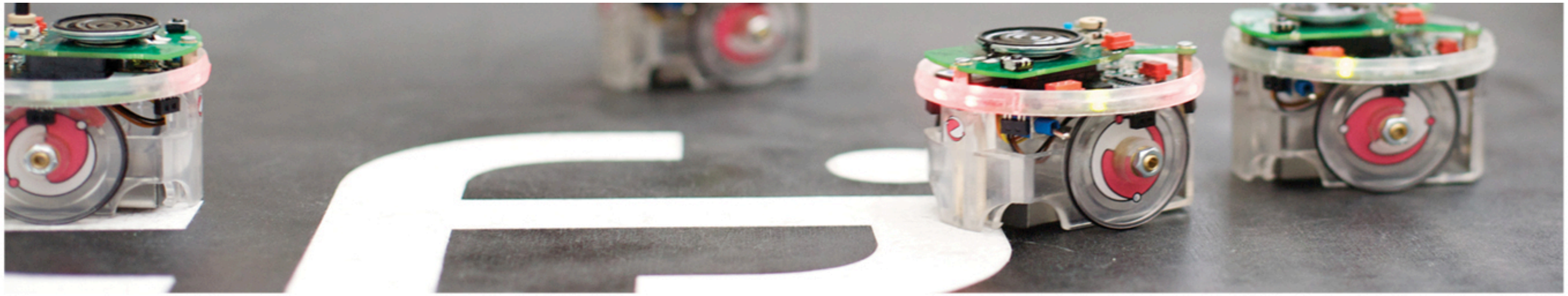


UiO : Institutt for informatikk
Det matematisk-naturvitenskapelige fakultet

Boolsk Algebra

Hva gikk vi gjennom forrige uke?

Læringsutbytte

- Kunnskapsmål:
  - Kunnskap om boolsk algebra

- Ferdighetsmål:
  - Kunne forenkle boolske uttrykk
  - Kunne implementere flerinputs-porter med bare 2-inputs porter

- Generelle kompetansemål:
  - Kunne uttrykke boolske funksjoner
  - Kjenne til porter som AND, OR, NOT, BUFFER, NAND, NOR, XOR

Hovedpunkter

- Toverdi boolsk algebra
- Huntington´s postulater
- Diverse teorem
- Boolske funksjoner med sannhetstabell
- Forenkling av uttryk
- Maksterm / minterm
- Flerinputsporter
- Designprosedyre med eksempler

Motivasjon

Design/analyse av digital logikk

Eksempler:

- Styring av heis
  - Input: bryter i dør (0/1) / vektsensor (0/1)
  - Output: styresignal til motor (0/1)

- Visualisering av tall på 7-segments display
  - Input: 3 bits binært signal (tall)
  - Output: “desimalt” lys (7 lamper)

Hvordan klarer vi det?

- Vi må bruke en “skreddersydd” algebra

Algebra - definisjon

- Algebra – Studie av algebraiske system
- Algebraisk system:
  - Mengde av elementer
  - Et sett av operasjoner på elementene (funksjoner) som tilfredsstiler en liste av aksiomer/postulater
- Aksiom/postulat – Regler som ikke skal bevises (utgangspunkt for algebraisk system)
- Eksempler på algebraiske system:
  - "Vanlig" algebra
  - Boolsk algebra
  - Matrise algebra

Hvorfor trenger vi Boolsk algebra?

- Muliggjør design av komplekse digitale systemer
- Muliggjør analyse av komplekse digitale systemer
- Forenkling av logiske uttrykk:
  - Gir enklere fysisk implementasjon

"Toverdi" Boolsk algebra

Definisjon:

- Mengde:

  {0, 1}

- 2 operasjoner på mengden over:

  OR "+"

  AND "•"

AND

x	y	x•y
0	0	0
0	1	0
1	0	0
1	1	1

OR

x	y	x+y
0	0	0
0	1	1
1	0	1
1	1	1

Husk: +/• har ingenting med addisjon og multiplikasjon å gjøre

Huntington postulater

(P0)	Mengden {0,1} er lukket under "+" og "•"		lukket
(P1)	$x + x = x$	$x \bullet x = x$	
(P2)	$x + 0 = x$	$x \bullet 1 = x$	ident.el.
(P2b)	$x + 1 = 1$	$x \bullet 0 = 0$	
(P5)	$x + x' = 1$	$x \bullet x' = 0$	komplem.
(P6)	$0 \neq 1$		minst 2 el.
(P3)	$x + y = y + x$	$x \bullet y = y \bullet x$	kommutativ
(P4)	$x \bullet (y + z) = x \bullet y + x \bullet z$	$x+(y \bullet z) = (x+y) \bullet (x+z)$	distributiv
(P5)	$(x')' = x$		

Dualitet for postulatene:

Presedens:

.

Basis-byggeblokker

AND

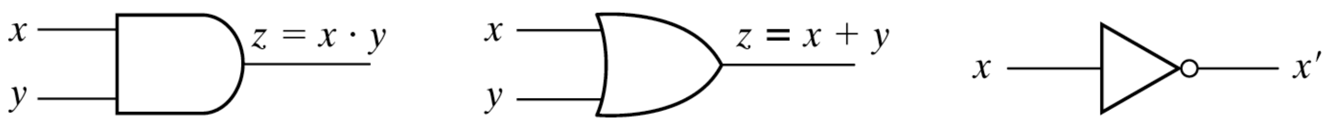
X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

OR

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

NOT

X	Y
0	1
1	0



(a) Two-input AND gate (b) Two-input OR gate (c) NOT gate or inverter

Teorem 1a $\quad x + x = x$

Bevis:

(P2)	$x + 0 = x$	$x \bullet 1 = x$
(P5)	$x + x' = 1$	$x \bullet x' = 0$
(P4)		$x+(y \bullet z) = (x+y) \bullet (x+z)$

Teorem 1b $\quad x \bullet x = x$

Bevis:

$$x \bullet x = x \bullet x + 0$$
$$x \bullet x = x \bullet x + x \bullet x'$$
$$x \bullet x = x \bullet (x + x')$$
$$x \bullet x = x \bullet 1$$
$$x \bullet x = x$$

(P2) $x + 0 = x \qquad x \bullet 1 = x$

(P5) $x + x' = 1 \qquad x \bullet x' = 0$

(P4) $x \bullet (y+z) = (x \bullet y) + (x \bullet z)$

Teorem 2a: $x + 1 = 1$

Bevis:		
	$x + 1 = 1 \bullet (x + 1)$	P2 / P3
	$x + 1 = (x + x') \bullet (x + 1)$	P5
	$x + 1 = x + (x' \bullet 1)$	P4
	$x + 1 = x + x'$	P2
	$x + 1 = 1$	P5

(P2)		$x \bullet 1 = x$
(P3)		$x \bullet y = y \bullet x$
(P5)	$x + x' = 1$	
(P4)		$x+(y \bullet z) = (x+y) \bullet (x+z)$

Flere teorem

Dualitet for ligninger utledet fra postulat:

kan bytte "• " med "+" hvis man bytter "0" med "1"

Teorem 2b: $x \bullet 0 = 0$

Bevis: $x + 1 = 1$ (teorem 2a, samt dualitet)

DeMorgans teorem

$$(x \bullet y)' =$$

$$(x + y)' =$$

På invertert form:

.

Regneregler - oversikt

$x + 0 = x$	$x \bullet 1 = x$
$x + x' = 1$	$xx' = 0$
$x + y = y+x$	$xy = yx$
$x + (y+z) = (x+y) + z$	$x\,(yz) = (xy)z$
$x(y+z) = xy + xz$	$x + (yz) = (x+y)(x+z)$
$x + x = x$	$x \bullet x = x$
$x + 1 = 1$	$x \bullet 0 = 0$
$x + xy = x$	$x(x+y) = x$
$(x+y)' = x'y'$	$(xy)' = x' + y'$

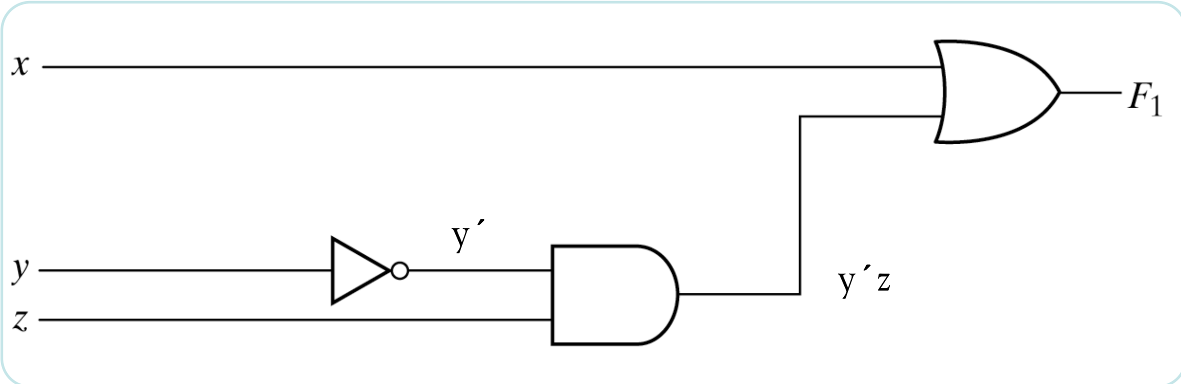
Boolske funksjoner

Tilordner en binær variabel en verdi bestemt av verdien på en eller flere andre binære variabler

Eksempel:

F = x + y´z

Direkte port-implementasjon:



Sannhetstabell

En boolsk funksjon kan visualiseres i en sannhetstabell

Eksempel:

$$F = x + y'z$$

x y z	F
0 0 0	0
0 0 1	1
0 1 0	0
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

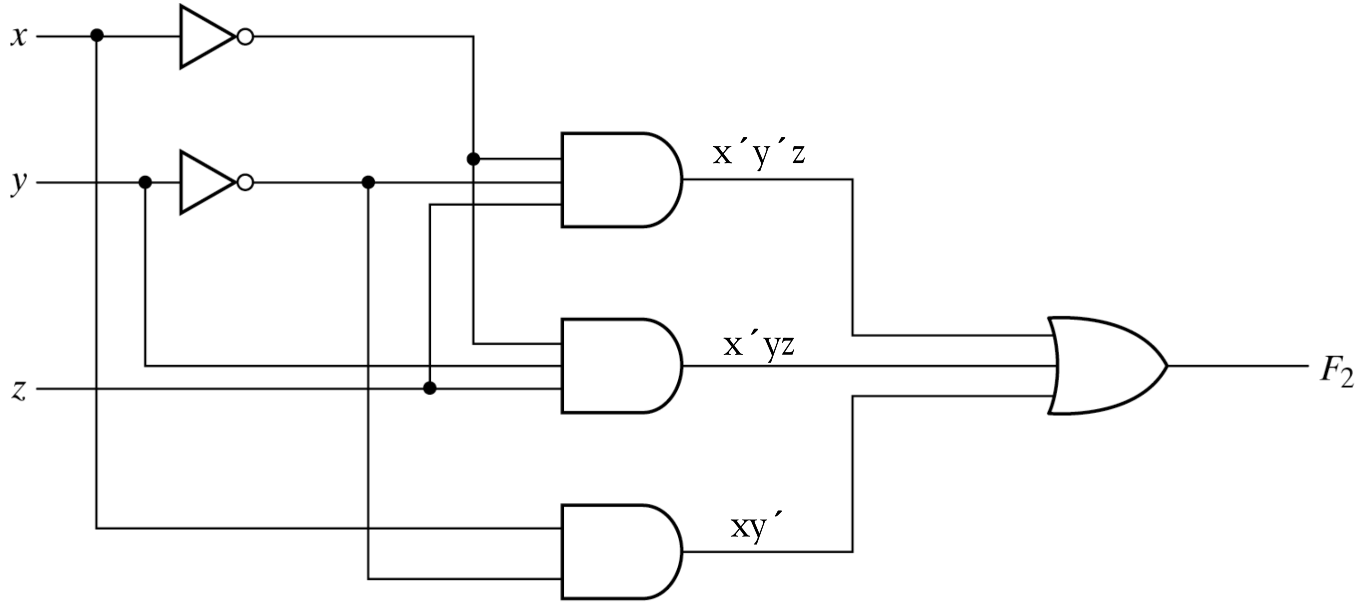
En gitt funksjon har *kun* en sannhetstabell

Men, en gitt sannhetstabell har uendelig mange funksjonsuttrykk

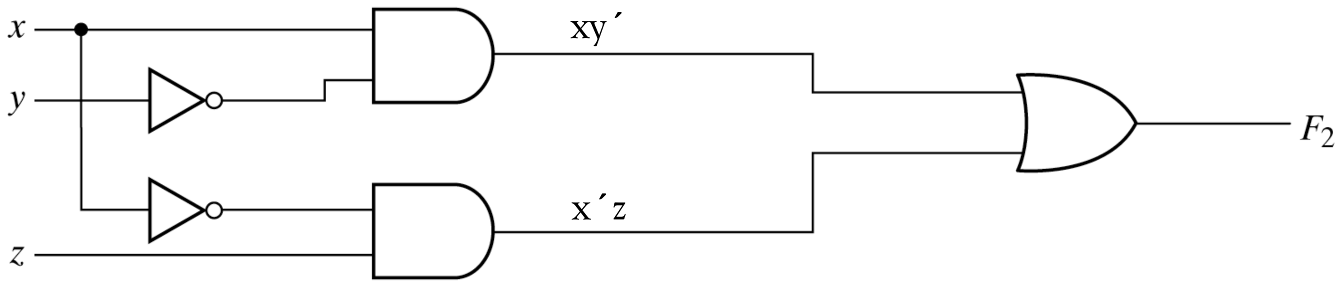
Forenkling av uttrykk

$$F = x'y'z + x'yz + xy'$$

Prosedyre:



(a) $F_2 = x'y'z + x'yz + xy'$



(b) $F_2 = xy' + x'z$

Forenklingseksempler

Eksempel:

$F = x(x'+y)$

Eksempel:

$F = x+x'y$

Eksempel:

$F = (x+y)(x+y')$

Forenklingseksempler II

Eksempel:

$F = xy+x'z+yz$

Eksempel:

$F = (x+y)(x'+z)(y+z)$

Komplement av funksjon

Inverterer begge sider og bruker DeMorgan

Eksempel:

$F = x'yz'+x'y'z$

Eksempel:

$F = x(y'z'+yz)$

Minterm

I en funksjon kan en binær variabel x opptre som x eller x´

En funksjon kan være gitt på ***"sum av produkt"*** form

Eksempel:

$F = xy + xy' + x$

Hvert "produktledd" som inneholder **alle** variablene kalles en **minterm**.

For to variable finnes det 4 forskjellige mintermer:

$xy + xy' + x'y + x'y'$

For 3 variable finnes det 2^3 forskjellige mintermer

Maksterm

En funksjon kan være gitt på “produkt av sum” form

Eksempel:

$$F = (x+y)(x+y')y$$

Hvert “summeledd” som inneholder alle variablene kalles en **maksterm**

For to variable finnes det 4 forskjellige makstermer:

$$(x+y)\ (x+y')\ (x'+y)\ (x'+y')$$

For n variable finnes det 2^n forskjellige makstermer

Sannhetstabell / mintermer

Hvis man genererer en funksjon ut i fra sannhetstabellen får man en sum av mintermer

<u>Eksempel:</u>

$$F = x'y'z + xy'z' + xyz'$$

x	y	z	F
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

En sannhetstabell kan sees på som en liste av mintermer

Notasjon

Mintermer har notasjon m_x

Maxtermer har notasjon M_x

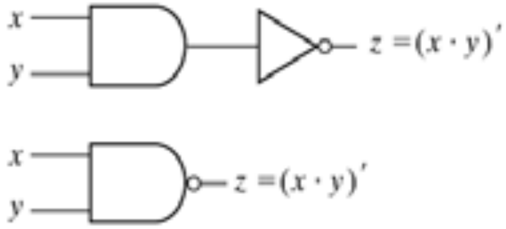
Funksjoner som bare består av sum/produkt av min/maxtermer (kanonisk form) har følengde notasjon:

$F(x,y,z) =$

$F(x,y,z) =$

			Minterms		Maxterms	
x	y	z	Term	Designation	Term	Designation
0	0	0	$x'y'z'$	m_0	$x + y + z$	M_0
0	0	1	$x'y'z$	m_1	$x + y + z'$	M_1
0	1	0	$x'yz'$	m_2	$x + y' + z$	M_2
0	1	1	$x'yz$	m_3	$x + y' + z'$	M_3
1	0	0	$xy'z'$	m_4	$x' + y + z$	M_4
1	0	1	$xy'z$	m_5	$x' + y + z'$	M_5
1	1	0	xyz'	m_6	$x' + y' + z$	M_6
1	1	1	xyz	m_7	$x' + y' + z'$	M_7

NAND / NOR



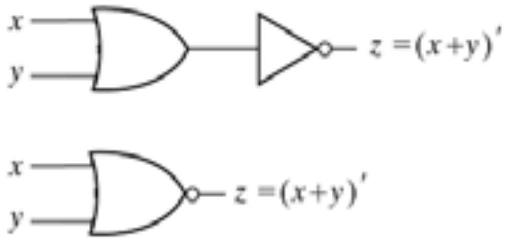
(a) Two-input NAND gate

NAND

X Y	Z
0 0	1
0 1	1
1 0	1
1 1	0

NOR

X Y	Z
0 0	1
0 1	0
1 0	0
1 1	0

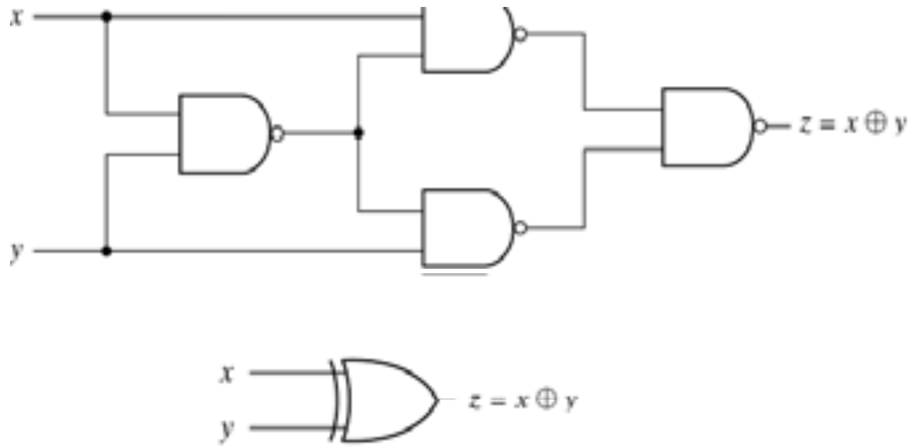


(a) Two-input NOR gate

XOR

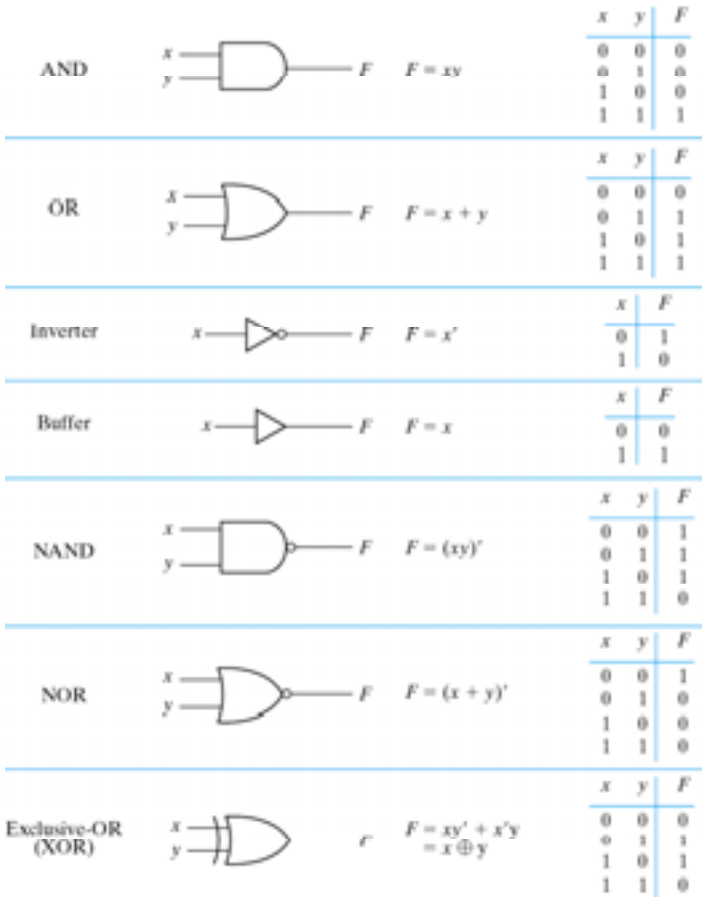
XOR

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0



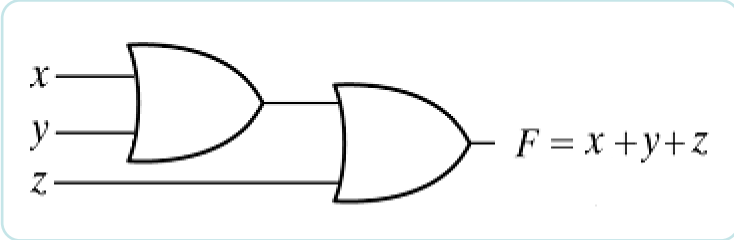
Exclusive-OR Implementations

2-inputs byggeblokker oversikt



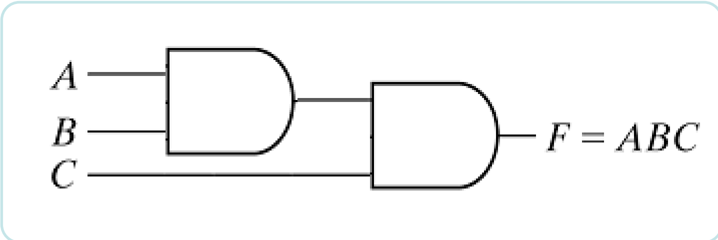
Flerinputs-porter

Flerinputs-
implementasjon
av AND / OR
kan lages direkte



OR

x y z	F
0 0 0	0
0 0 1	1
0 1 0	1
0 1 1	1
1 0 0	1
1 0 1	1
1 1 0	1
1 1 1	1

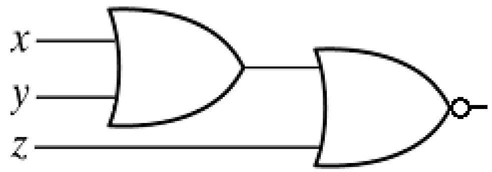


AND

A B C	F
0 0 0	0
0 0 1	0
0 1 0	0
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	0
1 1 1	1

Flerinputs-porter NAND/NOR

Flerinputs-
implementasjon
av NAND / NOR
kan <u>ikke</u> lages direkte

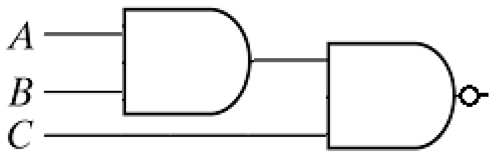


F = (x+y+z) ´

Mulig løsning

NOR

x	y	z	F
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0



F = (ABC)´

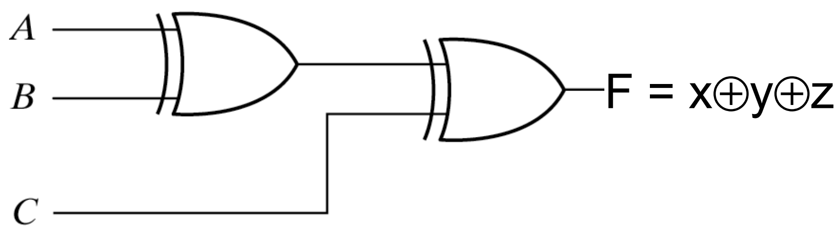
Mulig løsning

NAND

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Flerinputs-porter

Flerinputs implementasjon av XOR (oddefunksjon) kan lages direkte



(a) 3-input odd function

XOR

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Designeksempel 1

- Styring av heis

- Inngangssignaler fra sensor
  - K = Knapp trykket inn: 0/1
  - V = Overvekt: 0/1
  - D = Dør lukket: 0/1

- Utgangssignal til aktuator:
  - M = Motor på: 0/1

Sannhetstabell for designet

K	V	D	M
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Implementasjon på portnivå

Implementasjon kun med 2-inputs NAND

Bruk av 2-inputs NAND og NOR porter

Forenkling på portnivå

- Det er ikke alltid at det enkleste funksjonsuttrykket resulterer i den enkleste port-implementasjonen

- Ved forenkling på portnivå må man vite hvilke porter man har til rådighet, og så justere funksjonsuttrykket mot dette. (håndverk)

Generell design prosedyre

1. Bestem hvilke signal som er innganger og utganger

2. Sett opp sannhetstabell for alle inngangskombinasjoner

3. Generer funksjonsuttrykket som sum av mintermer

4. Tilpass / forenkle funksjonsuttrykket mot aktuelle porter