Opdracht 4.1 Binair optellen

a. $25_{dec} = 1 \times 16 + 1 \times 8 + 0 \times 4 + 0 \times 2 + 1 \times 1$
$25_{dec} = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
$25_{dec} = 11001_{bin}$

$11_{dec} = 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1$
$11_{dec} = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
$11_{dec} = 1011_{bin}$

b.
```
  11001
  01011 +
 ------
  12012
  20020
 100100
```

c. $100100_{bin} = 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$
$100100_{bin} = 32 + 0 + 0\ 4 + 0 + 0$
$100100_{bin} = 36_{dec}$
En 25 + 11 = 36 in het decimale stelsel.

Opdracht 4.2 Binair optellen

a.
```
00100101
10011110 +
--------
10111211
10112011
10120011
10200011
11000011
```

$00100101_{bin} = 0 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
$00100101_{bin} = 0 + 0 + 32 + 0 + 0 + 4 + 0 + 1$
$00100101_{bin} = 37_{dec}$

$10011110_{bin} = 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$
$00100101_{bin} = 128 + 0 + 0 + 16 + 8 + 4 + 2 + 0$
$00100101_{bin} = 158_{dec}$

$11000011_{bin} = 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
$00100101_{bin} = 128 + 64 + 0 + 0 + 0 + 0 + 2 + 1$
$00100101_{bin} = 195_{dec}$

158 + 37 = 195 decimaal.

b. $93 = 64 + 16 + 8 + 4 + 1$
$93_{dec} = 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times$

2^0
$93_{dec} = 1011101$

$109 = 64 + 32 + 8 + 4 + 1$
$93_{dec} = 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
$93_{dec} = 1101101$

```
 1011101
 1101101 +
 2112202
10113010
10121010
10201010
11001010
```
11001010

$11001010_{bin} = 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$
$11001010_{bin} = 128 + 64 + 0 + 0 + 8 + 0 + 2 + 0$
$11001010_{bin} = 202_{dec}$

93 + 109 = 202 decimaal.

c.
```
 01110011
 11000111 +
 12110122
 20110130
100110210
```
100111010

Het zijn negen bits. Dat past niet in eem byte.

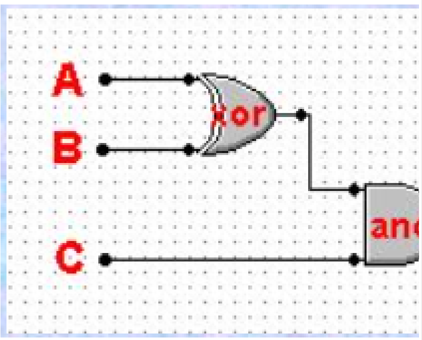
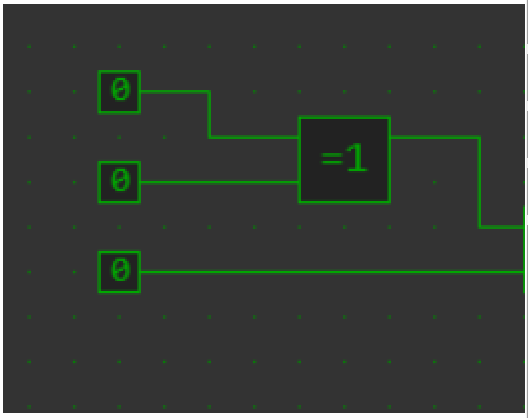
Opdracht 4.3 Binair optellen met logica

a. bit1 = 1 en bit2 = 1
b. bit1 = 1 of bit2 = 1

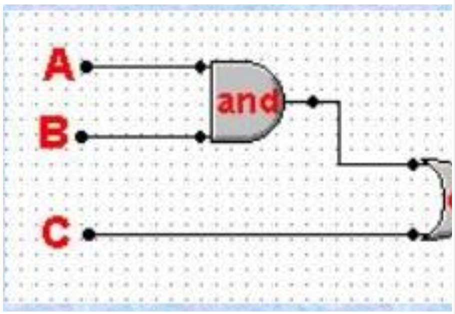
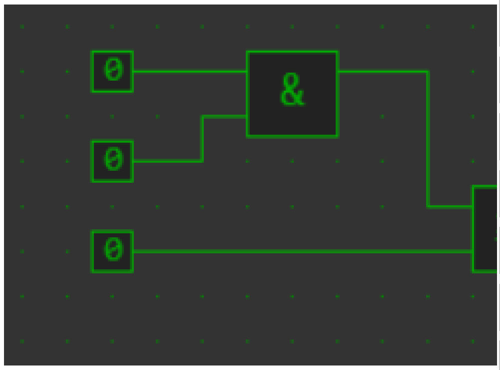
Opdracht 4.4 Logische schakelingen

a.

MultiMedia Logic	A	B	C	Uit	Simulator.io
	1	1	1	0	
	1	1	0	0	

	1	0	1	1	
	1	0	0	0	
	0	1	1	1	
	0	1	0	0	
	0	0	1	0	
	0	0	0	0	

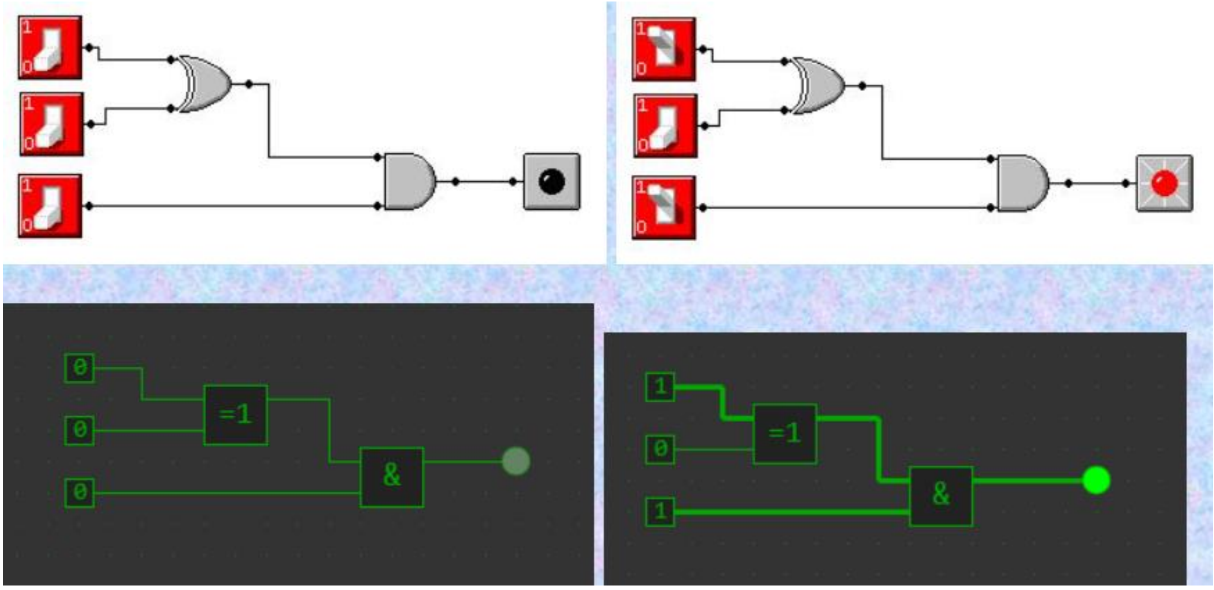
b.

MultiMedia Logic	A	B	C	Uit	Simulator.io
	1	1	1	1	
	1	1	0	1	
	1	0	1	1	
	1	0	0	0	
	0	1	1	1	
	0	1	0	0	
	0	0	1	1	
	0	0	0	0	

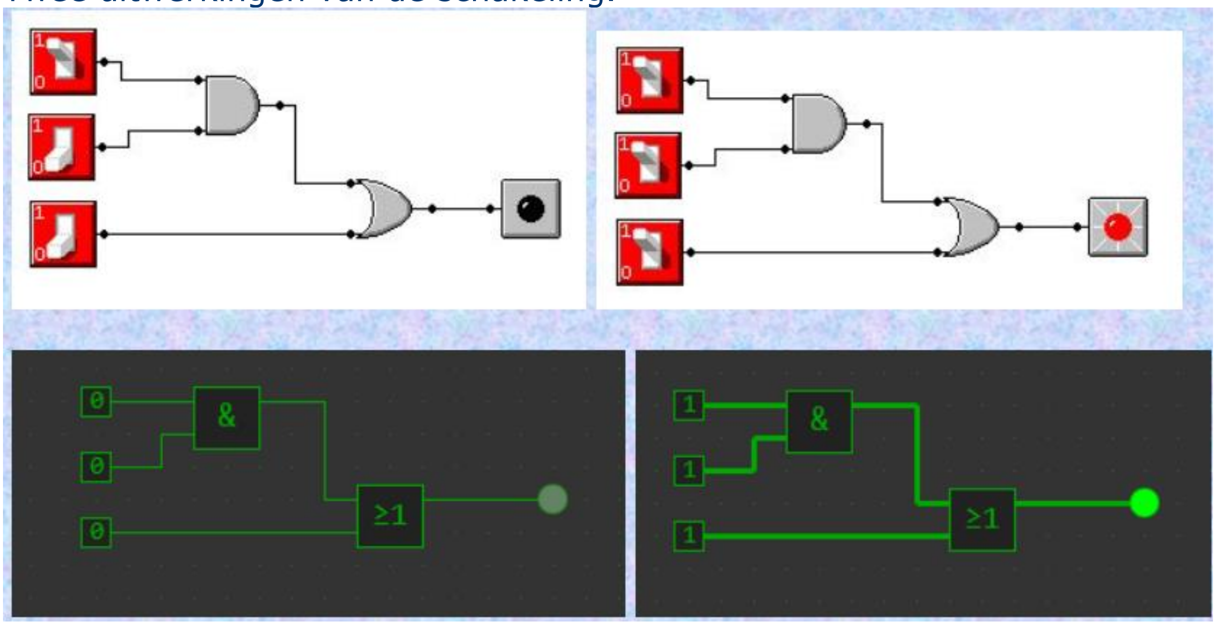
Opdracht 4.5 Een schakeling in simulator.io

Deze schakelingen zijn gebouwd met Multi Media Logic.

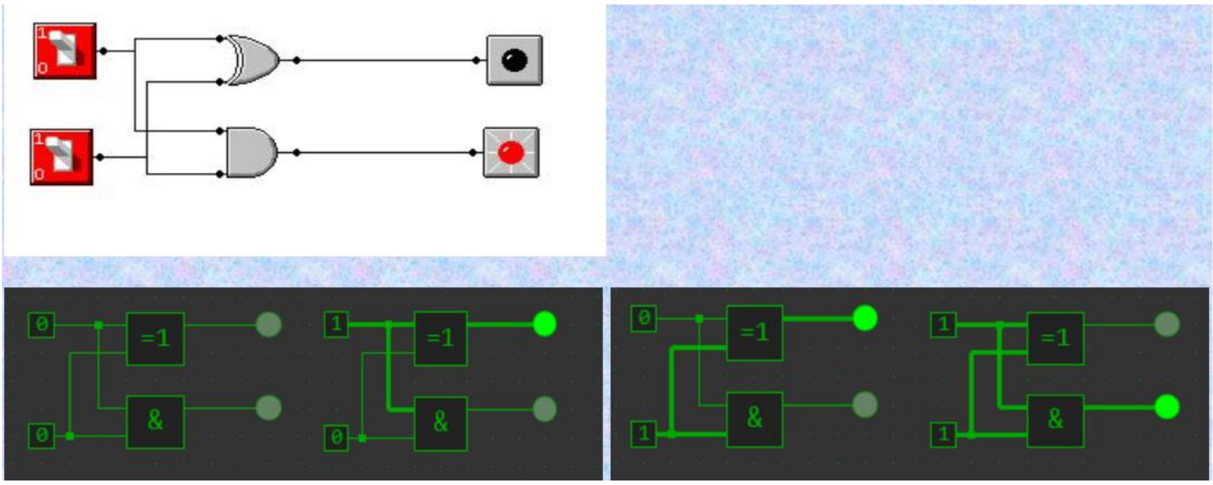
a. Twee uitwerkingen van de schakeling.



b. Twee uitwerkingen van de schakeling.



Opdracht 4.6 Een opteller van twee bits



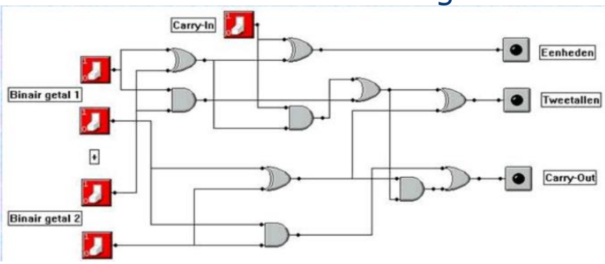
Als een van de twee schakelaars aan staat (een bit staat op 1), dan zal het bovenste lampje branden. Zijn beide schakelaars aan, dan brandt het onderste lampje.
Je kunt dat als volgt weergeven.

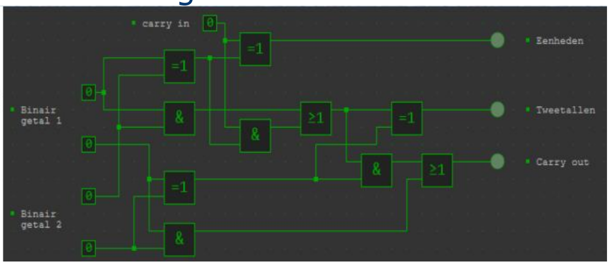
getal 1	getal 2	uitkomst binair	uitkomst decimaal
0	0	00	0
0	1	01	1

1	0	01	1
1	1	10	2

Extra.
Grotere binaire getallen optellen

De onderstaande schakeling telt twee twee-bits getallen



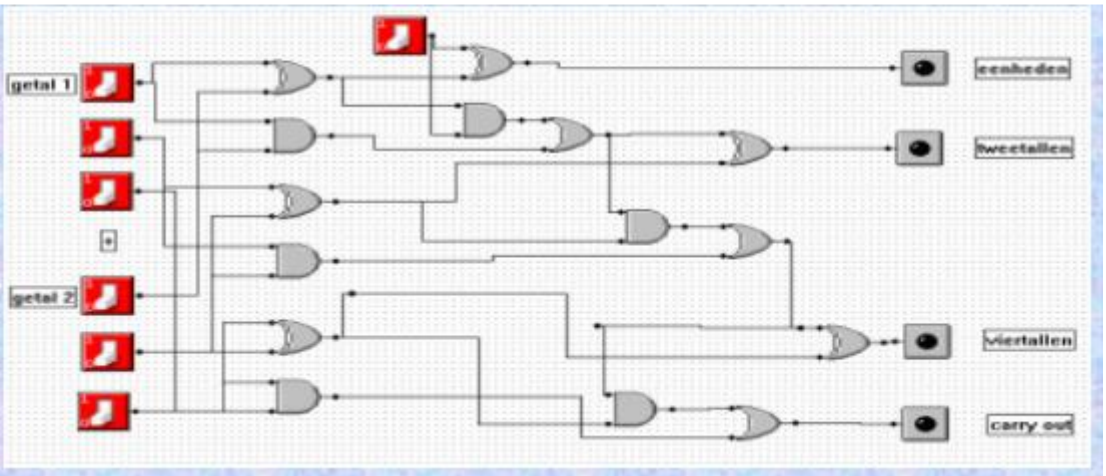


getal 1 binair	getal 2 binair	uitkomst binair	getal 1 decimaal	getal 2 decimaal	uitkomst decimaal
00	00	000	0	0	0
00	01	001	0	1	1
00	10	010	0	2	2
00	11	011	0	3	3
01	00	001	1	0	1
01	01	010	1	1	2
01	10	011	1	2	3
01	11	100	1	3	4
10	00	010	2	0	2
10	01	011	2	1	3
10	10	100	2	2	4

10	11	101	2	3	5
11	00	011	3	0	3
11	01	100	3	1	4
11	10	101	3	2	5
11	11	110	3	3	6

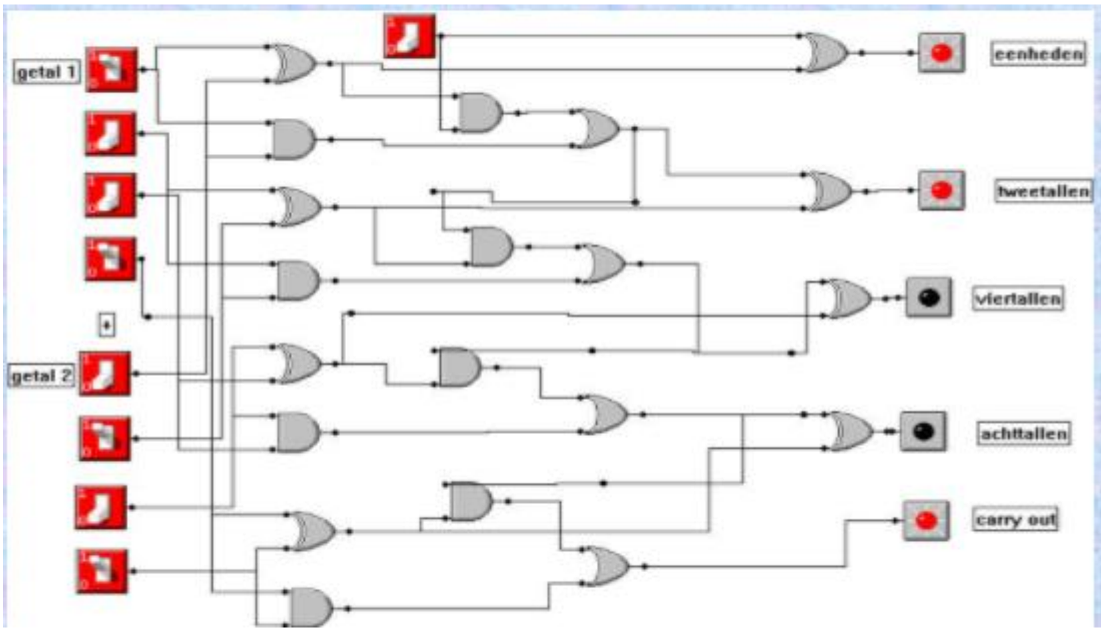
Opdracht 4.7 De carry van de full adder

De onderstaande schakeling telt twee drie-bits getallen op.



Opdracht 4.8 Opteller van vier bits

En de schakeling die twee vier-bits getallen optelt.



Opdracht 4.9 Binair optellen oefenen

a. 0101 1010 + 1011 1010 = 1 0001 0100
   Er is overflow.
b. $53_{dec} = 0011\ 0101_{bin}$
   $38_{dec} = 0010\ 0110_{bin}$
   $91_{dec} = 0101\ 1011_{bin}$

Opdracht 4.10 Binair vermenigvuldigen

a. $101_{bin} = 5_{dec}$
b. $1010_{bin} = 10_{dec}$
c. $10100_{bin} = 20_{dec}$
d. $101000_{bin} = 40_{dec}$
e. Een extra nul geeft een verdubbeling.
f. Een Arithmetic Shift Left verdubbelt een binair getal.
g. $27_{dec} = 1\ 1011_{bin}$
   $(4 \times 27)_{dec} = 110\ 1100_{bin}$
   $(8 \times 27)_{dec} = 1101\ 1000_{bin}$
   Tel nu 1 1011, 110 1100 en 1101 1000 op.

Opdracht 4.11*** Negatieve getallen in binaire notatie

a. 00000001 + 10000001 = 10000010
b. $10000010_{bin} = -2_{dec}$
c.
d. 00001001 + 11110111 = 00000000
e.

De uitkomst is 0.

```
f. 00000100 + xxxxxxxx = 00000000
g.   xxxxxxxx = 111111100
h.
```

Je moet elk bit omdraaien (een 0 wordt een 1 en een 1 een 0) en er 1 bij optellen.

```
i. 01100101 + 10011011 = 00000000
```

Zie ook Floating point conversion