)_b$?
 - 195
 - -61
 - ...?

Opgelet: *aan een bitrij kan je niet zien wat ze betekent; de context is belangrijk!*



Hogere Programmeertaal

- Een hogere programmeertaal laat je toe om de context van een bitrij te definiëren:
 - `uint i = 3000000000;`
 - `int j = -587489;`

10110010110100000101111000000000

11111111111101110000100100011111

2-complementsnotatie

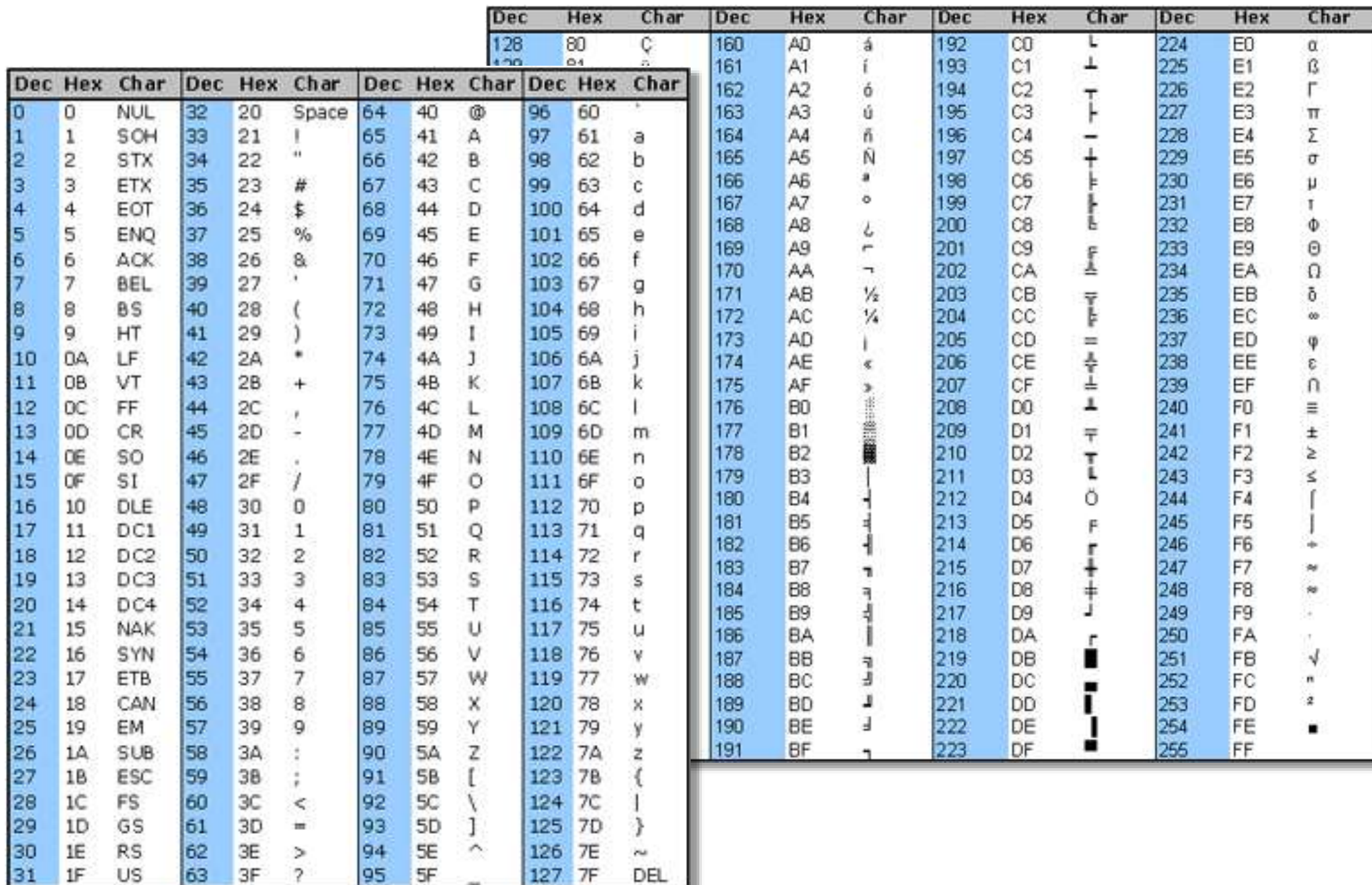


H1: Computers en gegevens

- Verwijzingen naar de hardware
- Binaire voorstelling van integers
- **Binaire voorstelling van tekst**
- Binaire voorstelling van reals

Binaire Voorstelling van Tekst

- Computergeheugen: enkel bits
 - Hoe stellen we tekst voor?
- ASCII (american standard code for information interchange)
 - (oude) standaard die letters en andere tekens mapt op getallen
 - Per karakter worden 8 bits (= byte) gebruikt
 - A 01000001
 - a 01100001
 - 1 00110001
 - ...



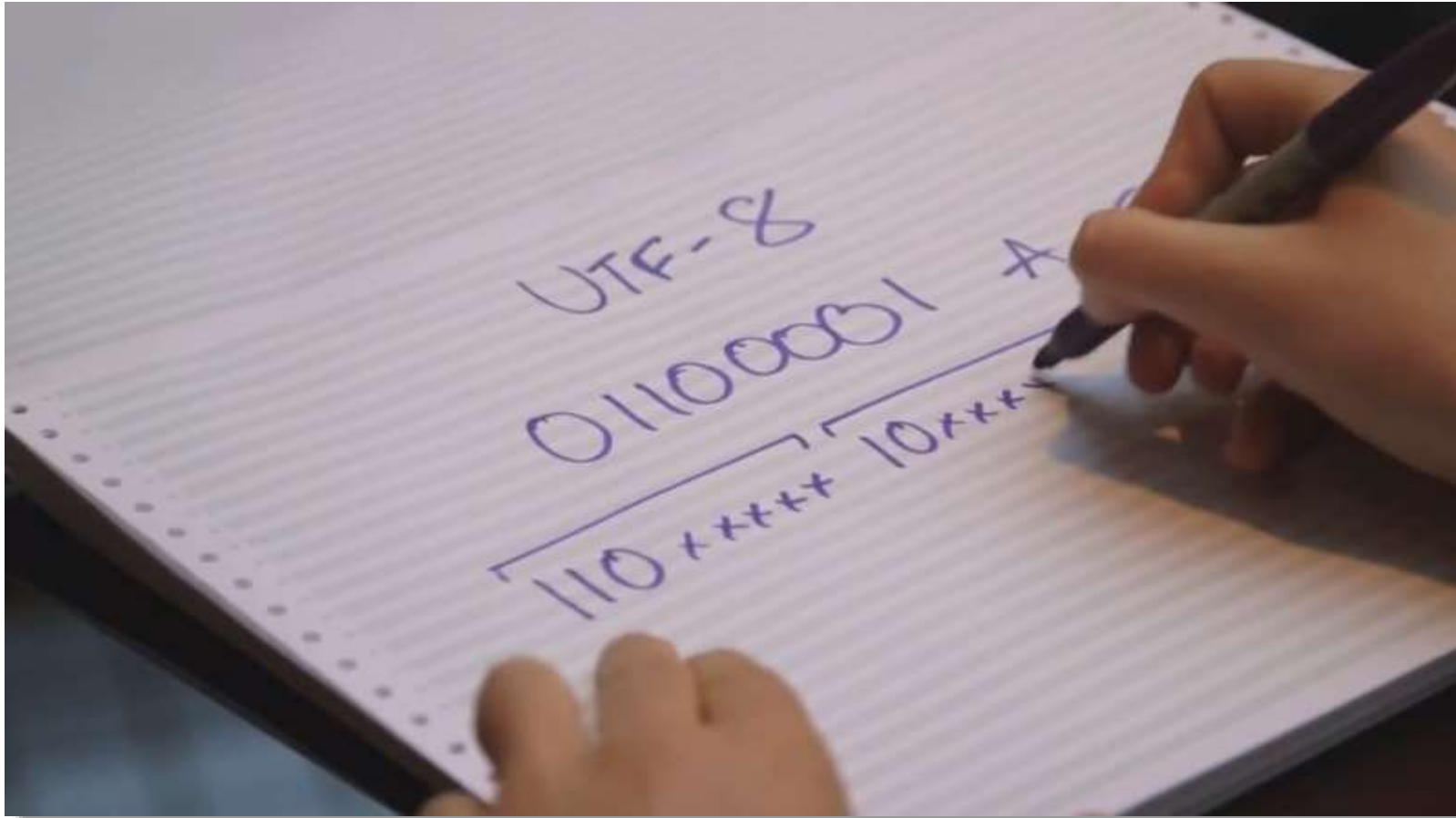


- Hoeveel ascii-tekens?
 - Oorspronkelijk eerste bit = 0: Latin
 - Ook â, é, è, ü, ...: eerste bit 1, Latin-1
 - Veel te weinig
- Unicode: ondersteuning voor meer karakters
 - **UTF-16:** 2 bytes per karakter
 - **UTF-32:** 4 bytes per karakter
 - **UTF-8:** variabel aantal bytes per karakter



UC Leuven
Limburg

Characters, Symbols and the Unicode Miracle





- Opeenvolging van karakters: **string**
- Vb.: "Het Spaanse graan heeft de orkaan ..."
staat in het geheugen als:
48 65 74 20 53 70 61 61 6E 73 65 20 ...
in opeenvolgende bytes

Opgelet: *als een gebruiker een getal intypt,
komt dit getal in het geheugen in ASCII-
vorm; met deze bytes kan je niet rekenen!*



ASCII en Binaire Getallen

- Stel, de gebruiker typt het getal **1811** in
 - In het geheugen komen de bytes **31 38 31 31**
 - In binair: **00110001 00111000 00110001 00110001**
 - Dit is *niet* de binaire voorstelling van 1811!
- Eerst omzetten naar de binaire voorstelling
 - $(31\ 38\ 31\ 31)_{\text{ascii}} = (1811)_d = (11100010011)_b$
- Dan rekenen...



H1: Computers en gegevens

- Verwijzingen naar de hardware
- Binaire voorstelling van integers
- Binaire voorstelling van tekst
- ~~Binaire voorstelling van reals~~
 - Behoort niet tot de leerstof