

# TKT-1212 Digitaalijärjestelmien toteutus, tentti/välikoe2, sivu 1/3, ma 9.5.2011

Laatinut: Erno Salminen. Laskinta saa käyttää. Tenttipaperille voi tehdä mitä lystää.

**HUOM!** Opiskelijoilla saa olla yksi A4-arkillinen omia muistiinpanoja mukana. Niitä ei tarvitse palauttaa.

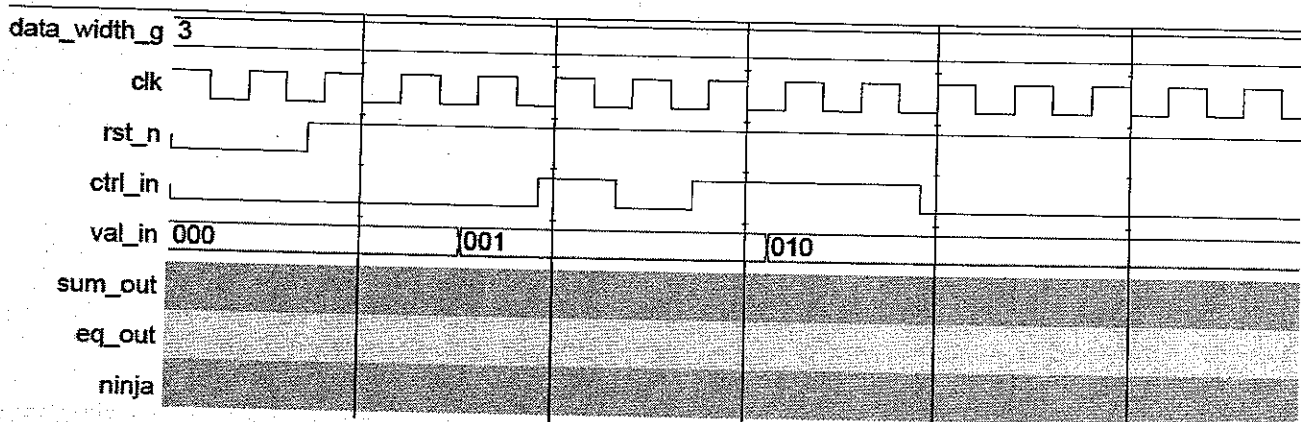
Jos suoritat tenttiä, vastaa kaikkiin kysymyksiin.

Jos suoritat 2. välikokeen, vastaa vain kysymyksiin 3-5.

Tekstin lisäksi suosi kaavoja, kuvia, taulukoita ja esimerkkejä vastauksissasi.

Logiikkakuvissa voit käyttää perusporttien (AND, OR...) lisäksi kiikkuja, multipleksereitä, yhteen-, vähennys- ja kertolaskuyksiköitä sekä vertailijoita. Nimeä kaikki signaalit ja merkitse niiden leveys selkeästi. Pyri vastaamaan tehtäviin oikeassa järjestyksessä (1a, 1b, 1c, 2a, ..., 5c).

1. Selitä seuraavat (ä 2p)
  - a) Design-under-test (DUT).
  - b) VHDL-kielen käsitteiden `package` ja `entity` ero.
  - c) Mitä eroa on käyttäkö `generic`-arvoja vai `constant`-arvoja?
2. Analysoi seuraavan sivun VHDL-koodia. Kellojakso on 10 ns.
  - a) Mitä tyyppiä prosessit ovat: sekventiaallinen vai kombinatorinen? (1p)
  - b) Tarkista ja korjaa prosessien herkkyysslistat. (1p)
  - c) Mitä virheitä tai epäilyttäviä kohtia koodissa on vielä b-kohdan jälkeenkin? (3p)
  - d) Täydennä oheinen ajoituskaavio suoraan koodin perusteella (=ilman korjauksia). Esitä simulaattorin mukainen ajoitus. (4p)



Nimi: \_\_\_\_\_  
Op. num. \_\_\_\_\_

TKT-1212 Digitaalijärjestelmien toteutus, tentti/välikoe2, sivu 2/3,  
ma 9.5.2011

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity tentti_k11 is
  generic (
    data_width_g : integer := 3);
  port (
    clk      : in  std_logic;
    rst_n    : in  std_logic;
    ctrl_in  : in  std_logic;
    val_in   : in  std_logic_vector (data_width_g-1 downto 0);
    sum_out  : out std_logic_vector (data_width_g-1 downto 0);
    eq_out   : out std_logic
  );
end tentti_k11;
```

architecture gatelevel of tentti\_k11 is

```
  signal ninja   : unsigned ( data_width_g-1 downto 0);
  signal sum_reg : unsigned ( data_width_g-1 downto 0);
```

begin

```
  mike : process (val_in, ninja, rst_n)
  begin
    if ninja = to_unsigned(3, data_width_g) then
      eq_out <= '1';
    else
      eq_out <= '0';
      sum_out <= std_logic_vector (unsigned(val_in) + ninja);
    end if;
  end process mike;
```

```
  patton : process (clk, rst_n, ninja)
  begin
    if rst_n = '0' then
      ninja <= "000";
    elsif clk'event and clk = '1' then
      if ctrl_in = '1' then
        ninja <= "000";
      else
        ninja <= ninja + 1;
      end if;
    end if;
```

```
  end if;
end process patton;
```

end gatelevel;

Nimi: \_\_\_\_\_

Op. num. \_\_\_\_\_

TKT-1212 Digitaalijärjestelmien toteutus, tentti/välikoe2, sivu 3/3,  
ma 9.5.2011

~~X~~ Analysoi edellisen sivun VHDL-koodia. Esitä siitä syntesoituva logiikka. Merkitse katkoviivoilla kustakin prosessista syntesoitunut logiikka. Merkitse jokainen koodissa esiintynyt signaali ja muuttuja näkyviin. Älä tee kirpun kokoista tai rumaa kuvaa, vaan tee selkeä ja siisti. (6p)

4. FPGA

~~X~~ Mitä tarkoittaa engl. termi *clock skew*? Miten se vaikuttaa FPGA-piirien rakenteeseen? (2p)

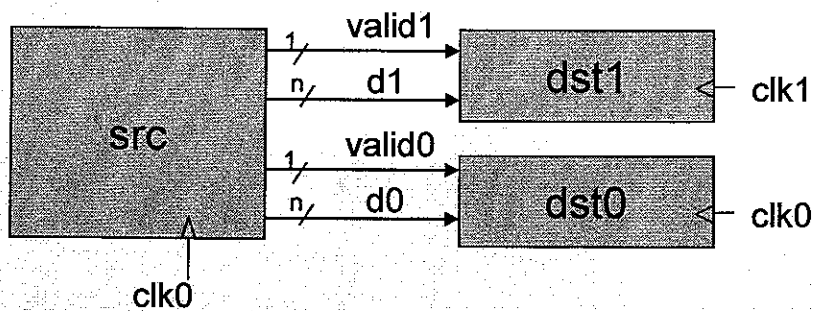
~~X~~ Miten SRAM-pohjainen FPGA eroaa kahdesta muusta perustyyppistä? (2p)

5. Analysoi ao. kytkentää.

~~X~~ Mitkä signaalit pitää synkronoida vai pitääkö mitään? Miksi? (2p)

~~X~~ Tullaanko esitetyillä signaaleilla toimeen? Miksi? (2p)

~~X~~ Mitä pitää huomioida tätä simuloitaessa? (1p)



Nimi: \_\_\_\_\_  
Op. num. \_\_\_\_\_

