

# UNIVERSITETET I OSLO

## Det matematisk-naturvitenskapelige fakultet

Eksamen i: INF1400

**Eksamensdag: 02. desember 2009**

**Tid for eksamen: kl. 09.00 – 13.00**

**Oppgavesettet er på 4 side(r)**

**Vedlegg: 0 sider**

**Tillatte hjelpemidler: Alle trykte og skriftlige, samt kalkulator**

*Kontroller at oppgavesettet er komplett før du begynner å besvare spørsmålene.*

### Oppgave 1. Binære tall

**Her godtar vi nesten alle svar. Dad et er skrivefeil i oppgaveteksten.**

$$61 = 111101$$

$$27 = 11011$$

$$61 = 00111101 \text{ (signed)}$$

$$-61 = 11000011$$

$$27 = 00011011 \text{ (signed)}$$

$$-27 = 11100101$$

$$\begin{aligned} 27 - 61 &= 27 + (-61) \\ &= 0001\ 1011 \\ &+ 1100\ 0011 \\ &= 1101\ 1110 = -(0010\ 0001 + 1) = -(0010\ 0010) \end{aligned}$$

$$-27 + 61 = 1110\ 0101 + 0011\ 1101 = (1)\ 0010\ 0010$$

$$-27 - 61 = -27 + (-61) = 1110\ 0101 + 1100\ 0011 = (1)\ 1010\ 1000 = -(0101\ 0111 + 1) = -(0101\ 1000)$$

Gjør om til signed 2'er komplement:

- a) 61
- b) 27

Gjør om til unsigned 2'er komplement:

- a) 61
- b) -61

- c) 27
- d) -27

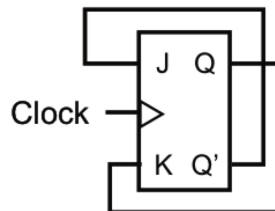
Gjør om tallene til unsigned 2'er komplement og vis utregning

- a)  $27 - 61$
- b)  $-27 + 61$
- c)  $-27 - 61$

## Oppgave 2. Sekvensiell logikk

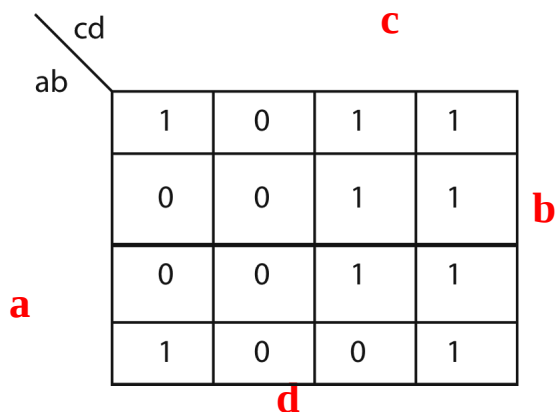
Hvis følgende krets har en initiell tilstand = 1, hva er da kretsens tilstand etter to klokkepuls? **Når vi snakker om initiell tilstand så er  $Q = 1$ , og etter to klokkepuls vil den være  $Q = 1$ .**

**Start:**  $Q = 1$   
**Klokkepuls 1:**  $Q = 0$   
**Klokkepuls 2:**  $Q = 1$



## Oppgave 3. Karnaugh diagram

- a) Forenkle følgende Karnaugh diagram, og finn det forenklete kombinatoriske uttrykket:



**Her skal de ha lært seg hvordan man betegner en karnaughdiagram. Det vil si hvilke område a, b, c og d dekker. Som vist på figuren over. F blir så:**

$$F = b'd' + bc + a'c$$

- b) Tegn Karnaugh diagrammet for funksjon, F, gitt under og finn det forenklete uttrykket.  
 $F(A,B,C,D) = \Sigma(0, 1, 2, 4, 5, 7, 11, 15)$

**Her blir forkortningen av  $F = a'c' + acd + a'b'd + bcd$**

Eller  
Eller  
Eller

$$F = a'c' + acd + a'b'd + a'bd$$

$$F = a'c' + acd + a'b'd' + a'c(b \text{ XNOR } d)$$

$$F = a'c' + acd + a \text{ XOR } b \text{ XOR } c$$

## Oppgave 4. Sekvensiell logikk

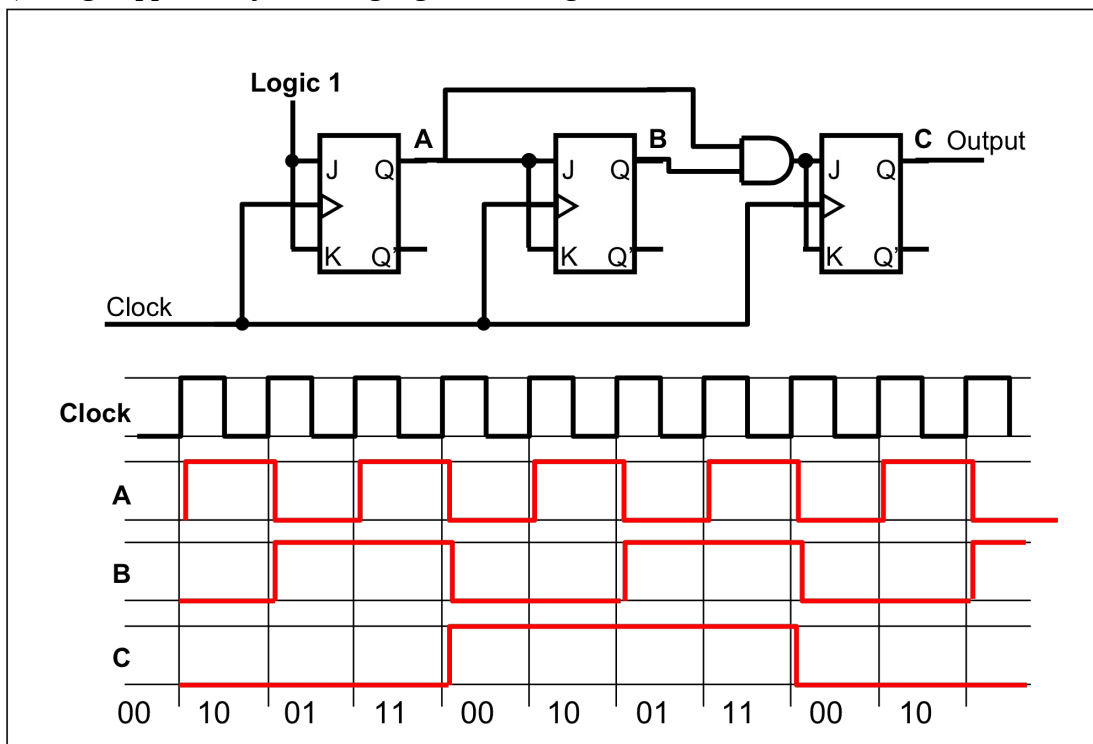
Den sekvensielle kretsen under har bare klokkesignal som eneste inngang (input).

a) Hvis systemet starter i tilstand 000, hva blir så de neste 9 tilstandene?

**Vi tenker oss  $ABC = 000$  og da blir de neste 9 tilstandene**

Clock	State	output
0	000	0
1	100	0
2	010	0
3	110	0
4	001	1
5	101	1
6	011	1
7	111	1
8	000	0
9	100	0

b) Tegn opp tidslinjen for utgangene A, B og C.



c) Hvis utgangen er C, hvilken funksjon utfører systemet?

**Noen vil svare en rippelteller, og det er i og for seg riktig, Men svaret vi er ute etter er at dette er en krets som deler klokkefrekvensen på 8.**

d) Hva begrenser klokkefrekvensen? Drøft hvor mange ledd kan man legge til.

**Det som begrenser max klokkefrekvensen er primært portforsinkelse og tiden det har å propagere signalet gjennom hver del ledd. Antall ledd man kan legge til er nærmest uendelig, da hvert ledd ikke er kun avhengig av det foregående. AND-porten vil sørge for at signalet propagerer raskt til ledd lengre fremme i rekken.**

# Oppgave 5. Boolsk algebra

Gitt den boolske funksjonen

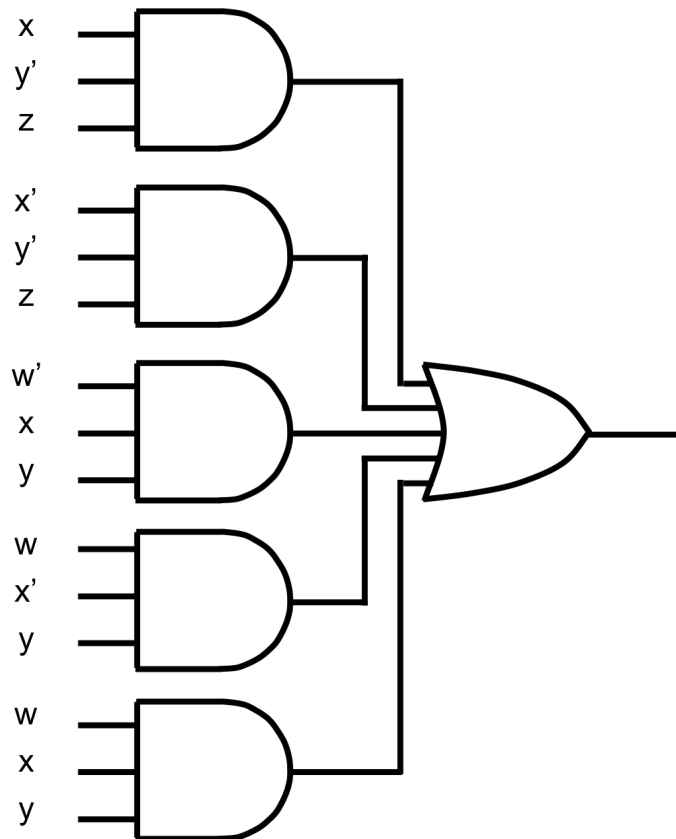
$$F = xy'z + x'y'z + w'xy + wx'y + wxy$$

a) Skriv opp sannhetsverditabellen

w	x	y	z	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
<hr/>				
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
<hr/>				
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
<hr/>				
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

b) Tegn det logiske diagrammet til F, bruk kun 2-inputs logiske porter.

b)



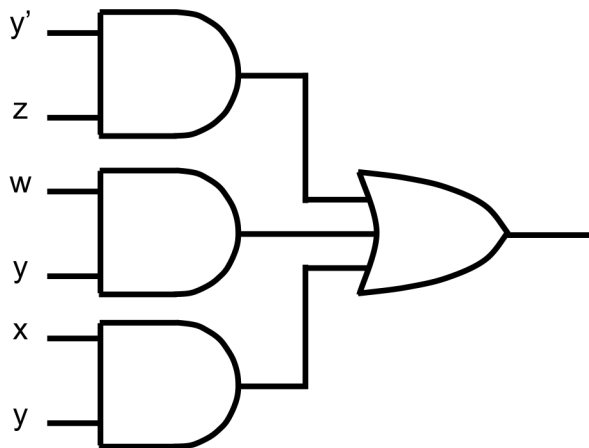
Det å gjøre om 3 inputs AND og 5 inputs OR til 2-inputs ansees å være lett å gjøre etter at man har tegnet funksjonen.

c) Reduser antall ledd ved hjelp av regnereglene for boolsk algebra

$$F(w, x, y, z) = xy'z + x'y'z + w'xy + wx'y + wxy$$

$$\begin{aligned}
 &(x + x')y'z + w'xy + wy(x' + x) \\
 &y'z + w'xy + wy \\
 &y'z + w'xy + w(y+xy) \\
 &y'z + wy + xy
 \end{aligned}$$

- d) Skriv opp sannhetsverditabellen til det reduserte uttrykket  
**Naturligvis blir den helt lik oppgave (a), hvis den blir forskjellig så har man gjort noe feil.**
- e) Tegn det logiske diagrammet til det reduserte uttrykket



The original design used 5 3-input ANDs plus a 5-input OR. This design uses 3 2-input ANDs plus a 3-input OR, eliminating  $1/3^{\text{rd}}$  of the gates and reducing the number of inputs on all of them by over  $1/3^{\text{rd}}$ .

## Oppgave 6. Boolsk algebra

- a) Gjør om til sum-av-produkter:  $(AB+C)(B+C'D)$

**Kan gjøre på to måter, (1) settes inn i karnaughdiagram og forkortes, eller (2) regnes for hånd:**

$$\begin{aligned}
 &= (AB + C)(B+C'D) = ABB + ABC'D + BC + CC'D \\
 &= AB + ABC'D + BC \\
 &= AB + BC
 \end{aligned}$$

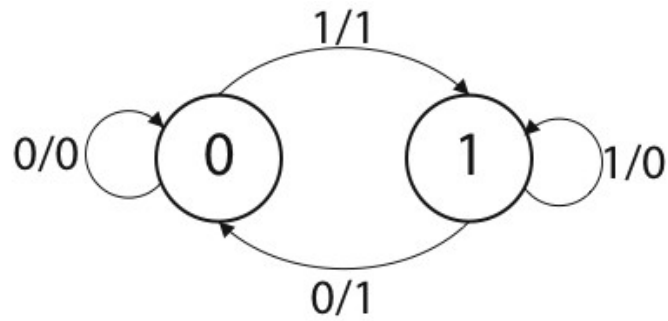
- b) Gjør om til produkt-av-sum:  $x' + x(x+y')(y+z')$

**Kan gjøres på to måter, (1) settes inn i karnaughdiagram og plukke ut 0'ere, eller (2) regne seg frem for hånd:**

$$\begin{aligned}
 &= x' + (x+yy')(x+y')(y+z') \\
 &= x' + (x+y)(x+y')(x+y')(y+z') \\
 &= x' + (x+y)(x+y')(y+z') \\
 &= (x'+x+y)(x'+x+y')(x'+y+z') \\
 &= (1+y)(1+y')(x'+y+z') \\
 &= (1)(1)(x' + y + z') \\
 &= (x'+y+z')
 \end{aligned}$$

## Oppgave 7. Tilstandsmaskin

- a) Lag tilstandstabellen for tilstandsdiagrammet under:



Q(t)	x	ut	Q(t+1)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

b) Tegn det logiske diagrammet som gjenspeiler oppførselen til tilstandsmaskinen.

## Oppgave 8. VHDL

a) Hvilken krets beskrives av følgende VHDL-kode? (Navn på signaler er tilfeldig valgt).

### JK FLIPFLOP

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity krets1 is
port (
    A : in std_logic;
    B : in std_logic;
    C : in std_logic;
    D : out std_logic
);
end krets1;
architecture behavioral of krets1 is
    signal E : std_logic;
begin
    process (C)
    begin
        if C = '1' and C'event then
            E <= (A and not (E)) or (not (B) and E);
        end if;
    end process;
    D <= E;
end behavioral;
  
```

b) Hvilken krets beskrives av følgende VHDL-kode? (Navn på signaler er tilfeldig valgt).

### 1-BITS FULL ADDER

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity krets2 is
    port(
        a : in std_logic;
        b : in std_logic;
        c : in std_logic;
        d : out std_logic;
        e : out std_logic
    );
end krets2;
architecture behavioral of krets2 is begin

    process(a,b,c) begin
        if a = '0' then
            if (b and c) = '1' then
                d <= '0';
                e <= '1';
            elsif (b or c) = '1' then
                d <= '1';
                e <= '0';
            else
                d <= '0';
                e <= '0';
            end if;
        elsif a = '1' then
            if (b and c) = '1' then
                d <= '1';
                e <= '1';
            elsif (b or c) = '1' then
                d <= '0';
                e <= '1';
            else
                d <= '1';
                e <= '0';
            end if;
        end if;
    end process;
end behavioral;

```