

Grundläggande Datorteknik

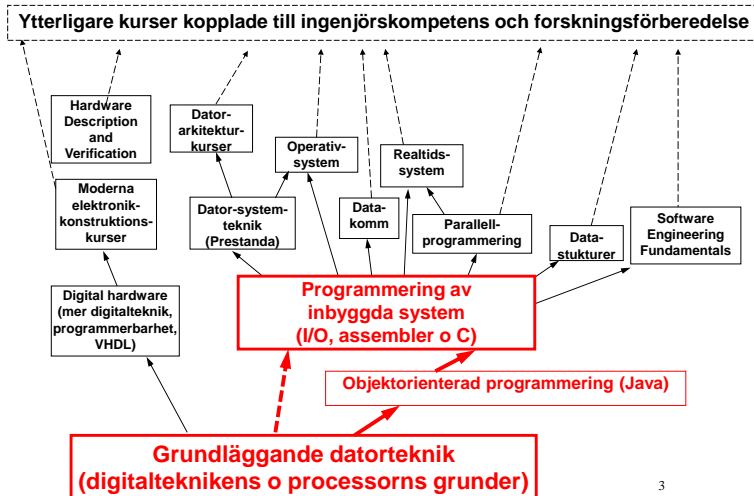
Välkommen!

LP1 EDA433 (IT1), DIT790 (GU)
LP2 EDA451 (D1), DIT790 (GU)
LP3 EDA217 (Z1), DIT790 (GU)

Kursens mål:

- Fatta hur en dator är uppbyggd (HDW)
- Fatta hur du programmerar den (SW)
- Fatta hur HDW o SW samverkar

Digital teknik Dator teknik



Kursens Hemsida

Sök via studieportalen

EDA433/452/217 (LP1,2,3)

eller

<http://www.cse.chalmers.se/edu/year/2012/course/EDA433>

<http://www.cse.chalmers.se/edu/year/2012/course/EDA452>

<http://www.cse.chalmers.se/edu/year/2013/course/EDA217>

Länkar till kursdokument

Senaste nytt

Kurs-PM

Veckoplanering

Laborationsinformation

Simulatorer

Schema Vanliga veckor LP1 HT12 It

Fö: Föreläsningar / Demonstration
 Sim: Handledd självverksamhet.
 LAB: Laboration

	Må	Ti	On	To	Fr
8			Demo HA4	SimBC 08.00-09.45 ES61/62	LAB C 08.00-11.45 LAB 4225
10		Fö HC4		LAB A 08.00-11.45 LAB 4225	LAB C 08.00-11.45 LAB 4225
13	Fö HA4	Sim A 15.15-17.00 ES61	Fö HB1	Sim E 13.15-15.00 ES61	LAB B 13.15-17.00 LAB 4225
15				Sim D 13.15-15.00 ES61	LAB D 13.15-17.00 LAB 4225
	SimStart Tis 4/9	Lab Start Tor 20/9		LAB E 17.15-21.00 LAB 4225	

5

Laborationsregler

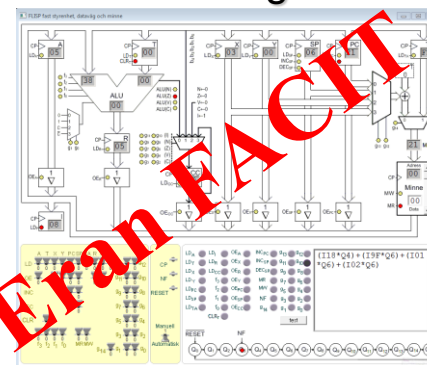
- Laborationerna innehåller hemuppgifter. (Se respektive laborations-PM). **Dessa uppgifter skall vara lösta och uppvisade före laborationstillfället.**
- Vid laborationstillfället delas ut **extra uppgifter** som du självständigt skall lösa för att bli godkänd.
- En handledare skall bedöma arbetet efter laborationen.
- **Kom i tid** och hasta ej igenom uppgifterna.

Är laborant p g a sjukdom eller annan angelägen orsak förhindrad att delta vid ett laborationstillfälle skall detta omedelbart meddelas till laborationschefen.

Simulator övningar

DigiFlisp
 Kopplingsboxen
 Datavägar
 Flisp

Eterm
 Flisp
 I/O
 Borrmaskin



6

Kurslitteratur

(ARB1) Roger Johansson: Arbetsbok för DigiFlisp. 2012.

(EXT) Kompletterande material CTH 2012
 Instruktionslista för FLISP-processorn.
 Laborations-PM nr 1-4. Inst för datorteknik, CTH, 2012.

(SIM) Simulatorer för digitala kretsar, FLISP. Programvara för PC (Windows).

(KMP) Johnson, Larsson & Arebrink: Grundläggande digital- och datorteknik.
 Kompendium, Inst för datorteknik, CTH, 2007.
 Del 1. Digital teknik

KIT

Fo1

Kursens mål:

- ▶ Konstruera en dator mha grindar och programmera denna

Veckans mål:

- ▶ Beskriva grindar och de verktyg som behövs under konstruktionsarbetet
- ▶ Hur kodas tal och tecken i datorn

Dagens mål:

- ▶ **Inledning till Grundläggande Datorteknik**
- ▶ Kunna använda binära tal
- ▶ Kunna omvandla mellan binära, hexadecimala och decimala tal
- ▶ Förstå innebörden av olika binära koder

**Läs smart!
Lär dig mer!**

DIGITAL- och DATORTEKNIK

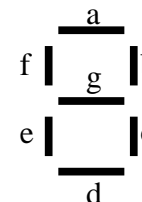
Ext 1

Vad är digitalteknik för något?

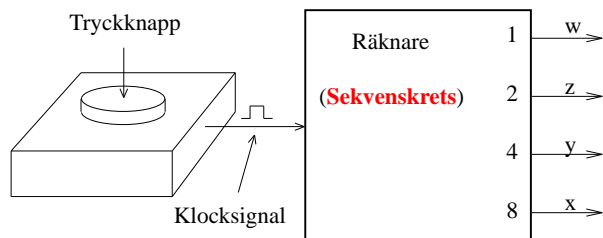
Digitalteknik = Sifferteknik



Sifferindikatorn har sju segment. Man kan utifrån välja vilka segment som skall synas och vilka som skall vara osynliga.



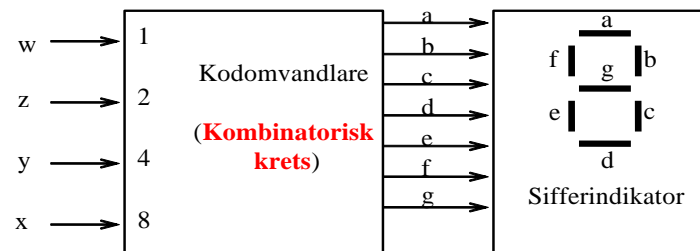
Exempel på digital krets:



Varje gång man trycker ned tryckknappen kommer det en puls (klocksignal) på ledningen som är kopplad till räknaren.

Exempel på digital krets:

Ext 1



Kodomvandlaren till vänster i figuren ovan är en digital krets. Den omvandlar ett binärt tal med siffrorna xyzw (ingångarna) till ett mönster av nollor och ettor på utgångarna abcdefg så att sifferindikatorn visar det binära talet $(xyzw)_2$ på decimal form.

I kursen visas metoder för att studera (analys) och konstruera (syntes) digitala kretsar liknande kodomvandlaren.

Fo1

Kursens mål:

- ▶ Använda en modern processor
- ▶ Konstruera en dator mha grindar och programmera denna

Veckans mål:

- ▶ Beskriva grindar och de verktyg som behövs under konstruktionsarbetet
- ▶ Hur kodas tal och tecken i datorn

Dagens mål:

- ▶ Inledning till Grundläggande Dator teknik
- ▶ **Kunna använda binära tal**
- ▶ Kunna omvandla mellan binära, hexadecimala och decimala tal
- ▶ Förstå innebörden av olika binära koder

**Läs smart!
Lär dig mer!**

Vad Repr ettorna o nollorna

100110100011101010101011000011

Vad representerar ettorna o nollorna?

- | Binärtal
- | Excess-kod
- | Gray-kod
- | NBCD
- | ASCII-kod

Begrepp vid binär kodning

begrepp	betydelse	exempel...
bit/bitar	minsta informationsenhet, kan anta två värden	0 eller 1
bitsträng binärt ord	sekvens av bitar	101100100001...
kodord	$K_7 K_6 K_5 K_4 K_3 K_2 K_1 K_0$ också ett binärt ord men med en fastställd kodning (betydelse)	1000001 = "A" (ASCII) 1000001 = 65 (naturligt tal) 1000001 = -127 (heltal)
ordlängd	antal bitar i ordet	
nibble	ordlängden 4 bitar	0101
byte	ordlängden 8 bitar	01011100

Decimala – positionssystemet; binära tal

SEK 215:-

Decimala tal
Bas 10
10 siffror



$$2 \cdot 10^2 + 1 \cdot 10^1 + 5 \cdot 10^0$$

~~251:-~~
~~521:-~~
~~152:-~~

215₁₀

Binära tal
Bas 2
2 siffror

$$10111_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$

$$16 + 0 + 4 + 2 + 1 = 23_{10}$$

Fo1

Kursens mål:

- ▶ Använda en modern processor
- ▶ Konstruera en dator mha grindar och programmera denna

Veckans mål:

- ▶ Beskriva grindar och de verktyg som behövs under konstruktionsarbetet
- ▶ Hur kodas tal och tecken i datorn

Dagens mål:

- ▶ Inledning till Grundläggande Datorteknik
- ▶ Kunna använda binära tal
- ▶ **Kunna omvandla mellan binära, hexadecimala och decimala tal**
- ▶ Förstå innebörden av olika binära koder

**Läs smart!
Lär dig mer!**

Talsystem
Basen 2, 8, 10, 16

Positionssystem
Ex: 214
(421)

$\beta = 2$	$\beta = 8$	$\beta = 10$	$\beta = 16$
binärt	oktalt	decimalt	hexa-decimalt
0	0	0	0
1	1	1	1
10	2	2	2
11	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F
10000	20	16	10
10001	21	17	11

S2.4

Omvandling av ett tal N med basen 10 till basen 2.

s2.8
Ext2

$$(N)_{10} \rightarrow (N)_2$$

Talet $(N)_{10}$ delas upp i heltalsdel $(N_H)_{10}$ och bråktalsdel $(N_B)_{10}$.

$$(N)_{10} = (N_H \cdot N_B)_{10}$$

Heltalsdelen och bråktalsdelen behandlas sedan var för sig.

Exempel

211.678_{10} till binärt

Heltalsdelen:

$(211)_{10}$ omvandlas till bas 2 genom successiva divisioner med 2.

kvot		rest	
$211/2 = 105$	+	$\frac{1}{2}$	$d_0 = 1$
$105/2 = 52$	+	$\frac{1}{2}$	$d_1 = 1$
$52/2 = 26$	+	0	$d_2 = 0$
$26/2 = 13$	+	0	$d_3 = 0$
$13/2 = 6$	+	$\frac{1}{2}$	$d_4 = 1$
$6/2 = 3$	+	0	$d_5 = 0$
$3/2 = 1$	+	$\frac{1}{2}$	$d_6 = 1$
$1/2 = 0$	+	$\frac{1}{2}$	$d_7 = 1$

S2.8
Ext2

Sluta

som ger $(211)_{10} = (11010011)_2$

Bråktalsdelen:

Talet $(0,678)_{10}$ omvandlas till bas 2 med successiva multiplikationer

	heltal	bråk		S2.12
$2 \cdot 0,678 =$	1	+ 0,356	$d_{-1} = 1$	Ext2
$2 \cdot 0,356 =$	0	+ 0,712	$d_{-2} = 0$	
$2 \cdot 0,712 =$	1	+ 0,424	$d_{-3} = 1$	
$2 \cdot 0,424 =$	0	+ 0,848	$d_{-4} = 0$	
$2 \cdot 0,848 =$	1	+ 0,696	$d_{-5} = 1$	
$2 \cdot 0,696 =$	1	+ 0,392	$d_{-6} = 1$	
		:	:	

Sluta
????

som ger $(0,678)_{10} = (0.101011.....)_2$

Kursens mål:

- ▶ Använda en modern processor
- ▶ Konstruera en dator mha grindar och programmera denna

Veckans mål:

- ▶ Beskriva grindar och de verktyg som behövs under konstruktionsarbetet
- ▶ Hur kodas tal och tecken i datorn

Dagens mål:

- ▶ Inledning till Grundläggande Datorteknik
- ▶ Kunna använda binära tal
- ▶ Kunna omvandla mellan binära, hexadecimala och decimala tal
- ▶ **Förstå innebörden av olika binära koder**

Läs smart!
Lär dig mer!

Hur generera GRAY-KOD?

Tabell 2.2. Graykoder.

S2.16

Decimal ordning	Kodord i trebitars Graykod	Kodord i fyrbitars Graykod
0	000	0000
1	001	0001
2	011	0011
3	010	0010
4	110	0110
5	111	0111
6	101	0101
7	100	0100
8		1100
9		1101
10		1111
11		1110
12		1010
13		1011
14		1001
15		1000

Tabell 2.1. Excess- 2^{n-1} kodning vid $n = 4$.

S2.15

Excess - kod

Ex koda bipolära storheter

Ex avkoda:

$1\ 1\ 0\ 0_2$ Excess- 2^{n-1}

$n=4$

$2^3+2^2+0+0 -2^3= 4_{10}$

Nivå $k\Phi$	Kodord i excess- 2^{n-1} -kod ($n=4$)
-8 Φ	0000
-7 Φ	0001
-6 Φ	0010
-5 Φ	0011
-4 Φ	0100
-3 Φ	0101
-2 Φ	0110
-1 Φ	0111
0	1000
1 Φ	1001
2 Φ	1010
3 Φ	1011
4 Φ	1100
5 Φ	1101
6 Φ	1110
7 Φ	1111

Decimal siffra	NBCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
	1010
	1011
	1100
	1101
	1110
	1111

Används EJ.

NBCD-kod

S2.18

Skriv $(53,27_{10})$ på NBCD-kod.

Alfanumeriska koder

ASCII-koden.

0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1	$b_3b_2b_1$	$b_3b_2b_1b_0$
NUL	DLE	SP	0	@	P	`	p		0 0 0 0
SOH	DC1	!	1	A	Q	a	q		0 0 0 1
STX	DC2	"	2	B	R	b	r		0 0 1 0
ETX	DC3	#	3	C	S	c	s		0 0 1 1
EOT	DC4	\$	4	D	T	d	t		0 1 0 0
ENQ	NAK	%	5	E	U	e	u		0 1 0 1
ACK	SYN	&	6	F	V	f	v		0 1 1 0
BEL	ETB	'	7	G	W	g	w		0 1 1 1
BS	CAN	(8	H	X	h	x		1 0 0 0
HT	EM)	9	I	Y	i	y		1 0 0 1
LF	SUB	*	:	J	Z	j	z		1 0 1 0
VT	ESC	+	;	K	[ä	{		1 0 1 1
FF	FS	,	<	L	\	ö			1 1 0 0
CR	GS	-	=	M]	å	~		1 1 0 1
S0	RS	.	>	N	^	n	~		1 1 1 0
S1	US	/	?	O	_	o		RUBOUT (DEL)	1 1 1 1

S2.27

I DAG

Fö2

S3.1

Veckans mål:

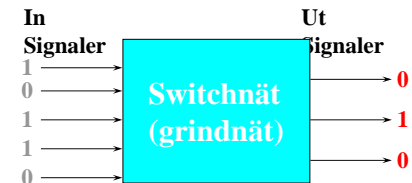
- ▶ Beskriva grindar och de verktyg som behövs under konstruktionsarbetet av datorn
- ▶ Hur kodas tal och tecken i datorn

Dagens mål:

- ▶ Switchnätsalgebra
- ▶ Kunna använda boolesk algebra
- ▶ Kunna använda funktionstabell
- ▶ Kunna utföra binär evaluering
- ▶ Kunna återge grindarnas sanningstabeller
- ▶ Kunna koppla ihop grindar

Läs smart!
Lär dig mer!

Switchnätsalgebra



Verktyg

- Boolesk algebra
- Funktionstabell
- Karnaughdiagram

Fö2

Dagens mål:

- ▶ Switchnätsalgebra
- ▶ **Kunna använda boolesk algebra**
- ▶ Kunna använda funktionstabell
- ▶ Kunna utföra binär evaluering
- ▶ Kunna återge grindarnas sanningstabeller
- ▶ Kunna koppla ihop grindar

**Läs smart!
Lär dig mer!**

3.2 Boolesk algebra

S3.7

Operatorer: **+**, **·**, **'**, Data: **0**, **1**

Räkneregler:

$0 + 0 = 0$	$0 \cdot 0 = 0$	$0' = 1$
$0 + 1 = 1$	$0 \cdot 1 = 0$	$1' = 0$
$1 + 0 = 1$	$1 \cdot 0 = 0$	↑ Icke
$1 + 1 = 1$	$1 \cdot 1 = 1$	
↑ Eller	↑ Och	

Satser inom Boolesk algebra.

S3.8

- | | | |
|---|-----------|------------------|
| 1. Kommutativa lagarna | 3. | $x + 0 = x$ |
| $x + y = y + x$ | | $x \cdot 1 = x$ |
| $x \cdot y = y \cdot x$ | 4. | $x + x' = 1$ |
| 2. Distributiva lagarna | | $x \cdot x' = 0$ |
| $x \cdot (y + z) = x \cdot y + x \cdot z$ | 5. | $x + 1 = 1$ |
| $x + (y \cdot z) = (x + y) \cdot (x + z)$ | | $x \cdot 0 = 0$ |
| 7. Associativa lagarna | 6. | $x + x = x$ |
| $x + (y + z) = (x + y) + z$ | | $x \cdot x = x$ |
| $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ | 9. | $(x')' = x$ |
| 8. De Morgans lagar | | |
| $(x + y)' = x' \cdot y'$ | | |
| $(x \cdot y)' = x' + y'$ | | |

Fö2

Dagens mål:

- ▶ Switchnätsalgebra
- ▶ Kunna använda boolesk algebra
- ▶ **Kunna använda funktionstabell**
- ▶ **Kunna utföra binär evaluering**
- ▶ Kunna återge grindarnas sanningstabeller
- ▶ Kunna koppla ihop grindar

**Läs smart!
Lär dig mer!**

Funktionstabell

S3.8

Satserna i Boolesk algebra kan enkelt bevisas med hjälp av **binär evaluering** och utnyttjande av räkneregler för en Boolesk algebra.

Enklast görs detta i tabellform i en s k **funktionstabell**.

$$\text{Visa att: } (x + y)' = x' \cdot y'$$

(Sats 8 De Morgans)

Sätt upp alla kombinationer av variablerna !!!!

Fö2

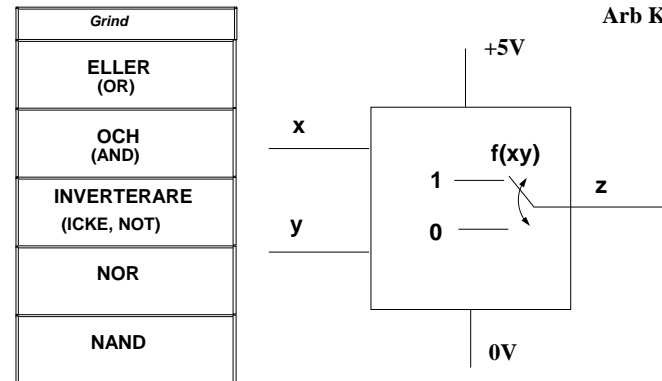
Dagens mål:

- ▶ Switchnätsalgebra
- ▶ Kunna använda boolesk algebra
- ▶ Kunna använda funktionstabell
- ▶ Kunna utföra binär evaluering
- ▶ **Kunna återge grindarnas sanningstabeller**
- ▶ Kunna koppla ihop grindar

**Läs smart!
Lär dig mer!**

Logikkrets.

S1.14
S3.18
Arb Kap 1

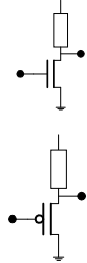


Några olika teknologier...

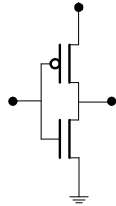
TTL
(Transistor-Transistor-Logic)



MOS
(Metal Oxide Silicon)



CMOS
(Complementary MOS)

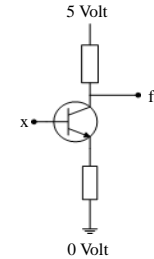


Mer om detta kommer i kursen Digitalteknik FK

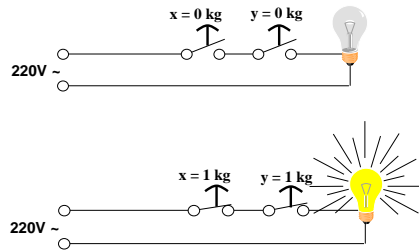
Funktionstabeller för grundläggande grindtyper

Sanningstabell

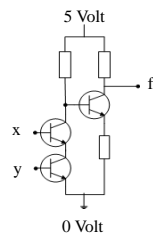
X	1	X'	X	f
Inverterare (NOT)				
	0	1	1	0
	1	0	0	1



AND - GRIND



x	y	f
0	0	0
0	1	0
1	0	0
1	1	1

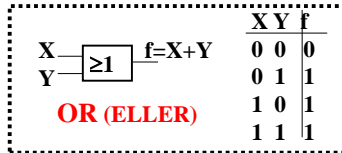


Funktionstabeller för grundläggande grindtyper

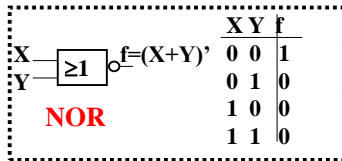
X	Y	$f = X \cdot Y$	X	Y	f
AND (OCH)					
	0	0	0	0	0
	0	1	0	1	0
	1	0	1	0	0
	1	1	1	1	1

X	Y	$f = (X \cdot Y)'$	X	Y	f
NAND					
	0	1	0	0	1
	0	1	0	1	1
	1	0	1	0	1
	1	1	1	1	0

Funktionstabeller för grundläggande grindtyper



S3.19



Logikkretssymboler för grundläggande logikoperationer.

S3.18
S1.14

Funktion	Grind	Grafisk symbol
$z = x + y$	ELLER (OR)	
$z = x \cdot y$	OCH (AND)	
$z = x'$	INVERTERARE (ICKE, NOT)	
$z = (x + y)'$	NOR	
$z = (x \cdot y)'$	NAND	

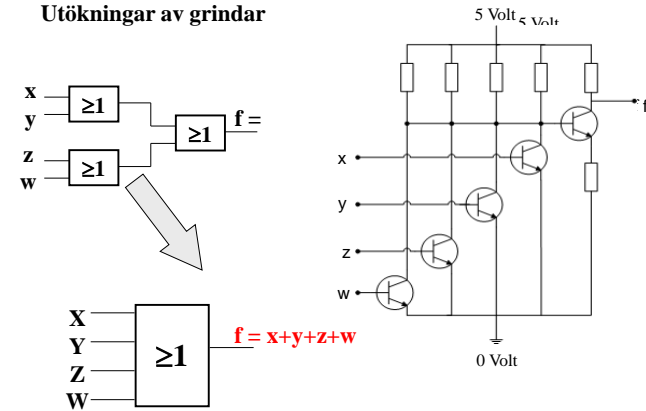
Fö2

Dagens mål:

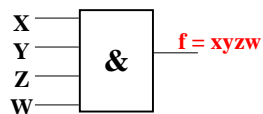
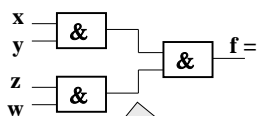
- ▶ Switchnätsalgebra
- ▶ Kunna använda boolesk algebra
- ▶ Kunna använda funktionstabell
- ▶ Kunna utföra binär evaluering
- ▶ Kunna återge grindarnas sanningstabeller
- ▶ **Kunna koppla ihop grindar**

**Läs smart!
Lär dig mer!**

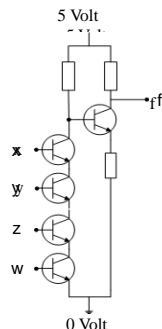
Utökningar av grindar



Utökningar av grindar

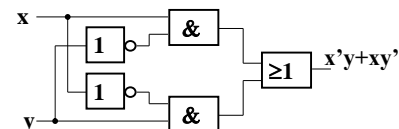


Grundläggande Datorteknik OH LV1



45

XOR, (Exclusive-OR)

Speciell operationssymbol \oplus Så: $x \oplus y = x'y + xy'$

X	Y	f
0	0	0
0	1	1
1	0	1
1	1	0

Grundläggande Datorteknik OH LV1

46

I DAG

Fö3

Veckans mål:

- ▶ Beskriva grindar och de verktyg som behövs under konstruktionsarbetet av datorn
- ▶ Hur kodas tal och tecken i datorn

Dagens mål:

- ▶ Kunna jobba med booleska uttryck och funktioner
- ▶ Kunna ange uttryck på Disjunktiv och Konjunktiv form
- ▶ Kunna skriva uttryck på Normal / Minimal form
- ▶ Kunna ange Mintermer och Maxtermer
- ▶ Kunna minimera funktioner med hjälp av Karnaughdiagram
- ▶ Kunna Analysera och Minimera grindnät
- ▶ Konstruera grindnät bestående av NAND och NOR logik

**Läs smart!
Lär dig mer!**

Grundläggande Datorteknik OH LV1

47

Booleska uttryck och Booleska funktioner

s3.14-17

Booleska uttryck som är ekvivalenta repr. samma **Booleska funktion**:

$f(x,y) = (x+y)'$ och $g(x,y) = x' \cdot y'$ utgör olika uttryck för

samma funktion (ty $(x+y)' = (x' \cdot y)'$ enligt sats 8). (jfr vanlig algebra)

Ett uttryck på **disjunktiv form** är en summa av termer, där

varje term är en Boolesk produkt av variabler (med eller utan prim):

T ex $f(x,y,w,z) = yz + wz + x'yz'$

Ett uttryck på **konjunktiv form** är en produkt av faktorer, där

varje faktor är en Boolesk summa av variabler (med eller utan prim):

T ex $g(x,y,w,z) = (y + w)(w' + z)(x' + z)$

Grundläggande Datorteknik OH LV1

48

I DAG

Fö3

Dagens mål:

- ▶ Kunna jobba med booleska uttryck och funktioner
- ▶ Kunna ange uttryck på Disjunktiv och Konjunktiv form
- ▶ **Kunna skriva uttryck på Normal / Minimal form**
- ▶ **Kunna ange Mintermer och Maxtermer**
- ▶ Kunna minimera funktioner med hjälp av Karnaughdiagram
- ▶ Kunna Analysera och Minimera grindnät
- ▶ Konstruera grindnät bestående av NAND och NOR logik

**Läs smart!
Lär dig mer!**

Exempel (Kalle forts)

Visa hur det Booleska uttrycket

$$f(x,y,z) = xy + yz + x'z$$

1) kan förenklas (minimeras) till

disjunktiv minimal form: (*Summa av Produkter*)

2) kan förenklas (minimeras) till

konjunktiv minimal form: (*Produkt av Summor*)

Exempel (Kalle forts)

Visa hur $f(x,y,z) = xy + yz + x'z$ kan förenklas till

disjunktiv (minimal) form: (*Summa av Produkter*)

$$\begin{aligned}
 f(x,y,z) &= xy + yz + x'z \\
 &= xy + yz1 + x'z && \text{sats 3} \\
 &= xy + yz(x + x') + x'z && \text{sats 4} \\
 &= xy + yzx + yzx' + x'z && \text{sats 2} \\
 &= xy + xyz + x'z + x'yz && \text{sats 1} \\
 &= xy1 + xyz + x'z1 + x'zy && \text{sats 3} \\
 &= xy(1 + z) + x'z(1 + y) && \text{sats 2} \\
 &= xy1 + x'z1 && \text{sats 5} \\
 &= xy + x'z && \text{sats 3}
 \end{aligned}$$

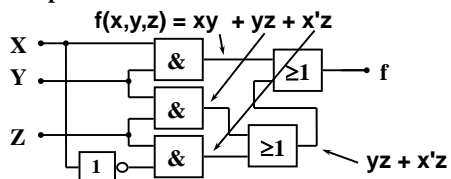
Exempel (Kalle forts)

Visa hur $f(x,y,z) = xy + yz + x'z$ kan förenklas till
konjunktiv (minimal) form: (*Produkt av Summor*)

$$\begin{aligned}
 f(x,y,z) &= xy + yz + x'z \\
 &= xy + yz + xx' + x'z && \text{sats 4} \\
 &= y(x + z) + x'(x + z) && \text{sats 2} \\
 &= (x + z)(y + x') && \text{sats 2}
 \end{aligned}$$

Grindnät för exempel Kalle

Ext 3

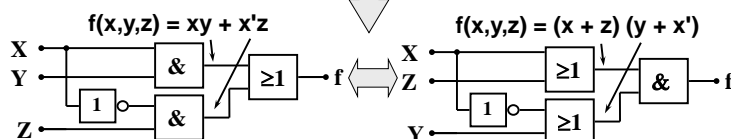


► Kunna skriva uttryck på Normal / Minimal form

Disjunktiv normal form och konjunktiv normal form

Disjunktiv (minimal) form
(Summa av Produkter)

Konjunktiv (minimal) form
(Produkt av summer)



Grundläggande Datorteknik OH LV1

53

Vårt gamla exempel Kalle: $f(x,y,z) = x y + y z + x'z$

Rita funktionstabell

(3 variabler - xyz - hur många rader i funktionstabellen?)

$x,y,z \Rightarrow 2^3 = 8$ rader

Grundläggande Datorteknik OH LV1

54

Disjunktiv normal form och konjunktiv normal form

Vårt gamla exempel Kalle: $f(x,y,z) = x y + y z + x'z$

Ext 3

Rita funktionstabell (3 variabler $\Rightarrow 2^3 = 8$ rader)

	x	y	z	xy	yz	x'z	f
0)	0	0	0	0	0	0	0
1)	0	0	1	0	0	1	1
2)	0	1	0	0	0	0	0
3)	0	1	1	0	1	1	1
4)	1	0	0	0	0	0	0
5)	1	0	1	0	0	0	0
6)	1	1	0	1	0	0	1
7)	1	1	1	1	1	0	1

Grundläggande Datorteknik OH LV1

55

Ext 3

x	y	z	f
0)	0	0	0
1)	0	0	1
2)	0	1	0
3)	0	1	1
4)	1	0	0
5)	1	0	0
6)	1	1	0
7)	1	1	1

Invariabelkombinationen (x, y, z) i varje rad i funktionstabellen är unik.

Ta fram en unik produkt av invariabler som ger $1 \cdot 1 \cdot 1 = 1$. (Tex ger rad 6 $x y z'$)

Ta fram produkter för samtliga rader där funktionsvärdet är 1 och addera dessa.

Summan blir ett uttryck för den Booleska funktionen $f = x'y'z + x'y z + x y z' + x y z$

Varje term är unik och innehåller samtliga invariabler.

Termerna kallas **mintermer** och funktionen sägs vara skriven på **disjunktiv normal form**.

Grundläggande Datorteknik OH LV1

56

x	y	z	f
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Invariabelkombinationen (x, y, z) i varje rad i funktionstabellen är unik.

Ta fram en unik summa av invariabler som ger $0+0+0=0$. (Tex ger rad 4 $x'+y+z$)

Ta fram summor för samtliga rader där funktionsvärdet är 0 och multiplicera dessa.

Produkten blir ett uttryck för den Booleska funktionen

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

Varje faktor är unik och innehåller samtliga invariabler. Faktorena kallas **maxtermer** och funktionen sägs vara skriven på **konjunktiv normal form**.

I DAG

Fö3

Dagens mål:

- ▶ Kunna jobba med booleska uttryck och funktioner
- ▶ Kunna ange uttryck på Disjunktiv och Konjunktiv form
- ▶ Kunna skriva uttryck på Normal / Minimal form
- ▶ Kunna ange Mintermer och Maxtermer
- ▶ **Kunna minimera funktioner med hjälp av Karnaughdiagram**
- ▶ Kunna Analysera och Minimera grindnät
- ▶ Konstruera grindnät bestående av NAND och NOR logik

**Läs smart!
Lär dig mer!**

Funktionstabell

x	y	z	f(x, y, z)
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Gray-kodat

Alternativ uppställning S3.25-27

x	y	z	f(x, y, z)
0	0	0	0
1	0	0	1
3	0	1	1
2	0	1	0
6	1	0	1
7	1	1	1
5	1	0	0
4	1	0	0

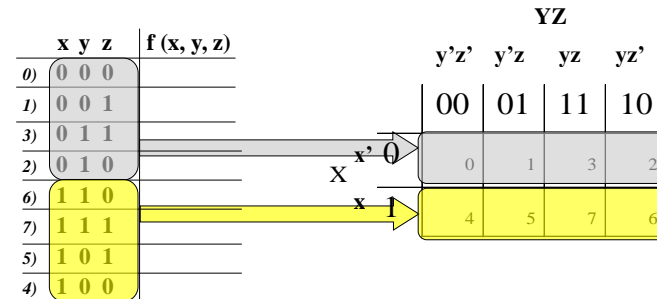
$$f = xy + yz + x'z$$

1) och 3) $x'y'z + x'yz = x'z(y' + y) = x'z$

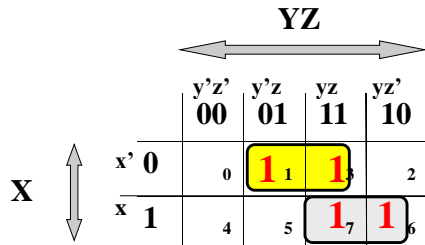
3) och 7) $x'yz + xyz = yz(x' + xy) = yz$

6) och 7) $xyz' + xyz = xy(z' + z) = xy$

Ännu bättre



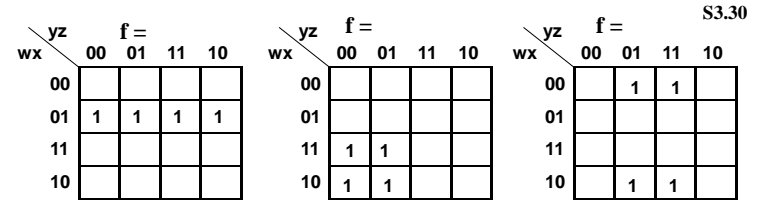
Karnaughdiagram



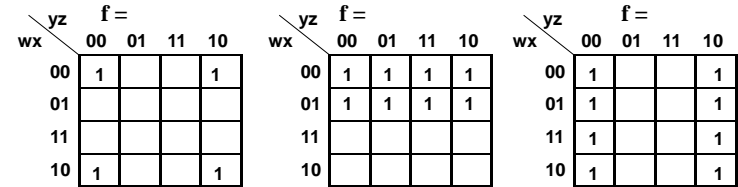
X är noll
Y är don't care
Z är ett
X'Z

X är ett
Y är ett
Z är don't care
XY

$$f_{\min} = X'Z + XY$$



figurerna 3.27 och 3.28



Sammanfattning

- NORMAL form ⇒ Funktionstabell
- MINIMAL form ⇒ Karnaughdiagram

- DISJUNKTIV (normal / minimal) form
 - Σ av Prod Ex: $(x'y)+(xw)+(xyw)$
 - Ettor
 - Mintermer: $(1 \cdot 1 \cdot 1) = 1$
 - NAND / NAND - logik

LEEDS P

- KONJUNKTIV (normal / minimal) form
 - Prod av Σ:or Ex: $(x+z)(x'+z+w)(z'+w')$
 - Nollor
 - Maxtermer: $(0+0+0) = 0$
 - NOR / NOR - logik

I DAG

Fö3

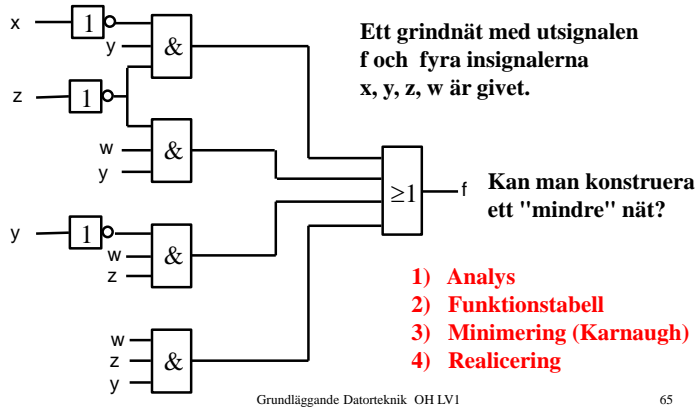
Dagens mål:

- ▶ Kunna jobba med booleska uttryck och funktioner
- ▶ Kunna ange uttryck på Disjunktiv och Konjunktiv form
- ▶ Kunna skriva uttryck på Normal / Minimal form
- ▶ Kunna ange Mintermer och Maxtermer
- ▶ Kunna minimera funktioner med hjälp av Karnaughdiagram
- ▶ Kunna Analysera och Minimera grindnät
- ▶ Konstruera grindnät bestående av NAND och NOR logik

Läs smart!
Lär dig mer!

Praktikfall, minimering av grindnät

Ext4



I DAG

Fö3

Dagens mål:

- ▶ Kunna jobba med booleska uttryck och funktioner
- ▶ Kunna ange uttryck på Disjunktiv och Konjunktiv form
- ▶ Kunna skriva uttryck på Normal / Minimal form
- ▶ Kunna ange Mintermer och Maxtermer
- ▶ Kunna minimera funktioner med hjälp av Karnaughdiagram
- ▶ Kunna Analysera och Minimera grindnät
- ▶ **Konstruera grindnät bestående av NAND och NOR logik**

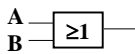
**Läs smart!
Lär dig mer!**

Grundläggande Datorteknik OH LV1

66

NAND-logik

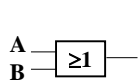
Om vi har A+B



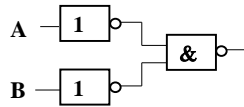
Hur realicera med NAND?

De Morgan:

$$\begin{aligned} (A+B)' &= A'B' \\ ((A+B)')' &= (A'B')' \\ A+B &= (A'B')' \end{aligned}$$



=

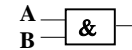


Grundläggande Datorteknik OH LV1

67

NOR-logik

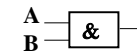
Om vi har A•B



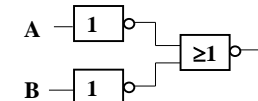
Hur realicera med NOR?

De Morgan:

$$\begin{aligned} (AB)' &= A' + B' \\ ((AB)')' &= (A' + B')' \\ AB &= (A' + B')' \end{aligned}$$



=



Grundläggande Datorteknik OH LV1

68