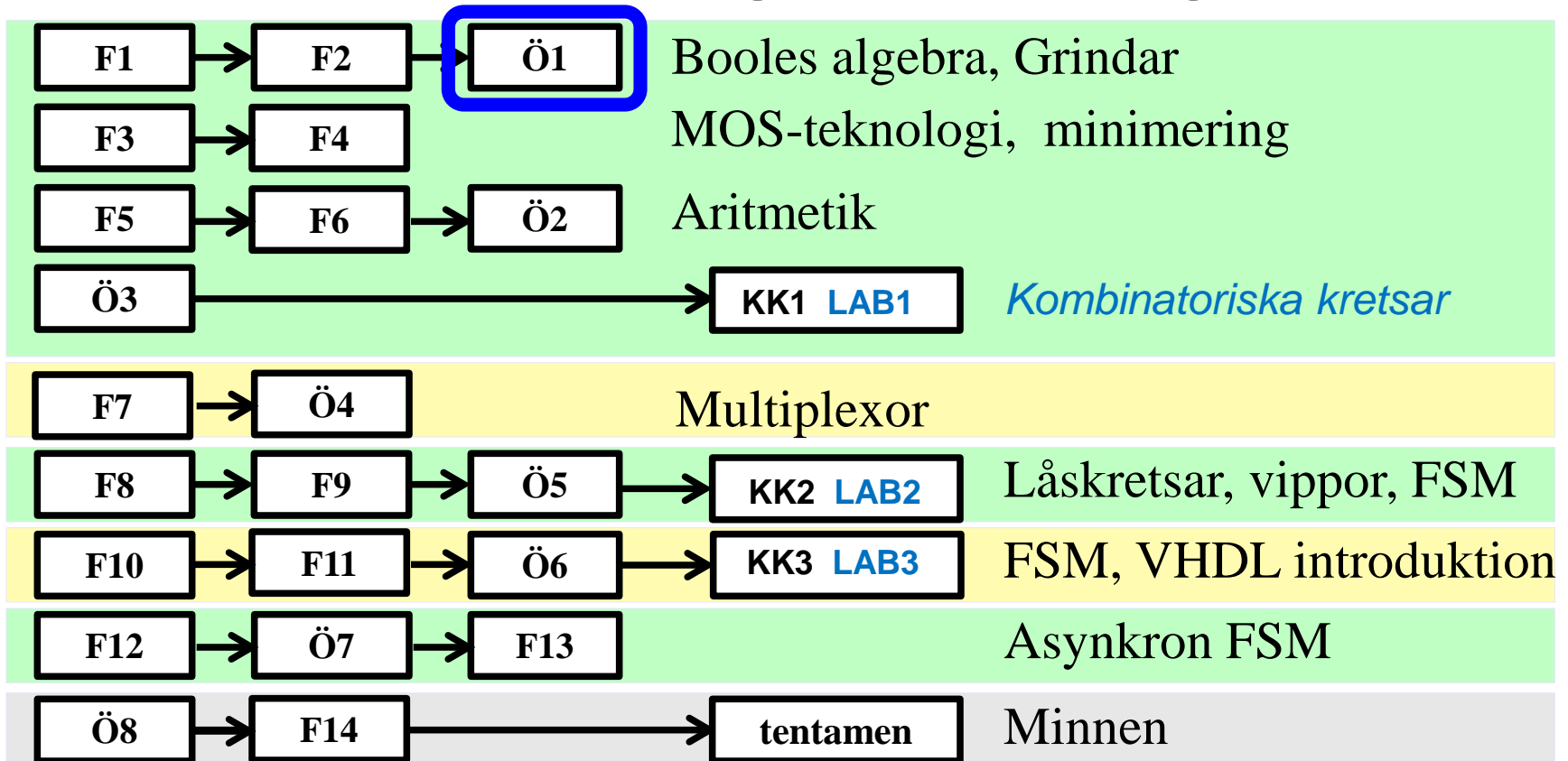
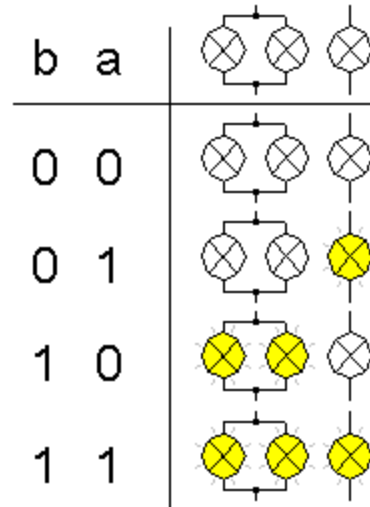
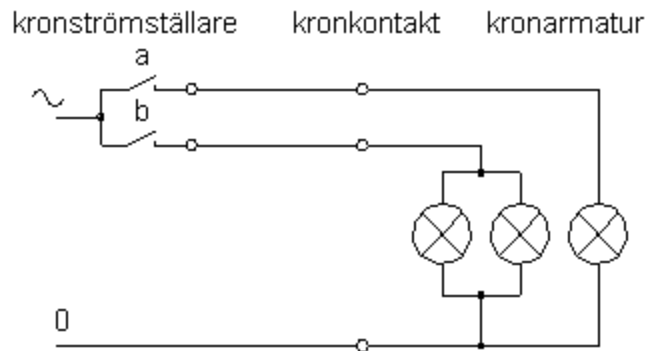


# IE1204 Digital Design

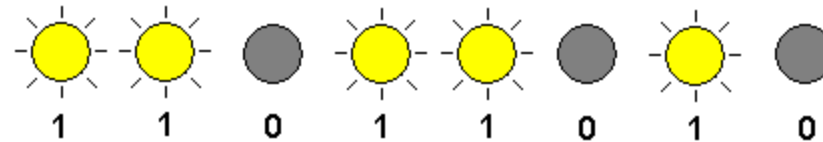


*Föreläsningar och övningar bygger på varandra! Ta alltid igen det Du missat!  
Läs på i förväg – delta i undervisningen – arbeta igenom materialet efteråt!*

# Kronljusströmställaren 0, 1, 2, 3



# Styr med binärkod



Lamptabla som visar 8-bitarstal, Byte

Bin → Dec

$$\begin{array}{cccccccc} 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ 128 & 64 & 32 & 16 & 8 & 4 & 2 & 1 \\ 128+64+0+16+8+0+2+0=218 \end{array}$$

Hex → Dec

$$\begin{array}{cc} 1101 & 1010 \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ D & A \\ 16^1 & 16^0 \\ 16 & 1 \\ 13 \cdot 16 + 10 \cdot 1 = 218 \end{array}$$

# Dec – Bin – Hex – Okt

Dec	Bin	Hex	Dec	Bin	Hex
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

**11011010**

**D      A                      3    3    2**  
  
**1101 1010                      11 011 010**

$$218_{10} = 11011010_2 = DA_{16} = 332_8$$

# ÖH 1.1c Decimaltal till Binärtal

binära vikter: 1024 512 256 128 64 32 16 8 4 2 1

$$71_{10} = ?_2$$

# ÖH 1.1c Decimaltal till Binärtal

binära vikter: 1024 512 256 128 64 32 16 8 4 2 1

$$71_{10} = ?_2$$

$$71_{10} = \\ (64+7 = 64+4+2+1) = 1000111_2$$

# ÖH 1.2a Binärtal till Decimaltal

binära vikter: 1024 512 256 128 64 32 16 8 4 2 1

$$101101001_2 = ?_{10}$$

# ÖH 1.2a Binärtal till Decimaltal

binära vikter: 1024 512 256 128 64 32 16 8 4 2 1

$$101101001_2 = ?_{10}$$

$$101101001_2 =$$

$$(2^8 + 2^6 + 2^5 + 2^3 + 2^0 = 256 + 64 + 32 + 8 + 1)$$

$$= 361_{10}$$



# ÖH 1.3c

## Binärt/Oktalt/Hexadecimalt

$$100110101_2 = ?_{16} = ?_8$$

# ÖH 1.3c

## Binärt/Oktalt/Hexadecimalt

$$100110101_2 = ?_{16} = ?_8$$

$$1\ 0011\ 0101_2 = 1\ 3\ 5_{16}$$

# ÖH 1.3c

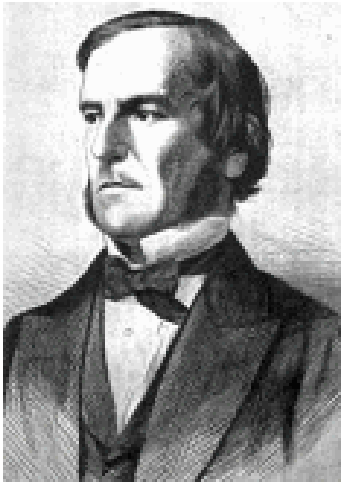
## Binärt/Oktalt/Hexadecimalt

$$100110101_2 = ?_{16} = ?_8$$

$$1\ 0011\ 0101_2 = 1\ 3\ 5_{16}$$

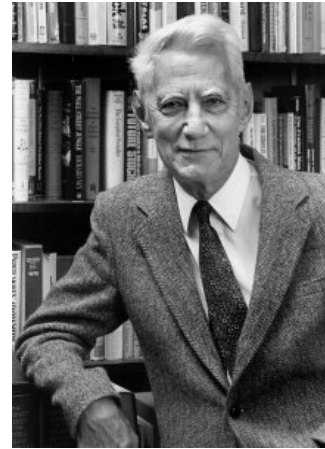
$$100\ 110\ 101_2 = 4\ 6\ 5_8$$

# Booles Algebra



George Boole  
matematiker  
(1815-1864)

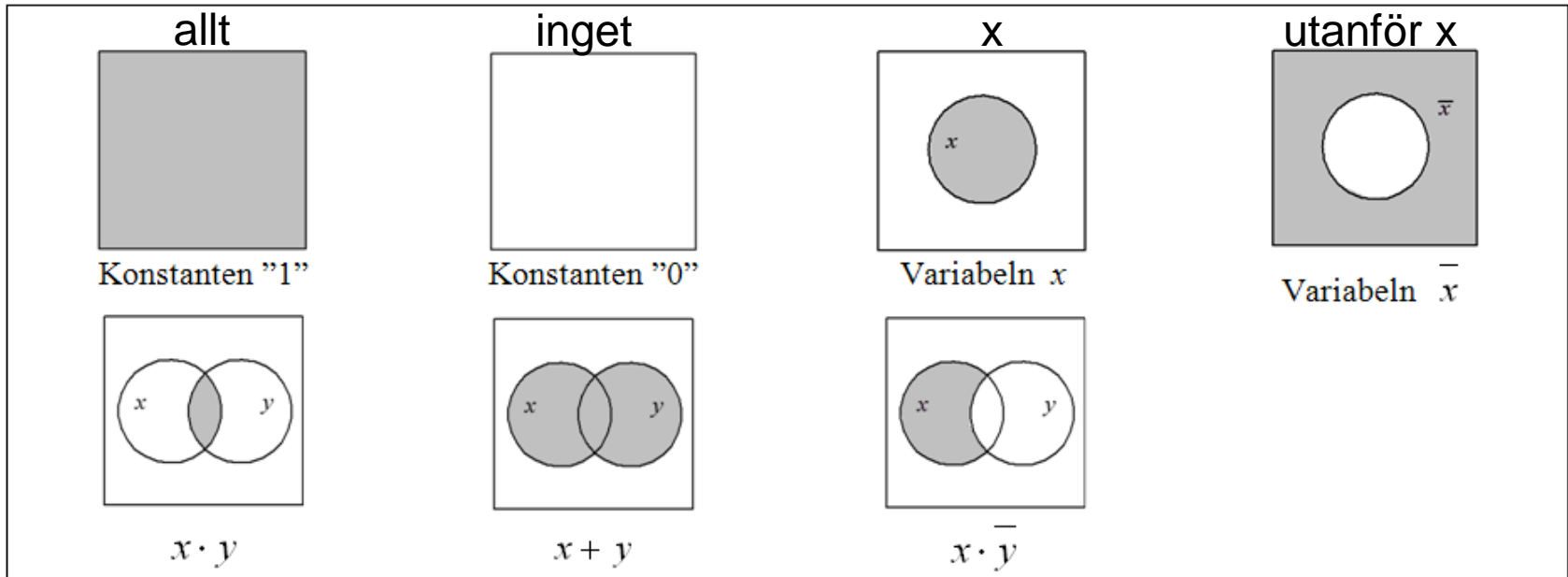
Genom att representera logiska uttryck på matematisk form, där sammanfogningsorden OR och AND motsvarade ett slags addition och multiplikation, blev det möjligt att med en algebra undersöka om komplicerade logiska utsagor och resonemang, i slutändan var sanna eller falska.



Claude Shannon  
matematiker/elektrotekniker  
(1916 –2001)

1938 "dammade" Claude Shannon av algebran och använde den till elektriska kontaktnät. Sedan dess är Booles algebra det huvudsakliga verktyget för all digital konstruktion.

# Venn-diagram



x gemensamt med y   x tillsammans med y   x tillsammans med utanför y

# ÖH 3.2 De Morgans lag med Venn-diagram

Bevisa De Morgans lag med hjälp av Venn-diagram.

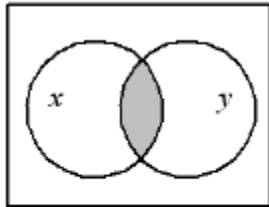
$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

# ÖH 3.2 De Morgan

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

# ÖH 3.2 De Morgan

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

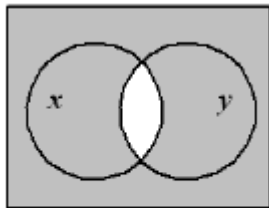


$x \cdot y$



# ÖH 3.2 De Morgan

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

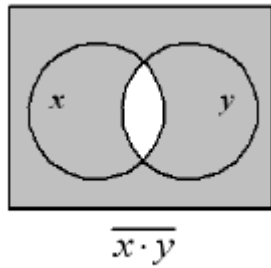


=

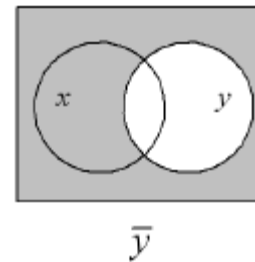
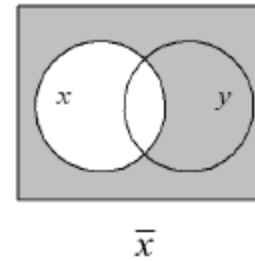
$\overline{x \cdot y}$

# ÖH 3.2 De Morgan

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$

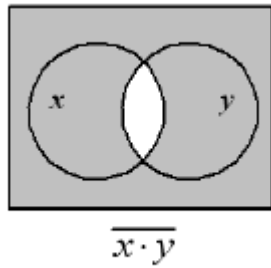


=

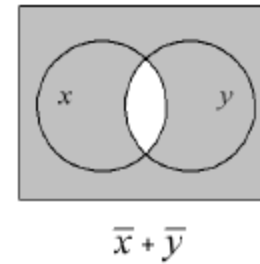


# ÖH 3.2 De Morgan

$$\overline{x \cdot y} = \bar{x} + \bar{y}$$



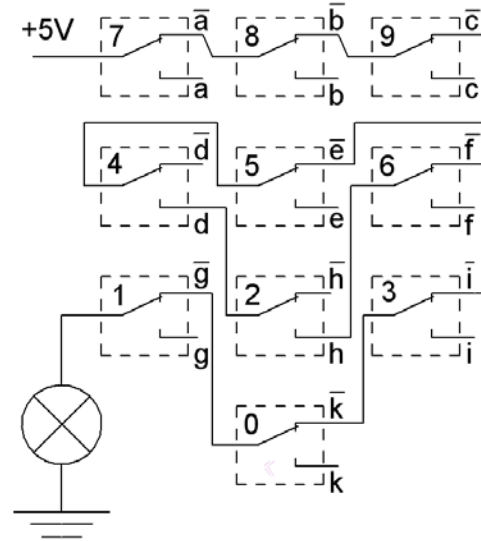
=



***Bevisat!***

# (ÖH 5.1) Hur öppnar man kodlåset? (=minterm)

Vilka knappar ska man samtidigt trycka på för att tända lampan? (= öppna kodlåset)

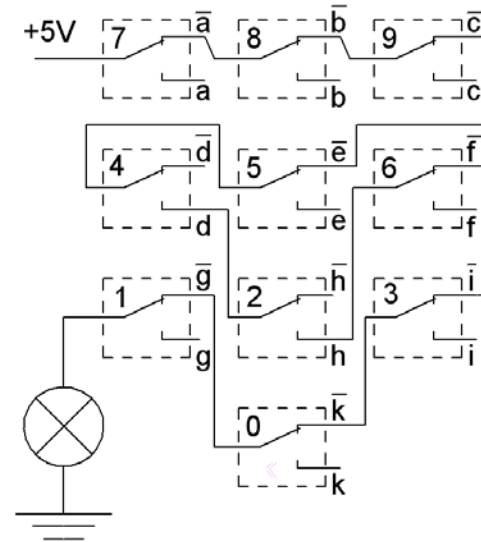


# (ÖH 5.1) Hur öppnar man kodlåset? (=minterm)



Vilka knappar ska man samtidigt trycka på för att tända lampan? (= öppna kodlåset)

**Svar:** 4, d och 2, h men samtidigt måste man *undvika* att trycka på a b c e f g i och k!

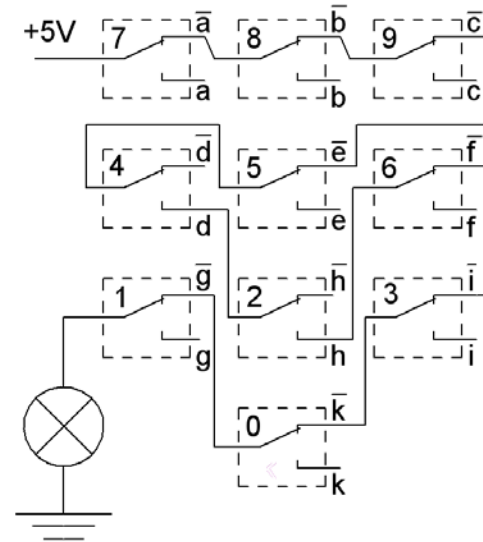


# (ÖH 5.1) Hur öppnar man kodlåset? (=minterm)



Vilka knappar ska man samtidigt trycka på för att tända lampan? (= öppna kodlåset)

**Svar:** 4, d och 2, h men samtidigt måste man *undvika* att trycka på a b c e f g i och k!



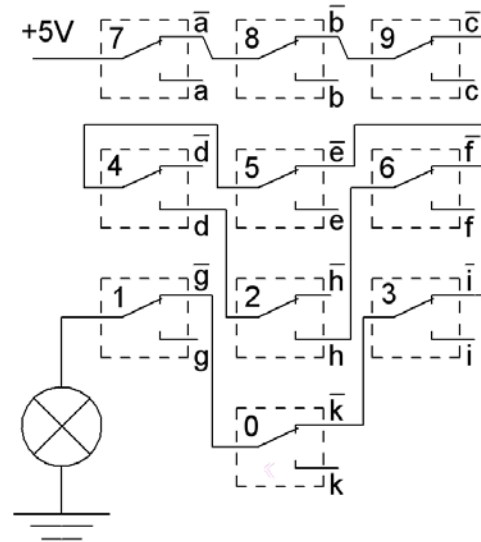
$$T = \bar{a} \cdot \bar{b} \cdot \bar{c} \cdot d \cdot \bar{e} \cdot \bar{f} \cdot \bar{g} \cdot h \cdot \bar{i} \cdot \bar{k}$$

# (ÖH 5.1) Hur öppnar man kodlåset? (=minterm)



Vilka knappar ska man samtidigt trycka på för att tända lampan? (= öppna kodlåset)

**Svar:** 4, d och 2, h men samtidigt måste man *undvika* att trycka på a b c e f g i och k!



$$T = \bar{a} \cdot \bar{b} \cdot \bar{c} \cdot \bar{d} \cdot \bar{e} \cdot \bar{f} \cdot \bar{g} \cdot \bar{h} \cdot \bar{i} \cdot \bar{k}$$

En produktterm där *alla* variabler ingår kallas för en **minterm**.

# ÖH 3.3 Venndiagram

## 3.3

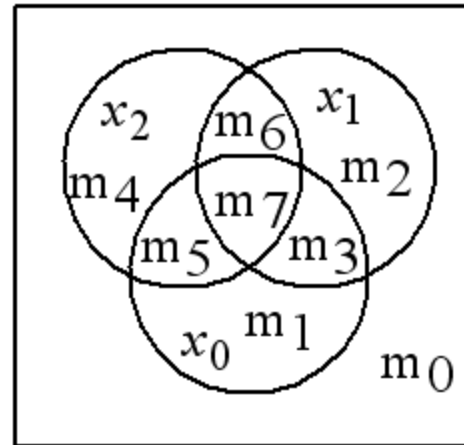
- Rita ett Venn-diagram för tre variabler och markera var sanningstabellens alla mintermer är placerade.
- Minimera funktionen med hjälp av Venn-diagram.

$$f = \overline{x_2} \overline{x_1} x_0 + \overline{x_2} x_1 x_0 + x_2 \overline{x_1} x_0 + x_2 x_1 \overline{x_0} + x_2 x_1 x_0$$



# ÖH 3.3a Sanningstabell - Venndiagram

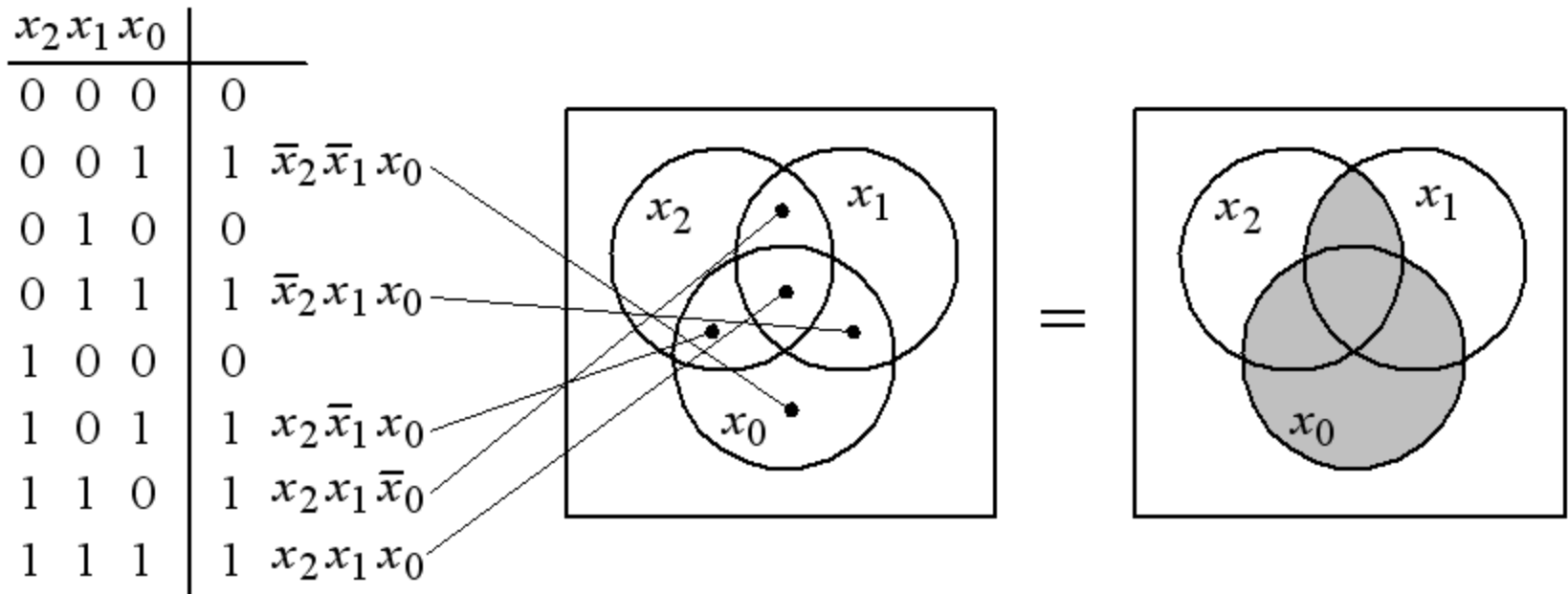
$x_2$	$x_1$	$x_0$	
0	0	0	$m_0$
0	0	1	$m_1$
0	1	0	$m_2$
0	1	1	$m_3$
1	0	0	$m_4$
1	0	1	$m_5$
1	1	0	$m_6$
1	1	1	$m_7$



# ÖH 3.3b förenklat uttryck

*Ursprungligt uttryck.*

$$f = \bar{x}_2 \bar{x}_1 x_0 + \bar{x}_2 x_1 x_0 + x_2 \bar{x}_1 x_0 + x_2 x_1 \bar{x}_0 + x_2 x_1 x_0$$

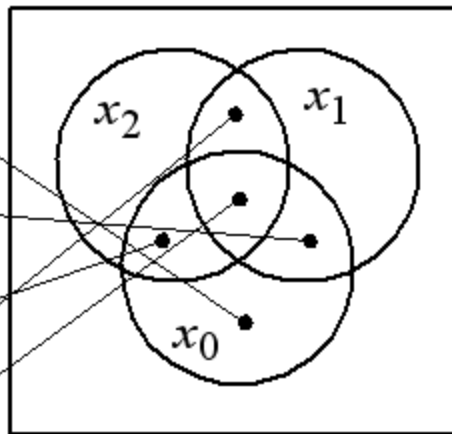


# ÖH 3.3b förenklat uttryck

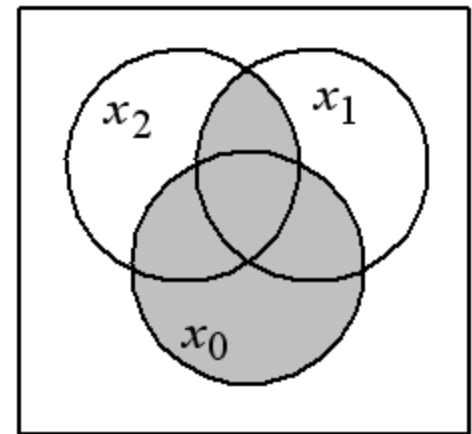
*Ursprungligt uttryck.*

$$f = \bar{x}_2 \bar{x}_1 x_0 + \bar{x}_2 x_1 x_0 + x_2 \bar{x}_1 x_0 + x_2 x_1 \bar{x}_0 + x_2 x_1 x_0$$

$x_2 x_1 x_0$	
0 0 0	0
0 0 1	1 $\bar{x}_2 \bar{x}_1 x_0$
0 1 0	0
0 1 1	1 $\bar{x}_2 x_1 x_0$
1 0 0	0
1 0 1	1 $x_2 \bar{x}_1 x_0$
1 1 0	1 $x_2 x_1 \bar{x}_0$
1 1 1	1 $x_2 x_1 x_0$



=



$$f = x_2 x_1 + x_0$$

**Förenklat!**

# Booles algebra räknelagar

Logisk addition "+", **OR**, och logisk multiplikation ".", **AND**, följer i stort sätt de vanliga normala algebraiska distributiva, kommutativa och associativa lagarna (med ett "udda" undantag).

<b>Distributiva lagarna</b>	$A \cdot (B + C) = A \cdot B + A \cdot C$ $A + (B \cdot C) = (A + B) \cdot (A + C)$ <i>Undantaget!</i>
<b>Kommutativa lagarna</b>	$A \cdot B = B \cdot A$ $A + B = B + A$
<b>Associativa lagarna</b>	$(A \cdot B) \cdot C = A \cdot (B \cdot C)$ $(A + B) + C = A + (B + C)$

# Förenklingsregler och teorem

## Förenklingsregler

$$A \cdot A = A \quad A \cdot 0 = 0 \quad A + 0 = A$$

$$A + A = A \quad A \cdot 1 = A \quad A + 1 = 1$$

## Några teorem

<b>Absorbtionslagarna</b>	$A + A \cdot B = A$ $A \cdot (A + B) = A$
<b>Konkensuslagen</b>	$A \cdot B + \bar{A} \cdot C =$ $A \cdot B + \bar{A} \cdot C + B \cdot C$
<b>de Morgans lagar</b>	$\overline{(A + B)} = \bar{A} \cdot \bar{B}$ $\overline{(A \cdot B)} = \bar{A} + \bar{B}$

# ÖH 4.1 (a, b, c, h) Booles algebra

## 4.1

Använd räknelagarna i den booleska algebran för att förenkla följande logiska uttryck:

a)  $f = a \cdot \bar{c} \cdot d + a \cdot d$

b)  $f = a \cdot (\bar{b} + \bar{a} \cdot c + a \cdot b)$

c)  $f = a + \bar{b} + \bar{a} \cdot b + \bar{c}$

d)  $f = (a + b \cdot \bar{c}) \cdot (\bar{a} \cdot \bar{b} + c)$

e)  $f = (a + \bar{b}) \cdot (\bar{a} + b) \cdot (a + b)$

f)  $f = \bar{a} \cdot \bar{b} \cdot c + a \cdot b \cdot c + \bar{a} \cdot b \cdot c$

g)  $f = \bar{a} \cdot b \cdot \bar{c} + \bar{a} \cdot b \cdot \bar{d} + c \cdot d$

h)  $f = a + (\overline{\overline{a \cdot b}})$

i)  $f = \overline{\overline{a + a \cdot b} + c}$

# ÖH 4.1a

$$f = a \cdot \bar{c} \cdot d + a \cdot d = \{\text{bryt ut ad}\} = a \cdot d \cdot (\bar{c} + 1) = a \cdot d$$

# ÖH 4.1b

$$\begin{aligned} f &= a \cdot (\bar{b} + \bar{a} \cdot c + a \cdot b) = a \cdot \bar{b} + a \cdot \bar{a} \cdot c + a \cdot a \cdot b = \\ &= a\bar{b} + 0 + a \cdot b = a \cdot (\bar{b} + b) = a \end{aligned}$$



# ÖH 4.1c

$$f = a + \bar{b} + \bar{a} \cdot b + \bar{c} =$$

# ÖH 4.1c

$$\begin{aligned} f &= a + \overline{b} + \overline{a} \cdot b + \overline{c} = a + (a + \overline{a}) \cdot \overline{b} + \overline{a} \cdot b + \overline{c} = \\ &= a + a \cdot \overline{b} + \overline{a} \cdot \overline{b} + \overline{a} \cdot b + \overline{c} = \\ &= a + a \cdot \overline{b} + \overline{a} \cdot (\overline{b} + b) + \overline{c} = \\ &= \dots a + \overline{a} \dots = 1 \end{aligned}$$

# ÖH 4.1h

$$f = a + \overline{\overline{a \cdot b}} =$$

# ÖH 4.1h

$$f = a + \overline{\overline{a \cdot b}} = \{deMorgan\} = a + \overline{\overline{a} \oplus \overline{b}} = a + b$$

# ÖH 4.4 Använd De Morgans lag

## 4.4

Förenkla nedanstående uttryck så långt som möjligt.

$$\overline{(a + b + \bar{c})(a + \bar{b} + \bar{c})(\bar{a} + \overline{bc} + \bar{b}\bar{c})}$$

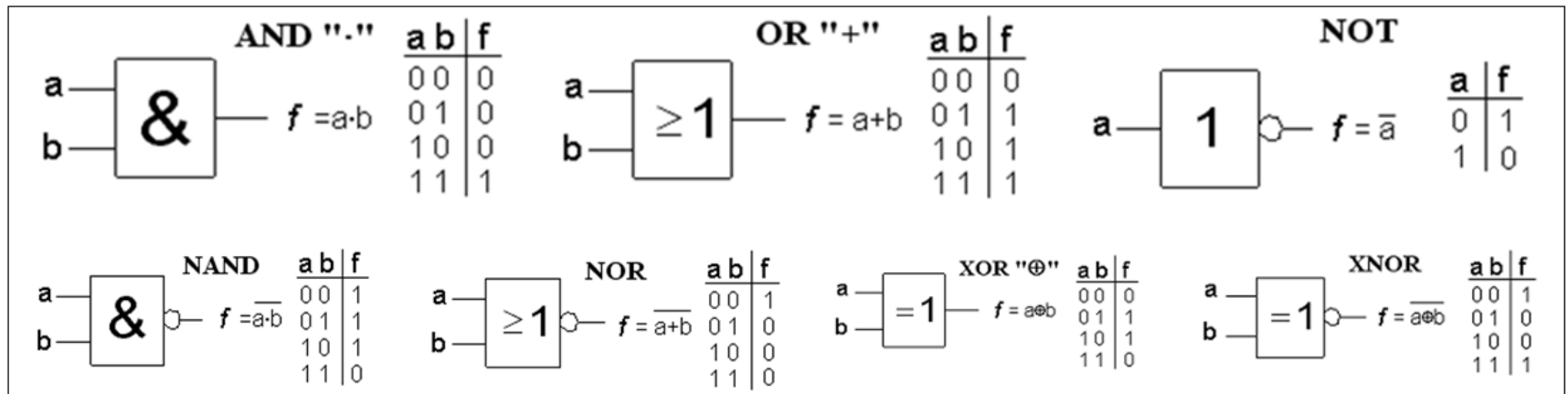
# ÖH 4.4

$$\begin{aligned} & \overline{(a + b + \bar{c})(a + \bar{b} + \bar{c})(\bar{a} + \overline{bc} + b\bar{c})} = \\ & \overline{(a + b + \bar{c})(a + \bar{b} + \bar{c})(\bar{a} + b + \bar{c} + b\bar{c})} = \\ & \overline{(a + b + \bar{c})(a + \bar{b} + \bar{c})(\bar{a} + b + \bar{c})} = \end{aligned}$$

Dubblera!

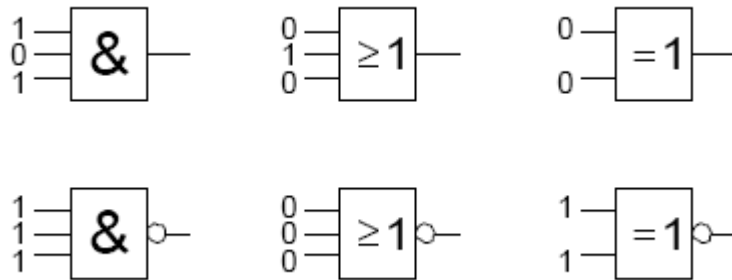
$$\begin{aligned} & \overline{(a + b + \bar{c})} + \overline{(a + \bar{b} + \bar{c})} + \overline{(\bar{a} + b + \bar{c})} = \\ & \overline{\overline{abc}} + \overline{abc} + \overline{a\bar{b}c} = \overline{\overline{abc}} + \overline{\overline{abc}} + \overline{abc} + \overline{a\bar{b}c} = \\ & \overline{ac}(\overline{b} + b) + \overline{bc}(a + \overline{a}) = \overline{ac} + \overline{bc} \end{aligned}$$

# Logikgrindar



# (ÖH 4.5a) Grundtyper

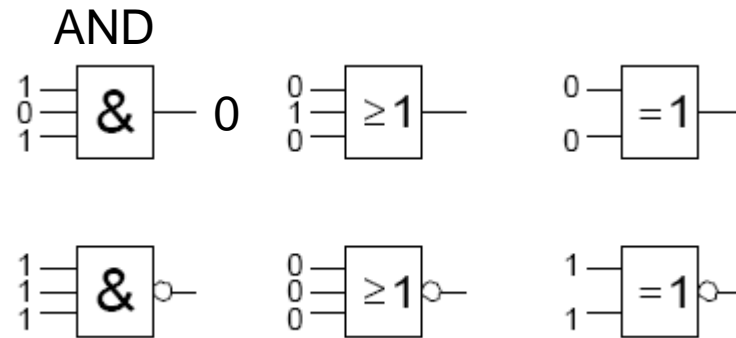
Ange namn och utsignal 1/0 för följande sex grindtyper när insignalerna är de som visas i figuren.





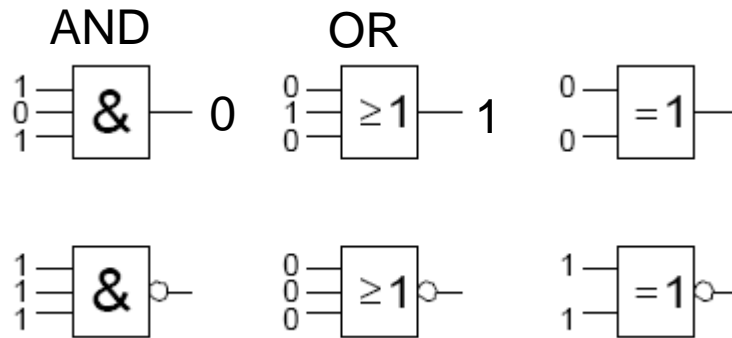
# (ÖH 4.5a) Grundtyper

Ange namn och utsignal 1/0 för följande sex grindtyper när insignalerna är de som visas i figuren.



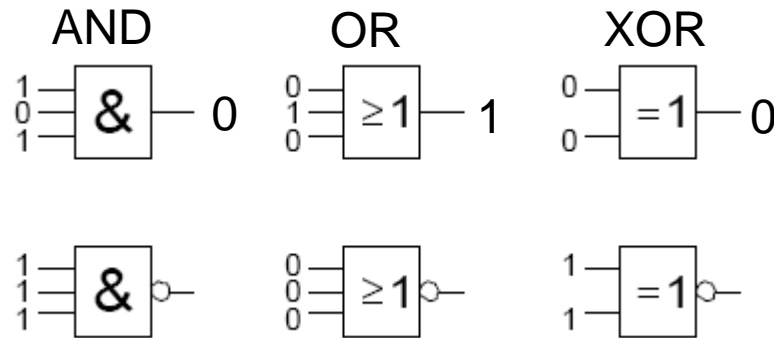
# (ÖH 4.5a) Grundtyper

Ange namn och utsignal 1/0 för följande sex grindtyper när insignalerna är de som visas i figuren.



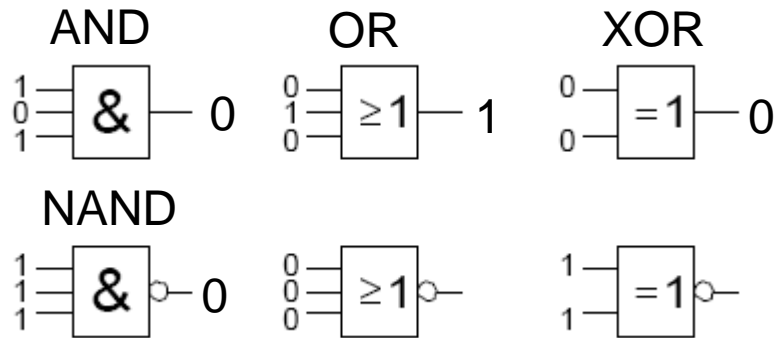
# (ÖH 4.5a) Grundtyper

Ange namn och utsignal 1/0 för följande sex grindtyper när insignalerna är de som visas i figuren.



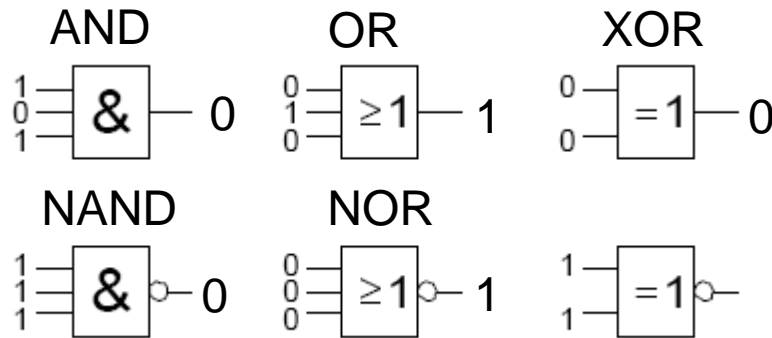
# (ÖH 4.5a) Grundtyper

Ange namn och utsignal 1/0 för följande sex grundtyper när insignalerna är de som visas i figuren.



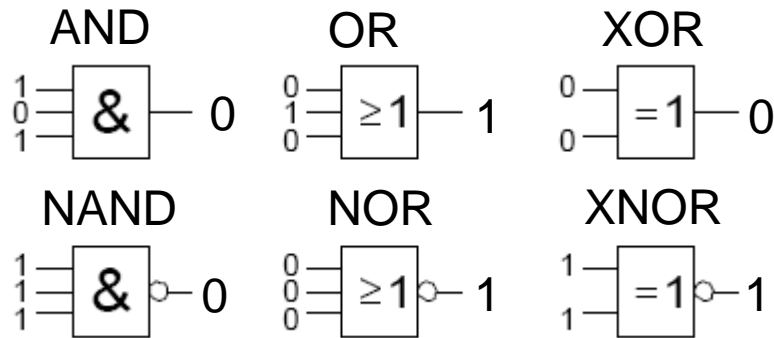
# (ÖH 4.5a) Grundtyper

Ange namn och utsignal 1/0 för följande sex grundtyper när insignalerna är de som visas i figuren.



# (ÖH 4.5a) Grundtyper

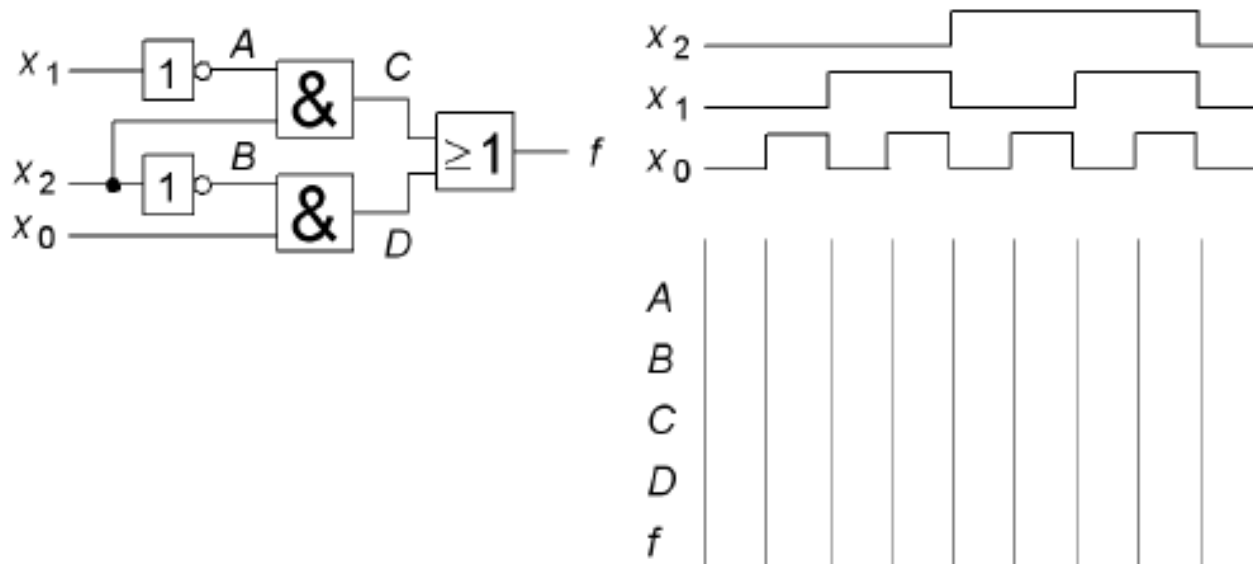
Ange namn och utsignal 1/0 för följande sex grundtyper när insignalerna är de som visas i figuren.



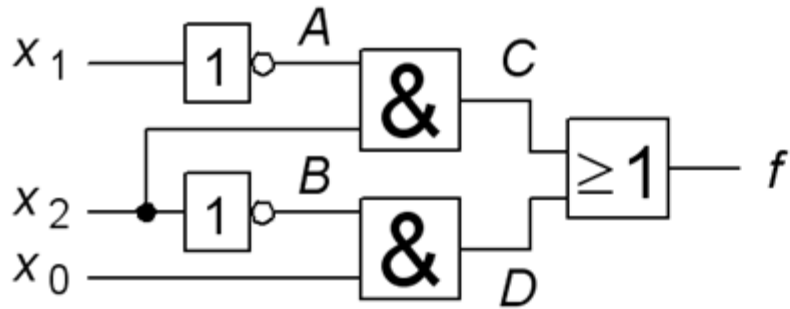
# ÖH 4.7 Tidsdiagram och sanningsstabell

## 4.7

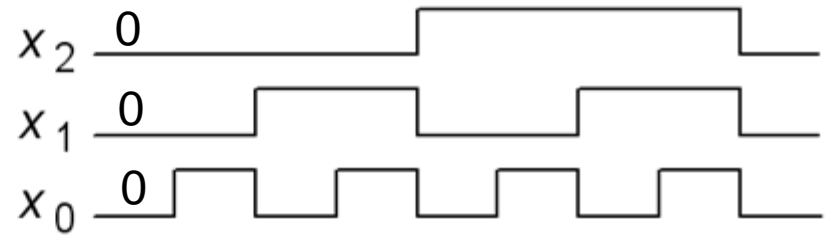
- a) Rita tidsdiagram över signalerna  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $f$ . Insignalerna till  $x_0$ ,  $x_1$ , och  $x_2$  har frekvensförhållandet 4:2:1 för att "svepa" igenom sanningsstabellens *alla* kombinationer i "rätt" ordning.
- b) Skriv upp sannings Tabellen för funktionen  $f$ .



# ÖH 4.7



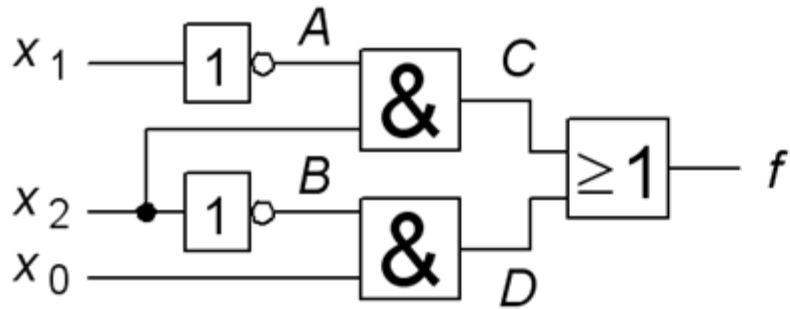
$x_2$	$x_1$	$x_0$	$f$
0	0	0	0
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	



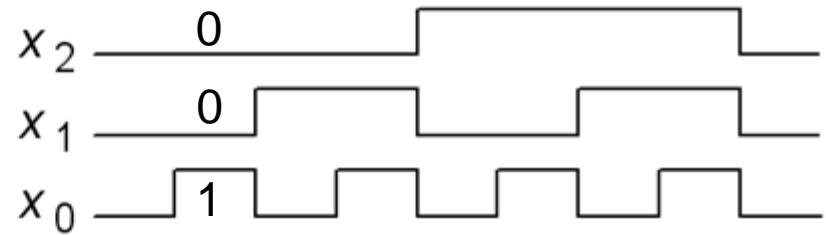
$A$	1							
$B$	1							
$C$	0							
$D$	0							
$f$	0							



# ÖH 4.7

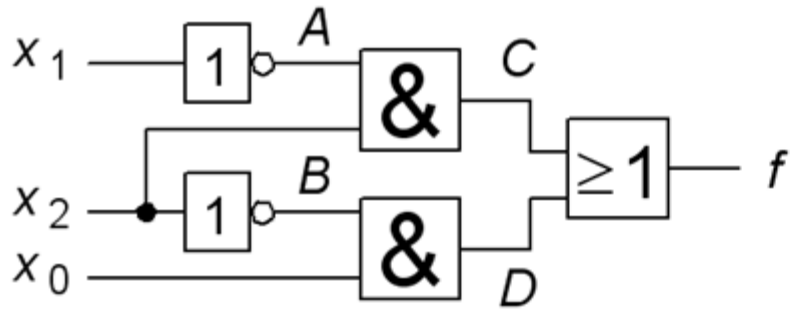


$X_2$	$X_1$	$X_0$	$f$
0	0	0	0
0	0	1	1
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

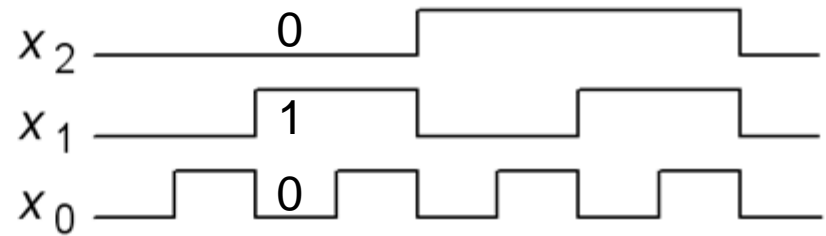


$A$	1								
$B$	1								
$C$	0								
$D$	1								
$f$	1								

# ÖH 4.7

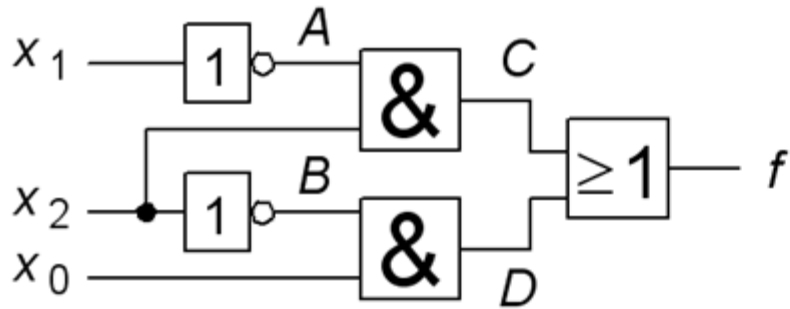


$X_2$	$X_1$	$X_0$	$f$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

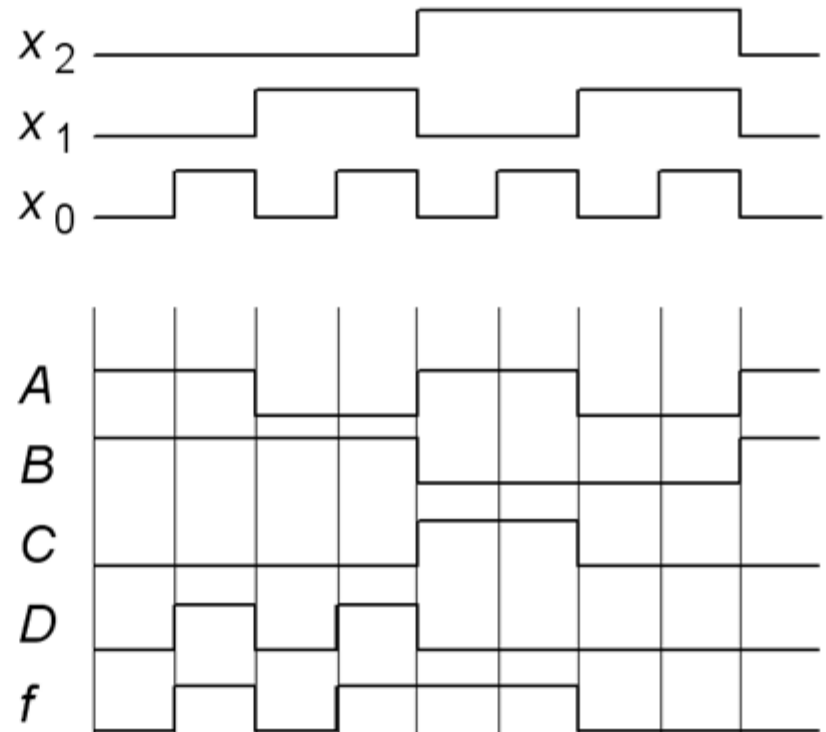


$A$		0						
$B$		1						
$C$		0						
$D$		0						
$f$		0						

# ÖH 4.7



$X_2$	$X_1$	$X_0$	$f$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

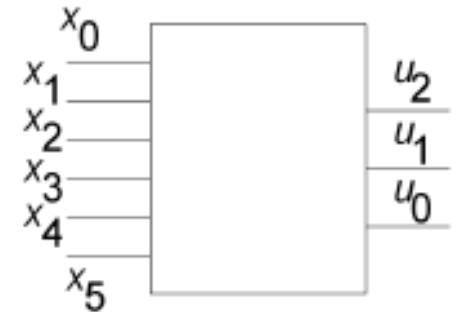


# ÖH 4.12 Från text till Boolska ekvationer

## 4.12

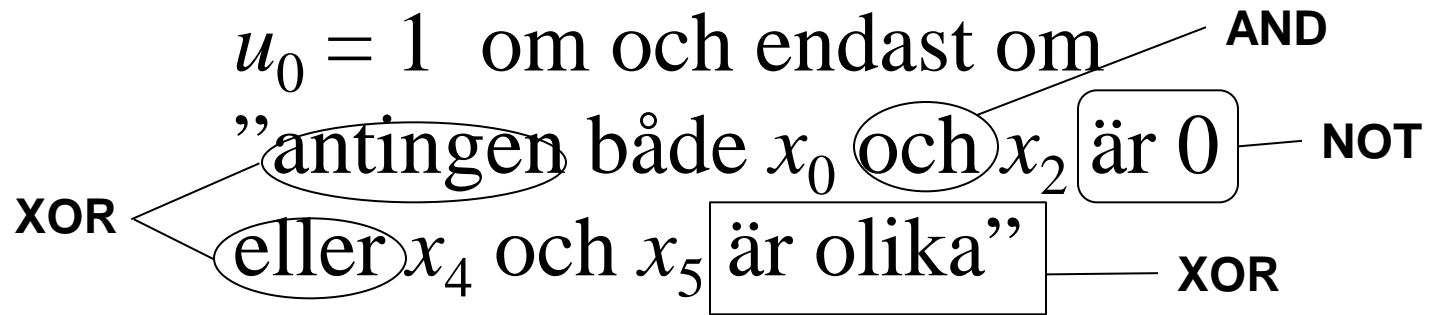
Ett kombinatoriskt nät med sex insignaler  $x_5, x_4, x_3, x_2, x_1, x_0$  och tre utsignaler  $u_2, u_1, u_0$ , beskrivs med text på följande sätt:

- $u_0 = 1$  om och endast om "antingen både  $x_0$  och  $x_2$  är 0 eller  $x_4$  och  $x_5$  är olika"
- $u_1 = 1$  om och endast om " $x_0$  och  $x_1$  är lika och  $x_5$  är inversen av  $x_2$ "
- $u_2 = 0$  om och endast om " $x_0$  är 1 och någon av  $x_1 \dots x_5$  är 0"



Beskriv nätet med Boolesk algebra och operationerna AND OR NOT XOR i stället.

# ÖH 4.12



$$u_0 = \bar{x}_0 \cdot \bar{x}_2 \oplus (x_4 \oplus x_5)$$

# ÖH 4.12

$u_1 = 1$  om och endast om

” $x_0$  och  $x_1$  är lika och  $x_5$  är inversen av  $x_2$ ”

XNOR

AND

XOR

$$u_1 = \overline{x_0 \oplus x_1} \cdot (x_5 \oplus x_2)$$

$$= (x_0 x_1 + \overline{x_0} \overline{x_1}) \cdot (x_5 \overline{x_2} + \overline{x_5} x_2)$$

# ÖH 4.12

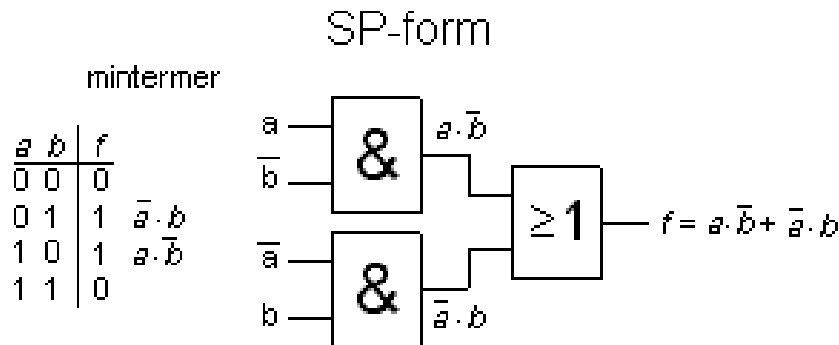
**NOT**  
 $u_2 = 0$  om och endast om  
” $x_0$  är 1 **och** någon av  $x_1 \dots x_5$  **är 0**”  
**AND** **OR** **NOT**

$$\begin{aligned}\bar{u}_2 &= x_0 \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4 + \bar{x}_5) \\ \Rightarrow u_2 &= \overline{x_0 \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4 + \bar{x}_5)} = \\ &= \bar{x}_0 + \overline{(\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4 + \bar{x}_5)} = \\ &= \bar{x}_0 + x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5\end{aligned}$$

# ÖH Logiknät SP-form

Alla logiska funktioner kan realiseras med hjälp av grindtyperna AND och OR kombinerade i *två* steg. Vi förutsätter här att ingångs-variablerna även finns i inverterad form, om inte så behöver man naturligtvis även inverterare NOT till detta.

## AND-OR logik, SP-form



Man kan realisera grindnätet direkt ur sanningstabellen. Varje "1" i tabellen är en *minterm*.

Funktionen blir *summan* av dessa mintermer. Man säger att funktionen är uttryckt på **SP-form** ( Summa av Produkter ).

*Men, det kan finnas mycket enklare grindnät med färre grindar som gör samma arbete.*



# ÖH 5.2 SP och PS normalform

## 5.2

En logisk funktion har följande sanningstabell:

<i>a b c</i>	<i>f</i>
0 0 0	1
0 0 1	0
0 1 0	0
0 1 1	0
1 0 0	1
1 0 1	1
1 1 0	0
1 1 1	1

Ange funktionen på SP-normalform (summa av produkttermer):

$$f(a, b, c) =$$

Ange funktionen på PS-normalform (produkter av summatermer):

$$f(a, b, c) =$$

# ÖH 5.2 SP-form

En logisk funktion har följande sanningstabell. Ange funktionen på SP-normalform (summa av produkter).

<i>a</i>	<i>b</i>	<i>c</i>	<i>f</i>
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

# ÖH 5.2 SP-form

En logisk funktion har följande sanningstabell. Ange funktionen på SP-normalform (summa av produkter).

$a$	$b$	$c$	$f$	
0	0	0	1	$\bar{a}\bar{b}\bar{c}$
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	1	$a\bar{b}\bar{c}$
1	0	1	1	$a\bar{b}c$
1	1	0	0	
1	1	1	1	$abc$

# ÖH 5.2 SP-form

En logisk funktion har följande sanningsstabell. Ange funktionen på SP-normalform (summa av produkter).

$a$	$b$	$c$	$f$	
0	0	0	1	$\bar{a}\bar{b}\bar{c}$
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	1	$a\bar{b}\bar{c}$
1	0	1	1	$a\bar{b}c$
1	1	0	0	
1	1	1	1	$abc$

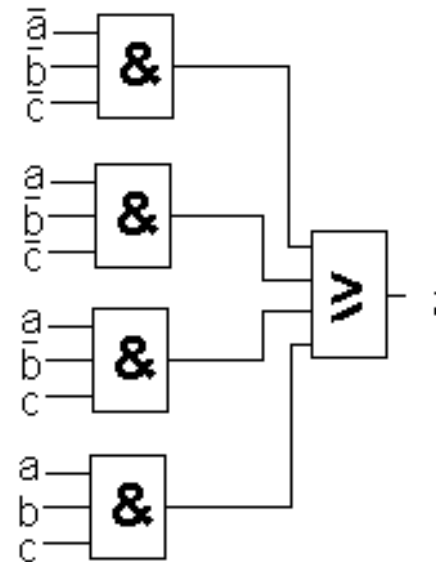
$$f = \bar{a} \cdot \bar{b} \cdot \bar{c} + a \cdot \bar{b} \cdot \bar{c} + a \cdot \bar{b} \cdot c + a \cdot b \cdot c$$

# ÖH 5.2 SP-form

En logisk funktion har följande sanningstabell. Ange funktionen på SP-normalform (summa av produkter).

<i>a</i>	<i>b</i>	<i>c</i>	<i>f</i>	
0	0	0	1	$\bar{a}\bar{b}\bar{c}$
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	1	$a\bar{b}\bar{c}$
1	0	1	1	$a\bar{b}c$
1	1	0	0	
1	1	1	1	$abc$

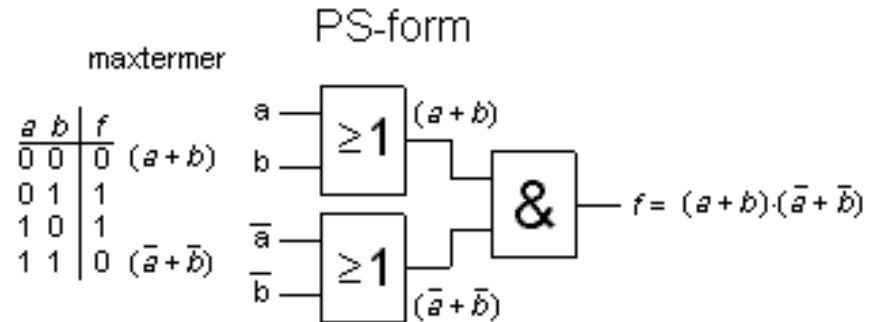
$$f = \bar{a} \cdot \bar{b} \cdot \bar{c} + a \cdot \bar{b} \cdot \bar{c} + a \cdot \bar{b} \cdot c + a \cdot b \cdot c$$



# ÖH Logiknät PS-form

Alternativt kan man inrikta sig på sanningstabellens 0:or. Om ett grindnät återger funktionens 0:or korrekt så är ju även 1:orna rätt!

## OR-AND logik, PS-form



Om således funktionen ska vara "0" för en viss variabelkombination (a,b) tex. (0,0) så bildar man summan ( $a + b$ ). Den summan kan ju bara bli "0" för kombinationen (0,0).

En sådan summa kallas för en *maxterm*. Funktionen uttrycks som en produkt av alla sådana maxtermer. Varje maxterm bidrar med en 0:a från sanningstabellen. Funktionen sägs vara uttryckt på **PS-form** ( Produkt av Summor ).

# ÖH 5.2 PS-form

En logisk funktion har följande sanningstabell. Ange funktionen på PS-normalform (produkt av summer).

<i>a</i>	<i>b</i>	<i>c</i>	<i>f</i>
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

# ÖH 5.2 PS-form

En logisk funktion har följande sanningstabell. Ange funktionen på PS-normalform (produkt av summor).

$a$	$b$	$c$	$f$	
0	0	0	1	
0	0	1	0	$(a+b+\bar{c})$
0	1	0	0	$(a+\bar{b}+c)$
0	1	1	0	$(a+\bar{b}+\bar{c})$
1	0	0	1	
1	0	1	1	
1	1	0	0	$(\bar{a}+\bar{b}+c)$
1	1	1	1	



# ÖH 5.2 PS-form

En logisk funktion har följande sanningstabell. Ange funktionen på PS-normalform (produkt av summor).

$a$	$b$	$c$	$f$	
0	0	0	1	
0	0	1	0	$(a+b+\bar{c})$
0	1	0	0	$(a+\bar{b}+c)$
0	1	1	0	$(a+\bar{b}+\bar{c})$
1	0	0	1	
1	0	1	1	
1	1	0	0	$(\bar{a}+\bar{b}+c)$
1	1	1	1	

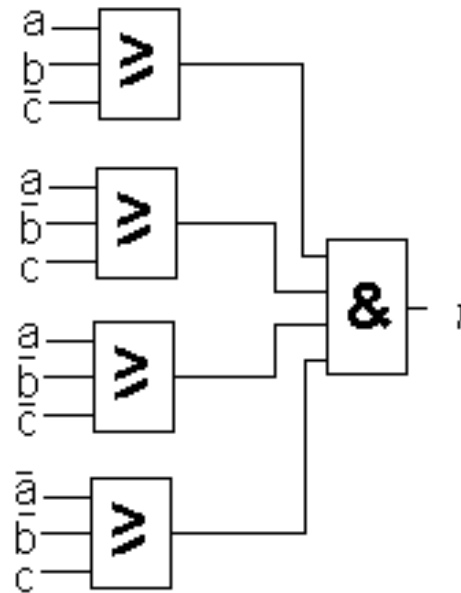
$$f = (a+b+\bar{c}) \cdot (a+\bar{b}+c) \cdot (a+\bar{b}+\bar{c}) \cdot (\bar{a}+\bar{b}+c)$$

# ÖH 5.2 PS-form

En logisk funktion har följande sanningstabell. Ange funktionen på PS-normalform (produkt av summor).

$a$	$b$	$c$	$f$	
0	0	0	1	
0	0	1	0	$(a+b+\bar{c})$
0	1	0	0	$(a+\bar{b}+c)$
0	1	1	0	$(a+\bar{b}+\bar{c})$
1	0	0	1	
1	0	1	1	
1	1	0	0	$(\bar{a}+\bar{b}+c)$
1	1	1	1	

$$f = (a + b + \bar{c}) \cdot (a + \bar{b} + c) \cdot (a + \bar{b} + \bar{c}) \cdot (\bar{a} + \bar{b} + c)$$

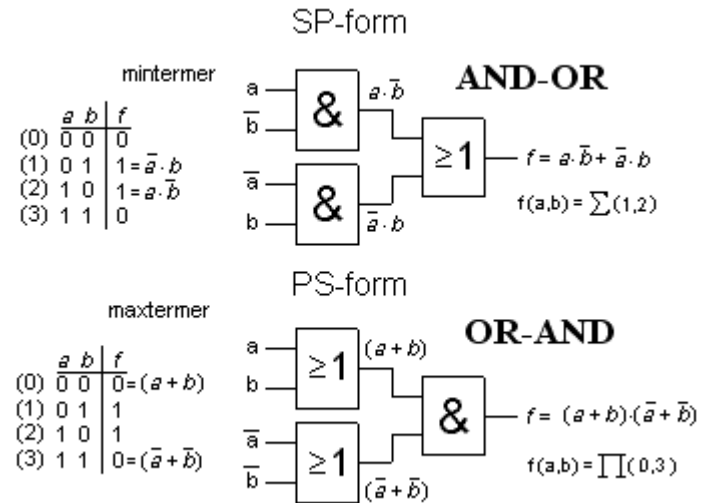


# $\Sigma$ och $\Pi$

SP och PS-formerna brukar förenklat uttryckas genom en *uppräknig* av de ingående maxtermernas/mintermernas ordningsnummer:

$$f(a,b) = \Sigma m(1,2)$$

$$f(a,b) = \Pi M(0,3)$$



# ÖH 5.3 SP och PS -form

## 5.3

En minimerad funktion är angiven på SP form (Summa av Produkter).

Ange samma funktion med mintermer som SP, respektive med maxtermer som PS (Produkt av Summer).

$$f(x, y, z) = x\bar{y} + y\bar{z} + \bar{x}z$$

# ÖH 5.3

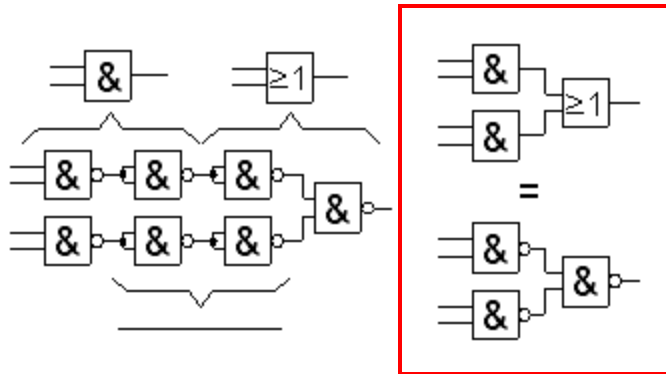
$$\begin{aligned}f(x, y, z) &= x\bar{y} + y\bar{z} + \bar{x}z \\&= x\bar{y}(z + \bar{z}) + (x + \bar{x})y\bar{z} + \bar{x}(y + \bar{y})z = \\&= x\bar{y}z + x\bar{y}\bar{z} + x\bar{y}z + \bar{x}y\bar{z} + \bar{x}y\bar{z} + \bar{x}y\bar{z}\end{aligned}$$

$$\Rightarrow f(x, y, z) = \sum m(001, 010, 011, 100, 101, 110) = \sum m(1, 2, 3, 4, 5, 6)$$

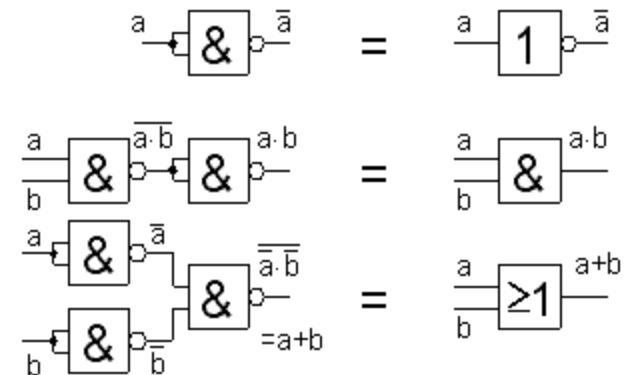
$$\Rightarrow f(x, y, z) = \prod M(0,7) = (x + y + z)(\bar{x} + \bar{y} + \bar{z})$$

# Komplett logik NAND-NAND

OR AND och NOT går att framställa med NAND-grindar. För logik-funktioner på SP-form kan man byta AND-OR grindarna mot NAND-NAND "rakt av".



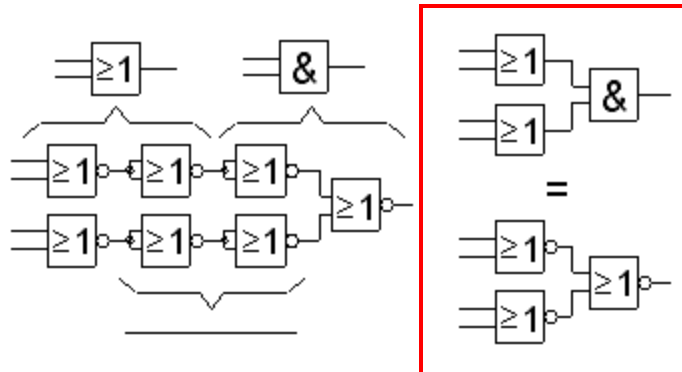
## NAND-NAND



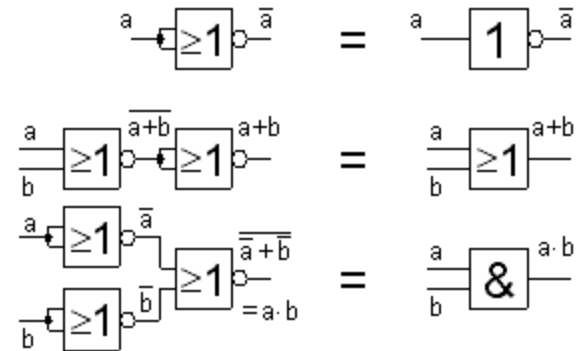
Kostnaden i antal grindar blir densamma!

# Komplett logik NOR-NOR

OR AND och NOT går även att framställa med NOR-grindar. För logikfunktioner på PS-form kan man byta OR-AND grindarna mot NOR-NOR grindar "rakt av".



NOR-NOR

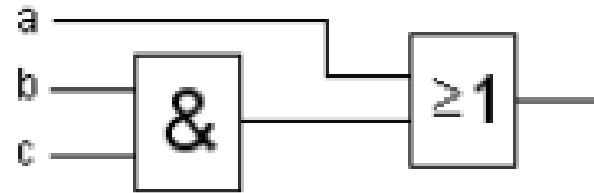


Kostnaden i antal grindar blir densamma!

# ÖH 5.5 NAND-grindar

## 5.5

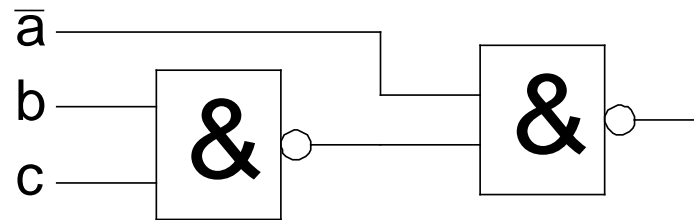
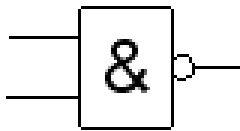
Rita vidstående AND/OR-nät som ett NAND/NAND-nät.





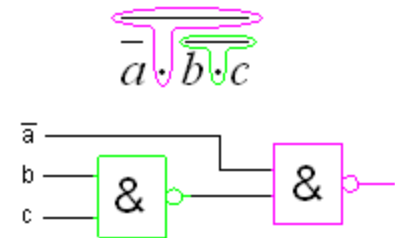
# ÖH 5.5

byt till  
NAND-grindar



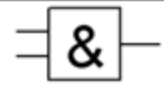
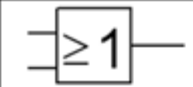
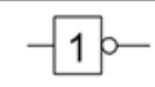
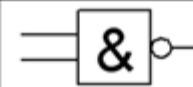
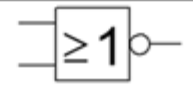
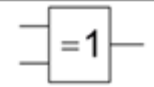
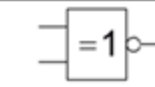
*Algebraiskt:*

$$a + b \cdot c = \overline{\overline{a + b \cdot c}} = \overline{\overline{a} \cdot \overline{b \cdot c}}$$



# (ÖH 4.11) Europeiska och Amerikanska symboler

*Testa dig själv ...*

# (ÖH 4.11) Europeiska och Amerikanska symboler

*Testa dig själv ...*

AND	OR	NOT	NAND	NOR	XOR	XNOR
