

**Tentti: 9.12.2011**
**Tehtävä 1**

Kuvassa 1 on esitetty työn alla oleva logiikkasuunnitelma. Tehtäväsi on suunnitella siihen spesifikaation täyttävä ulkoinen ohjausosa, joka toimii Mooren tilakoneen periaatteella. Kuvan 1 lohko suorittaa laskentaa datalle osissa. Lohkon sisällä on ohjausosa, joka tuottaa DUD:n lähdöt. Lohko ottaa vastaan dataa kolme kertaa. Kaikki tuleva data tulossa *data\_in* näkyy rinnakkain lohkolle, joten se voidaan ladata lohkon yhdessä kellojaksossa.

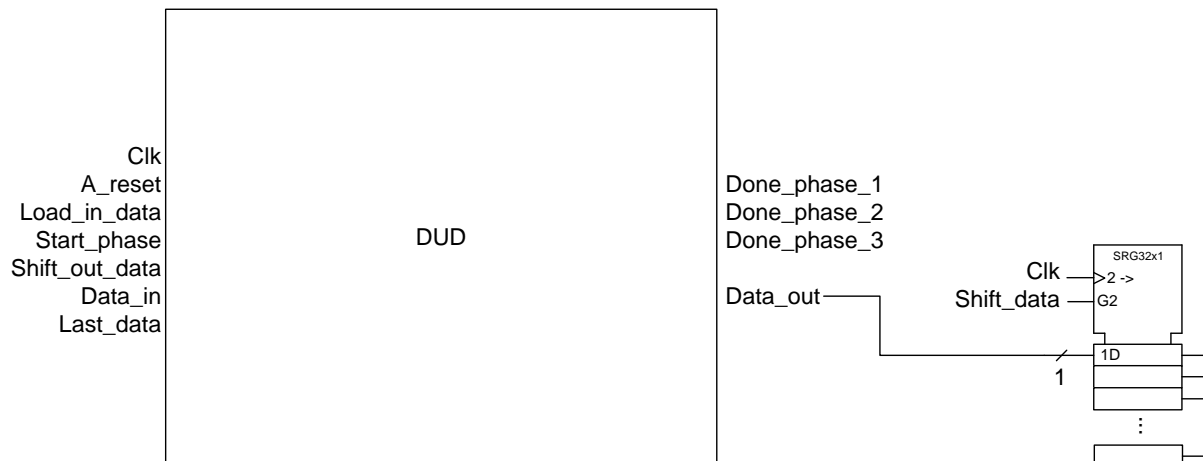
Normaalisti ohjaus on *idle*-tilassa, eikä tällöin mitään tapahdu. Kun tulo *start\_phase\_1* käy ylhäällä kellojakson ajan, on laskentaan tarvittava data näkyvissä tulossa *data\_in*. Tuleva data pitää ensin ladata lohkon tulolla *load\_in\_data*, tämän jälkeen laskenta käynnistetään tulolla *start\_phase*.

Kun laskennan 1. vaihe on määrittelemättömän ajan jälkeen valmis, nousee lähtö *done\_phase\_1* ylös kellojakson ajaksi. Tästä 50  $\mu$ s:n päästä tiedetään, että uusi data on valmiina tulossa. Edellä mainitulla tavalla data voidaan ladata lohkon ja käynnistää uusi laskenta.

Kun laskennan 2. vaihe on määrittelemättömän ajan jälkeen valmis, nousee lähtö *done\_phase\_2* ylös kellojakson ajaksi. Tästä tuntemattoman ajan jälkeen tulo *last\_data* nousee ylös kellojakson ajaksi indikoimaan viimeisen datapaketin valmiutta tulossa *data\_in*. Edellä mainitulla tavalla kolmas laskennan vaihe voidaan suorittaa.

Kun laskennan 3. vaihe on määrittelemättömän ajan jälkeen valmis, nousee lähtö *done\_phase\_3* ylös kellojakson ajaksi. Tämän jälkeen data on valmiina lohkon DUD sisällä, josta se pitää siirtää sarjamuodossa kuvassa 1 näkyvään siirtorekisteriin. Tulo *shift\_out\_data* siirtää lähtödataa samalla periaatteella, kuin siirtorekisterissäkin. Kun kaikki 32 bittiä on siirretty, käy lähtö *all\_done* kellojakson ajan ylhäällä ja ohjaus palautuu *idle*-tilaan.

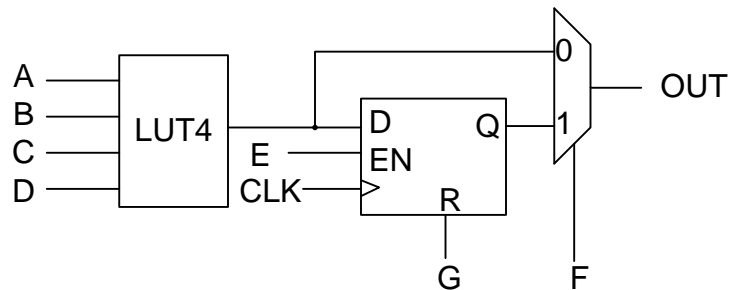
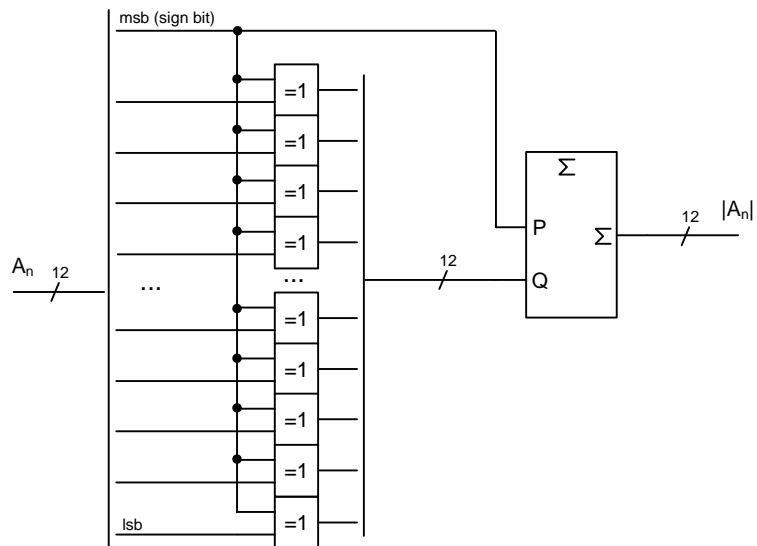
Piirrä ohjauksen tilakaavio ja RTL yksiselitteisesti.



**Kuva 1. Keskeneräinen suunnitelma.**

**Tentti: 9.12.2011**
**Tehtävä 2**

- Montako D-kiikkua edellisen tehtävän ohjausosa tarvitsee?
- Arvio mahdollisimman tarkasti, kuinka monta FPGA:n logiikkaelementtiä edellisen tehtävän ohjausosa tarvitsee. Logiikkaelementti kuvattuna kuvassa 2.
- Kirjoita kuvan 3 täydellinen VHDL-malli. Tulot ja lähdöt *signed*-tietotyyppiä.


**Kuva 2. Logiikkaelementin rakenne.**

**Kuva 3. RTL kuvaus vailla VHDL-mallia.**

**Tentti: 9.12.2011**
**Tehtävä 3**

Suorita **normaalin aritmetiikan** laskusääntöjä noudattaen kukin seuraavista neljästä laskutoimituksesta kahteen kertaan siten, että ensin luvut ( $X$  ja  $Y$ ) tulkitaan etumerkittömiksi (*unsigned*) ja toisella kertaa etumerkillisiksi (*signed*) 2:den komplementti –esitys muodossa oleviksi binäärivektoreiksi.

Oletetaan reaalilukuja esittävät binäärivektorit ovat:  $X = 1011.01$  ja  $Y = 0.1011$  (huomaa binääripiste!)

- a)  $Z1 = X + Y$
- b)  $Z2 = X - Y$
- c)  $Z3 = X * Y$
- d)  $Z4 = X / Y$

Esitä laskutoimituksien vaiheet mahdollisimman yksityiskohtaisesti ykkösien ja nollien avulla. Selitä erityisesti 2:den komplementti –muodossa olevien lukujen yhteydessä mitä valmistelevia toimenpiteitä pitää tehdä ennen kuin laskutoimitukset voidaan suorittaa.

**Tehtävä 4**

- a)  $X = 10011101010$  ja  $Y = 1100101$ . Suorita kertolasku  $Z = YX$  modulo-2 –aritmetiikan laskusääntöjä noudattaen. Esitä laskutoimituksen eri vaiheet ensin nollien ja ykkösien avulla ja toisen kerran polynomimuotoisien lausekkeiden avulla.
- b) Piirrä modulo-2-kertojan RTL-arkkitehtuurikuva. Kertojan pitää kertoa sarjamuodossa syötetty luku vakiolla  $Y = 1100101$ .
- c) Kirjoita kohdan b) arkkitehtuuria vastaava VHDL-malli.