

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i:	INF1400 – Digital teknologi
Eksamensdag:	5. desember 2005
Tid for eksamen:	9-12
Vedlegg:	Ingen
Tillatte hjelpemidler:	Alle trykte og skriftlige samt kalkulator
Oppgavesettet er på 2 sider	

Kontroller at oppgavesettet er komplett før du begynner å besvare spørsmålene

Oppgave 1 – Karnaughdiagram (vekt 15%)

1a Bruk Karnaughdiagram for å forenkle følgende funksjon

$$Y = A' B' C' D' + A' B' C D' + ABCD'$$

1b Anta følgende: hvis både $A=1$ og $B=0$ og $C=1$ samtidig, kan ikke D være lik 0. Vis hvordan denne antagelsen kan gi et enklere uttrykk for Y .

Oppgave 2 – Boolsk algebra (vekt 14%)

2a Vis hvordan man kan forenkle følgende uttrykk ved å bruke den distributive loven

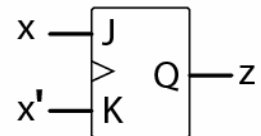
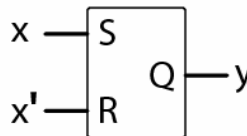
$$F = A + (B \cdot A')$$

2b Bruk DeMorgan's teorem for å forenkle følgende uttrykk (reducere antall inverteringer)

$$G = (A' + (B' C'))'$$

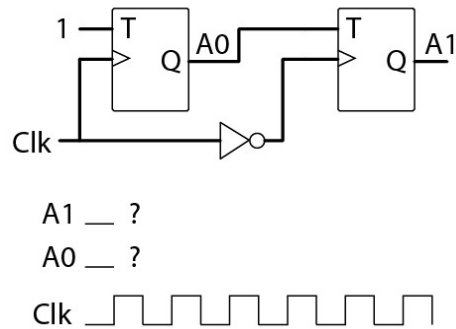
Oppgave 3 – Sekvensiell logikk (vekt 14%)

I figurene til høyre er det vist to komponenter. Beskriv forskjeller / likheter i virkemåte. Hva gjør de to komponentene når x varierer (anta at nødvendige aktive klokkesignaler er koblet til).



Oppgave 4 – Sekvensielle systemer (vekt 14%)

I figuren til høyre er det vist en sekvensiell krets. Skisser hvordan signalene A0 og A1 forandrer verdi i løpet av 6 klokke (Clk) perioder. Anta at A0 og A1 begge er 0 i utgangspunktet



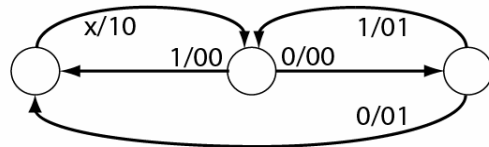
A1 __ ?

A0 __ ?



Oppgave 5 – Tilstandsmaskin (vekt 16%)

5a Til høyre er det vist et tilstandsdiagram for en tilstandsmaskin. Finn ut hvor mange innganger/utganger denne maskinen har, velg tilstandskoder og tegn opp en tilstandstabell basert på diagrammet. Hva betyr "x" i figuren? Anta at systemet alltid vil starte opp i tilstanden helt til venstre i diagrammet.



5b Tegn opp denne tilstandsmaskinen basert på D flip-flop'er og nødvendige porter.

Oppgave 6 – Minne/adder (vekt 13%)

Vi ønsker å lage en binær adder som kan addere to binære tall A og B. Tall A er representert ved 2-bit, mens tall B er bare representert ved 1-bit. Adderen skal ikke ha mente (carry) inn og trenger 3-bit ut. Vi har en ROM minnekrets som kan lagre 8 stk. 4-bits ord. Adresseringen av ROM'en foregår på vanlig måte (binært). Vi vil bruke denne ROM-kretsen til å implementere hele adderen, og vi vil ikke bruke noe ekstra logikk utenom denne ROM-kretsen. Vis hvordan signalene A og B må kobles til ROM-kretsen. Spesifiser hva innholdet i ROM-kretsen (innholdet i hver minnecelle) må være for at ROM-kretsen skal summere A og B riktig.

ord nr.0	?	?	?	?
ord nr.1	?	?	?	?
ord nr.2	?	?	?	?
ord nr.3	?	?	?	?
ord nr.4	?	?	?	?
ord nr.5	?	?	?	?
ord nr.6	?	?	?	?
ord nr.7	?	?	?	?

Oppgave 7 – VHDL (vekt 14%)

7a VHDL koden til høyre beskriver en mye brukt krets. Forklar hva denne kretsen gjør.

7b Skisser grafisk med en tegning av porter og ledninger hvordan denne kretsen kan kobles opp.

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity krets is
    Port ( A : in std_logic_vector(1 downto 0);
          WL : out std_logic_vector(3 downto 0));
end krets;

architecture Behavioral of krets is
begin
    process(A)
    begin
        case A is
            when "00" =>
                WL <= "0001";
            when "01" =>
                WL <= "0010";
            when "10" =>
                WL <= "0100";
            when others =>
                WL <= "1000";
            end case;
        end process;
    end Behavioral;

```

Lykke til!