



Academie voor Technology, Innovation &
Society Delft
Academie voor ICT & Media

Inleiding Digitale Techniek

Week 1 – Poorten, vereenvoudigen, dualiteit, De Morgan
Jesse op den Brouw
INLDIG/2020-2021

DE HAAGSE
HOGESCHOOL

Gebeurtenissen in de digitale techniek

Jaar	Naam	Gebeurtenis/uitvinding
1623	Schickard	Eerste mechanisch apparaat voor optellen en aftrekken
1673	Leibniz	Mechanisch apparaat voor optellen, aftrekken, vermenigvuldigen en delen
1804	Jacquard	Weefgetouw met ponskaartbesturing
1833	Babbage	Difference Engine
1854	Boole	Verhandeling over logica
1890	Hollerith	Ponskaartverwerking volkstelling VS
1906	De Forest	Vacuumbuis
1918	Scherbius	Enigma codeermachine

Gebeurtenissen in de digitale techniek

Jaar	Naam	Gebeurtenis/uitvinding
1936	Turing*)	Turing-machine
1937	Shannon*)	Theoretische verhandeling digitaal ontwerpen
1939	Atanasoff e.a.	Eerste elektronische computer
1941	Zuse	Eerste elektronische computer (binair)
1946	Eckert e.a.	ENIAC
1948	Shockley e.a.	Transistor
1951	IBM	Eerste generatie computers (buizen)
1959	IBM	Tweede generatie computers (transistoren)
1960	Kilby e.a.	Eerste IC

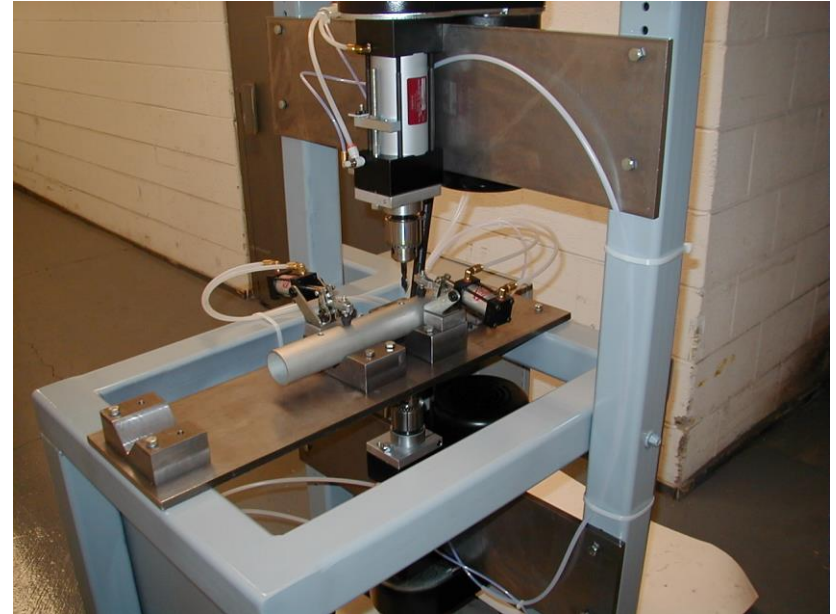
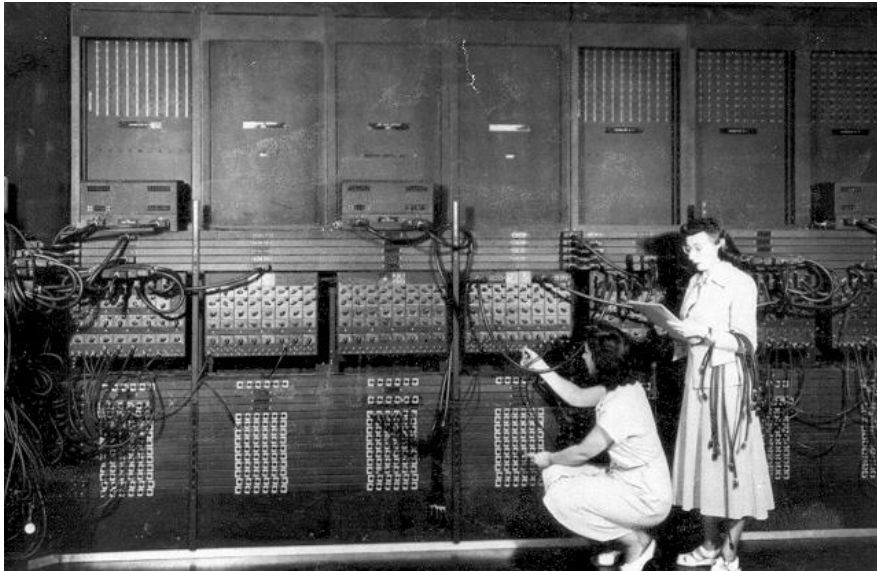
*) http://en.wikipedia.org/wiki/Alan_Turing
http://en.wikipedia.org/wiki/Claude_Shannon

Gebeurtenissen in de digitale techniek

Jaar	Naam	Gebeurtenis/uitvinding
1964	IBM	Derde generatie computers met IC's
1971	Intel	Eerste microprocessor (4004, 4 bits, 2300 transistoren)
1974	Intel, Motorola	8080/85, 6800 e.a. (8 bits)
1978	Intel	8086 (16 bits, 29000 transistoren)
1979	Motorola	68000 (32 bits, 68000 transistoren)
1993	Intel	Pentium
1995	Intel	Pentium Pro 200 MHz
2010	Sun	1 miljard transistoren op een IC, 16-Core Sparc T3
2014	Intel	15-Core Xeon Ivy Bridge-EX (5 miljard transistoren)

Oorsprong

- Automatisering productieprocessen
- Uitvoeren van complexe berekeningen



Industriële automatisering

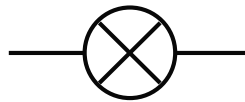
- In de industriële automatisering wordt gebruik gemaakt van sensoren en actuatoren. Denk aan: drukknoppen, eindmelders, lampen, motor.



Schakelaar gesloten (n.c.)

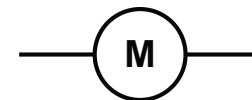
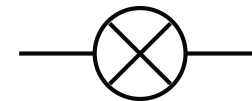


Schakelaar open (n.o.)



Binaire signalen

- Aan de toestanden van bv. schakelaars en motoren kunnen twee waarden gekoppeld worden: 0 en 1.
- Schakelaars (altijd in rust getekend):
 - In rust: 0
 - Actief: 1
- Lampen:
 - Brandt niet: 0
 - Brandt: 1
- Motoren:
 - Uit: 0
 - Draait: 1



Heggenschaar

- Hoe wordt een heggenschaar bediend?



- Met twee schakelaars, beide moeten geactiveerd worden om de motor van de heggenschaar te starten.

Heggenschaar

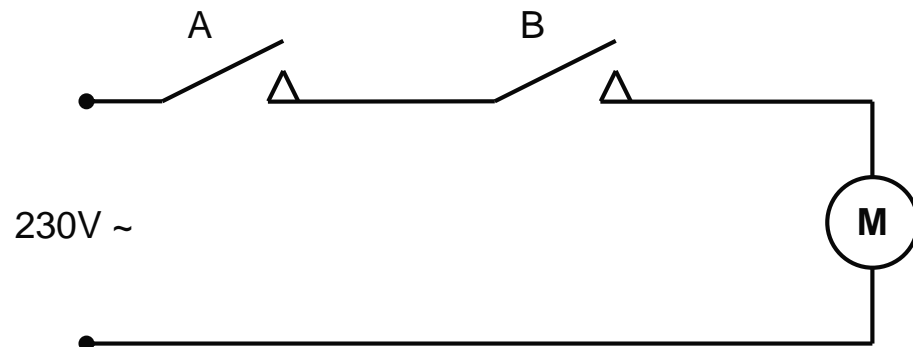


Schakelaar open: 0

Schakelaar gesloten : 1

Motor uit: 0

Motor aan: 1



Waarheidstabel

- Alle combinaties kunnen in een tabel gezet worden.
- Dit wordt een waarheidstabel genoemd.
- Aan elke combinatie kan een betekenis gekoppeld worden.

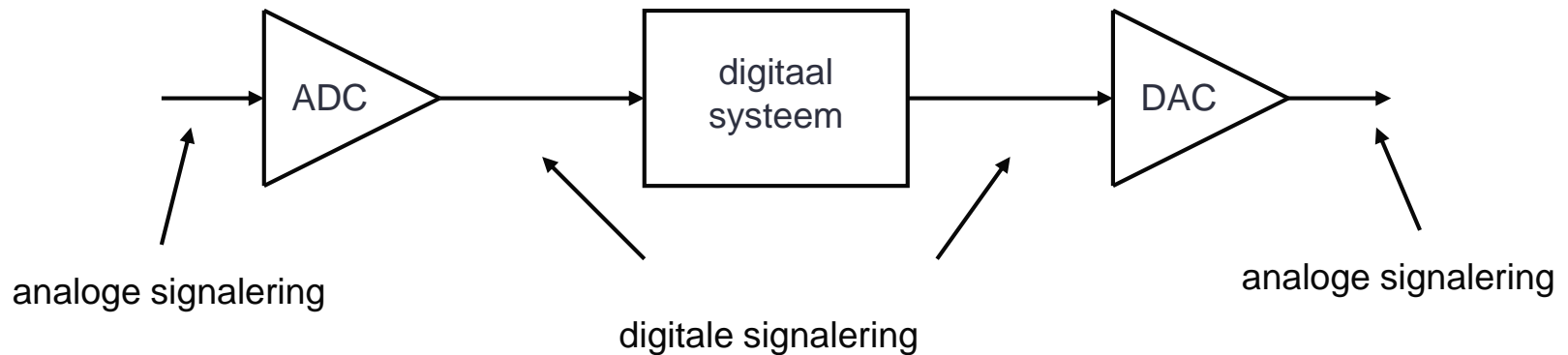
A	B	M	Betekenis
0	0	0	schakelaars open, motor uit
0	1	0	schak. A open, schak. B gesloten, motor uit
1	0	0	schak. A gesloten, schak. B open, motor uit
1	1	1	schakelaars gesloten, motor aan.

Digitalisering

- Bekende fysische grootheden (stroom, spanning, druk, licht, hoek, etc.) worden omgezet naar getallen.
- Het is eigenlijk een transformatie van de fysische grootheden naar getallen die in principe niet binair (tweewaardig) hoeven te zijn.
- De digitale informatie kan met behulp van elektronica (computers) bewerkt worden en weer naar de fysische grootheden terug getransformeerd worden.
- Dus: fysiek $\xrightarrow{\text{omzetting}}$ getallen $\xrightarrow{\text{omzetting}}$ fysiek

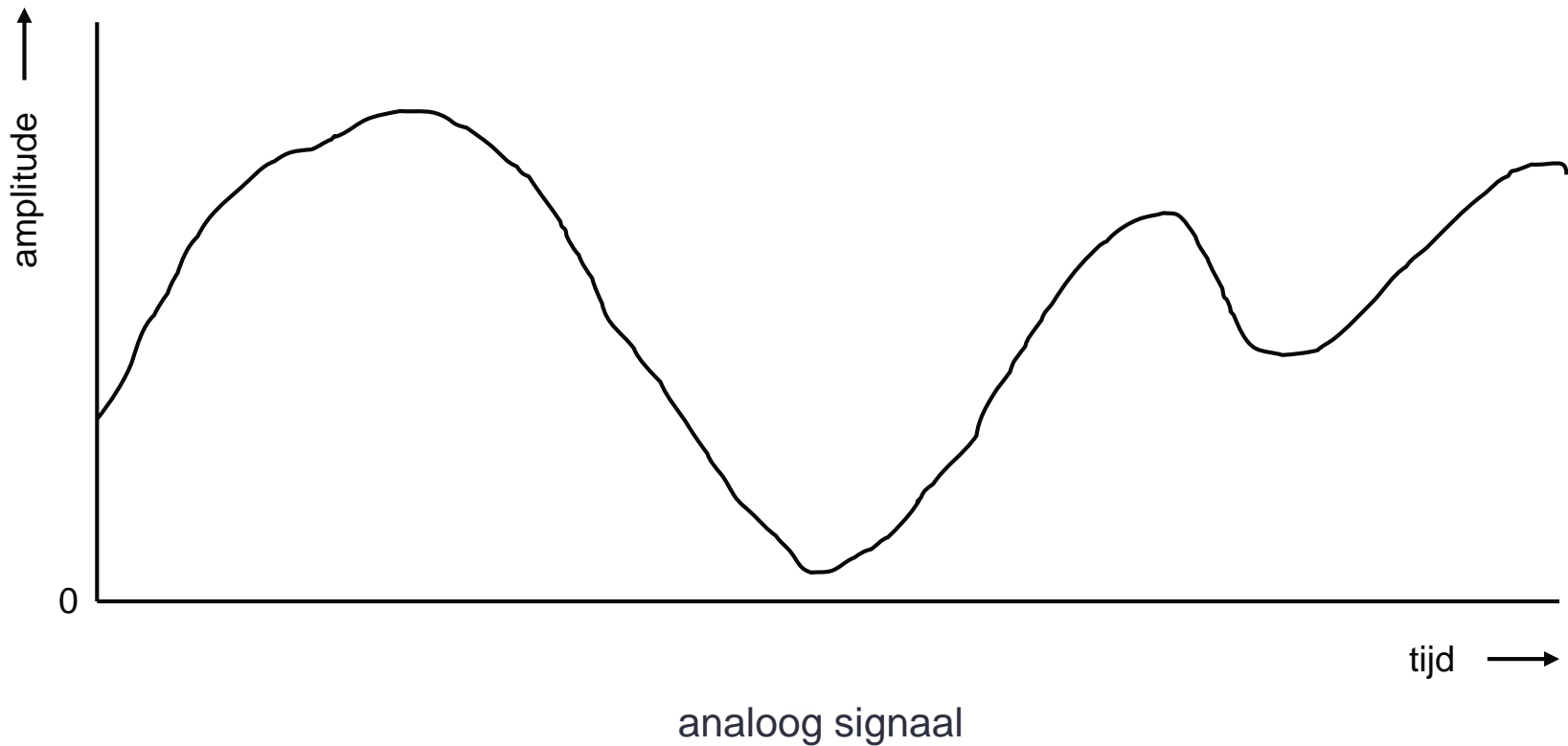
Digitalisering

- Transformatie gebeurt met behulp van Analog-Digital Converter (ADC) en Digital-Analog Converter (DAC).



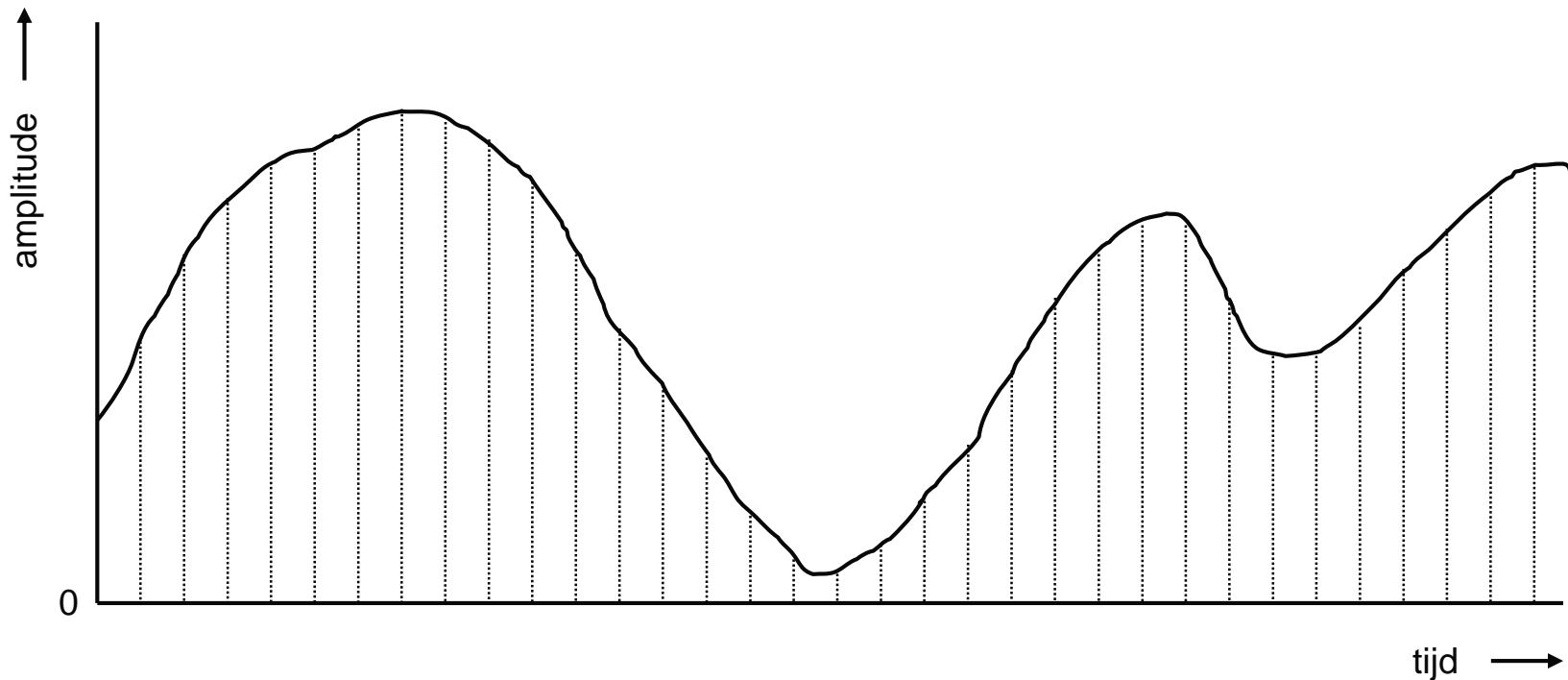
Digitalisering

- Voorbeeld: geluidsregistratie



Digitalisering

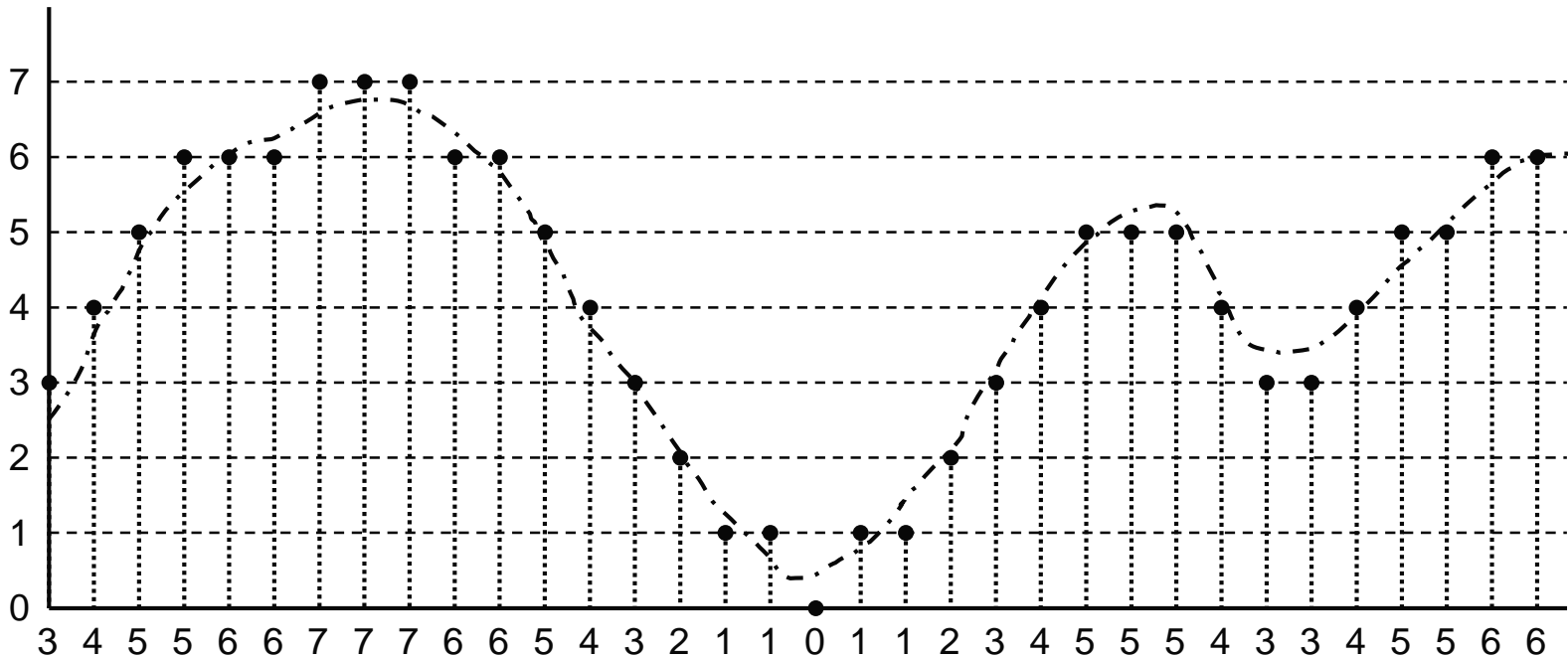
- Het analoge signaal wordt bemonsterd (gesampled), *equidistante* tijdstippen.



Voor CD-kwaliteit worden 44100 samples per seconde gemaakt (22050 Hz)

Digitalisering

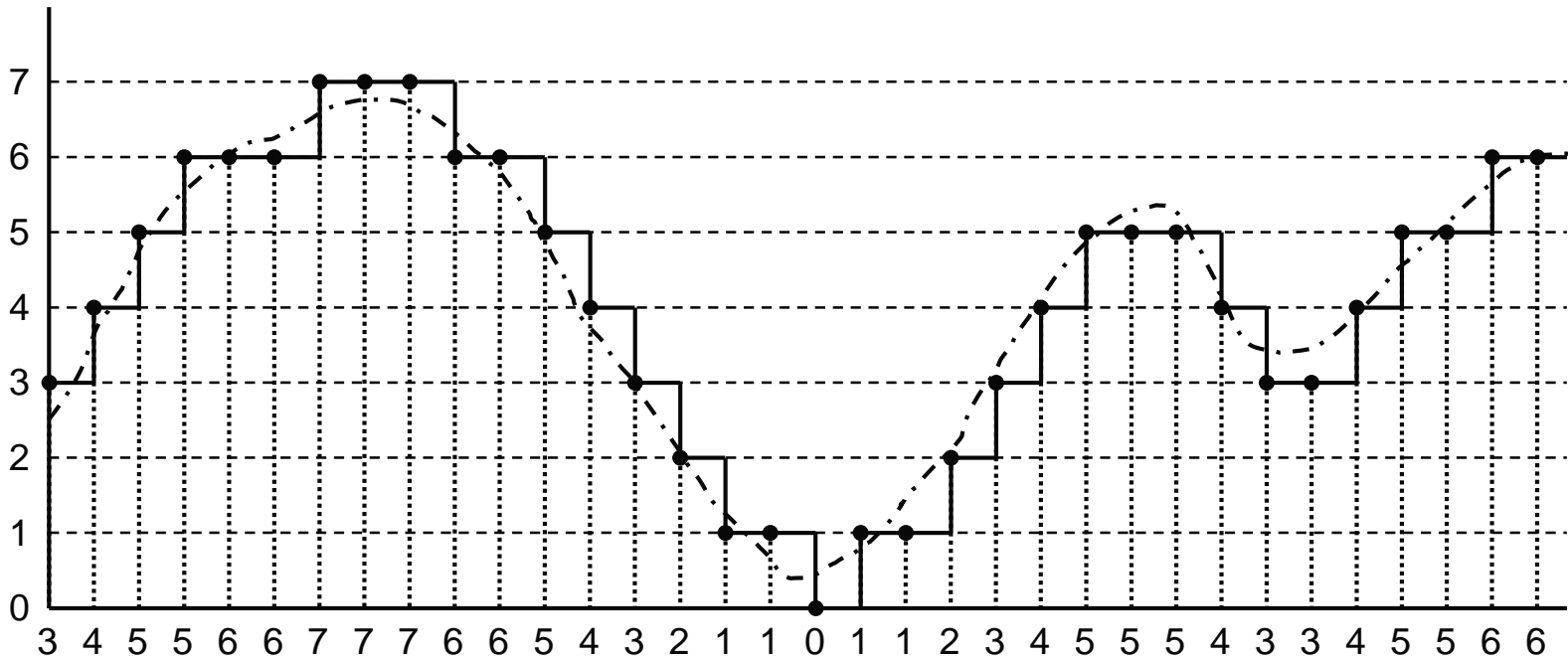
- Verdeling van het discreet-analoge signaal in acht gebieden, introduceert afrondingsfouten.



Een CD heeft 65536 kwantiseringsniveau's (16 bit).

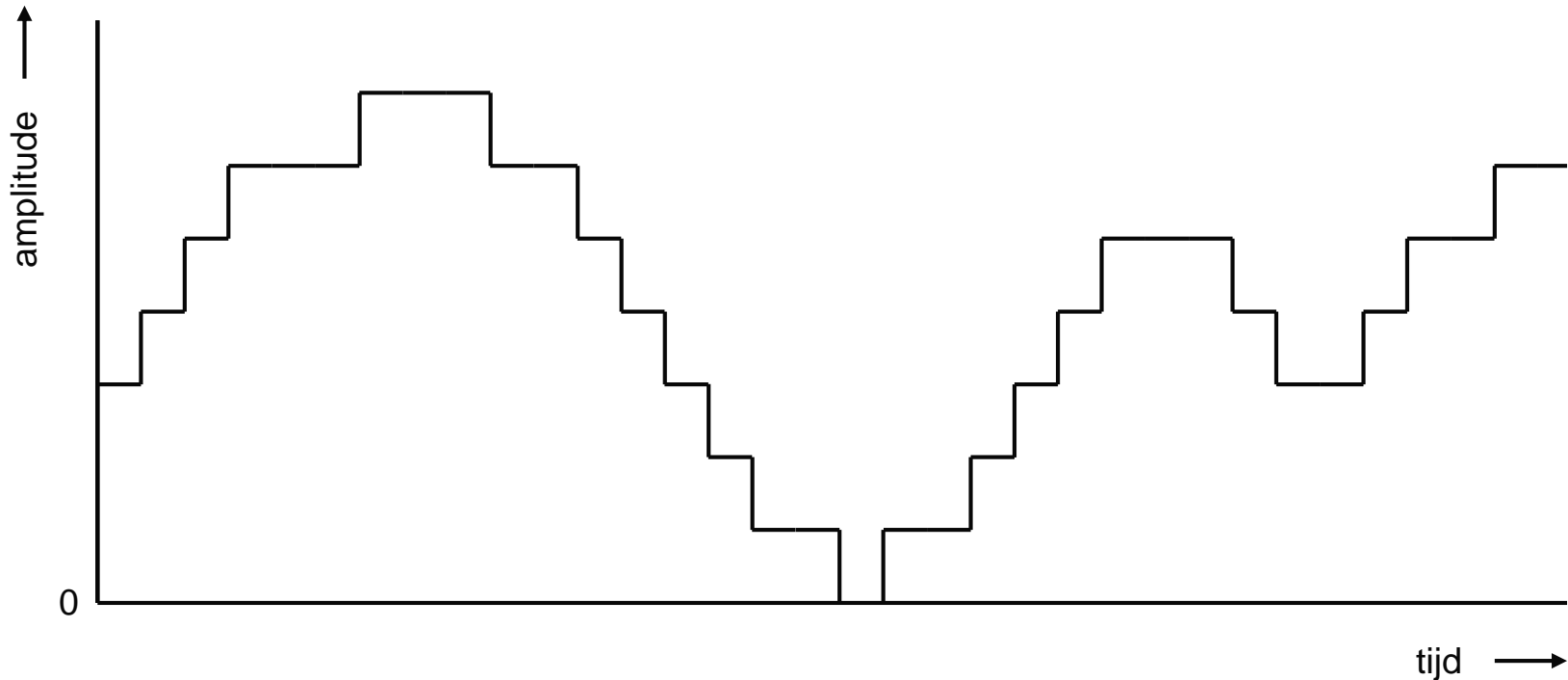
Digitalisering

- Het geanalogueerde muzieksignaal.



Digitalisering

- Reconstructie van het analoge signaal.



Digitalisering

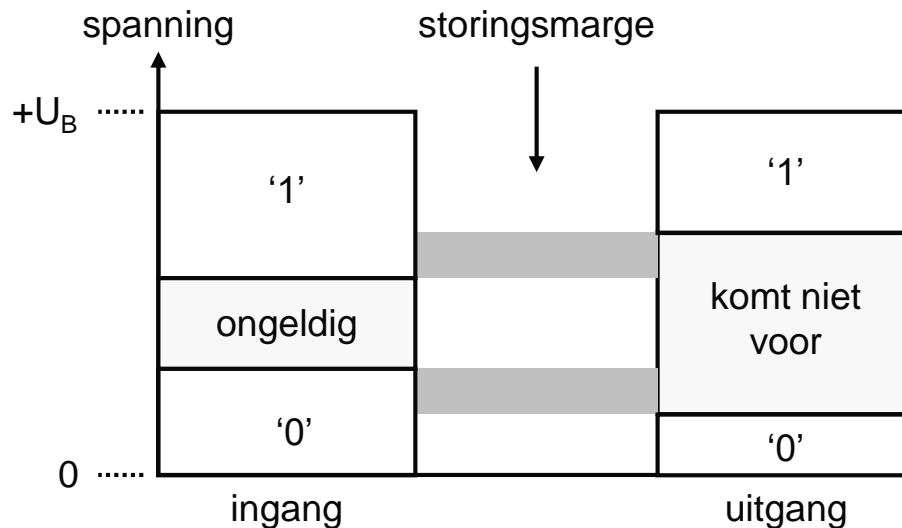
- Waarom zo moeilijk: fysiek → getallen → fysiek ?
- Verwerking
 - informatie is gemakkelijk te bewerken (versleutelen, comprimeren, foutdetectie, foutcorrectie, complexe signaalbewerkingen);
 - gemakkelijk informatie op te slaan voor latere bewerking;
 - informatie kan langere tijd bewaard worden met behoud van kwaliteit (reproduceerbaarheid van informatie);
 - betrouwbaarheid m.b.t. storende signalen;
 - snelheid;
 - parallel verwerken van informatie.

Digitalisering

- Waarom zo moeilijk: fysiek → getallen → fysiek ?
- Ontwerpen
 - snel ontwerpen en testen m.b.v. tools (beschrijvingstaal, simulatie);
 - eenvoudig te leren, geen uitgebreide wiskundige kennis nodig*;
 - uitbreidbaarheid (vooral in reconfigureerbare logica);
 - hoge integratiedichtheid (veel functionaliteit op een klein oppervlak);
 - goede testvoorzieningen aan te brengen.
 - gebruik van standaard blokken (IP-cores; Ethernet, USB, processoren, DSP, RAM).
- De wiskunde voor digitale schakelingen is beperkt, maar de te ontwikkelen schakeling kan wel voortkomen uit een wiskundige functie, bv. digitale filtering.

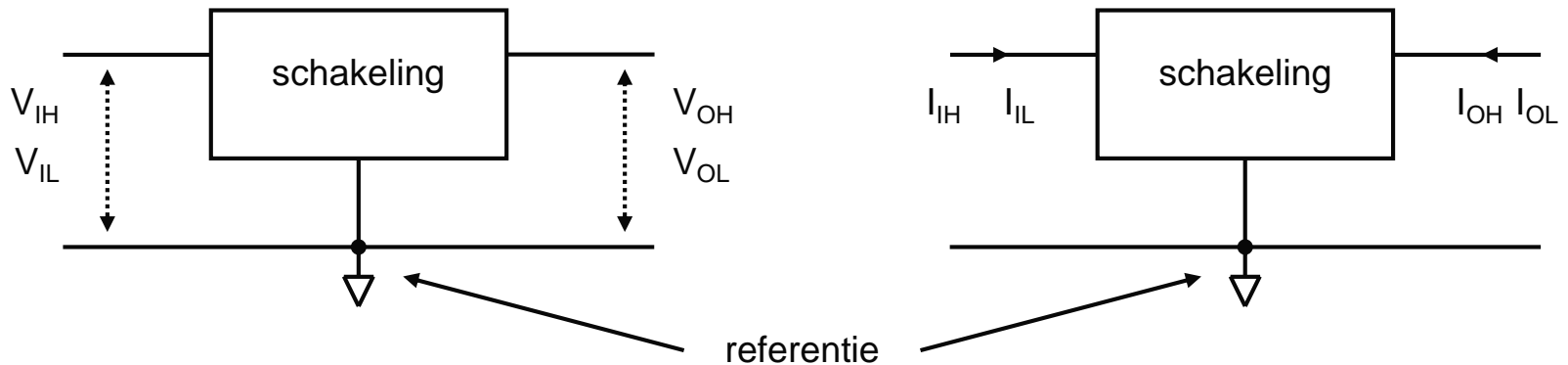
Elektrische eigenschappen

- Een digitale schakeling werkt met spanningen en stromen. Voor het aangeven van een logische waarde wordt een spanningsgebied gebruikt. Een storingsmarge tussen ingang en uitgang is noodzakelijk om variaties in voedings- en signaalspanningen op te vangen.



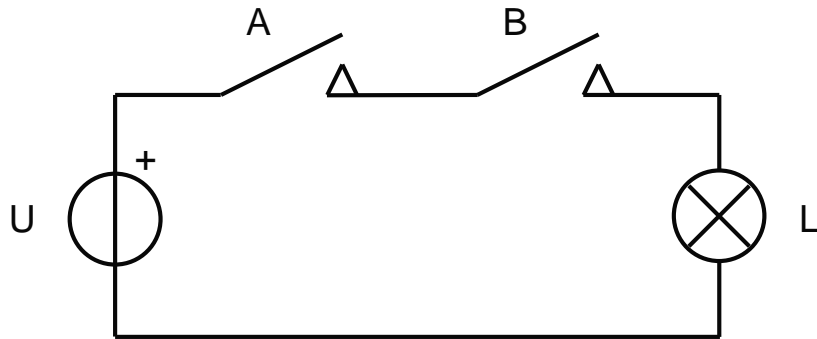
Elektrische eigenschappen

- De fabrikant geeft in *datasheets* aan hoe de spanningen en stromen zijn gedefinieerd. Alle spanningen zijn gegeven ten opzichte van de referentie. De stromen zijn naar binnen toe gedefinieerd.



Logische schakelingen met schakelaars

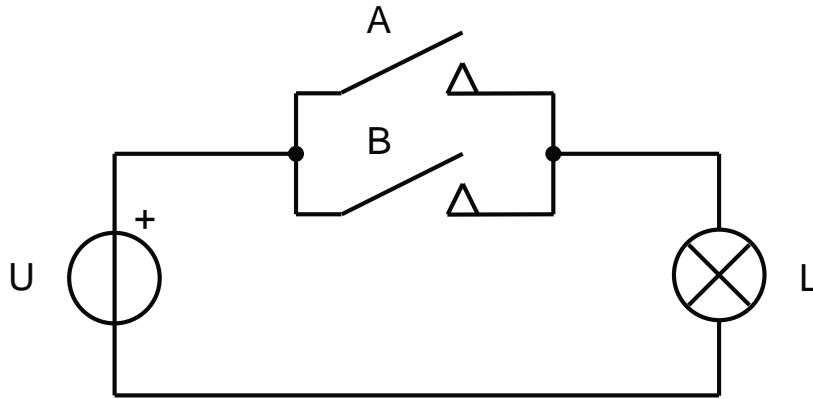
- Met schakelaars zijn logische schakelingen te maken. Een schakelaar in rust wordt gezien als logisch 0. Een geactiveerde schakelaar wordt gezien als 1. We kunnen alle mogelijkheden weergeven in een *waarheidstabel*.



logische AND

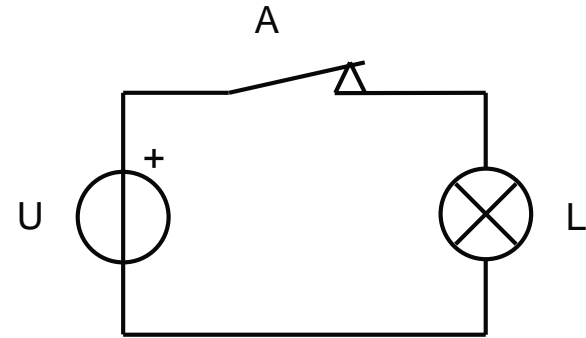
A	B	L
0	0	0
0	1	0
1	0	0
1	1	1

Logische schakelingen met schakelaars



logische OR

A	B	L
0	0	0
0	1	1
1	0	1
1	1	1

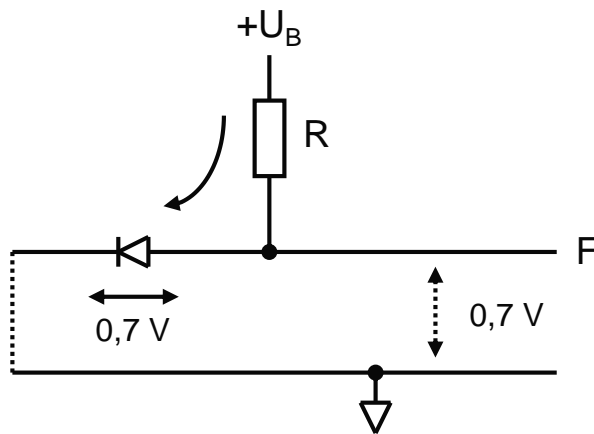


logische NOT

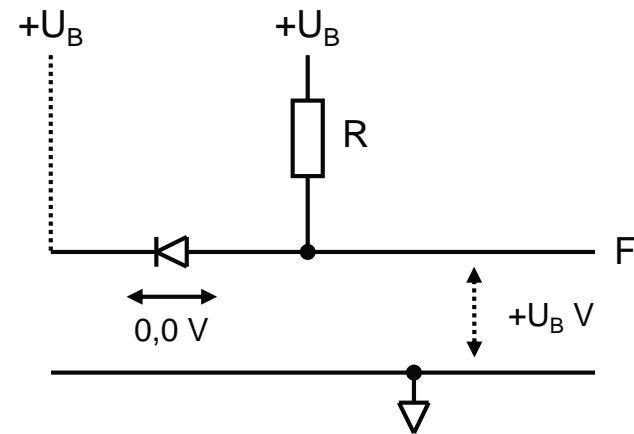
A	L
0	1
1	0

Logische schakelingen met diodes

- Met diodes en weerstanden zijn logische schakelingen te maken. Een diode is een elektronische component die de stroom slechts een kant op geleidt. De werkspanning in *doorlaat* is 0,7 V.



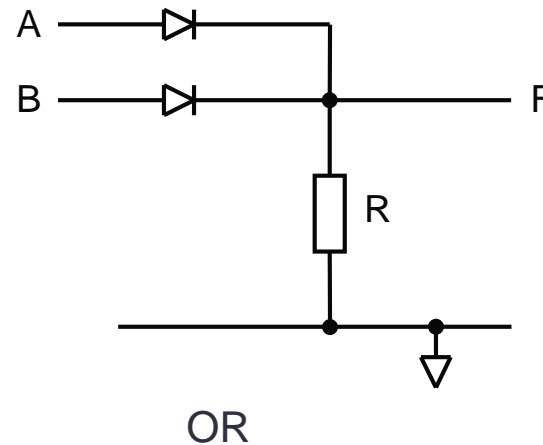
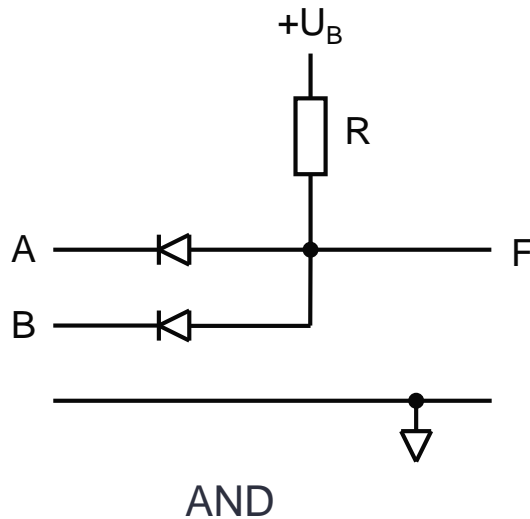
diode in doorlaat



diode in sper

Logische schakelingen met diodes

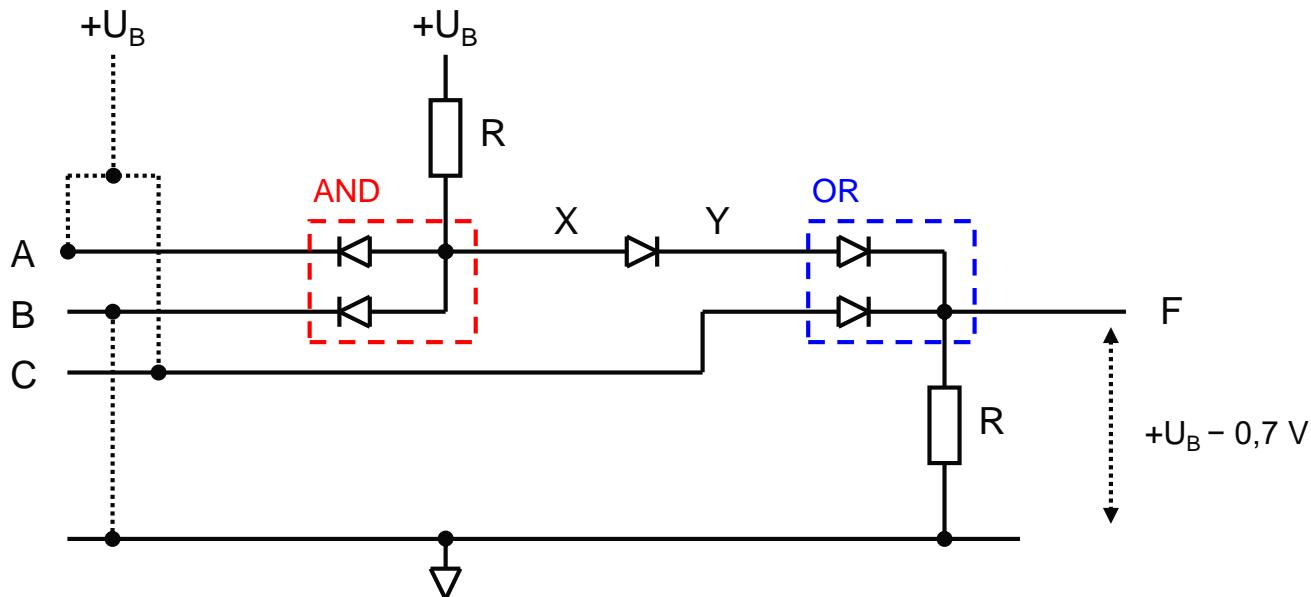
- Hieronder zijn de AND- en OR-schakeling met diodes gegeven.



- Het is niet mogelijk om met alleen diodes en weerstanden een NOT te maken.

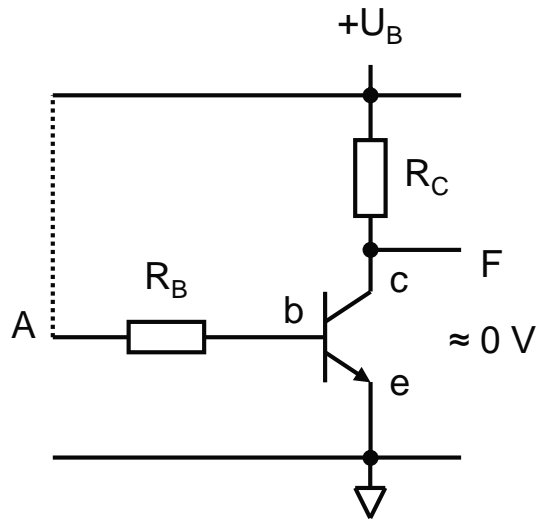
Logische schakelingen met diodes

- Er kunnen combinaties van AND en OR gemaakt worden. Functie onderstaande schakeling: $F = (A \cdot B) + C$

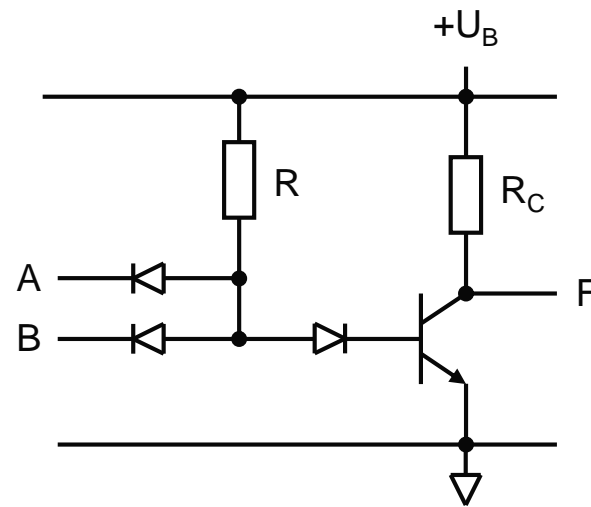


Logische schakelingen met transistoren

- Met transistoren en diodes kunnen logische schakelingen gemaakt worden. Links een inverter, rechts een NOT-AND- of NAND-poort.

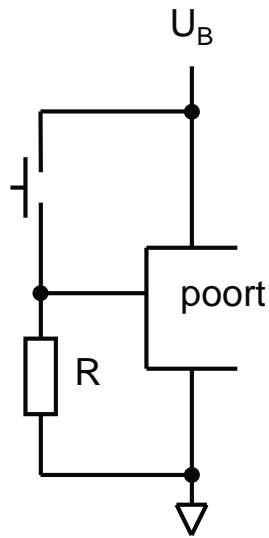


inverter

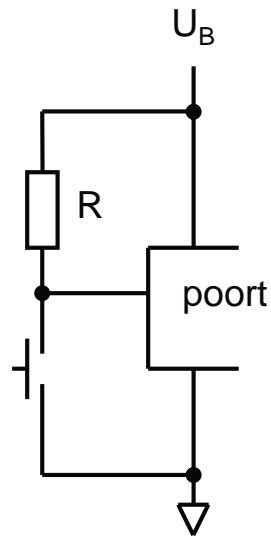


NAND

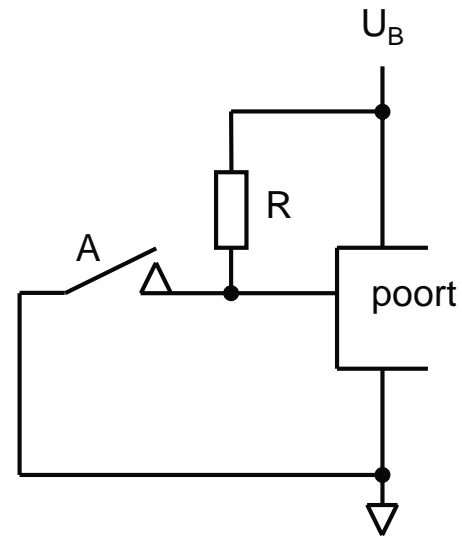
Actief hoog en laag



actief hoog

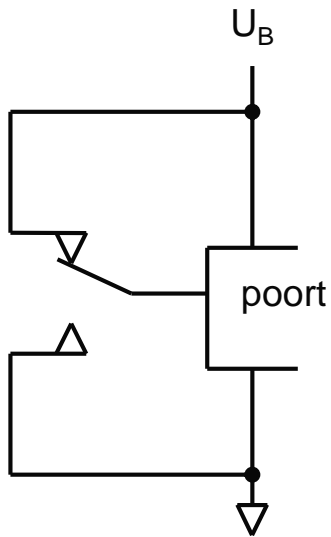


actief laag

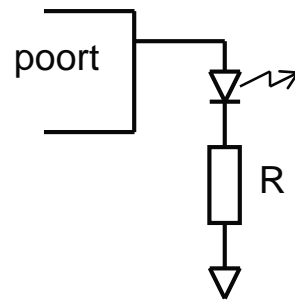


actief laag

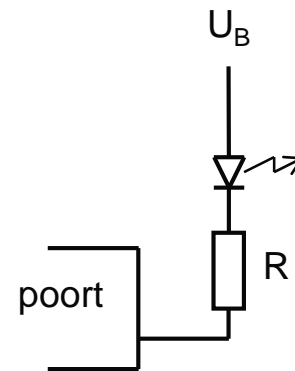
Actief hoog en laag



twee actieve standen



actief hoog



actief laag

Poorten

- Digitale schakelingen worden verreweg het meest ontworpen met elektronische componenten.
- Andere mogelijkheden:
 - pneumatische componenten (lucht)
 - hydraulische componenten (vloeistof)
 - fotonische componenten (licht)
- Digitale schakelingen worden opgebouwd met behulp van *poorten*.
- Eerst worden de elementaire poorten besproken.

Poorten

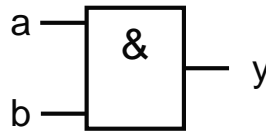
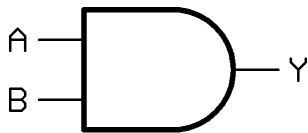
- Elke digitale schakeling kan worden opgebouwd met behulp van drie typen poorten:
- AND (en)
- OR (of)
- NOT (niet, inverter)
- Andere veel voorkomende poorten:
 - Buffer, EXOR, NAND, NOR, EXNOR

Poorten

- Op de volgende slides worden de elementaire poorten besproken.
- Gegeven wordt:
- Naam en beschrijving
- Poortsymbolen
 - Amerikaanse of vormsymbolen, worden veel in CAD-programma's gebruikt
 - IEC/IEEE symbolen, wordt veel in Nederlandse literatuur gebruikt
- Waarheidstabel
- Wiskundige functie
- Uitspraak

AND

- Uitgang y geeft alleen een 1 af als beide ingangen een 1 zijn, anders geeft uitgang y een 0 af.



a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

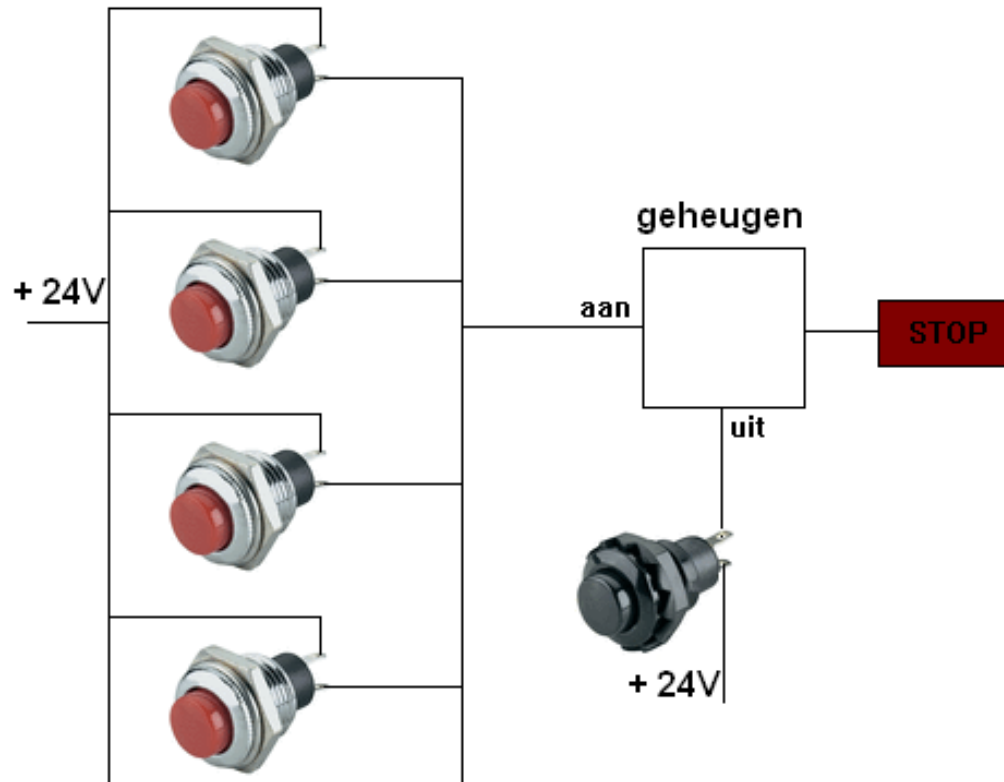
- Functie: $y = a \cdot b = ab$
- Uitspraak: “ y is gelijk aan a and b ”

Busstopschakeling

- Hoe werkt een busstopschakeling?
- Eén of meerdere stopschakelaars moeten worden ingedrukt om de bus te laten stoppen.



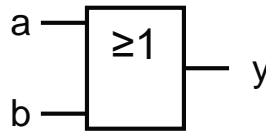
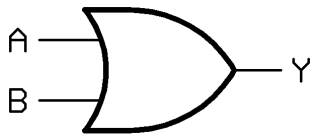
Busstopschakeling



- Geheugenelement wordt later besproken.

OR

- Uitgang y geeft een 1 af als één of meer ingangen 1 zijn, anders geeft uitgang y een 0 af.



a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

- Functie: $y = a + b$
- Uitspraak: “y is gelijk aan a or b”

Noodverlichting

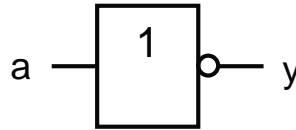
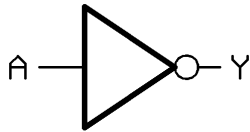
- Hoe werkt een noodverlichting?



- Als spanning uitvalt, dan verlichting aan.

NOT (inverter)

- Uitgang y geeft een 0 af ingang a een 1 is, anders geeft uitgang y een 1 af.



a	y
0	1
1	0

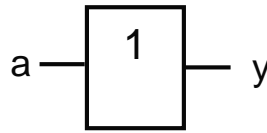
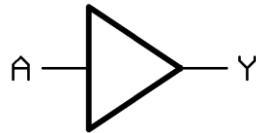
- Functie: $y = \bar{a}$
- Uitspraak: “ y is gelijk aan not a ”

Nog meer poorten

- Hiervoor zijn de drie basispoorten besproken. Hiermee kan elke digitale schakeling worden ontworpen en gebouwd.
- In de praktijk worden nog andere poorten gebruikt.
- Deze poorten zijn niet strikt noodzakelijk, wel heel handig. Ze worden gebruikt bij veel voorkomende bewerkingen.
- Al deze poorten kunnen worden opgebouwd uit AND, OR en NOT.

Buffer

- Uitgang y geeft een 1 af ingang a een 1 is, anders geeft uitgang y een 0 af.

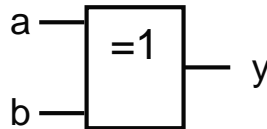
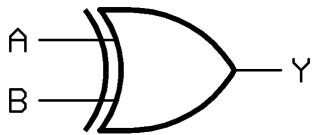


a	y
0	0
1	1

- Functie: $y = a$
- Uitspraak: “y is gelijk aan a”

EXOR

- Uitgang y geeft een 1 af als precies één ingang 1 is, anders geeft uitgang y een 0 af.

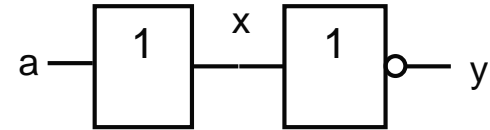
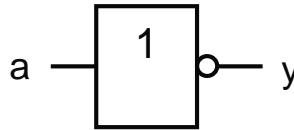
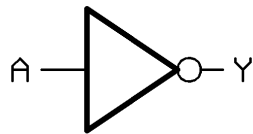


a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

- Functie: $y = a \oplus b$
- Uitspraak: “ y is gelijk aan a exor b ”

NOT buffer

- Symbool en schema



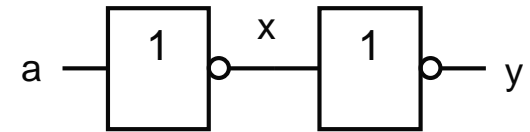
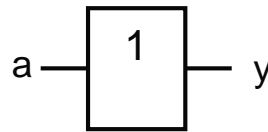
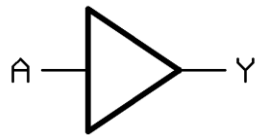
- Functie: $x = a$ $y = \bar{x}$
 $y = \bar{a}$

a	x	y
0	0	1
1	1	0

- Uitspraak: “y is gelijk aan not a”

NOT NOT

- Symbool en schema



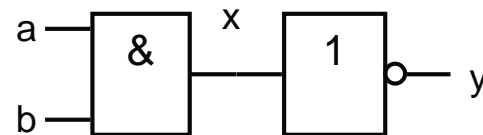
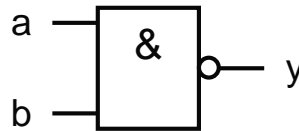
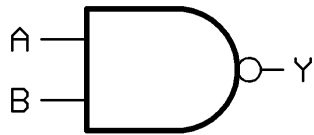
- Functie: $x = \bar{a}$ $y = \bar{x}$
 $y = a$

a	x	y
0	1	0
1	0	1

- Uitspraak: “y is gelijk aan a”

NOT AND (NAND)

- Symbool en schema



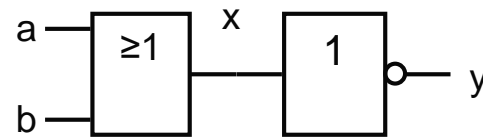
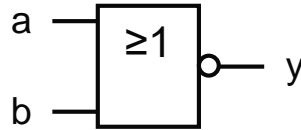
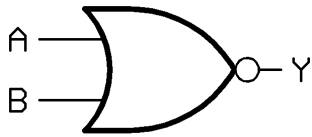
- Functie: $y = \overline{x}$ $x = a \cdot b$
 $y = \overline{a \cdot b}$

- Uitspraak: “y is gelijk aan a nand b”

a	b	x	y
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

NOT OR (NOR)

- Symbool en schema



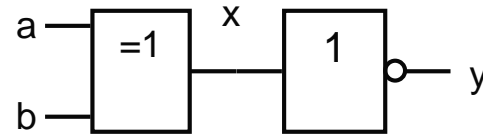
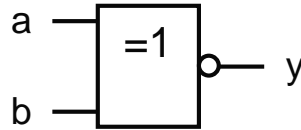
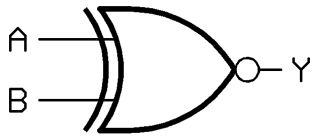
- Functie: $y = \overline{x}$ $x = a + b$
 $y = \overline{a + b}$

- Uitspraak: “y is gelijk aan a nor b”

a	b	x	y
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

NOT EXOR (EXNOR)

- Symbool en schema



- Functie: $y = \bar{x}$ $x = a \oplus b$
 $y = \overline{a \oplus b}$

- Uitspraak: “y is gelijk aan a exnor b”

a	b	x	y
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

Over OR en EXOR

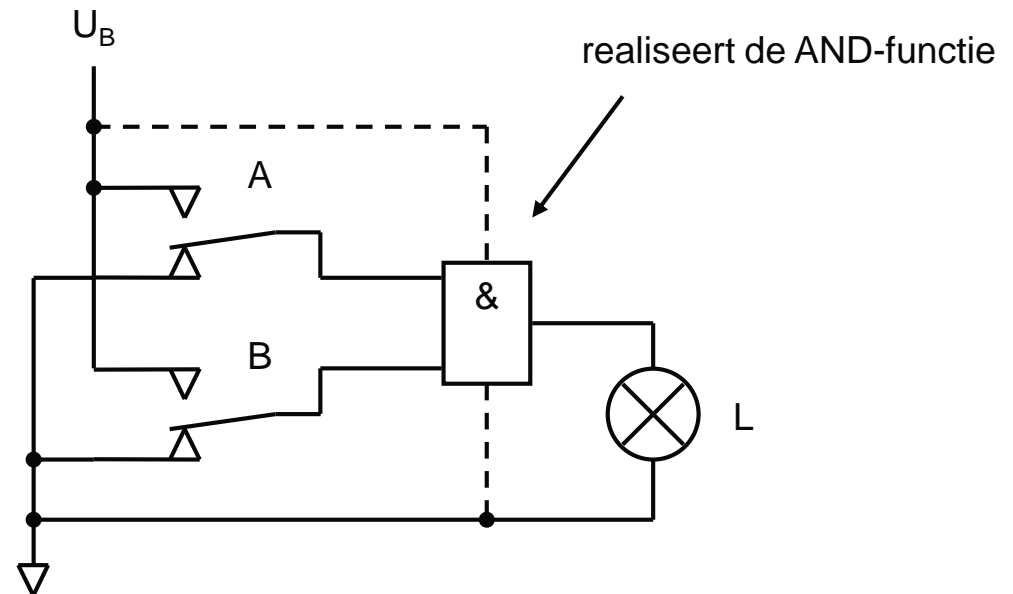
- De OR wordt ook wel *inclusive OR* genoemd omdat deze ook een 1 levert als beide ingangen 1 zijn.
- Óf de ene ingang, óf de andere ingang óf allebei.
- Voor de EXOR (*exclusive OR*) geldt: óf de ene ingang, óf de andere ingang, maar niet allebei.
- Andere kenmerken: kop óf munt, ik eruit óf jij eruit.

Overzicht

buffer	$y = a$ “a”	NOT	$y = \bar{a}$ “not a”
AND	$y = a \cdot b$ “a and b”	NAND	$y = \overline{a \cdot b}$ “a nand b”
OR	$y = a + b$ “a or b”	NOR	$y = \overline{a + b}$ “a nor b”
EXOR	$y = a \oplus b$ “a exor b”	EXNOR	$y = \overline{a \oplus b}$ “a exnor b”

Poorten

- Deze schakeling realiseert de AND-functie met behulp van een poort.
- De lamp brandt als beide schakelaars geactiveerd zijn.
- Vertaling spanningen voor ingangen en uitgang:
'0' = 0 V
'1' = U_B V

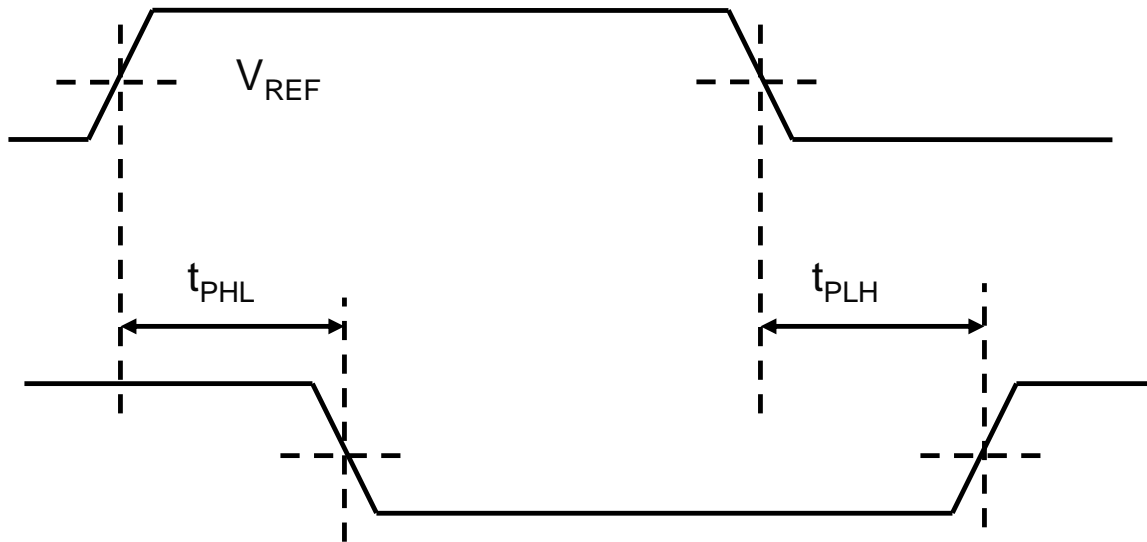


Timing

- Een verandering op de ingang van een component geeft als resultaat een verandering op de uitgang.
- Dit gaat niet oneindig snel; de transistoren in een IC hebben parasitaire capaciteiten en die moeten worden opgeladen of ontladen.
- Het duurt dus even voordat de uitgangsverandering zichtbaar wordt.
- Deze tijd wordt vertragingstijd (*propagation delay*) genoemd.

Timing

- De fabrikant geeft in de *datasheet* hoeveel deze tijden bedragen, met meestal daarbij de meetopstelling en -instellingen.
- De relaties tussen ingangen en uitgangen worden getekend in een tijdvolgordediagram (timing diagram).



V_{REF} = de referentiespanning voor de tijdmeting ($\frac{1}{2} V_{DD}$)

t_{PHL} = vertragingstijd voor uitgang hoog naar laag

t_{PLH} = vertragingstijd voor uitgang laag naar hoog

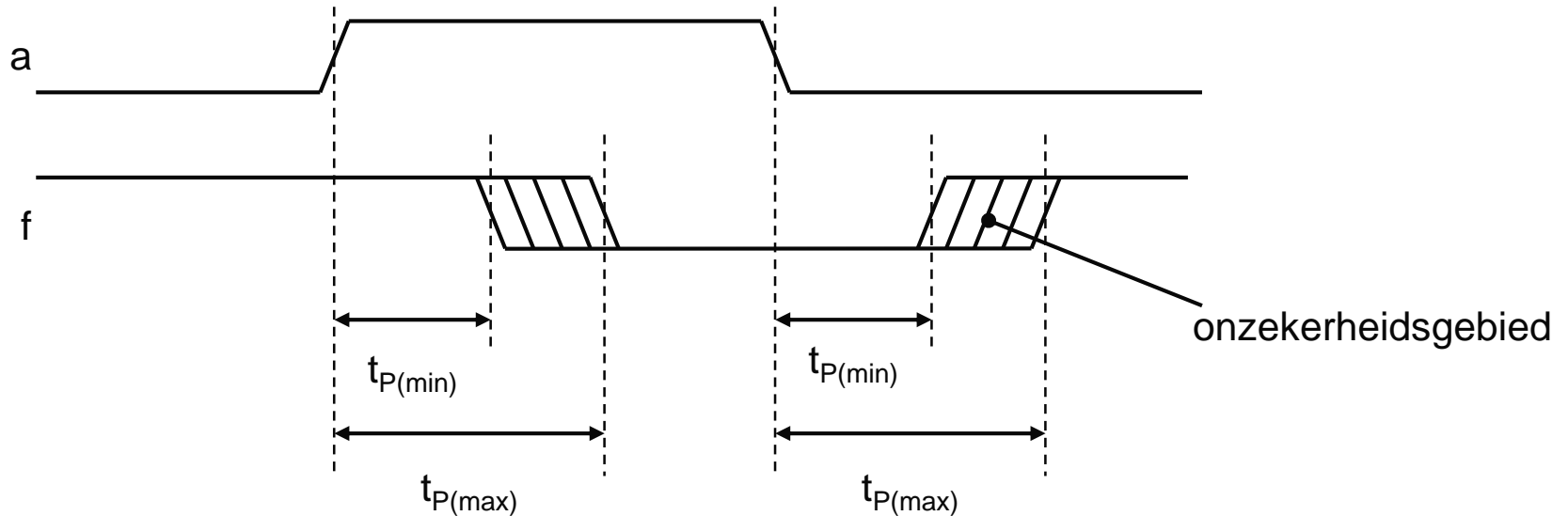
Timing

- Bij componenten (losse poorten, library-modules) worden de volgende tijden vermeld:
- $t_{P(\min)}$ De fabrikant garandeert dat de uitgang nooit eerder verandert dan de minimale vertragingstijd.
- $t_{P(\max)}$ De fabrikant garandeert dat de uitgang nooit later verandert dan de maximale vertragingstijd.
- $t_{P(\text{typ})}$ De typische vertragingstijd, “nice to know”, maar er kan niet mee ontworpen worden.

nb: in plaats van t_p ook wel t_{pD}

Timing

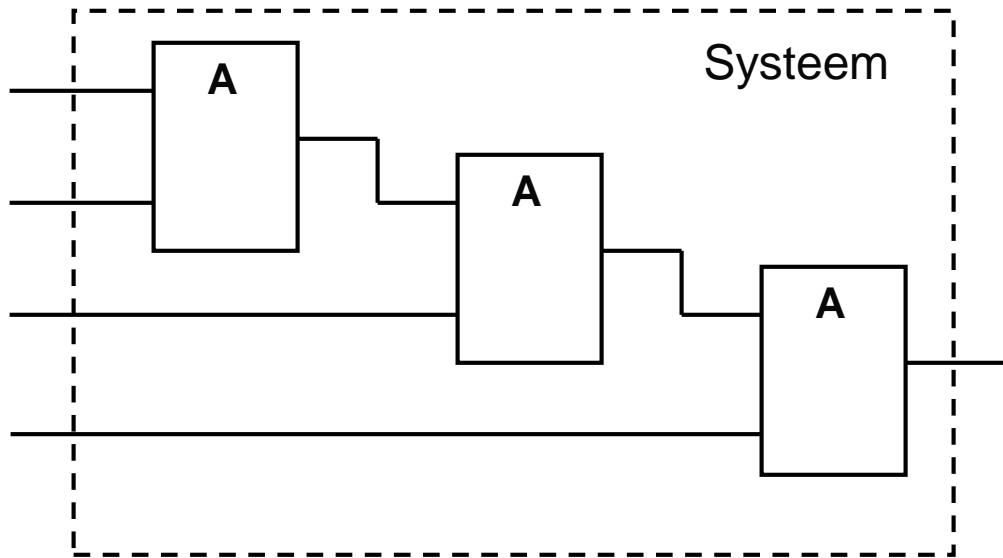
- Hieronder het timingdiagram van een NOT-poort.



- In het gearceerde gedeelte is de definitieve waarde van de uitgang nog onzeker.

Timing

- Gegeven de volgende schakeling:



$$t_{P(\min)}(A) = 4 \text{ ns.}$$

$$t_{P(\max)}(A) = 9 \text{ ns.}$$

- Vraag: wat zijn de $t_{P(\min)}$ en $t_{P(\max)}$ van dit systeem?

$$t_{P(\max)}(\text{systeem}) = 27 \text{ ns.}$$

$$t_{P(\min)}(\text{systeem}) = 4 \text{ ns.}$$

Timing

- Voorbeeld van een 74F00 quad 2-input NAND.

testcondities

AC ELECTRICAL CHARACTERISTICS

SYMBOL	PARAMETER	TEST CONDITION	LIMITS						UNIT	
			$V_{CC} = +5.0V$ $T_{amb} = +25^{\circ}C$ $C_L = 50pF, R_L = 500\Omega$			$V_{CC} = +5.0V \pm 10\%$ $T_{amb} = 0^{\circ}C \text{ to } +70^{\circ}C$ $C_L = 50pF, R_L = 500\Omega$		$V_{CC} = +5.0V \pm 10\%$ $T_{amb} = -40^{\circ}C \text{ to } +85^{\circ}C$ $C_L = 50pF, R_L = 500\Omega$		
			MIN	TYP	MAX	MIN	MAX	MIN		MAX
t_{PLH} t_{PHL}	Propagation delay Dna, Dnb to \bar{Q}_n	Waveform 1	2.4 2.0	3.7 3.2	5.0 4.3	2.4 2.0	6.0 5.3	2.0 1.5	6.5 6.0	ns

↑ ingangen naar uitgangen
 ↑ testsignaal

Frequentie & tijden

- In onderstaande tabel zijn enkele frequenties en bijbehorende tijden gegeven volgens de bekende formule $f = 1/T$.

Freq (not)	Freq (eng)	Tijd (not)	Tijd (eng)
1 Hz	1 Hz	1 s = 1000 ms	1 s
1 kHz	10^3 Hz	1 ms = 1000 μ s	10^{-3} s
1 MHz	10^6 Hz	1 μ s = 1000 ns	10^{-6} s
10 MHz	$10 \cdot 10^6$ Hz	100 ns = 0,1 μ s	$100 \cdot 10^{-9}$ s
100 MHz	$100 \cdot 10^6$ Hz	10 ns	$10 \cdot 10^{-9}$ s
1 GHz	10^9 Hz	1 ns = 1000 ps	10^{-9} s
1 THz	10^{12} Hz	1 ps	10^{-12} s

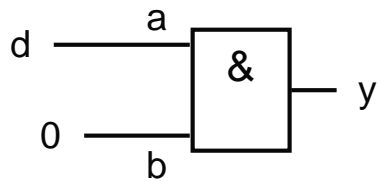
k = kilo, M = mega, G = giga, T = tera
m = milli, μ = micro, n = nano, p = pico

Vereenvoudigen

- We bekijken de AND- en OR-poort nog eens nader.
- Eén van de ingangen wordt aan een logische 0 of 1 gelegd, of de ingangen worden aan met elkaar verbonden, of één ingang krijgt de inverse van de andere ingang.
- Hierdoor vereenvoudigt de schakeling zich.
- Het levert een praktisch bewijs van enkele wiskundige stellingen (axioma's en theorema's).

AND revisited

- Wat gebeurt er als één ingang van de AND verbonden wordt aan een logische 0?

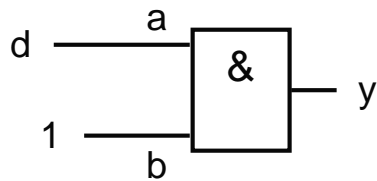


- Functie: $y = d \cdot 0 = 0$

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

AND revisited

- Wat gebeurt er als één ingang van de AND verbonden wordt aan een logische 1?

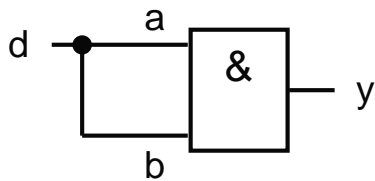


- Functie: $y = d \cdot 1 = d$

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

AND revisited

- Wat gebeurt er als de ingangen van de AND met elkaar verbonden worden?

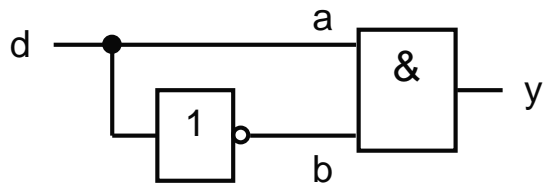


- Functie: $y = d \cdot d = d$

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

AND revisited

- Wat gebeurt er als één van de ingangen van de AND geïnverteerd wordt aangeboden vanuit hetzelfde signaal?

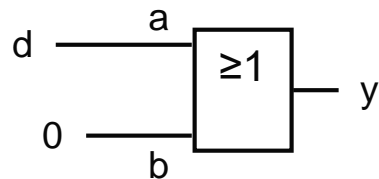


- Functie: $y = d \cdot \bar{d} = 0$

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

OR revisited

- Wat gebeurt er als één ingang van de OR verbonden wordt aan een logische 0?

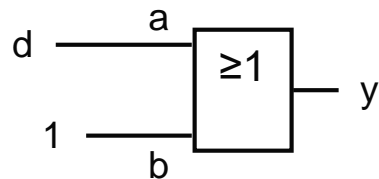


- Functie: $y = d + 0 = d$

a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

OR revisited

- Wat gebeurt er als één ingang van de OR verbonden wordt aan een logische 1?

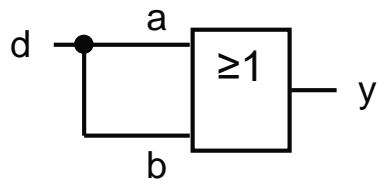


- Functie: $y = d + 1 = 1$

a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

OR revisited

- Wat gebeurt er als de ingangen van de OR met elkaar verbonden worden?

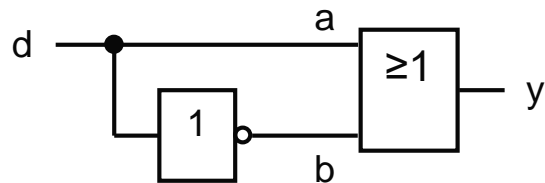


- Functie: $y = d + d = d$

a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

OR revisited

- Wat gebeurt er als één van de ingangen van de OR geïnverteerd wordt aangeboden vanuit hetzelfde signaal?



- Functie: $y = d + \bar{d} = 1$

a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

Alles op een rij

- Op deze slide een overzicht van alle vereenvoudigingen

$$y = d \cdot 0$$

$$y = 0$$

$$y = d + 0$$

$$y = d$$

$$y = d \cdot 1$$

$$y = d$$

$$y = d + 1$$

$$y = 1$$

$$y = d \cdot d$$

$$y = d$$

$$y = d + d$$

$$y = d$$

$$y = d \cdot \bar{d}$$

$$y = 0$$

$$y = d + \bar{d}$$

$$y = 1$$

$$y = \bar{\bar{d}}$$

$$y = d$$

Meer en meer

- AND- en OR-poorten kunnen ook gebouwd worden met meer dan twee ingangen. Praktisch worden poorten gebouwd met maximaal vier ingangen.

AND3	$y = a \cdot b \cdot c = abc$	$y \leq a \text{ AND } b \text{ AND } c;$
AND4	$y = a \cdot b \cdot c \cdot d = abcd$	$y \leq a \text{ AND } b \text{ AND } c \text{ AND } d;$
OR3	$y = a + b + c$	$y \leq a \text{ OR } b \text{ OR } c;$
OR4	$y = a + b + c + d$	$y \leq a \text{ OR } b \text{ OR } c \text{ OR } d;$

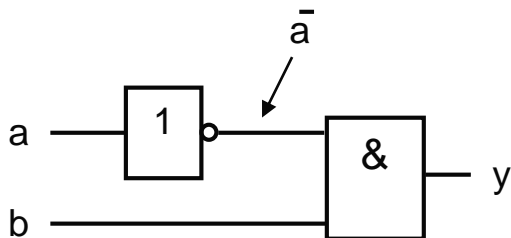
Vereenvoudigen

- Met behulp van de optimalisaties van de vorige slides kunnen we digitale poortschakelingen vereenvoudigen.
- De eenvoudigste vorm is de vorm die het minst aantal poorten bevat en met het minst aantal ingangen per poort.
- Het vinden van de eenvoudigste vorm heet *minimaliseren*.
- Er zijn drie manieren om te vereenvoudigen:
 - Wiskunde (schakelalgebra)
 - Karnaughdiagrammen (grafisch)
 - Computerprogramma (computer-algoritme: Quine-McCluskey, Espresso)

Vereenvoudigen

- Gegeven onderstaande waarheidstabel. Wat is hiervan de functie en het schema?

- Functie: $y = \bar{a} \cdot b$



a	b	y
0	0	0
0	1	1
1	0	0
1	1	0

- Uitgang $y = 1$ als $a = 0$ en $b = 1$, anders $y = 0$;

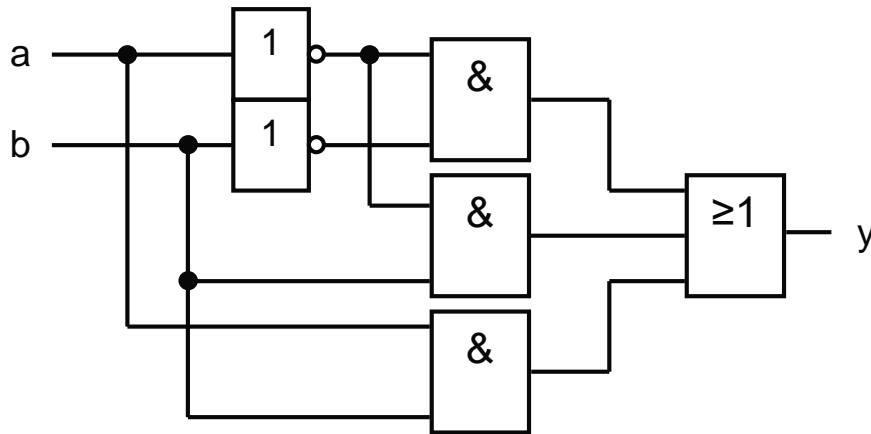
Vereenvoudigen

- Gegeven onderstaande waarheidstabel. Wat is hiervan de functie en het schema?
- Functie: $y = \bar{a} \cdot \bar{b} + \bar{a} \cdot b + a \cdot b$
- Uitgang $y = 1$ als $a = 0$ en $b = 0$
of $a = 0$ en $b = 1$
of $a = 1$ en $b = 1$
anders $y = 0$

a	b	y
0	0	1
0	1	1
1	0	0
1	1	1

Vereenvoudigen

- De schakeling kan als volgt gebouwd worden:



- Kosten: 2x NOT, 3x AND, 1x OR3
- Ingangen: 11

Opmerking: ingangen/uitgangen is bedrading (wires) op een IC.

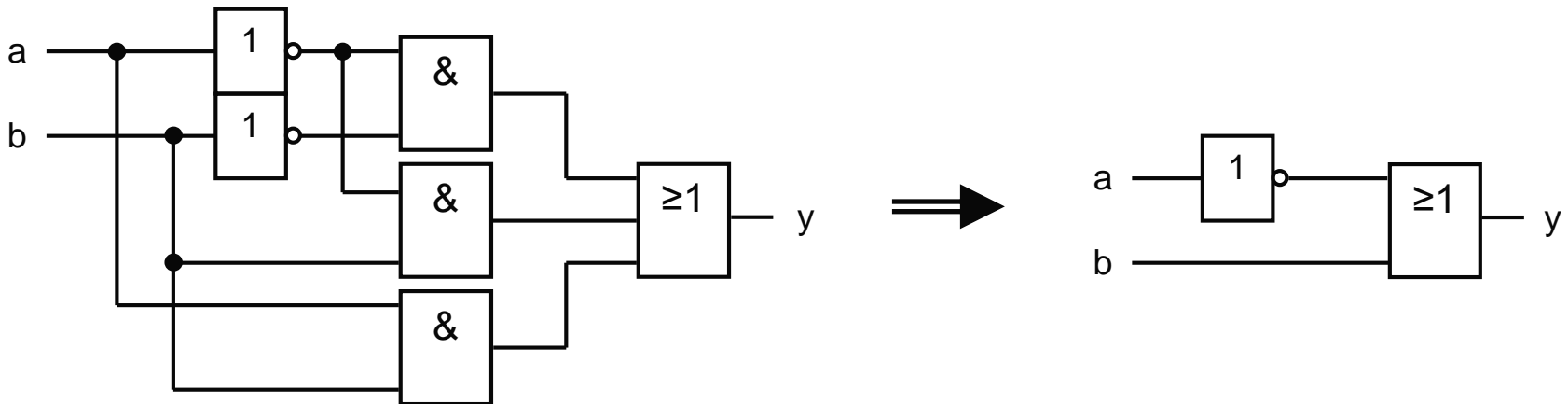
Vereenvoudigen

- Eenvoudig is te zien dat de functie een 1 levert als a 0 is.
- Dus: $\bar{a} \cdot \bar{b} + \bar{a} \cdot b = \bar{a}$
- Ook is snel te zien dat de uitgang een 1 levert als b 1 is.
- Dus: $\bar{a} \cdot b + a \cdot b = b$
- Opvallend: de term $\bar{a} \cdot b$ wordt twee keer gebruikt.

a	b	y
0	0	1
0	1	1
1	0	0
1	1	1

Vereenvoudigen

- De functie is dus als volgt:
$$y = \overline{a} \cdot \overline{b} + \overline{a} \cdot b + a \cdot b$$
$$= \overline{a} + b$$
- Het bijbehorend schema:



Dualiteit

- De definitie van een AND-poort is dat de uitgang een 1 afgeeft als beide ingangen een 1 zijn, anders geeft de uitgang een 0 af.

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

$$y = a \cdot b$$

Dualiteit

- Een alternatief zou zijn om de AND via de 0-en te beschrijven.
- De uitgang geeft een 0 als één of meer ingangen 0 zijn. Vanuit de 0-en gezien is het een OR-functie.
- Hieruit volgt het *dualiteitsprincipe*: bij verwisseling van 0 en 1, en van \cdot en $+$, blijft de gelijkheid intact:

$$0 = 0 \cdot 0$$

$$1 = 1 + 1$$

$$0 = 0 \cdot 1$$

$$1 = 1 + 0$$

$$0 = 1 \cdot 0$$

$$1 = 0 + 1$$

$$1 = 1 \cdot 1$$

$$0 = 0 + 0$$

De Morgan

- Als van een functie alle variabelen worden geïnverteerd, en alle ANDs en ORs worden verwisseld, blijft de functie hetzelfde.

$$y = a \cdot b \quad \Rightarrow \quad \bar{y} = \bar{a} + \bar{b}$$
$$y = a + b \quad \Rightarrow \quad \bar{y} = \bar{a} \cdot \bar{b}$$

- Dit zijn de stellingen van De Morgan.
- Het laat zien dat AND en OR uitwisselbaar zijn.

De Morgan

- Meestal worden de stellingen van De Morgan als volgt gegeven:

$$y = a \cdot b \quad \Leftrightarrow \quad y = \overline{\overline{a + b}}$$

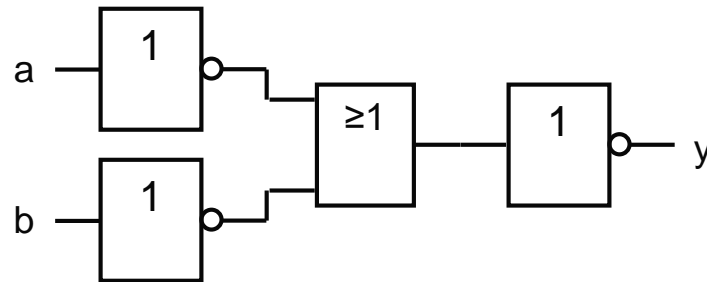
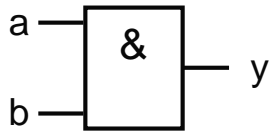
$$y = \overline{a \cdot b} \quad \Leftrightarrow \quad y = \overline{\overline{a} + \overline{b}}$$

$$y = a + b \quad \Leftrightarrow \quad y = \overline{\overline{a \cdot b}}$$

$$y = \overline{a + b} \quad \Leftrightarrow \quad y = \overline{\overline{a} \cdot \overline{b}}$$

De Morgan

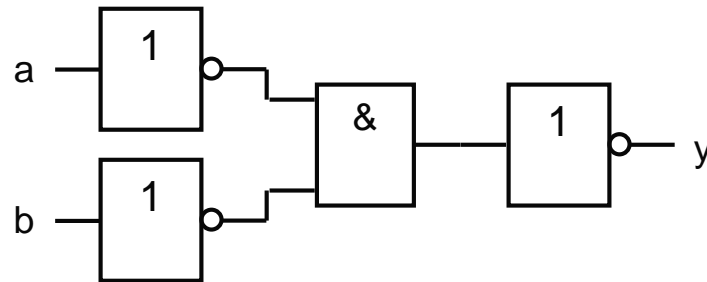
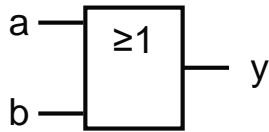
- Een AND-poort is om te zetten in een OR-poort:



- Functie: $y = a \cdot b = \overline{\overline{a} + \overline{b}}$

De Morgan

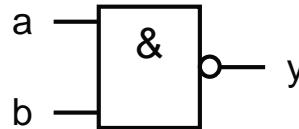
- Een OR-poort is om te zetten in een AND-poort:



- Functie: $y = a + b = \overline{\overline{a} \cdot \overline{b}}$

Universele bouwstenen

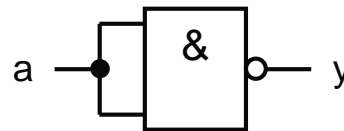
- De NOT-AND of NAND is een universele digitale bouwsteen.
- *Elke* digitale schakeling kan met alleen NANDs worden gemaakt.
- De NOT-OR of NOR is een universele digitale bouwsteen.
- *Elke* digitale schakeling kan met alleen NORs worden gemaakt.
- We bekijken alleen de NAND.



NOT met NAND

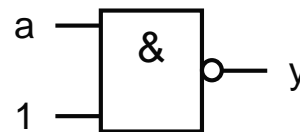
- Een NOT kan op twee manieren worden gebouwd m.b.v. een NAND:

- Ingangen met elkaar verbonden:



$$y = \overline{a \cdot a} = \overline{a}$$

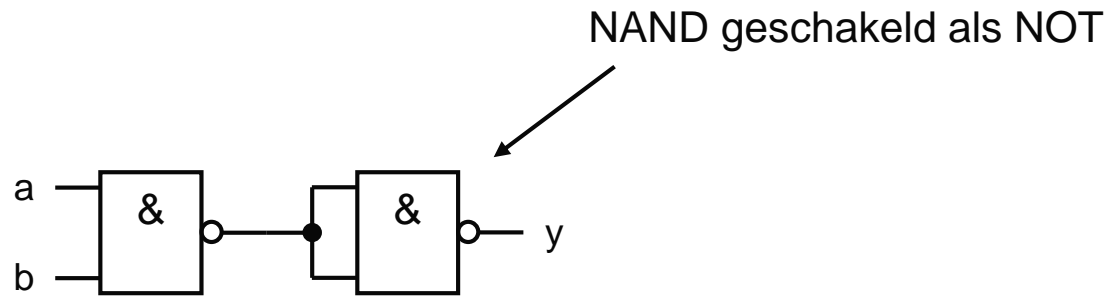
- Eén ingang aan logische 1:



$$y = \overline{a \cdot 1} = \overline{a}$$

AND met NANDs

- Een AND kan met twee NANDs worden gemaakt door één NAND te schakelen als NOT:



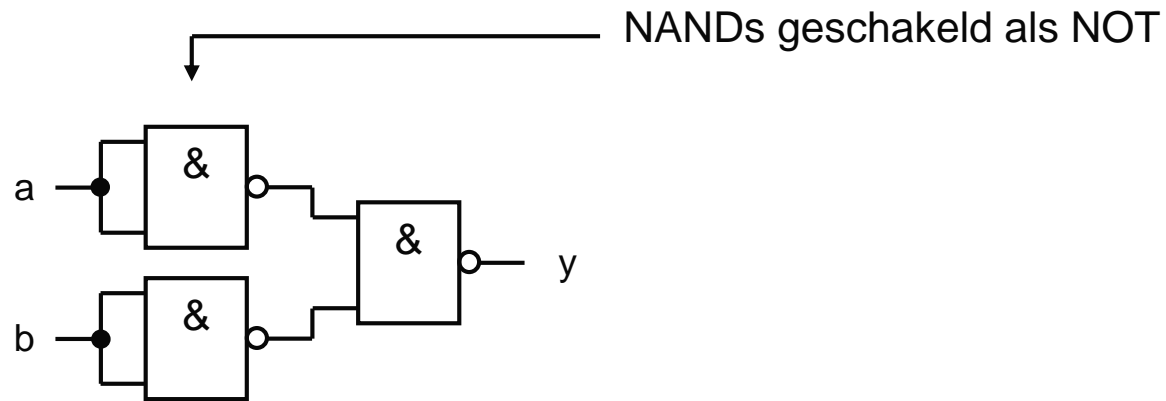
dubbele ontkenning

$$y = \overline{\overline{(a \cdot b)} \cdot \overline{(a \cdot b)}} = \overline{\overline{(a \cdot b)}} = a \cdot b$$

samennemen

OR met NANDs

- Een OR kan met drie NANDs worden gemaakt door twee NANDs als NOT te schakelen:

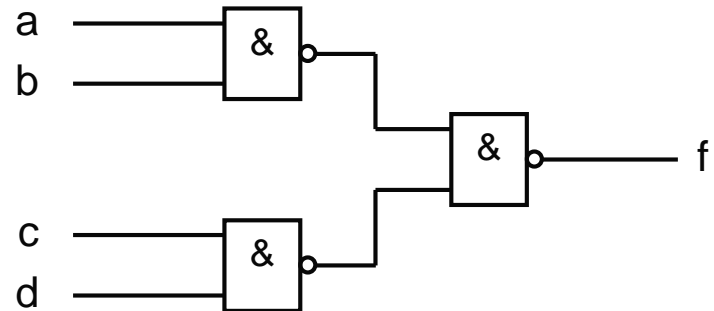
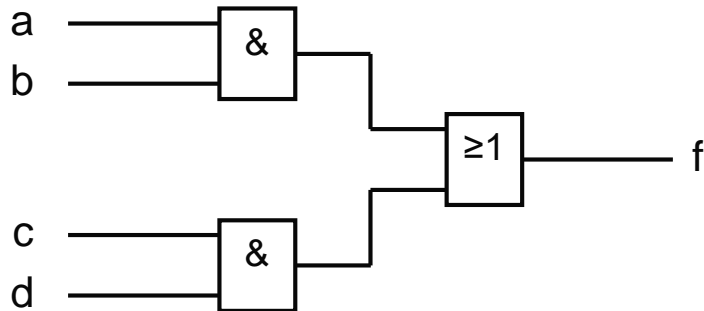


$$y = \overline{\underbrace{(a \cdot a)}_{\text{samennemen}} \cdot \underbrace{(b \cdot b)}_{\text{samennemen}}} = \overline{\overline{a} \cdot \overline{b}} = a + b$$

De Morgan

Functie met NANDs

- Elke functie in de vorm van $f = a \cdot b + c \cdot d$ kan worden omgezet naar een functie met alleen NANDs.
- Dus: $f = a \cdot b + c \cdot d \rightarrow f = \overline{\overline{a \cdot b} \cdot \overline{c \cdot d}}$
- Dit betekent dat een schakeling bestaande uit AND-OR kan worden omgezet naar een schakeling met alleen NANDs.



Literatuur

- De volgende boeken zijn gebruikt:

Digitale techniek, van probleemstelling tot realisatie deel 1; A.P. Thijssen; 5^e druk.

Digital Design, Principles and Practices; J.F. Wakery; 3rd Ed.

Contemporary Logic Design; R.H. Katz; 2nd Ed.

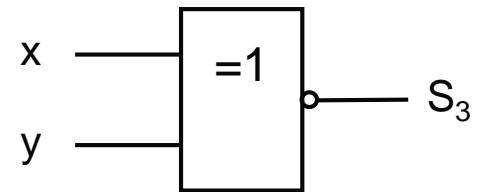
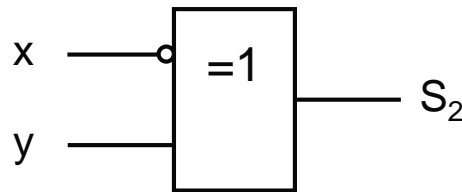
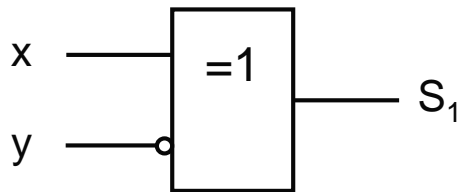
- Dictaat Digitale Techniek, hoofdstuk 1.

EXOR en EXNOR

- Een EXOR met één geïnverteerde ingang is identiek aan een EXNOR:

$$S = x \oplus \bar{y} = \bar{x} \oplus y = \overline{x \oplus y}$$

- De onderstaande schakelingen zijn dus logisch identiek:



EXOR en EXNOR

- Het is ook aan te tonen. Er worden drie functies S_1 , S_2 en S_3 uitgewerkt die alle dezelfde uitkomst geven.

$$S_1 = x \oplus \bar{y} = x \cdot \bar{\bar{y}} + \bar{x} \cdot \bar{y} = x \cdot y + \bar{x} \cdot \bar{y}$$

$$S_2 = \bar{x} \oplus y = \bar{x} \cdot \bar{\bar{y}} + \bar{\bar{x}} \cdot y = \bar{x} \cdot \bar{y} + x \cdot y = x \cdot y + \bar{x} \cdot \bar{y}$$

$$\begin{aligned} S_3 &= \overline{x \oplus y} = \overline{x \cdot \bar{y} + \bar{x} \cdot y} = \overline{x \cdot \bar{y}} \cdot \overline{\bar{x} \cdot y} = (\bar{x} + y) \cdot (x + \bar{y}) \\ &= \bar{x} \cdot x + \bar{x} \cdot \bar{y} + y \cdot x + y \cdot \bar{y} = 0 + \bar{x} \cdot \bar{y} + x \cdot y + 0 \\ &= x \cdot y + \bar{x} \cdot \bar{y} \end{aligned}$$



Academie voor Technology, Innovation &
Society Delft
Academie voor ICT & Media

De Haagse Hogeschool, Delft
+31-15-2606311
J.E.J.opdenBrouw@hhs.nl
www.dehaagsehogeschool.nl

DE HAAGSE
HOGESCHOOL