

INF3430/4431 Høsten 2015

# **Laboppgave 1**

## **Designflyt og VHDL**

**NB! Før dere starter les: "Obligatoriske og andre innleveringer ved Institutt for Informatikk" på siden <http://www.mn.uio.no/ifi/studier/admin/obliger/index.html>**

Målet med den første laboppgaven er å øve oss i VHDL og bli kjent med designflyten og utviklingsverktøyene Questa (Modelsim) og Xilinx ISE.

**NB! Bruk deres private område på nettet (M-disk) for å lagre kilde- og resultatfiler. Dersom nettverksresponsen er dårlig opprett et midlertidig område på lokal harddisk. Husk å slette dette området etterpå.**

**Alle innleverte VHDL-filer skal følge retningslinjene for indentering som beskrevet i kokeboken.**

**Tabell 1. Vedlagte filer. Comp\_first.do må fylles ut med egen path.**

first.vhd	VHDL-kildefil oppgave 1
tb_first.vhd	VHDL-testbenk oppgave 1
comp_first.do	Modelsim kommandofil for kompilering
sim_first.do	Modelsim kommandofil for simulering
delay.vhd	VHDL-kildefil oppgave 3
tb_delay.vhd	VHDL-testbenk oppgave 3
variables_vs_signals.vhd	VHDL-kildefil oppgave 3
tb_variables_vs_signals.vhd	VHDL-testbenk oppgave 3
dff.vhd	VHDL-kildefil oppgave 4

## Oppgave 1. Designflyt

### a).

Følg opplegget i kokeboken "Designflyt og utviklingsverktøyene Modelsim og Xilinx ISE"

Alle nødvendige filer hentes fra websidene til INF3430.

Simulerer kretsen på RTL-nivå og Post-Route (uten timing). Etterpå skal designet lastes ned på testkortet.

Før dere programmerer kretsen ta en titt i PAD-rapporten for å sjekke at pinnenummerene er riktig plassert. Dette betyr å sjekke opp mot pinnetilordning gitt i "Starter Kit Board Users guide":

[http://www.xilinx.com/support/documentation/boards\\_and\\_kits/ug130.pdf](http://www.xilinx.com/support/documentation/boards_and_kits/ug130.pdf)

Sjekk *Post Place and Route Static Timing Report* for å sjekke at "timing constraints" er tilfredsstillt.

### Godkjenning:

Ingen spesielle krav.

### b).

Forandre VHDL-koden slik at telleren blir en opp/ned teller ved å legge til et signal, "up".

Signalet "up" skal tilordnes SW6 (pin K14). Dersom up='1' skal telleren telle oppover og dersom up='0' skal telleren telle nedover.

I tillegg skal vi legge til et signal "min\_count" som gir en puls med varighet 1 klokkeperiode når telleren går til 0 på vei nedover. Min\_count skal tilordnes lysdioden LD6. Max\_count skal gjøres om slik at den gir en puls på max. verdi på vei oppover.

Legg til nødvendig endring i UCF-filen til Xilinx ISE for at up og min\_count skal tilordnes rett pinnenummer.

Gjør om testbenken slik at vi får verifisert at telleren fungerer riktig. Det er f.eks. greit med en tellesekvens fra 0 til 15, for deretter å snu telleretning og telle ned til 0.

Utfør Post-Route simulering (uten timing), verifiser pinneplassering og programmer kretsen når alt synes korrekt.

### **Godkjenning:**

Endret VHDL-fil, testbenk, endret UCF-fil og PAD-rapport.

## **Oppgave 2.**

I denne deloppgaven skal dere konstruere en 2 til 4-bits dekode. I VHDL læreboka brukes concurrent statements i eksemplet i kapittel 4, men dere skal bruke case statement i en process isteden. Utgangene på dekodeen skal være aktivt lave. Dekodeen skal implementeres på testkortet. Bruk SW1 og 2 som input og LD1, 2, 3 og 4 som output.

### **Godkjenning:**

VHDL-designfil, testbenk, do-fil for kompilering og visning av waveform, samt UCF-fil. Oppgaven må demonstreres for labveileder for å kunne bli godkjent.

## **Oppgave 3**

I denne deloppgaven skal vi se på forskjellen mellom signaler og variabler i VHDL.

### **a).**

Simuler den vedlagte koden i delay.vhd og tb\_delay.vhd. Når endrer outdata signalet seg og hva fører til denne forsinkelsen?

### **b).**

Endre delay.vhd slik at alle variabler blir erstattet med signaler (TIPS: signaler kan ikke deklarerer inne i en prosess). I tillegg skal tb\_delay.vhd endres slik at kretsen bare er i reset fra tiden 100 ns til 200 ns (TIPS: endre bl. a. til verdien '1' fra tid null) og inndata skal nå endre seg fra "00000000" ved 0 ns (dvs. start) til "11110000" ved tid 300 ns og til "00001111" ved tid 400 ns. Når endrer outdata signalet seg nå? Hvorfor er outdata lik "UUUUUUUU" ved tid 50 ns?

### **c).**

I denne deloppgaven skal den vedlagte koden i variables\_vs\_signals.vhd og tb\_variables\_vs\_signals.vhd simuleres. Hvorfor blir outdata(7 downto 6) alltid lik outdata(3 downto 2)? Hvorfor blir outdata(5 downto 4) ulik outdata(1 downto 0)?

### **d).**

Nå skal signalene sig1 og sig2 fjernes fra sensitivitetslista i variables\_vs\_signals.vhd. Hvorfor får outdata(7 downto 6) og outdata(3 downto 2) andre verdier enn i oppgave c?

## **Oppgave 4**

I denne deloppgaven skal vi se på instansiering av komponenter. Den vedlagte koden i dff.vhd er et register med asynkron reset hvor registeret får verdien '0' når signalet rst\_n er aktivt lavt (dvs. får verdien '0').

### **a).**

Lag et 8 bits shift register ved å instansiere komponenten dff 8 ganger, og kall denne komponenten for shift8. Bruk "named association" ved port map. Lag en testbenk som simulerer shiftregisteret.

**b).**

Lag et 32 bits shiftregister ved bruk av komponenten dff og generate statementet. Kall denne komponenten for shift32. Bruk "named association" ved port map. Lag en testbenk som simulerer shiftregisteret.

**c).**

Lag et N bits shiftregister ved å bruke generic statementet. Kall denne komponenten for shiftn. Lag en testbenk som simulerer en instansiering av shiftn med lengden satt til 64 bit.

**Godkjenning:**

VHDL kildefil og testbenk for de enkelte delspørsmålene