

**Tentti: 11.6.2011**

**Tehtävä 1**

Tehtävänäsä on suunnitella dataosa, joka toteuttaa seuraavat aritmeettiset funktiot:

$$C = A + B, \quad D = C * K, \quad E = A * K.$$

A, B ovat 24-bittisiä kahden komplementtilukuja ja K on kokonaisluku välillä 1-15. Kokonaisluku K ei tule data-osalle, vaan ohjaus-osalle. Mitoita tarvittavat bittilevydet lähdöille C, D ja E, kun laskennassa säilytetään täysi tarkkuus. Esitä bittilevydet myös RTL:n sisällä perusteluineen! Piirrä tarvittava RTL dataosalle.

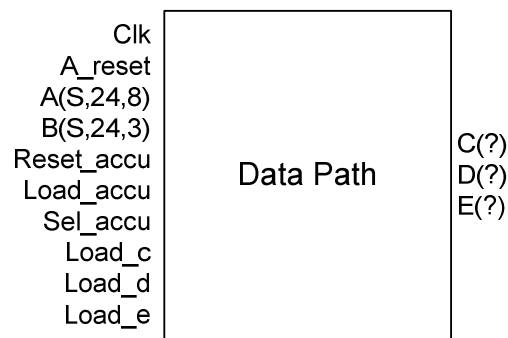
Suurten bittilevyksien johdosta data-osassa ei saa käyttää kertolaskulohkoja, data-osassa saa käyttää vain kahta summainta. Käytössäsi on kuvan 1 mukaiset liitynnät.

Sovitaan, että vektorin esitysmuoto (S),N,D tarkoittaa seuraavaa: Mikäli esitysmuodon edessä on 'S', on vektori etumerkillinen, kahden-komplementtivektori. 'N' kertoo bittien lukumäärän kokonaisuudessaan kyseessä olevalle vektorille, ja 'D' kokonaislukubittien lukumäärän.

Esimerkiksi vektori "S,24,3" tarkoittaa 24-bittistä kahden komplementtilukuja, jossa on 3 kokonaislukubittiä ja 20 murto-osabittiä. RTL:n sisällä bittilevyksiä voidaan ilmaista samalla tavalla.

$$A = S,24,8$$

$$B = S,24,3$$

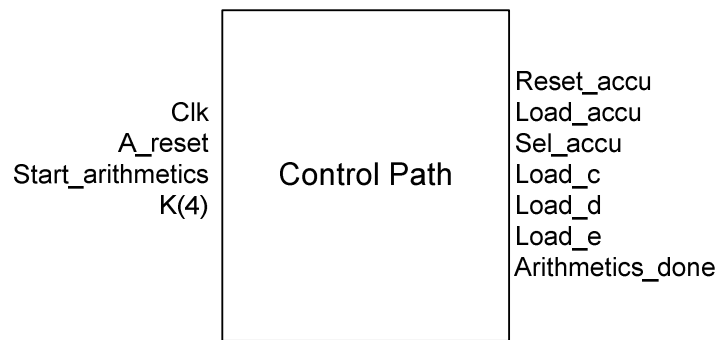


Kuva 1. Suunniteltavan aritmetiikkalohkon data-osa.

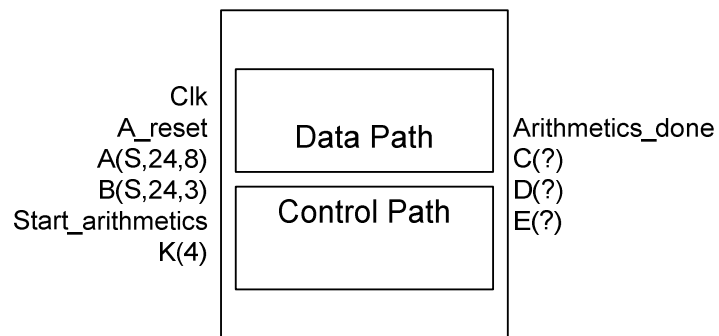
### Tehtävä 2

Tehtävänäsi on suunnitella edellisen tehtävän data-osalle ohjaus-osa. Tilakoneen perusrungon on oltava Mooren tilakone, laskureita saa käyttää. Toiminnan kuvaus:

Kun "start\_arithmetics" käytetään ylhäällä yksi kellojakso, laskee suunniteltu aritmetiikkalohko vaadittavat tulokset lähtöihin. Kun laskenta on valmis, nostaa lohko "arithmetics\_done" lähdön ylös yhden kellojakson ajaksi. a\_reset on nolla-aktiivinen. Kokonaisluku K, joka määrittää kertomat data-osassa, tulee ohjaus-osaan. Kuvassa 2 on ohjausosalle käytössä olevat liittymät



Kuva 2. Suunniteltavan aritmetiikkalohkon ohjausosa.



Kuva 3. Suunniteltava aritmetiikkalohko.

### Tehtävä 3

- a) Luettele digilogiikan synkronisuuden tunnusmerkit.
- b) Mitä ominaisuuksia digilogiikan suunnitelmasta (arkkitehtuurikaavio, VHDL malli) pitää löytyä, jotta sen voitaisiin sanoa olevan rekisterisiirtotasolla (RTL = *Register Transfer Level*)?
- c) Mitä CMOS-teknologialla toteutetun integroidun piirin piirikuvioinnin viivanleveydellä tarkoitetaan?
- d) Mitä integroidun digitaalipiirin piipinta-alan arvioinnissa käytetty *Cell Factor* –tarkoittaa?
- e) Selitä minkälainen FPGA-piirin hakutaulukko on **loogiselta** rakenteeltaan?

**Tentti: 11.6.2011**

**Tehtävä 4**

Alla olevaan listaukseen on otettu laajemmasta VHDL-mallista kaksi VHDL-prosessia, jotka kuvaavat tilakoneen toimintaa.

- a) Laadi mallin perusteella tilakoneen **tilakaavio** ja piirrä tilakoneen **RTL-arkkitehtuuri**.
- b) Selvitä onko tilkone **Moore**- vai **Mealy**-tyyppinen. Perustele valintasi. Pelkkä arvaus ei riitä.
- c) Mihin tarkoitukseen kuvattua tilakonetta voidaan käyttää?

```

State_machines_sync: process
(clk, reset, next_state_of_SM0) is
begin
  if reset = '0' then
    state_of_SM0 <= wait_for_button_pressed;
  elsif clk = '1' and clk'event
  then
    state_of_SM0 <= next_state_of_SM0;
  end if;
end process;
--
Switching_bounce_elimination_control_State_Machine_0:
process (clk, state_of_SM0, delay, pressed) is
begin
  case state_of_SM0 is
  when wait_for_button_pressed =>
    if pressed = '1'
    then next_state_of_SM0 <= wait_for_rising_delay;
    else next_state_of_SM0 <= wait_for_button_pressed;
    button <= '0'; cnt_en <= '0';
    end if;
  when wait_for_rising_delay =>
    if delay = '1' and pressed = '1'
    then next_state_of_SM0 <= wait_for_button_release;
    else next_state_of_SM0 <= wait_for_rising_delay;
    button <= '1'; cnt_en <= '1';
    end if;
  when wait_for_button_release =>
    if pressed = '0'
    then next_state_of_SM0 <= wait_for_falling_delay;
    else next_state_of_SM0 <= wait_for_button_release;
    button <= '1'; cnt_en <= '0';
    end if;
  when wait_for_falling_delay =>
    if delay = '1' and pressed = '0'
    then next_state_of_SM0 <= wait_for_button_pressed;
    else next_state_of_SM0 <= wait_for_falling_delay;
    button <= '0'; cnt_en <= '1';
    end if;
  end case;
end process;

```