

## Les A-03 Binaire en hexadecimale getallen

In deze les wordt behandeld hoe getallen kunnen worden voorgesteld door informatie die bestaat uit reeksen 0-en en 1-en. We noemen deze informatie **digitale informatie**. Het woord digitaal is afgeleid van het Engelse woord “digit”, dat cijfer, nummer, teen of vinger betekent.

### 3.1 Digitale informatie

Een computer is een elektronisch apparaat. Zowel het transportmedium (de belijning waarover de stroom loopt) als het geheugen van de computers (miljarden schakelaars die aan of uit kunnen staan) kunnen slechts in de twee toestanden (0 en 1) verkeren. Een informatiedrager voor computers moet dus per definitie bestaan uit 0-en en 1-en.

Allerlei soorten informatie moeten door 0-en en 1-en voorgesteld kunnen worden. Niet alleen getallen, maar ook tekst, geluid, plaatjes, enz. Dat betekent dat er afspraken zijn gemaakt op welke manier al die informatie wordt gecodeerd met 0-en en 1-en

De informatie eenheid van de computer is de **bit** (binary digit), het kleinst mogelijke stukje informatie. Een bit heeft de waarde 0 of 1. Met één enkele bit is het al mogelijk om informatie te coderen, bijvoorbeeld:

- 0                    1
- uit (0)            aan (1)
- nee (0)            ja (1)
- onwaar (0)        waar (1)
- wit (0)             zwart (1)

Met één bit kunnen we dus informatie coderen die in twee toestanden kan verkeren. Als we bits gaan combineren hebben we altijd een veelvoud van 2 aan mogelijkheden:

- 1 bit:                2 mogelijkheden ( 0, 1 )
- 2 bits:              4 mogelijkheden ( 00, 01, 10 en 10 )
- 3 bits:              8 mogelijkheden ( 000, 001, 010, 100, 011, 101, 110, 111 )
- enz.

Computers werken met veelvoud van 8-bits. Een achttal bits wordt ook wel een **byte** (by eight) genoemd. We gebruiken de byte om aan te geven hoe omvangrijk een bepaalde hoeveelheid digitale informatie is.

- **byte:**                1 B = 8 bits
- **kilobyte:**        1 kB = 1024 bytes
- **megabyte:**       1 MB = 1024 kB = 1048576 bytes = 8388608 bits
- **gigabyte:**        1 GB = 1024 MB = 1073741824 bytes = 8589934592 bits
- **terabyte:**        1 TB = 1024 GB = 24 x 1024 bytes = 8796093022208 bits

Let op dat het gebruik van de aanduiding “kilo”, “mega” en “giga” en “tera” in de computerwereld dus enigszins afwijkt van die in de wiskunde en natuurkunde.

Een Word-bestand van 40 kb bestaat dus uit  $40 \times 1024 = 40960$  bytes = 327680 bits, dus ruim 300000 0-en en 1-en.

### 3.2 Decimale getallen

Wij zijn gewend om te rekenen met decimale getallen.

#### **De waarde van decimale getallen**

Een **decimaal getal** is een reeks van cijfers, waarbij elk cijfer gekozen kan worden uit de cijfers 0 t/m 9. We staan er niet bij stil, maar in het getal 12243 heeft de ene “2” niet dezelfde betekenis als de andere. Het is voor ons een automatisme dat elk cijfer uit het getal **12243** een veelvoud is van een macht van 10. Zo heeft de eerste “2” uit het getal **12243** de waarde 2000 en de volgende “2” de waarde 200. Een decimaal getal is dus opgebouwd uit machten van 10:

$$\begin{aligned} \mathbf{1} &= 10000 = 1 \times 10000 = 1 \times 10^4 \\ \mathbf{2} &= 2000 = 2 \times 1000 = 2 \times 10^3 \\ \mathbf{2} &= 200 = 2 \times 100 = 2 \times 10^2 \\ \mathbf{4} &= 40 = 4 \times 10 = 4 \times 10^1 \\ \mathbf{3} &= 3 = 3 \times 1 = 3 \times 10^0 \end{aligned}$$

Een eenvoudige manier om de waarde van het getal 12243 voor te stellen is:

<b>1</b>	<b>2</b>	<b>2</b>	<b>4</b>	<b>3</b>
10000	1000	100	10	1

Het achterste cijfer heeft altijd de waarde van de eenheid, het volgende cijfer de waarde van tiental, het volgende cijfer de waarde van honderdtal, enz.

#### **Rekenen met decimale getallen**

Ook bij het rekenen met decimale getallen gaan we uit van allerlei automatismen. Bij een optelling van decimale getallen gaan we als volgt te werk:

$$\begin{array}{r} 12243 \\ 68925 \\ \hline + \\ 81168 \end{array}$$

De regel “één onthouden” als een optelling van twee cijfers de 10 overschrijdt (zoals bij het optellen van 2 en 9) is een regel die je op de basisschool hebt geleerd en inmiddels als het ware automatisch toepast. Zou dat anders werken als je in een ander getalstelsel als het decimale stelsel gaat rekenen?

### 3.3 Binaire getallen

#### De waarde van binaire getallen

Ons gebruik van het tientallig of decimale getalstelsel vindt zijn oorsprong in het feit dat wij met tien vingers kunnen rekenen. Een computer heeft niet zoveel vingers. Zijn micro-elektronica kent slechts twee toestanden: uit en aan. De bouwstenen van getallen in de computer zijn daarom een 0 (uit) en een 1 (aan); we noemen deze bouwstenen ook wel bits (binary digits). Een **binair getal** is een reeks van cijfers, waarbij elk cijfer gekozen kan worden uit de cijfers 0 en 1. Zo is het getal 00110100 dus een binair getal dat bestaat uit 8 bits. Welke decimale waarde hoort bij dit getal?

Net zoals in decimale getallen elk cijfer een veelvoud is van een macht van 10 is in binaire getallen elk cijfer een veelvoud van een macht van 2. De waarde van het binaire getal **00110100** is dus te berekenen en wel als volgt:

$$\begin{aligned} 0 &= 0 \times 2^7 = 0 \times 128 \\ 0 &= 0 \times 2^6 = 0 \times 64 \\ 1 &= 1 \times 2^5 = 1 \times 32 \\ 1 &= 1 \times 2^4 = 1 \times 16 \\ 0 &= 0 \times 2^3 = 0 \times 8 \\ 1 &= 1 \times 2^2 = 1 \times 4 \\ 0 &= 0 \times 2^1 = 0 \times 2 \\ 0 &= 0 \times 2^0 = 0 \times 1 \end{aligned}$$

Het binaire getal 00110100 heeft dus de waarde  $4 + 16 + 32 = 52$ .

Een eenvoudige manier om de waarde van een binair getal te berekenen is om de bijbehorende machten van 2 onder de bits te zetten:

<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>
128	64	32	16	8	4	2	1

Het grootste getal dat je met acht bits dus kunt maken is dus het getal:

$$11111111 = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255$$

#### Rekenen met binaire getallen

Nu je kennis hebt gemaakt met binaire getallen wordt het tijd om ook binair te leren rekenen. We tellen de getallen 00110100 (52) en 01110111 (119) op. Het resultaat is 10101011 (171):

$$\begin{array}{r} 00110100 \\ 01110111 \\ \hline 10101011 \end{array} +$$

↓

$1 + 1 = 0$  omdat  $1$  (keer 8) +  $1$  (keer 8) samen  $0$  (keer 8) en  $1$  (keer 16) is

De regel “één onthouden” (als een optelling van twee cijfers de 2 overschrijdt) kan je dus ook hier toepassen !

## OPDRACHTEN

### Opdracht 3.1

Maak van de volgende decimale getallen een binair getal:

- a) 23
- b) 57
- c) 69
- d) 199

### Opdracht 3.2

Maak van de volgende binaire getallen een decimaal getal:

- a) 01010101
- b) 00100100
- c) 11110111
- d) 00001000

### Opdracht 3.3

Tel de volgende binaire getallen binair bij elkaar op.

- a) 11001100 en 00110011
- b) 00101001 en 01110101

### Opdracht 3.4

Welk probleem treedt er op bij de volgende binaire optelling ?

$$11010011 + 11101110$$

### Opdracht 3.5

Wat is het grootste getal dat je kan maken met 16 bits ?

### Opdracht 3.6

Vermenigvuldig 00110100 (52) met 2. Welk binair getal is de uitkomst ?

Wat valt je op?

### Opdracht 3.7

Bereken:

$$\begin{array}{r} 01110110 \\ 00110101 \\ \hline \dots\dots\dots \end{array} -$$

Met welke regel vergelijkbaar met het rekenen met decimale getallen krijg je hier te maken ?

### 3.4 Hexadecimale getallen

#### De waarde van hexadecimale getallen

Een getalstelsel dat wij bij het computergebruik ook tegenkomen is het hexadecimale talstelsel. Hierbij worden niet de bouwstenen “0 en 1” (binair) of “0 t/m 9” (decimaal) gebruikt maar de bouwstenen “0 t/m 9 en letters A t/m F”.

<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Een **hexadecimaal getal** is een reeks van cijfers, waarbij elk cijfer gekozen kan worden uit de cijfers 0 t/m 9 en letters A t/m F. Zo is het getal 2AF3 dus een hexadecimaal getal dat bestaat uit 4 cijfers. Welke decimale waarde hoort bij dit getal?

Net zoals in decimale getallen elk cijfer een veelvoud is van een macht van 10 is en in binaire getallen elk cijfer een veelvoud is van een macht van 2 is in hexadecimale getallen elk cijfer een veelvoud van een macht van 16. De waarde van het hexadecimale getal **2AF3** is dus te berekenen en wel als volgt:

$$\begin{array}{rclcl}
 \mathbf{2} & = & 2 \times 16^3 & = & 2 \times 4096 & = & 8192 \\
 \mathbf{A} & = & 10 \times 16^2 & = & 10 \times 256 & = & 2560 \\
 \mathbf{F} & = & 15 \times 16^1 & = & 15 \times 16 & = & 240 \\
 \mathbf{3} & = & 3 \times 16^0 & = & 3 \times 1 & = & 3
 \end{array}$$

Het hexadecimale getal 2AF3 heeft dus de waarde  $3 + 240 + 2560 + 8192 = 10995$

Een eenvoudige manier om de waarde van een hexadecimaal getal te berekenen is om de bijbehorende machten van 16 onder de bits te zetten:

<b>2</b>	<b>A</b>	<b>F</b>	<b>3</b>
4096	256	16	1

$$\begin{aligned}
 & 3 \times 1 + F \times 16 + A \times 256 + 2 \times 4096 = \\
 & 3 \times 1 + 15 \times 16 + 10 \times 256 + 2 \times 4096 = 10995
 \end{aligned}$$

#### Hexadecimale geheugenadressen

Waar kom je deze getallen tegen? De computer geeft bij elk aangesloten apparaat aan via welke geheugenplaatsen van het intern geheugen hij het apparaat aan kan spreken (invoerbereik en uitvoerbereik). Zo kan je bijvoorbeeld opzoeken met start / instellingen / configuratiescherm / systeem / computer / invoer/uitvoer in hexadecimale notatie vinden via welke geheugenplaatsen de processor gegevens van met het toetsenbord uitwisselt.

## OPDRACHTEN

**Opdracht 3.8**

Maak van het volgende hexadecimale getal een decimaal getal: C2B3

**Opdracht 3.9**

Maak van het volgende decimale getal een hexadecimaal getal: 7824

**Opdracht 3.10**

Wat is het grootste hexadecimale getal van 8 cijfers ?

### 3.5 Samenvatting

De elektronische schakelaars en belijning in de computer kunnen slechts werken met 0-en en 1-en. We noemen informatie die bestaat uit 0-en en 1-en ook wel **digitale informatie**.

De **bit** is de kleinst mogelijke drager van digitale informatie en heeft de waarde 0 of 1.

We gebruiken de byte (dat zijn 8 bits) als informatie eenheid. We geven hoeveelheden digitale informatie aan in **kilobytes**, **megabytes** en **gigabytes**.

We zijn gewend om te rekenen met **decimale getallen** (10-tallig getallenstelsel). Computers rekenen met **binaire getallen** (2-tallig getallenstelsel) of **hexadecimale getallen** (16-tallig getallenstelsel)

Met het volgende **omrekenchema** kunnen we van decimale getallen binaire getallen maken en andersom:

Welke decimale waarde heeft het binaire getal 10110101 ?

<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
128	64	32	16	8	4	2	1

$$1 \times 128 + 0 \times 64 + 1 \times 32 + 1 \times 16 + 0 \times 8 + 1 \times 4 + 1 \times 1 = 181$$

Welk binair getal hoort er bij de decimale getal 102 ?

<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>
128	64	32	16	8	4	2	1

$$102 = 64 + 32 + 4 + 2, \text{ dus het binaire getal} = 01100110$$

Met het volgende omrekenchema kunnen we van decimale getallen hexadecimale getallen maken en andersom:

Welke decimale waarde heeft het hexadecimale getal 2AF3 ?

<b>2</b>	<b>A</b>	<b>F</b>	<b>3</b>
4096	256	16	1

$$2 \times 4096 + A \times 256 + F \times 16 + 3 \times 1 = \\ 2 \times 4096 + 10 \times 256 + 15 \times 16 + 3 \times 1 = 10995$$

Welk hexadecimaal getal hoort er bij het decimale getal 8157 ?

<b>1</b>	<b>F</b>	<b>D</b>	<b>D</b>
4096	256	16	1

$$8157 = 1 \times 4096 + 15 \times 256 + 13 \times 16 + 13 \times 1 = \\ 1 \times 4096 + F \times 256 + D \times 16 + D \times 1 = 1FDD$$

**3.6 ANTWOORDEN****Opdracht 3.1**

Maak van de volgende decimale getallen een binair getal:

- a) 23 = 00010111  
 b) 57 = 00111001  
 c) 69 = 01000101  
 d) 199 = 11000111

**Opdracht 3.2**

Maak van de volgende binaire getallen een decimaal getal:

- a) 01010101 = 85  
 b) 00100100 = 36  
 c) 11110111 = 247  
 d) 00001000 = 8

**Opdracht 3.3**

Tel de volgende binaire getallen bij elkaar op:

- a)  $\begin{array}{r} 11001100 \\ 204 \end{array} + \begin{array}{r} 00110011 \\ 51 \end{array} = \begin{array}{r} 11111111 \\ = 255 \end{array}$   
 b)  $\begin{array}{r} 00101001 \\ 41 \end{array} + \begin{array}{r} 01110101 \\ 117 \end{array} = \begin{array}{r} 10011110 \\ = 158 \end{array}$

**Opdracht 3.4**

Welk probleem treedt er op bij de volgende binaire optelling?

$$\begin{array}{r} 11010011 \quad (211) \\ 11101110 \quad (238) \\ \hline 111000001 \quad (449) \end{array} +$$

Er is één symbool extra nodig! We noemen dit verschijnsel “**overflow**”.

**Opdracht 3.5**

Wat is het grootste getal dat je kan maken met 8 bits?

$$128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255 = 2^8 - 1$$

Om de laatste stap te begrijpen moet je inzien dat 255 precies één minder is dan de macht van 2 die volgt na 128, dat is 256.

En wat is het grootste getal dat je kan maken met 16 bits?

$$2^{16} - 1 = 65535$$

**ANTWOORDEN****Opdracht 3.6**

Vermenigvuldig 00110100 (52) met 2. Welk binair getal is de uitkomst?

Wat valt je op?

00110100 "x 2" ofwel  $52 \times 2 = 104$  wordt 01101000,  
de enen schuiven een plek op!

**Opdracht 3.7**

Bereken:

$$\begin{array}{r} 01110110 \quad (118) \\ 00110101 \quad (53) \\ \hline 01000001 \quad (65) \end{array} -$$

Met welke regel vergelijkbaar met het rekenen met decimale getallen krijg je hier te maken?

De regel die je ook hier kan toepassen is dat je bij het aftrekken van een 1 van een 0 een 1 mag "lenen" van de volgende positie.

**Opdracht 3.8**

Maak van het volgende hexadecimale getal een decimaal getal: C2B3

<b>C</b>	<b>2</b>	<b>B</b>	<b>3</b>
4096	256	16	1

$$\begin{aligned} C \times 4096 + 2 \times 256 + B \times 16 + 3 \times 1 &= \\ 12 \times 4096 + 2 \times 256 + 11 \times 16 + 3 \times 1 &= 49843 \end{aligned}$$

**Opdracht 3.9**

Maak van het volgende decimale getal een hexadecimaal getal: 7824

<b>1</b>	<b>E</b>	<b>9</b>	<b>0</b>
4096	256	16	1

$$\begin{aligned} 7824 &= 1 \times 4096 + 14 \times 256 + 9 \times 16 + 0 \times 1 = \\ 1 \times 4096 + E \times 256 + 9 \times 16 + 0 \times 1 &= 1E90 \end{aligned}$$

**Opdracht 3.10**

Wat is het grootste hexadecimale getal van 8 cijfers?

$$16^8 - 1 = 16777216 - 1 = 16777215$$