COS
Computersystemer

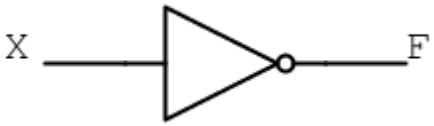
Lektion #2

Morten Hansen
moh@sdu.dk

Funktioner af enkelte bit

"Ikke" funktionen. Negationen. (NOT).

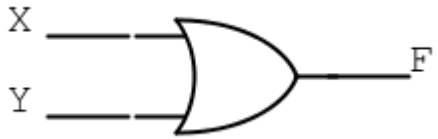
$F = \overline{X} = NOT\ X$



X	F
0	1
1	0

"Eller" funktionen. (OR).

$F = X + Y = X\ OR\ Y$



X	Y	F
0	0	0
0	1	1
1	0	1
1	1	1

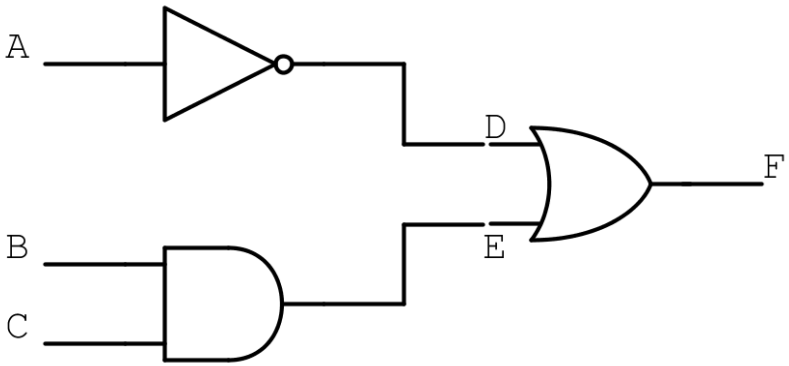
"Og" funktionen. (AND).

$F = X \cdot Y = X\ AND\ Y$

X	Y	F
0	0	0
0	1	0
1	0	0
1	1	1

Morten Hansen
moh@sdu.dk

Kombinerede funktioner



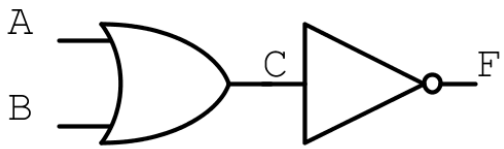
$$F = \overline{X} + Y \cdot Z$$

Input					F
A	B	C	D	E	F
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	0	1	1

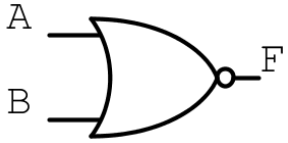
Morten Hansen
moh@sdu.dk

NOR og NAND

NOR – Not OR

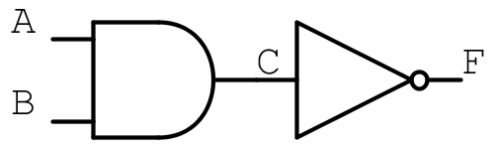


In			Out
A	B	C	F
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

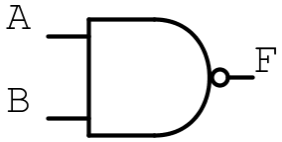


$$\begin{cases} F = \overline{C} \\ C = A + B \end{cases}$$
$$\Downarrow$$
$$F = \overline{A + B}$$

NAND – Not AND



In			Out
A	B	C	F
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

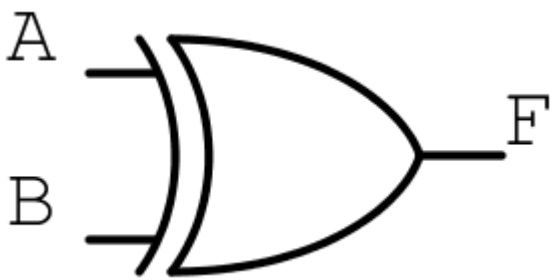


$$\begin{cases} F = \overline{C} \\ C = A \cdot B \end{cases}$$
$$\Downarrow$$
$$F = \overline{A \cdot B}$$

Morten Hansen
moh@sdu.dk

XOR

OR: ”Den ene eller den anden eller begge to”.
XOR: ”Den ene eller den anden med ikke begge to”.
XOR: De to input er forskellige.



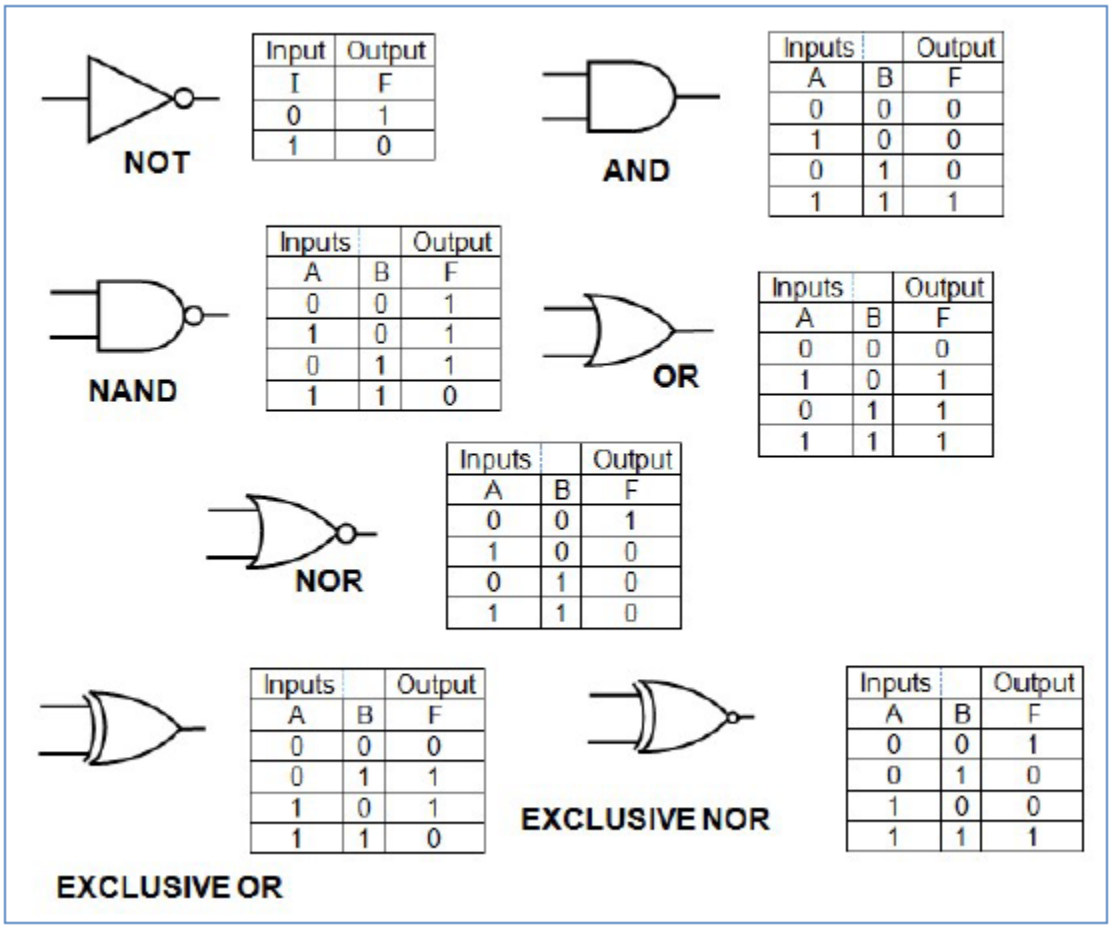
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

$$F = A \oplus B = A\ XOR\ B$$

Morten Hansen
moh@sdu.dk

Logiske gates

Logiske Gates og deres sandhedstabeller:



Input	Output
I	F
0	1
1	0

Inputs		Output
A	B	F
0	0	0
1	0	0
0	1	0
1	1	1

Inputs		Output
A	B	F
0	0	1
1	0	1
0	1	1
1	1	0

Inputs		Output
A	B	F
0	0	0
1	0	1
0	1	1
1	1	1

Inputs		Output
A	B	F
0	0	1
1	0	0
0	1	0
1	1	0

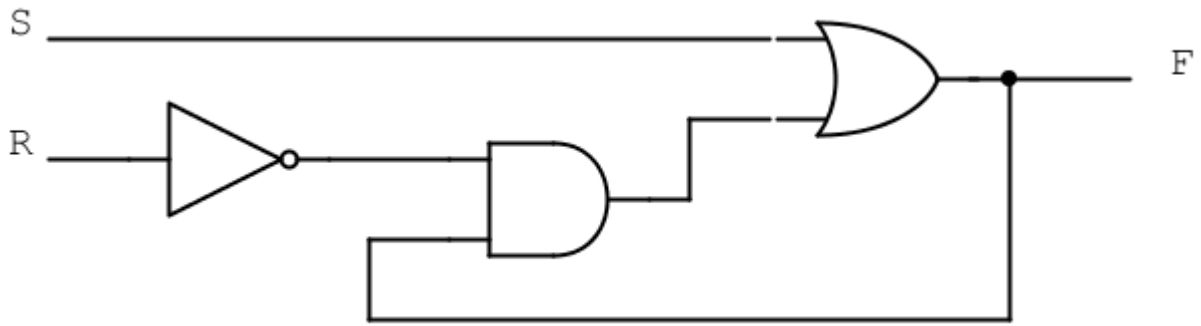
Inputs		Output
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

Inputs		Output
A	B	F
0	0	1
0	1	0
1	0	0
1	1	1

Se:
Note på BB.

Morten Hansen
moh@sdu.dk

Feed-back



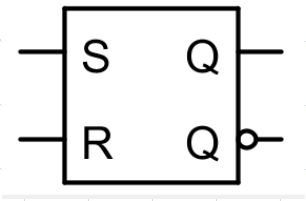
$$F := f(S, R, F)$$
$$\Downarrow$$
$$Hukommelse\ (Memory)$$

Morten Hansen
moh@sdu.dk

Flip-flops.

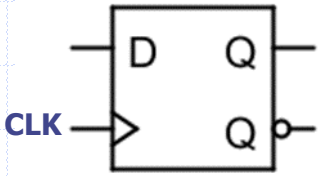
2 almindeligt forekommende flip-flop's

SR flip-flop:



Input		Output
S	R	Q
0	0	Last Q
0	1	0
1	0	1
1	1	Udefineret

D flip-flop:

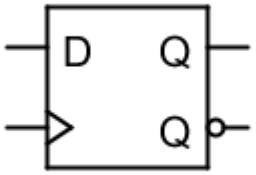


Input		Output
D	CLK	Q
X	0	Last Q
X	1	Last Q
0	↑	0
1	↑	1

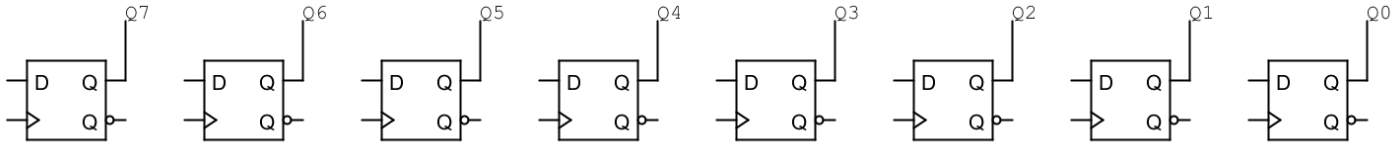
Morten Hansen
moh@sdu.dk

D-flip-flop

En abstraktion



8 flip-flop's – 8 bit

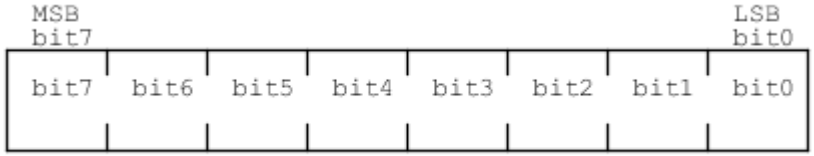


Morten Hansen
moh@sdu.dk

Bits, bytes og main memory

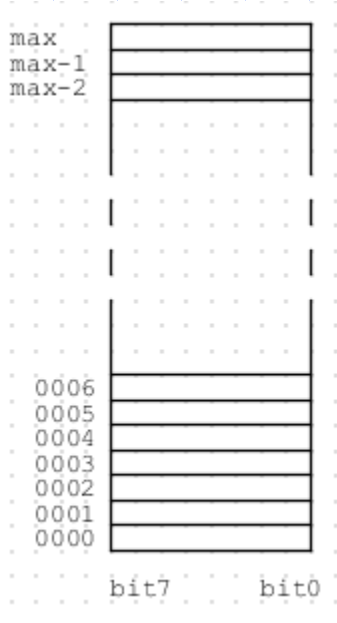
Bit:

- Mindst mulige mængde information.
- Kan antage 1 ud af 2 mulige værdier:
- 0 eller 1



Byte:

- Består af 8 bit.
- Kan antage 1 ud af 2^8 mulige værdier.
- =256 mulige, forskellige værdier, mønstre eller koder.



Main memory

- Består af en række celler
- Hver celle kan indeholde 8 bit information
- Hver celle har en adresse

Morten Hansen
moh@sdu.dk

Repræsentation af tekst

- 1 byte kan antage 256 forskellige værdier.
- Der er 28 karakterer i alfabetet.
- Store og små karakterer + tal + special tegn + kontroltegn -> ca. 127 forskellige tegn.
- For en nemheds skyld bruger vi 8 bit/karakter.
- ASCII tabel.
- For at skrive en tekst skal vi altså bruge en byte pr. tegn => En memorycelle pr. tegn.

Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII
32	00100000	040	20	SP	64	01000000	100	40	@
33	00100001	041	21	!	65	01000001	101	41	A
34	00100010	042	22	"	66	01000010	102	42	B
35	00100011	043	23	#	67	01000011	103	43	C
36	00100100	044	24	$	68	01000100	104	44	D
37	00100101	045	25	%	69	01000101	105	45	E
38	00100110	046	26	&	70	01000110	106	46	F
39	00100111	047	27	'	71	01000111	107	47	G
40	00101000	050	28	(	72	01001000	110	48	H
41	00101001	051	29	)	73	01001001	111	49	I
42	00101010	052	2A	*	74	01001010	112	4A	J
43	00101011	053	2B	+	75	01001011	113	4B	K
44	00101100	054	2C	,	76	01001100	114	4C	L
45	00101101	055	2D	-	77	01001101	115	4D	M
46	00101110	056	2E	.	78	01001110	116	4E	N
47	00101111	057	2F	/	79	01001111	117	4F	O
48	00110000	060	30	0	80	01010000	120	50	P
49	00110001	061	31	1	81	01010001	121	51	Q
50	00110010	062	32	2	82	01010010	122	52	R
51	00110011	063	33	3	83	01010011	123	53	S
52	00110100	064	34	4	84	01010100	124	54	T
53	00110101	065	35	5	85	01010101	125	55	U
54	00110110	066	36	6	86	01010110	126	56	V
55	00110111	067	37	7	87	01010111	127	57	W
56	00111000	070	38	8	88	01011000	130	58	X
57	00111001	071	39	9	89	01011001	131	59	Y

Morten Hansen
moh@sdu.dk

Talsystemer

10-tals systemet (decimal)

Basetal = 10

10^2	10^1	10^0	
2	9	3	= 293

Det binære talsystem

Basetal = 2

2^2	2^1	2^0	
1	0	1	= 5

16 talsystemet

Basetal = 16

16^2	16^1	16^0	
1	E	4	= 484

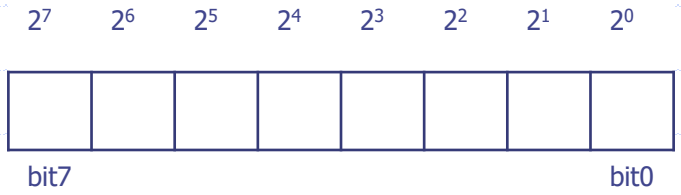
Morten Hansen
moh@sdu.dk

Hexadecimale cifre

Bit				Hex
3	2	1	0	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	B
1	1	0	0	C
1	1	0	1	D
1	1	1	0	E
1	1	1	1	F

Morten Hansen
moh@sdu.dk

8 bit binære koder



- 8 bit: bit 0 – bit 7
- $2^8 = 256$ mulige koder.
- Mindst mulige tal: $0 = 00000000_2$
- Størst mulige tal: $255 = 11111111_2$
- Eks.: $10100011_2 = 128 + 32 + 2 + 1 = 163$

Morten Hansen
moh@sdu.dk

Binær addition

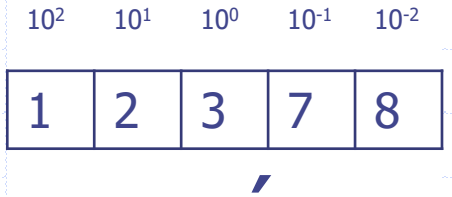
```
  1111
  00110100      52
+ 00101110    + 46
  --------      --
  01100010      98
```


Morten Hansen
moh@sdu.dk

Brøker (Fixed point)

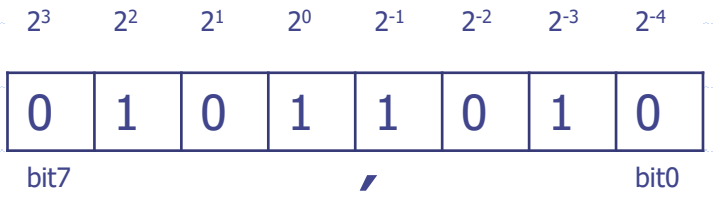
Brøker i 10-tals systemet:

123,78



Brøker i det binære system:

4.4 system:



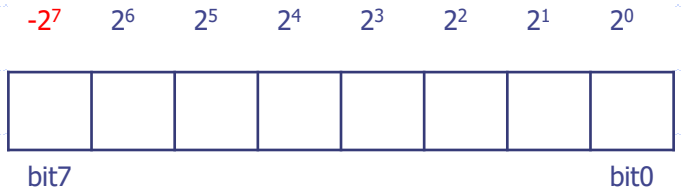
$101{,}101_2 = 4 + 1 + 0{,}5 + 0{,}125 = 5{,}625$

Størst mulige tal: $15{,}9375 = 11111111_2$

Mindst mulige tal: $0 = 000000000_2$

Morten Hansen
moh@sdu.dk

2's complement



- Mulighed for negative tal
  - Størst mulige tal: 127 = 01111111_2
  - Mindst mulige tal: -128 = 10000000_2
- Addition
- Overflow

Addition af 2'compl. Tal:

$$\begin{array}{rr} 00110000_2 & 48 \\ + \ 00010010_2 & 18 \\ \hline 01000010_2 & 66 \end{array}$$

Pas på overflow:

$$\begin{array}{rr} 01010000_2 & 80 \\ + \ 00111110_2 & 62 \\ \hline 10001110_2 & -114 \end{array}$$

Morten Hansen
moh@sdu.dk

Excess koder

Bit mønster	Værdi
1111	7
1110	6
1101	5
1100	4
1011	3
1010	2
1001	1
1000	0
0111	-1
0110	-2
0101	-3
0100	-4
0011	-5
0010	-6
0001	-7
0000	-8

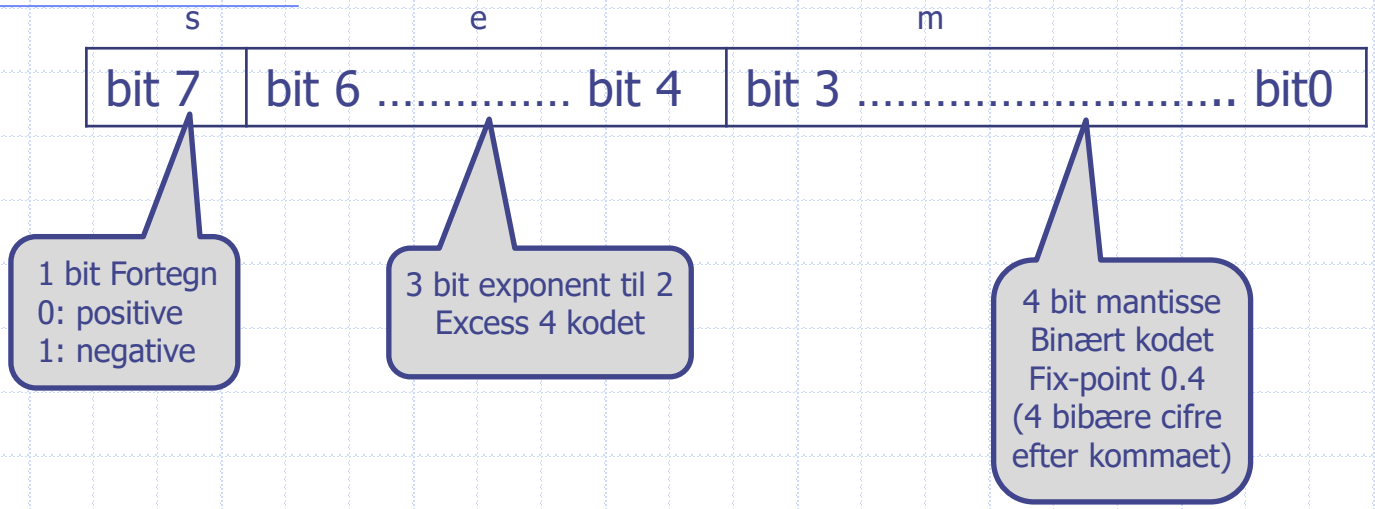
4 bit Excess 8 tabel

Bit mønster	Værdi
111	3
110	2
101	1
100	0
011	-1
010	-2
001	-3
000	-4

3 bit Excess 4 tabel

Morten Hansen
moh@sdu.dk

Floating point



$$Værdi = s \cdot m \cdot 2^e$$

Eks.:

$11011100_2 =$

$-1 \cdot 0{,}75 \cdot 2^1 = 1{,}5$

Range:
7,5..0,0039, 0, -0,0039..7,5

Pas på afrunding!

Eks.:

$3{,}125 \approx 01101100_2 = 3{,}00$

$3{,}125 \approx 01101101_2 = 3{,}25$

Morten Hansen
moh@sdu.dk

Spørgsmål?

Morten Hansen
moh@sdu.dk