

FACULTEIT DER NATUURWETENSCHAPPEN, WISKUNDE EN INFORMATICA

Wouter Geraedts

Processen & Processoren

Radboud Universiteit Nijmegen





Overzicht

Welkom op het 2^e werkcollege van Processen & Processoren!

- Overzicht van resultaten
- Opmerkingen over inleveren
- Uitwerkingen vorige opgavenserie
- Behandelen oefenopgaven



Resultaten

Todo: statistieken



Opmerkingen over inleveren

De volgende keer kijk ik de volgende uitwerkingen niet meer na:

- Uitwerkingen niet in PDF of plain text
- Uitwerkingen die te laat ingeleverd zijn
- E-mails met de verkeerde subject-header

Omdat dit de eerste opgavenserie is vergeef ik 't



Uitwerkingen vorige opgavenserie

Hoe gaan we dit doen?

- Ik laat antwoord zien
- Als je iets niet snapt stel je een vraag
- We gaan er snel door heen
- Later komen deze slides ook online te staan



Opgave 1

“Converteer de volgende getallen naar decimaal, binair, octaal en hexadecimaal”

- 37_{dec}
- 256_{dec}
- 1201_{dec}
- $101110101100_{\text{bin}}$
- $95A_{\text{hex}}$
- 712_{oct}



Opgave 1 ₂

Origineel	Decimaal	Binair	Octaal	Hexadecimaal
37_{dec}	37_{dec}	100101_{bin}	45_{oct}	25_{hex}
256_{dec}	256_{dec}	100000000_{bin}	400_{oct}	100_{hex}
1201_{dec}	1201_{dec}	10010110001_{bin}	2261_{oct}	$4B1_{hex}$
101110101100_{bin}	2988_{dec}	101110101100_{bin}	5654_{oct}	BAC_{hex}
$95A_{hex}$	2394_{dec}	100101011010_{bin}	4532_{oct}	$95A_{hex}$
712_{oct}	458_{dec}	11001010_{bin}	712_{oct}	$1CA_{hex}$



Opgave 2

Beargumenteer waarom de multiplicatie van restklassen onafhankelijk is van de representant.



Opgave 2

Beargumenteer waarom de multiplicatie van restklassen onafhankelijk is van de representant.

Omdat de factoren n behouden blijven na multiplicatie, onafhankelijk van de kleinste representanten.



Opgave 2

Beargumenteer waarom de multiplicatie van restklassen onafhankelijk is van de representant.

Omdat de factoren n behouden blijven na multiplicatie, onafhankelijk van de kleinste representanten.

$$[k + an]_n \cdot [l + bn]_n = [k \cdot l + k \cdot bn + l \cdot an + an \cdot bn]_n = [k \cdot l]_n$$



Opgave 3

Converteer naar binaire getallen en bereken:

- $11_{\text{dec}} + 43_{\text{dec}}$
- $74_{\text{dec}} - 12_{\text{dec}}$
- $(-25_{\text{dec}}) + 90_{\text{dec}}$



Opgave 3 ₂

$$11_{\text{dec}} + 43_{\text{dec}}$$



Opgave 3 ₂

$$11_{\text{dec}} + 43_{\text{dec}} = 1011_{\text{bin}} + 101011_{\text{bin}}$$



Opgave 3 ₂

$$11_{\text{dec}} + 43_{\text{dec}} = 1011_{\text{bin}} + 101011_{\text{bin}}$$

$$\begin{array}{r} 001011 \\ 101011 + \\ \hline 110110 \end{array}$$



Opgave 3 ₃

7₄dec – 12_{dec}



Opgave 3 ₃

$$74_{\text{dec}} - 12_{\text{dec}} = 1001010_{\text{bin}} - 1100_{\text{bin}}$$



Opgave 3 ₃

$$74_{\text{dec}} - 12_{\text{dec}} = 1001010_{\text{bin}} - 1100_{\text{bin}}$$

Maar... twee-complement!

$$01001010_{\text{bin}} - 00001100_{\text{bin}}$$



Opgave 3 ₃

$$74_{\text{dec}} - 12_{\text{dec}} = 1001010_{\text{bin}} - 1100_{\text{bin}}$$

Maar... twee-complement!

$$01001010_{\text{bin}} - 00001100_{\text{bin}}$$

01001010

00001100 -

00111110



Opgave 3 ₄

$$(-25_{\text{dec}}) + 90_{\text{dec}}$$



Opgave 3 ₄

$$(-25_{\text{dec}}) + 90_{\text{dec}} = (-11001_{\text{bin}}) + 1011010_{\text{bin}}$$



Opgave 3 ₄

$$(-25_{\text{dec}}) + 90_{\text{dec}} = (-11001_{\text{bin}}) + 1011010_{\text{bin}}$$

Maar... twee-complement!

$$11100111_{\text{bin}} + 01011010_{\text{bin}}$$

11100111

01011010 +

01000001



Opgave 4a

Bij binaire getallen in een-complement kan het gebeuren dat door een additie van twee erg grote of erg kleine getallen een som ontstaat die niet meer in die bits past.

Vraag: Hoe kan men dit geval herkennen?



Opgave 4a

Bij binaire getallen in een-complement kan het gebeuren dat door een additie van twee erg grote of erg kleine getallen een som ontstaat die niet meer in die bits past.

Vraag: Hoe kan men dit geval herkennen?

Antwoord:

- 2 grote getallen \rightarrow negatief
- 2 kleine getallen \rightarrow positief

(net als bij twee-complement)



Opgave 4b

De som van twee getallen in een-complement berekent men niet op dezelfde manier als bij getallen in twee-complement of met restklassen.

Leg uit hoe twee getallen in een-complement opgeteld kunnen worden.



Opgave 4b

De som van twee getallen in een-complement berekent men niet op dezelfde manier als bij getallen in twee-complement of met restklassen.

Leg uit hoe twee getallen in een-complement opgeteld kunnen worden.

Antwoord: tel de carry na optelling bij het resultaat erbij.



Opgave 4c

Converteer $10_{\text{dec}} - (-31_{\text{dec}})$ naar binaire getallen in een-complement en bereken.



Opgave 4c

Converteer $10_{\text{dec}} - (-31_{\text{dec}})$ naar binaire getallen in een-complement en bereken.

$$10_{\text{dec}} - (-31_{\text{dec}}) = 1010_{\text{bin}} - (-11111)_{\text{bin}}$$



Opgave 4c

Converteer $10_{\text{dec}} - (-31_{\text{dec}})$ naar binaire getallen in een-complement en bereken.

$$10_{\text{dec}} - (-31_{\text{dec}}) = 1010_{\text{bin}} - (-11111)_{\text{bin}}$$

Maar... een-complement!

$$0001010_{\text{bin}} - 1100000_{\text{bin}}$$



Opgave 4c

Converteer $10_{\text{dec}} - (-31_{\text{dec}})$ naar binaire getallen in een-complement en bereken.

$$10_{\text{dec}} - (-31_{\text{dec}}) = 1010_{\text{bin}} - (-11111)_{\text{bin}}$$

Maar... een-complement!

$$0001010_{\text{bin}} - 1100000_{\text{bin}}$$

$$\begin{array}{r} 0001010 \\ 1100000 - \\ \hline 0101001 \end{array}$$



Opgave 5a

Geef voorbeelden van niet-eenduidige “divisie” als je strict in restklassen modulo 256 rekent.



Opgave 5a

Geef voorbeelden van niet-eenduidige “divisie” als je strict in restklassen modulo 256 rekent.

$$\begin{aligned} [0] \cdot [128] &= [0] \quad \Rightarrow \quad \frac{[0]}{[128]} = [0] \\ [2] \cdot [128] &= [256] = [0] \quad \Rightarrow \quad \frac{[0]}{[128]} = [2] \end{aligned}$$



Opgave 5a

Geef voorbeelden van niet-eenduidige “divisie” als je strict in restklassen modulo 256 rekent.

$$[0] \cdot [128] = [0] \Rightarrow \frac{[0]}{[128]} = [0]$$

$$[2] \cdot [128] = [256] = [0] \Rightarrow \frac{[0]}{[128]} = [2]$$

$$[5] \cdot [2] = [10] \Rightarrow \frac{[10]}{[2]} = [5]$$

$$[133] \cdot [2] = [266] = [10] \Rightarrow \frac{[10]}{[2]} = [133]$$



Opgave 5b

Geef ook voorbeelden waarbij de “quotient” niet bestaat.



Opgave 5b

Geef ook voorbeelden waarbij de “quotient” niet bestaat.

Oftewel: er is geen restloze deling mogelijk.



Opgave 5b

Geef ook voorbeelden waarbij de “quotient” niet bestaat.

Oftewel: er is geen restloze deling mogelijk.

Let op: delen door [0] mag vaak!



Opgave 5b

Geef ook voorbeelden waarbij de “quotient” niet bestaat.

Oftewel: er is geen restloze deling mogelijk.

Let op: delen door [0] mag vaak!

- $\frac{[x]}{[2]}$ waarbij $x \in 1, 3, 5, 7, \dots$
- Vergelijkbare oplossingen



Opgave 5c

Onder welke voorwaarden kan één van die twee situaties optreden?



Opgave 5c

Onder welke voorwaarden kan één van die twee situaties optreden?

Bij $\frac{[a]_n}{[b]_n} = [c]_n$ of $\frac{a}{b} \equiv c \pmod{n}$:



Opgave 5c

Onder welke voorwaarden kan één van die twee situaties optreden?

Bij $\frac{[a]_n}{[b]_n} = [c]_n$ of $\frac{a}{b} \equiv c \pmod{n}$:

Enkel resultaat: $\gcd([b], n) = 1$

Ambigu resultaat: $\frac{[a]}{\gcd([b], n)}$ bestaat

Geen resultaat: anders



Opgave 5c

Onder welke voorwaarden kan één van die twee situaties optreden?

Bij $\frac{[a]_n}{[b]_n} = [c]_n$ of $\frac{a}{b} \equiv c \pmod{n}$:

Enkel resultaat: $\gcd([b], n) = 1$

Ambigu resultaat: $\frac{[a]}{\gcd([b], n)}$ bestaat

Geen resultaat: anders

Antwoord: $\gcd([b], n) \neq 1$



Oefenopgaven

Op naar de oefenopgaven!



Oefenopgaven

Op naar de oefenopgaven!

Een paar opmerkingen:



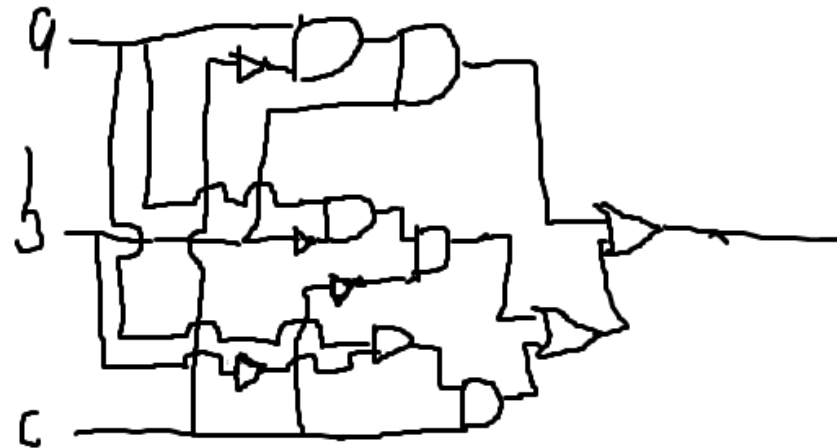
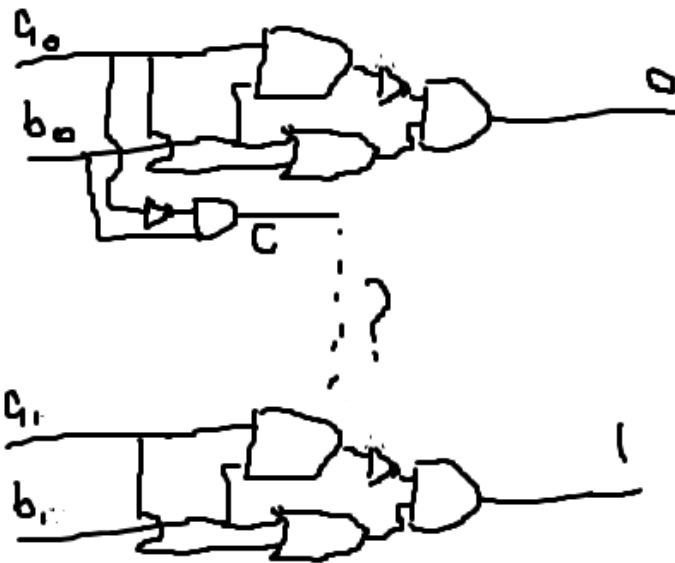
Oefenopgaven

Op naar de oefenopgaven!

Een paar opmerkingen:

- Gebruik een geschikt programma
- Zorg dat ik de circuit-symbolen kan lezen
- Maak desnoods ASCII-art

Oefenopgaven ²





Oefenopgaven ³





Opgave 1

Breng de onderstaande formules in disjunctieve normaalvorm.

$$\neg(a \wedge \neg b)$$

$$a \wedge b \wedge \neg c$$

$$(a \vee c) \wedge (\neg a \vee b)$$



Opgave 1₂

Vraag: $\neg(a \wedge \neg b)$ in DNF



Opgave 1 ₂

Vraag: $\neg(a \wedge \neg b)$ in DNF

a	b	$\neg(a \wedge \neg b)$
0	0	1
0	1	0
1	0	1
1	1	1



Opgave 1 ₂

Vraag: $\neg(a \wedge \neg b)$ in DNF

a	b	$\neg(a \wedge \neg b)$
0	0	1
0	1	0
1	0	1
1	1	1

Antwoord: $(\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge b)$



Opgave 1₃

Vraag: $a \wedge b \wedge \neg c$ in DNF



Opgave 1 ₃

Vraag: $a \wedge b \wedge \neg c$ in DNF

a	b	c	$(a \wedge b \wedge \neg c)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0



Opgave 1 ₃

Vraag: $a \wedge b \wedge \neg c$ in DNF

a	b	c	$(a \wedge b \wedge \neg c)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

Antwoord: $a \wedge b \wedge \neg c$



Opgave 1 ₄

Vraag: $(a \vee c) \wedge (\neg a \vee b)$ in DNF



Opgave 1 ₄

Vraag: $(a \vee c) \wedge (\neg a \vee b)$ in DNF

a	b	c	$(a \wedge b \wedge \neg c)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



Opgave 1 ₄

Vraag: $(a \vee c) \wedge (\neg a \vee b)$ in DNF

a	b	c	$(a \wedge b \wedge \neg c)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Antwoord:

$$(\neg a \wedge \neg b \wedge c) \vee (\neg a \wedge b \wedge c) \vee (a \wedge b \wedge \neg c) \vee (a \wedge b \wedge c)$$

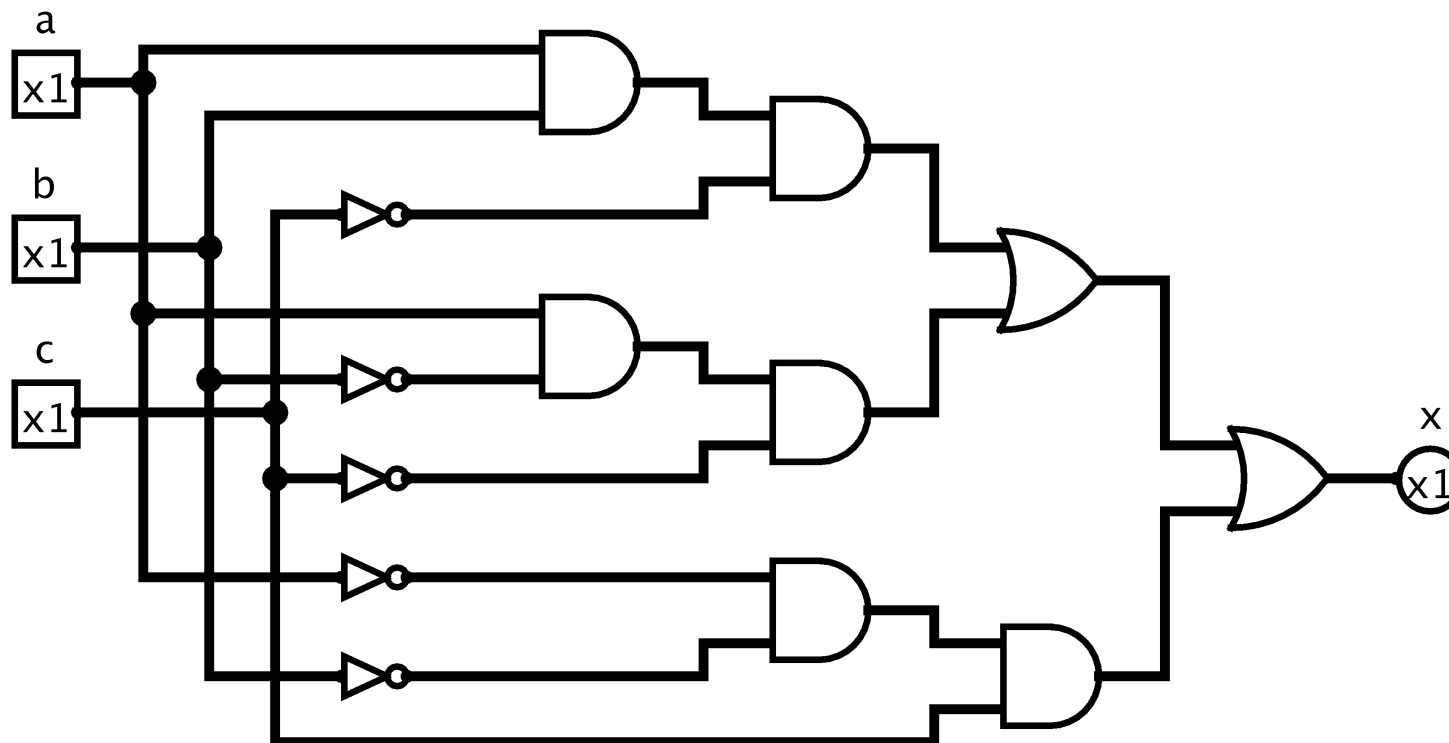


Opgave 2a

Ontwerp een circuit dat de volgende formule berekent.

$$(a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$

Opgave 2a ₂



$$(a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$



Opgave 2b

Probeer het circuit te vereenvoudigen, zodat je minder transistoren gebruikt.

(NOT = 1 transistor, NAND en NOR = 2 transistoren, AND en OR = 3 transistoren. Je mag ook gates met meer dan twee ingangen gebruiken, maar die hebben per extra ingang één extra transistor nodig, b.v. AND met 3 ingangen = 4 transistoren.)



Opgave 2b ₂

- 5 NOT-gates: $5 \times 1 = 5$ transistoren
- 6 AND-gates: $3 \times 6 = 18$ transistoren
- 2 OR-gates: $3 \times 2 = 6$ transistoren

Wat in totaal neerkomt op $5 + 18 + 6 = 29$ transistoren.



Opgave 2b ³

Als eerste moeten we de formule versimpelen:

$$(a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$



Opgave 2b ₃

Als eerste moeten we de formule versimpelen:

$$(a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$

$$(a \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$



Opgave 2b ₃

Als eerste moeten we de formule versimpelen:

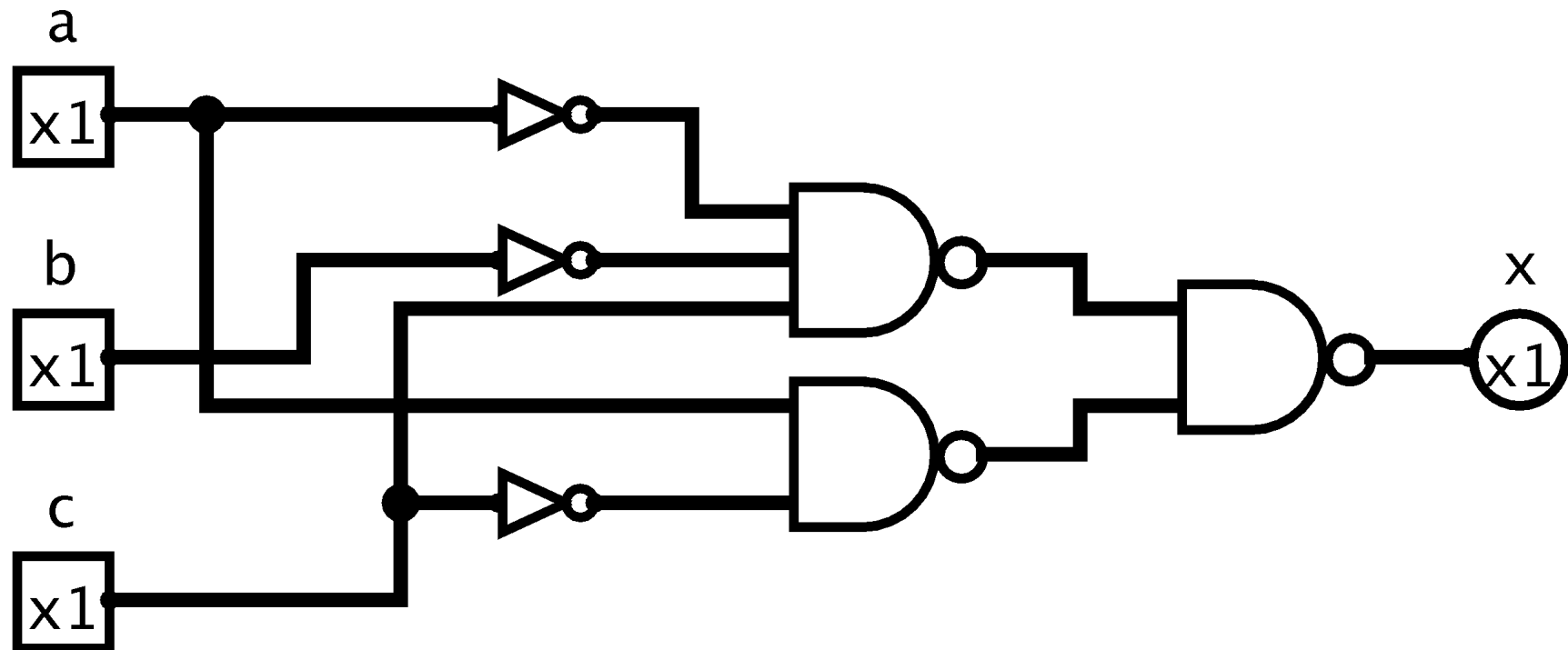
$$(a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$

$$(a \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$$

Vervolgens passen we DeMorgan toe:

$$\neg(\neg(\neg a \wedge \neg b \wedge c) \wedge \neg(a \wedge \neg c))$$

Opgave 2b ₄



$$\neg(\neg(\neg a \wedge \neg b \wedge c) \wedge \neg(a \wedge \neg c))$$



Opgave 2b ⁵

- 3 NOT-gates: $3 \times 1 = 3$ transistoren
- 2 2-NAND-gates: $2 \times 2 = 4$ transistoren
- 1 3-NAND-gate: $1 \times (2 + 1) = 3$ transistoren

In totaal zijn dit $3 + 4 + 3 = 10$ transistoren.



Opgave 3

In de cockpit van een vliegtuig zit een lampje dat aangeeft of de piloot gevaarlijk langzaam vliegt.

Om de kans op loos alarm te verkleinen, zitten er liefst drie computers die elk onafhankelijk uitrekenen of het lampje aan of uit moet gaan. Het echte lampje gaat aan als minimaal twee computers de uitvoer „aan” produceren.

Vraag: geef een circuit dat de drie computers met het lampje verbindt.



Opgave 3 ₂

Implementeer de meerderheidsfunctie.



Opgave 3 ₂

Implementeer de meerderheidsfunctie.

a	b	c	meerderheid
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Opgave 3 ₂

Implementeer de meerderheidsfunctie.

a	b	c	meerderheid
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Formule:

$$(\neg a \wedge b \wedge c) \vee (a \wedge \neg b \wedge c) \vee (a \wedge b \wedge \neg c) \vee (a \wedge b \wedge c)$$



Opgave 3 ₃

Formule versimpelen:

$$(\neg a \wedge b \wedge c) \vee (a \wedge \neg b \wedge c) \vee (a \wedge b \wedge \neg c) \vee (a \wedge b \wedge c)$$



Opgave 3 ₃

Formule versimpelen:

$$(\neg a \wedge b \wedge c) \vee (a \wedge \neg b \wedge c) \vee (a \wedge b \wedge \neg c) \vee (a \wedge b \wedge c)$$

$$(b \wedge c) \vee (a \wedge c) \vee (a \wedge b)$$



Opgave 3 ₃

Formule versimpelen:

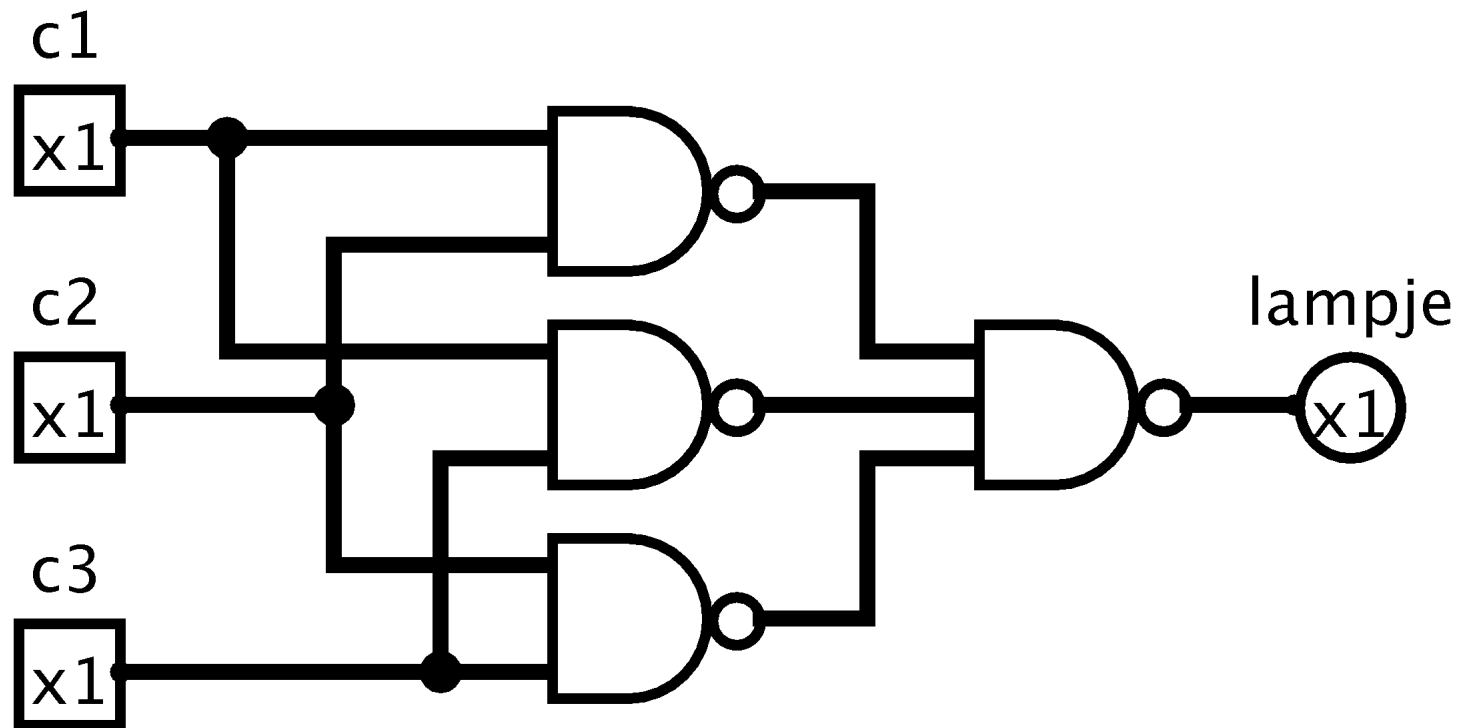
$$(\neg a \wedge b \wedge c) \vee (a \wedge \neg b \wedge c) \vee (a \wedge b \wedge \neg c) \vee (a \wedge b \wedge c)$$

$$(b \wedge c) \vee (a \wedge c) \vee (a \wedge b)$$

Vervolgens passen we DeMorgan toe:

$$\neg(\neg(b \wedge c) \wedge \neg(a \wedge c) \wedge \neg(a \wedge b))$$

Opgave 3 ₄



$$\neg(\neg(b \wedge c) \wedge \neg(a \wedge c) \wedge \neg(a \wedge b))$$



Opgave 4

Ontwerp een circuit dat twee getallen van twee bits van elkaar aftrekt, $(a_1a_0) - (b_1b_0)$.



Opgave 4 ₂

Hoe trek je twee bits van elkaar af?



Opgave 4₂

Hoe trek je twee bits van elkaar af?

a	b	R	C
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



Opgave 4 ₂

Hoe trek je twee bits van elkaar af?

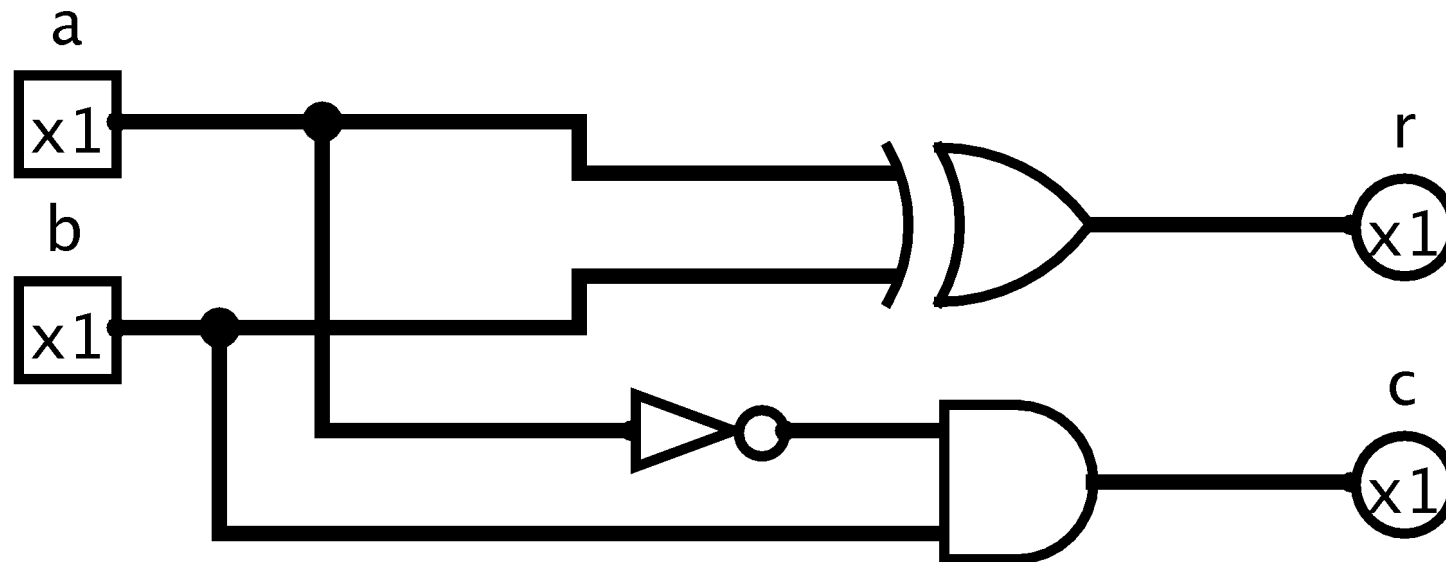
a	b	R	C
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

$R \Rightarrow \text{XOR}$

$C \Rightarrow \text{NOT} + \text{AND}$

Opgave 4₃

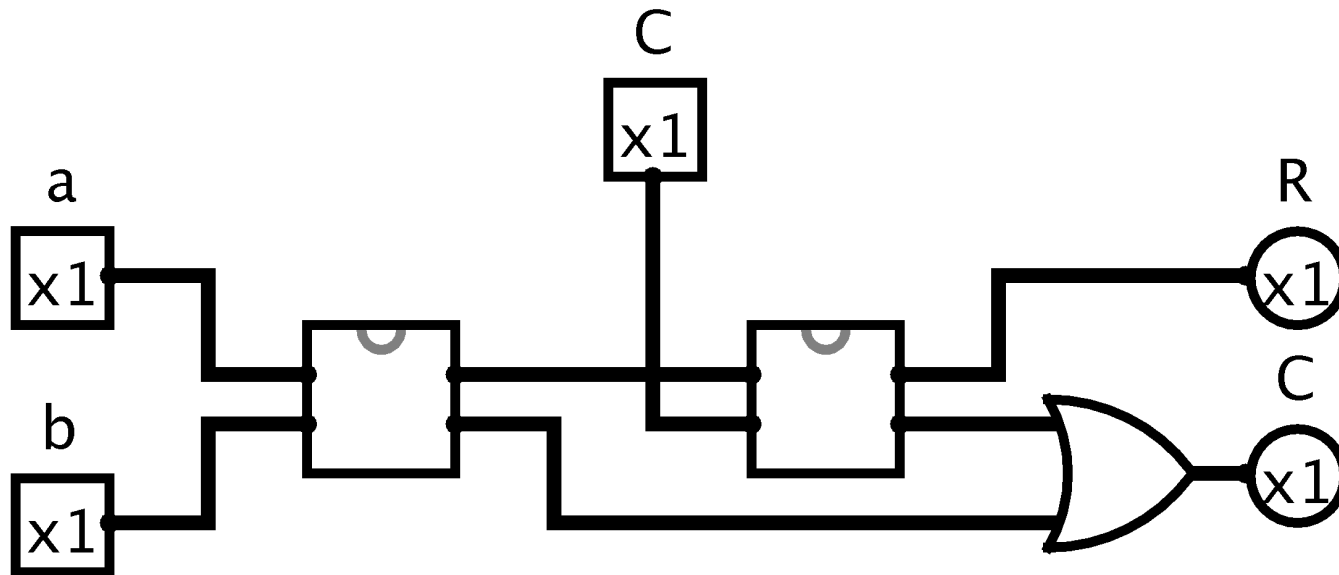
Dit noem je een half-subtractor:





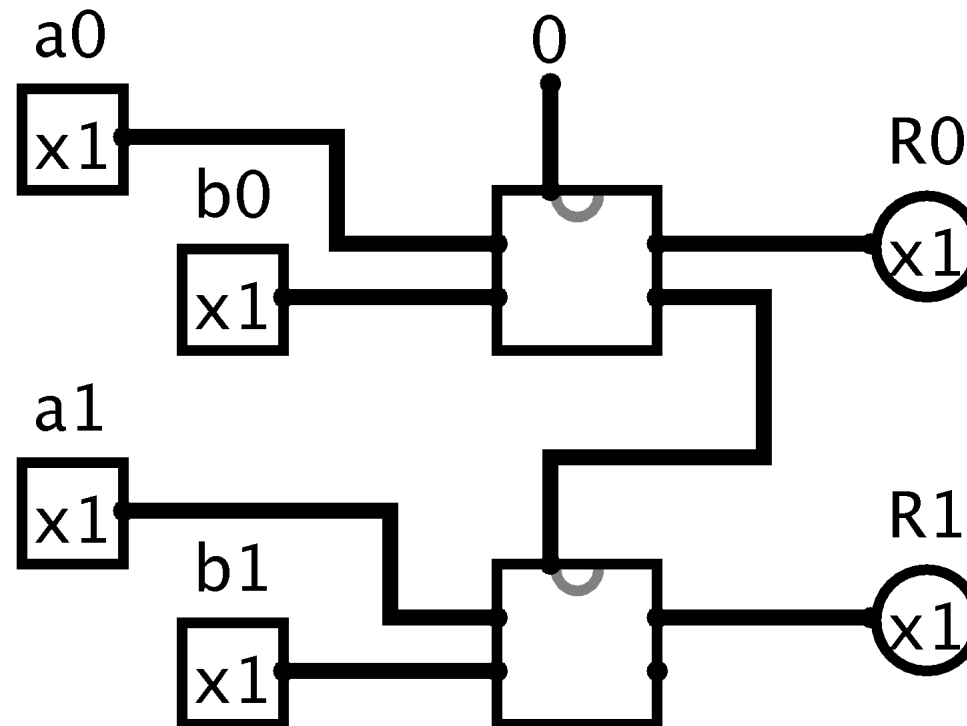
Opgave 4 ₄

En hoe gebruikt men de carry? Met een volledige subtractor!



Opgave 4 ₅

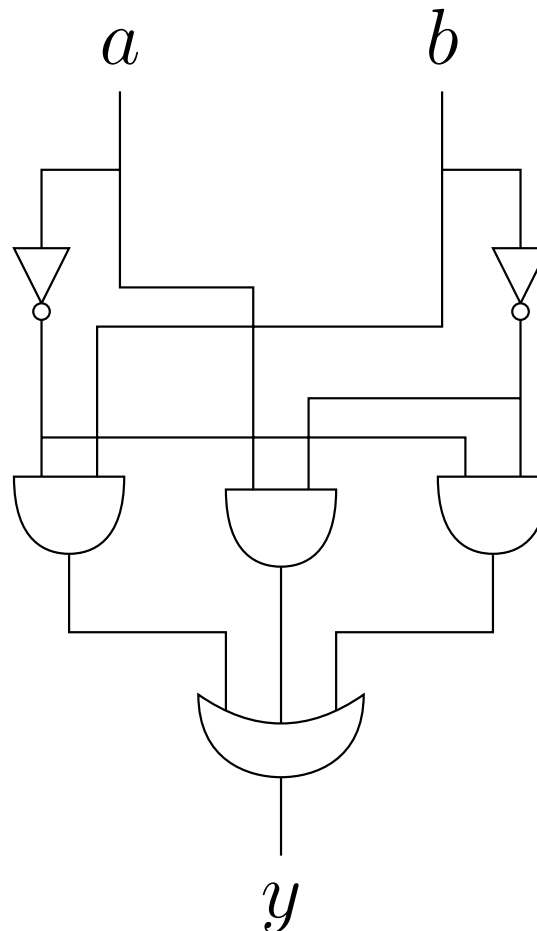
Meerdere subtractors voor 2-bits aftrekken:





Opgave 5

Welke functie berekent het volgende circuit?





Opgave 5 ₂

a	b	f
0	0	1
0	1	1
1	0	1
1	1	0



Opgave 5 ₂

a	b	f
0	0	1
0	1	1
1	0	1
1	1	0

Antwoord: $\neg(a \wedge b)$ ofwel de NAND



Einde

Fin