

# TSEA83 : Datorkonstruktion, 8hp

## Fö1

Introduktion

# Fö1 : Agenda

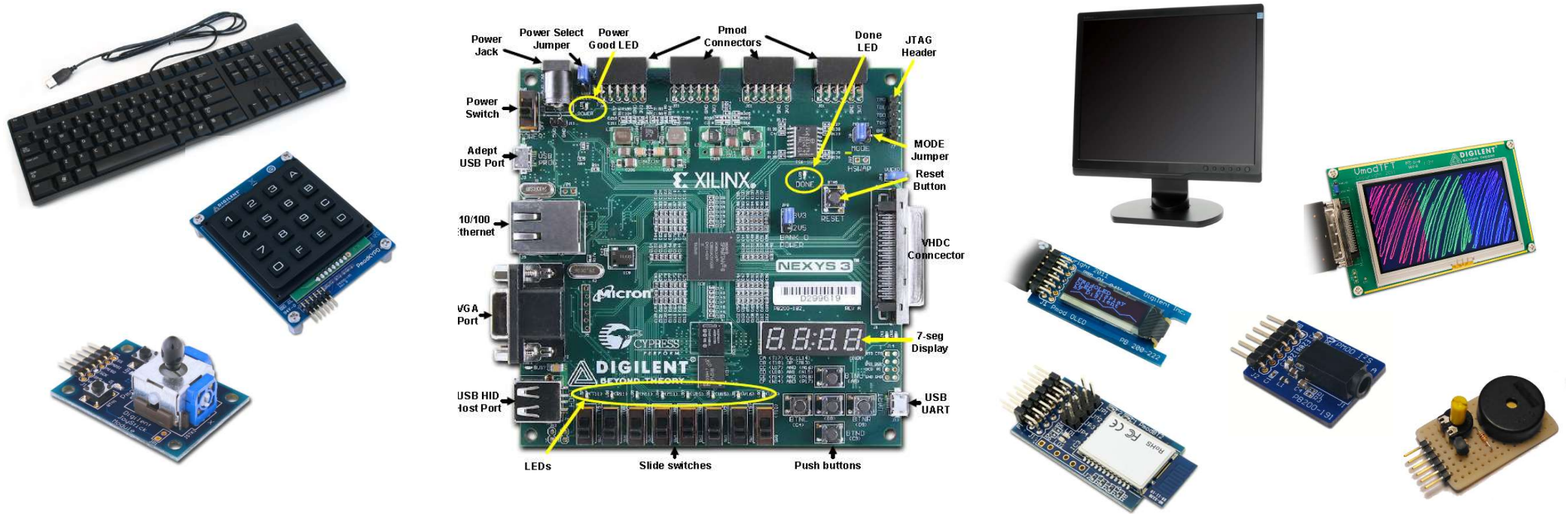
- Inledning
  - Mål, Resultat från tidigare år, Förändringar till årets kurs
- Formalia
  - Planering, Föreläsningar, Laborationer, Kurslitteratur
- VHDL
  - VHDL, Simulering, FPGA-kort
- Projekt
  - Exempel på tidigare projekt
- Distansarbete
  - Hur kan man arbeta hemifrån?

# Inledning

Mål, Resultat från tidigare år, Förändringar till årets kurs

# Mål

- Ni ska i grupper om 4 teknologer konstruera en inbyggd dator



## Resultat från tidigare år

- 86 studenter gick kursen 2021
- 14 studenter svarade på utvärderingen i Evaluate
- Helhetsbetyget var 4,86 (4,69 2020)

## Förändringar till årets kurs

- Veckovis tidrapport/statusrapport under projektet i VT2
- Föreläsningar på distans, labbar på plats
- Projektet i VT2, går förhoppningsvis på plats

# Formalia

Planering, Föreläsningar, Laborationer, Kurslitteratur

# Planering 2022, VT1 (2hp)

vecka	föreläsning	laboration	projekt
3	1. Intro 2. Mikroprog I 3. Mikroprog II		
4	4. Pipelining 5. Cache	1. Mikroprog	
5	6. Minnen +Bussar 7. Grafik +Proj	2. Pipelining	
6	8. VHDL I 9. VHDL II		Bilda 4-grupp klar må kl 16
7	10. VHDL III	3. UART (VHDL)	Kravspec 0.1 inlämn. må kl 16
8	11. ALU	4. VGA (VHDL)	Kravspec 1.0 klar må kl 16
9			Designspec 0.1 inlämn. må kl 16
10			Designspec 1.0 klar må kl 16

# Planering 2022, VT2 (6hp)

vecka	händelse	hållpunkt	projekt
11	Omtenta-P ht2		
12	Tenta-P VT1		
13	Fö11:Extra		Konstruktion
14		Tid/Status må kl 16	Konstruktion
15	(Påsk)	Tid/Status må kl 16	Konstruktion
16		Tid/Status må kl 16	Konstruktion
17	(Valborg+1Maj)	Milstolpe måndag	Konstruktion
18		Tid/Status må kl 16	Konstruktion
19		Tid/Status må kl 16	Konstruktion
20		<b>Examination må + ti</b>	



1. Intro
2. Mikroprogrammering 1
3. Mikroprogrammering 2
4. Pipelining
5. Cache
6. Minnen + Bussar
7. Grafik + Projekt
8. VHDL 1
9. VHDL 2
10. VHDL 3
11. ALU/Projektstart

} CPU

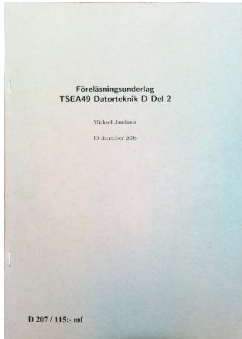
} VHDL är språket  
i projektet

Alla PP-slides på hemsidan  
+Föreläsningsanteckningar

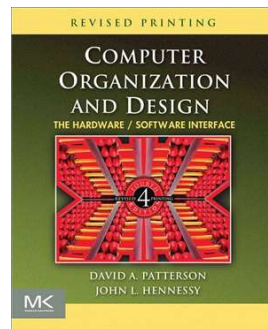
- 1. Mikroprogrammering (2 + 2 tim)**
  - Kan göras på egen hand och endast redovisas på labben
- 2. Pipelining (2 + 2 tim)**
  - Kan göras på egen hand och endast redovisas på labben
- 3. UART (seriell kommunikation) (2 + 2 tim)**
  - Kodning och simulering kan göras på egen hand.  
Syntetisering kräver FPGA-kort i labsal
- 4. VGA (2 + 2 tim)**
  - Kodning och simulering kan göras på egen hand.  
Syntetisering kräver FPGA-kort i labsal

Alla lab-pM på hemsidan

# Kurslitteratur - Datorteknikdelen



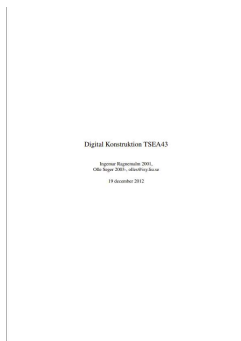
Michael Josefsson : Datorteknik D del 2  
→ mikroprogrammering, **pdf**



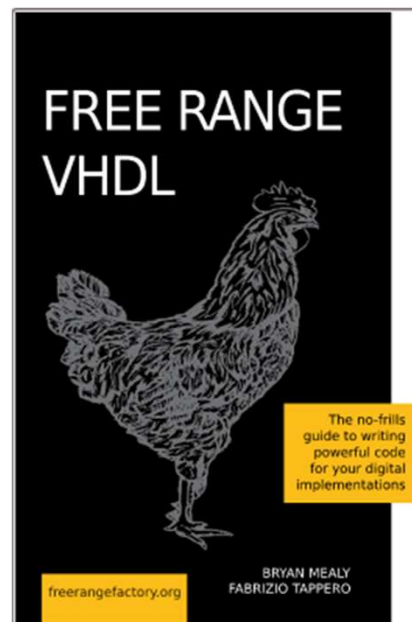
Patterson, Hennessy: Computer Organization and design  
→ pipelining, **e-bok**



Olle Roos: Grundläggande datorteknik  
→ mikroprogrammering



Ingemar Ragnemalm : Digital Konstruktion  
→ pdf



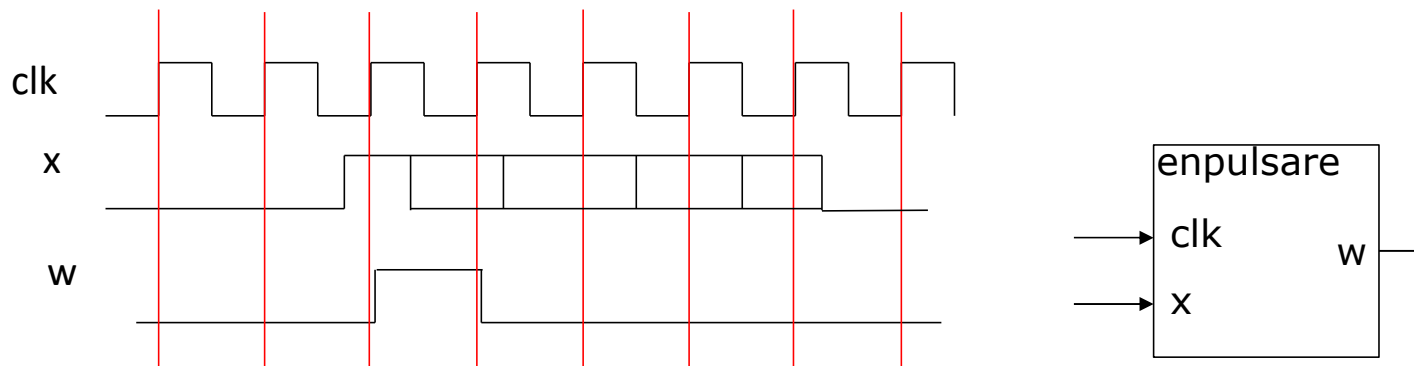
Mealy, Tappero  
Free Range VHDL  
→ pdf

# VHDL

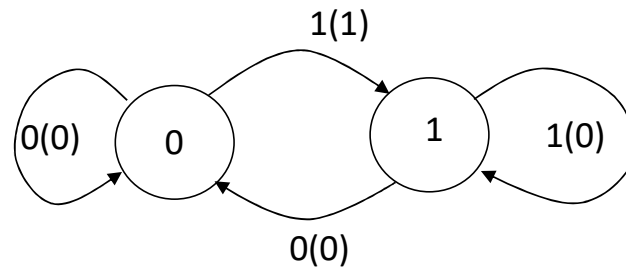
VHDL, Simulering, FPGA-kort

Jag vill bygga en "synkronisering och enpulsare".  
När asynkron insignal går hög (0->1) ska en synkroniserad utpuls produceras. Pulsens längd ska vara 1 CP. Inpulsen måste vara hög vid minst en klockflank.

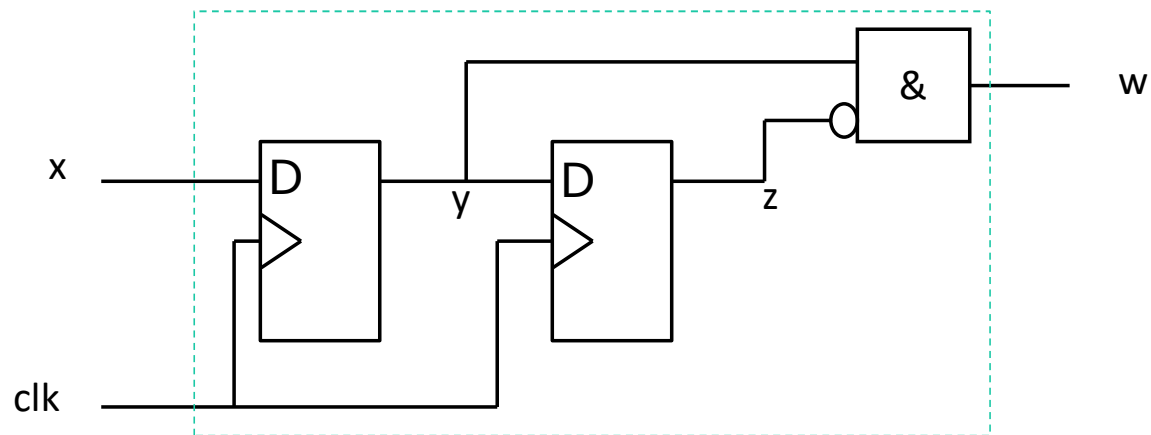
Tidsdiagram



a) Synkvippa + tillståndsgraf



b) Hårdvaruschema direkt



# VHDL : 3.VHDL-kod för b)

```
entity enpulsare is
  port(x,clk : in std_logic;
        w : out std_logic);
end enpulsare;

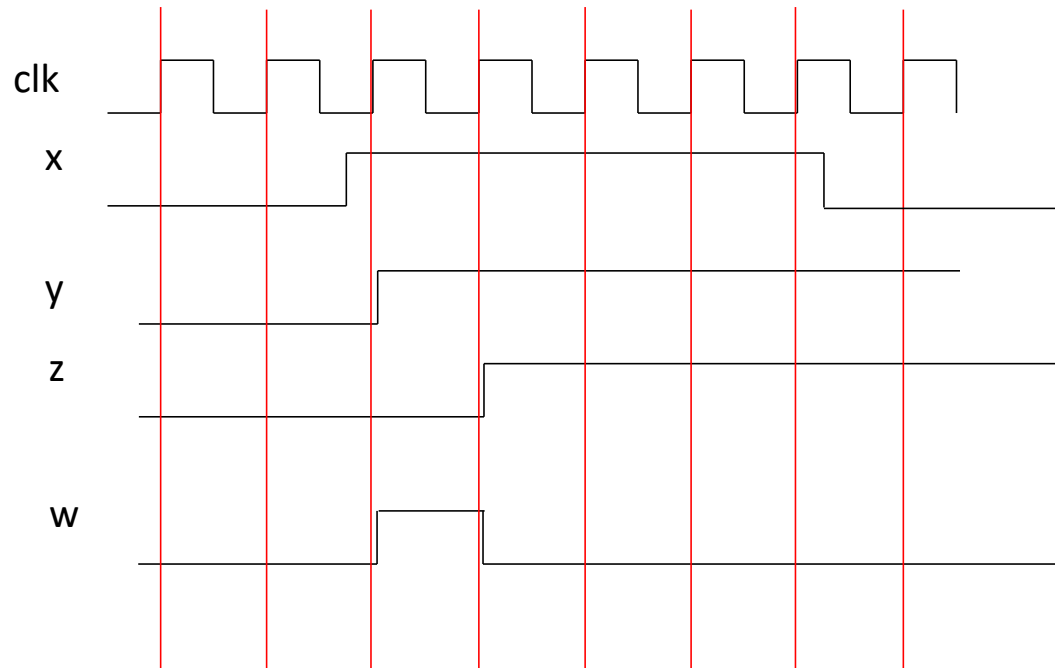
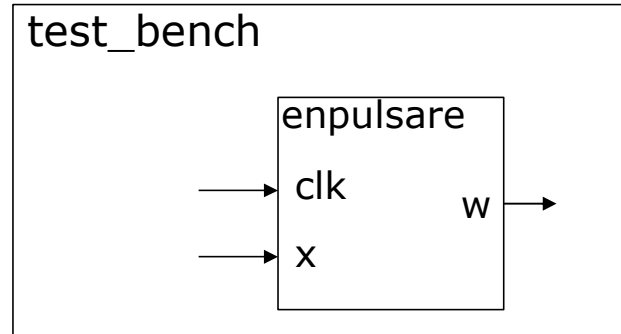
architecture func of enpulsare is
  signal y,z : std_logic;
begin
  w <= y and not z;          -- kombinatoriken

  process(clk)
  begin
    if rising_edge(clk) then
      y <= x;                -- synk-vippan
      z <= y;                -- tillståndsvippan
    end if;
  end process;

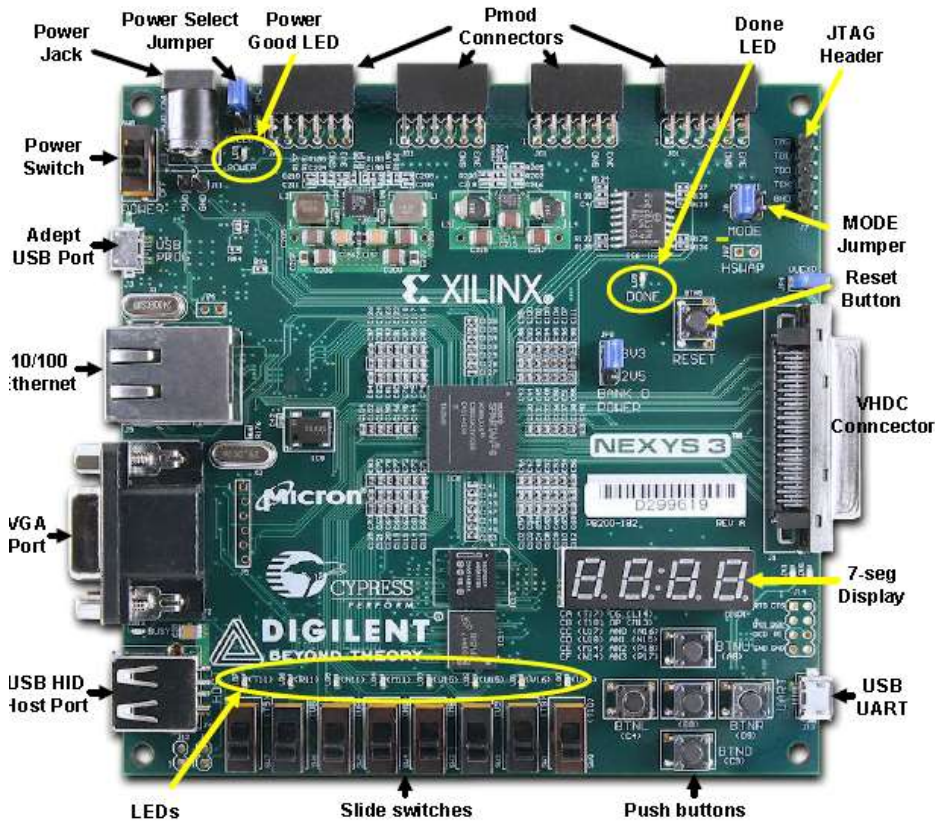
end architecture;
```



# VHDL : 4.Simulering med ModelSim



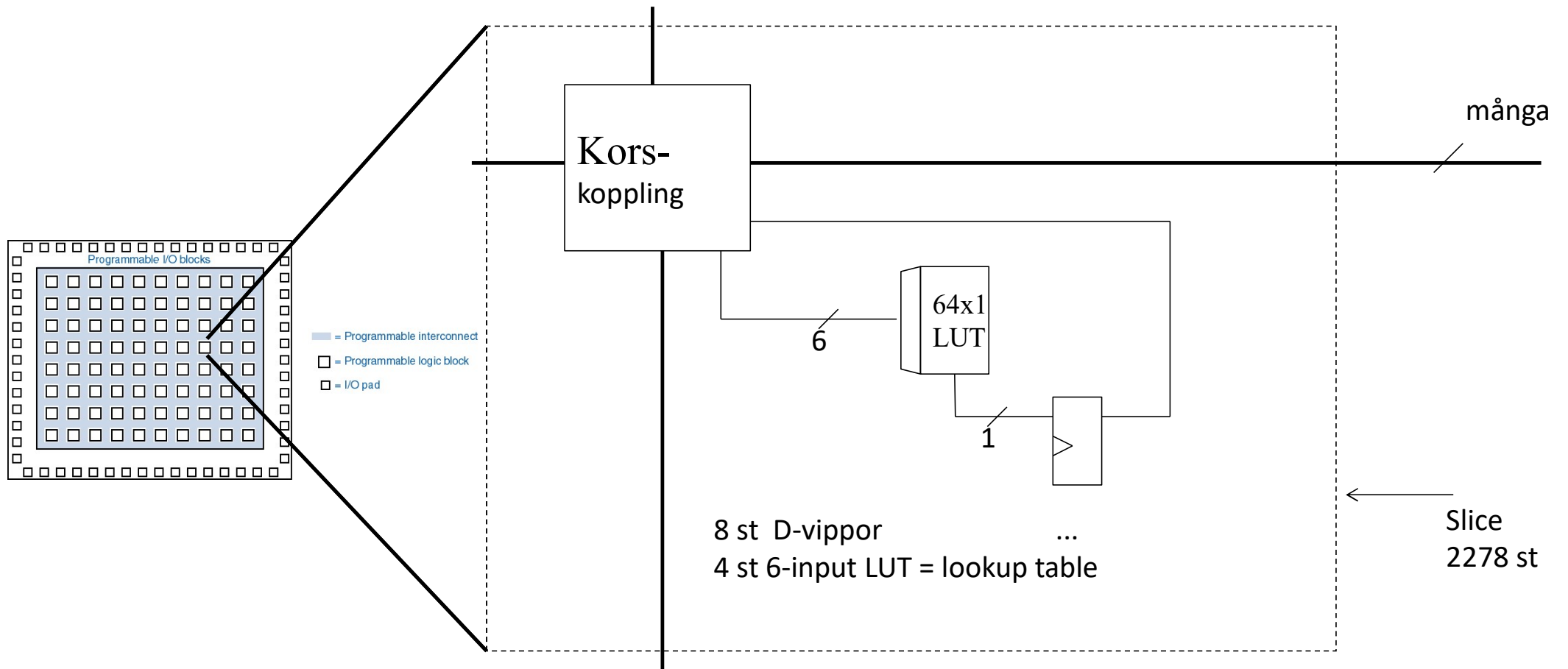
# VHDL : 5.Syntetisering till Nexys 3



FPGA (Field Programmable Gate Array)

- 18224 vippor
- 9112 LUTs (Look Up Table)(64x1 RAM)
- 32 st 2kB blockRAM
- 32 st 18x18 multiplikatorer + (MAC)

# VHDL : FPGA Spartan-6 LX16

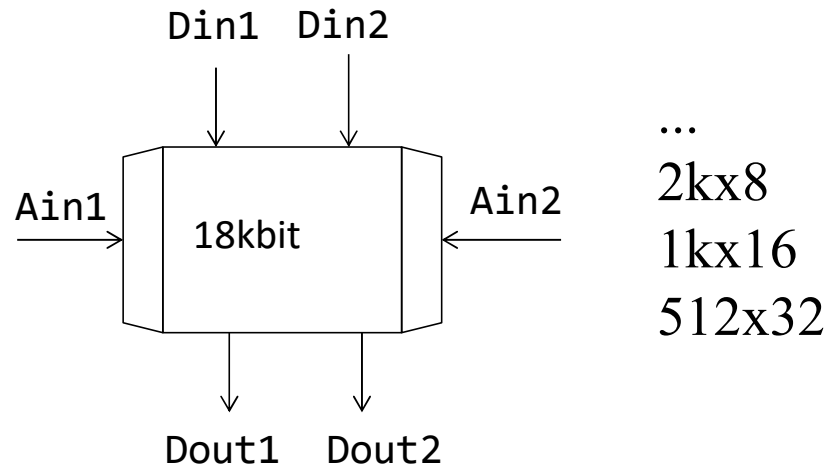


# VHDL : FPGA Spartan-6 LX16

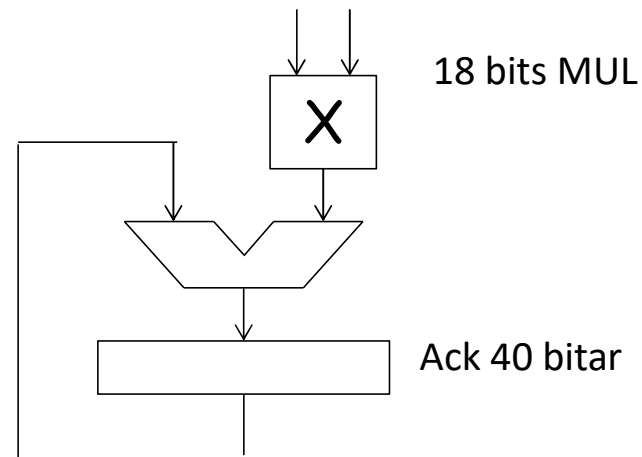
Innehåller också

32 st block RAM  
a 2kB a 9 bit

32 st DSP slices



...  
2kx8  
1kx16  
512x32



# VHDL : Hur går det till?

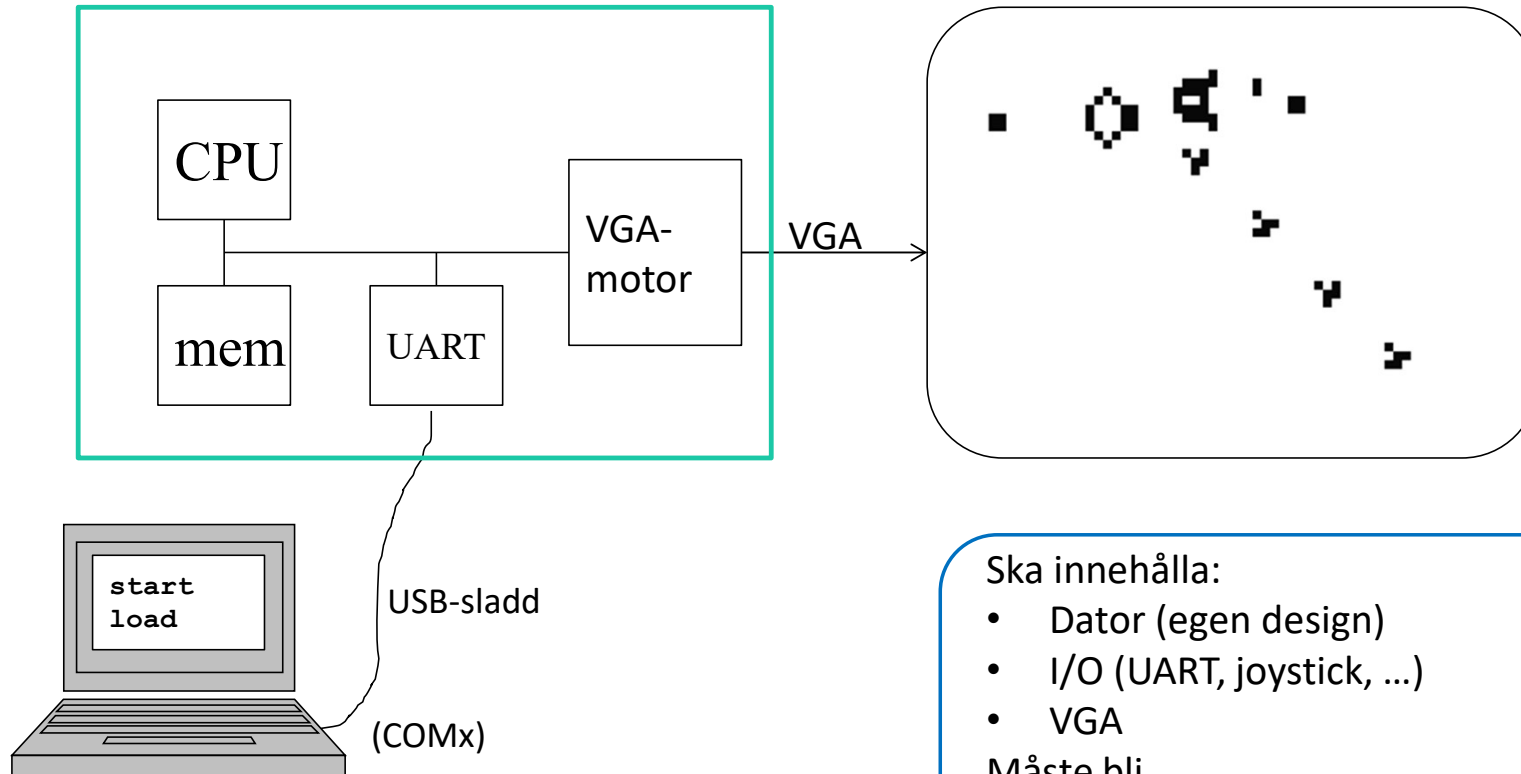
- Skriv VHDL med Emacs. Vi kör Linux.
- **make lab.sim**
  - ModelSim hoppar igång
- **make lab.bitgen**
  - synthesize = ”booleska ekvationer”
  - translate = ”LUTtar och vippor”
  - place = ”placera allting i schema”
  - route = ”koppla ihop allting”
- **make lab.prog**
  - filen design.bit programmeras i FPGA'n

# Projekt

Exempel på tidigare projekt

# Vad kan man bygga?

"en grafikdator, som gör Game of Life"



Ska innehålla:

- Dator (egen design)
- I/O (UART, joystick, ...)
- VGA

Måste bli

- demonstrerbar

# Vad har dom gjort förut?

Tetris  
Drag race  
Snake  
2048  
Music Mania  
Mandelbrot  
Ritprogram  
Grafisk räknare  
Pong  
Pac-man  
Minesweeper

Achtung die kurve!  
Breakout  
Frogger  
Othello  
Synth  
Bomberman  
Space invaders  
Asteroids  
Tron  
[Counter Strike 2D](#)  
[GameBoy](#)  
[MUX-OS](#)

<https://www.youtube.com/watch?v=HThvHjRpshE>

<https://www.youtube.com/watch?v=EjhpCFiEj40>

<https://www.youtube.com/watch?v=9t4dSRWNqOM&feature=youtu.be>



# Distansarbete

Hur kan man arbeta hemifrån?

Se här:

<http://www.isy.liu.se/edu/kurs/TSEA83/distansarbete/>

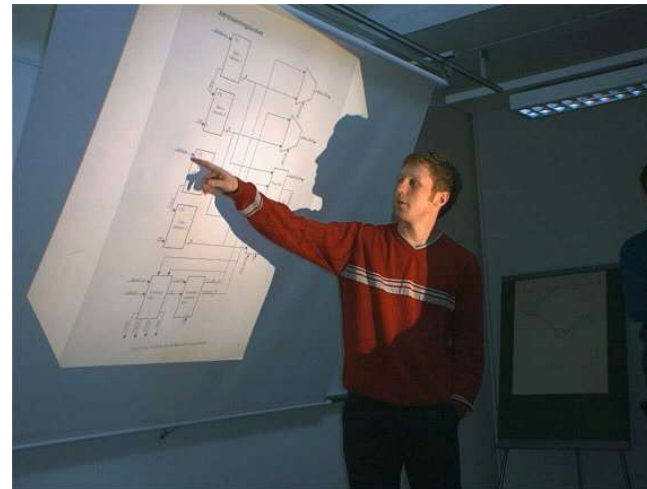
# Sammanfattning

## VT1

- Föreläsningar
- Laborationer
- Bilda 4-grupp (må v.6)
- Skriv en kravspecifikation
  - **Vad** ska datorn göra
- Skriv en designspecifikation
  - **Hur** ska datorn konstrueras

## VT2

- Konstruktion i 7 veckor
- Demo av fungerande dator (enl. kravspec)
- Föredrag v.20 (15 min)
- Skriv en Teknisk rapport



# Vad behöver ni göra nu?

- Registrera er på kursen TSEA83
- Anmäl er till laborationerna, i Lisam
  - 2 tillfällen per lab, i grupper om 2 studenter

Anders Nilsson

[www.liu.se](http://www.liu.se)