

Tentti: 16.11.2012
Tehtävä 1

Oleta binäriluvut $A = 100101$ ja $B = 11001$.

- Laske käsin paperille mod-2 aritmetiikkaa noudattaen $Y = A \times B$ ja $X = Y/B$ nollija ja ykkösiä käyttäen.
- Tee samat laskutoimitukset polynomiesitysmuotoa käyttäen.
- Piirrä vakiolla 11001 kertovan mod-2 kertojen RTL arkkitehtuuri.
- Kirjoita piirtämäsi RTL-arkkitehtuuria vastaava VHDL-malli.
- Arvioi piirtämäsi RTL-arkkitehtuurin perusteella mod-2 kertojan looginen monimutkaisuus (*logic complexity*).
- Mikä on suunnitelmasi kriittinen ajoituspolku (critical timing path) ja suurin synkroninen kellotaajuus millä logiikkaa voidaan ajaa. ($F = 32$ nm, $FO4 = 15$ ps. Oleta, että ekvivalenttiportin viive on $1.5 FO4$).
- Piirrä kokonaistehonkulutusta (staattinen + dynaaminen) kuvaava käyrä kytkentäaktiivisuuden funktiona. Portin tulokapasitanssi $C_{in} = 0.5$ fF, $V_{dd} = 1V$ ja staattinen tehonkulutus 150 nW/gate

Assume two binary numbers $A = 100101$ and $B = 11001$.

- Calculate by mod-2 arithmetic $Y = A \times B$ and $X = Y/B$ using zeros and ones in your hand written presentation.
- Do same calculations as earlier by means of polynomial presentation of A , B , Y , and X .
- Draw an RTL architecture of a serial mod-2 multiplier, which multiplies serial input by a constant 11001.
- Write VHDL models of RTL architecture described in c) .
- Estimate the logic complexity of the mod-2 multiplier you have designed.
- Find the critical timing path and maximum clock frequency ($F = 32$ nm, $FO4 = 15$ ps, Assume that equivalent gate's delay is $1.5 FO4$).
- Draw a plot of the total power consumption as a function of switching activity . C_{in} of a gate is 0.5 fF, $V_{dd} = 1V$ and static power consumption 150 nW/gate

Tehtävä 2

Kirjoita VHDL-malli (*entity* ja VHDL *architecture*), joka vastaa kuvassa 1 esitettyä QPSK moduloidun konstellaatioenkoodauksen toteuttaavaa RTL-arkkitehtuuria (datapolku ja kontrolli). Taulukossa T1 Q:n ja I:n numeroarvot on esitetty etumerkki-itseisarvo formaatissa. Käytä lukujen kahden komplementtesitysmuotoa VHDL-mallissasi.

Write a VHDL model (entity and VHDL architecture) of the RTL architecture (data path and control) of QPSK Constellation Encoding logic described in Fig. 1. I and Q are presented in sign and magnitude format in Table T1. Change the format into 2's complement presentation for the VHDL model.

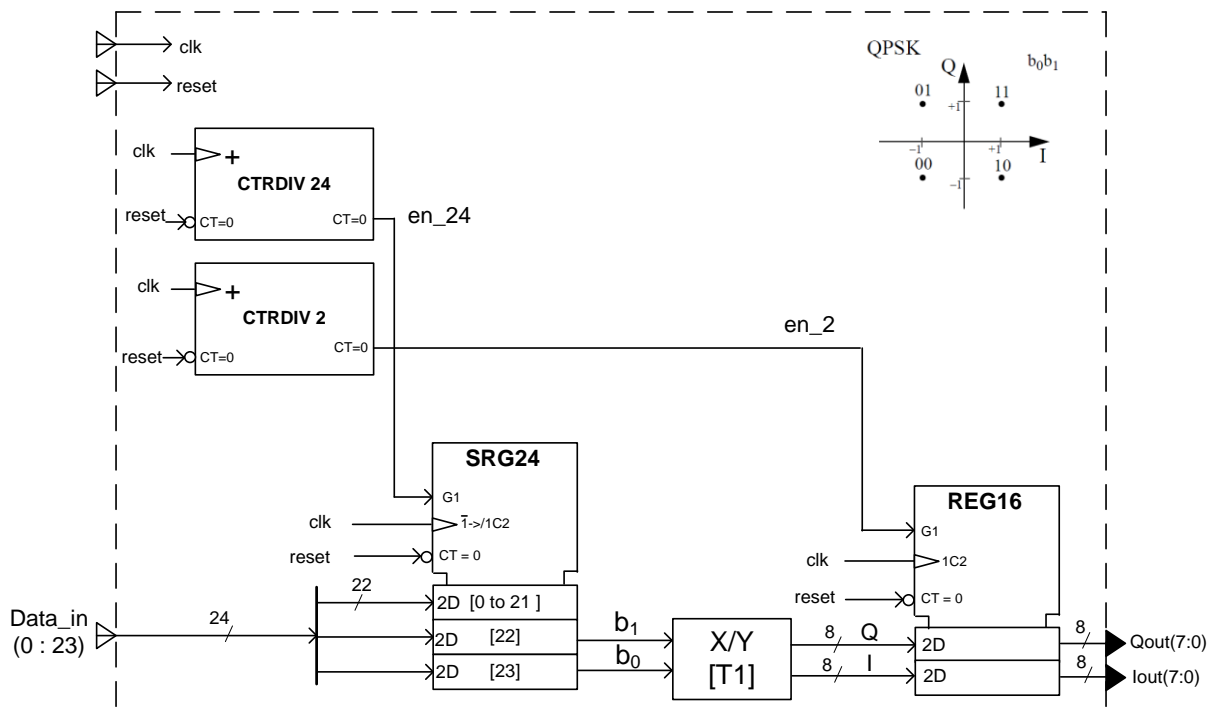


TABLE T1

| b₀ | b₁ | Q | I | Q | I |
|----------------------|----------------------|-----------|-----------|-----------------|-----------------|
| 0 | 0 | -1 | -1 | 10000001 | 10000001 |
| 0 | 1 | +1 | -1 | 00000001 | 10000001 |
| 1 | 0 | -1 | +1 | 10000001 | 00000001 |
| 1 | 1 | +1 | +1 | 00000001 | 00000001 |

Kuva 1. RTL.

Tehtävä 3

Kuvan 2 arkkitehtuurikaaviota voidaan käyttää toteuttamaan kumulatiivisen summan

$$S = \sum_{i=0}^{i=k-1} a_i$$

laskeminen. Oleta, että $k = 200$ ja numerot a_i ovat etumerkillisiä **kokonaislukuja** (signed) lukualueessa välillä $-512 \dots +511$.

- Määritä parametrien r , n , m , $x1$, $x2$, p ja q arvot.
- Täydennä kuvan 2 RTL-arkkitehtuurikaavio määrittämälläsi lukuarvoilla ja muilla tarvittavilla lisäyksillä.
- Miten parametrit ja arkkitehtuuri muuttuisivat, jos luvut a_i olisivat kahden etumerkillisen (signed) 8-bitin binäärivektorin avulla esitettyjä **kompleksilukuja**.

Architecture template of Fig. 2 could be used to implement the computing of cumulative sum

$$S = \sum_{i=0}^{i=k-1} a_i$$

Assume that $k = 200$ and numbers a_i are **integer numbers** (signed) in the range -512 to $+511$.

- Determine values of the parameters r , n , m , $x1$, $x2$, p and q .
- Complete the RTL description of Fig.2.
- How the things would change if the numbers (a_i) were complex numbers presented as two 8-bit signed bit vectors.

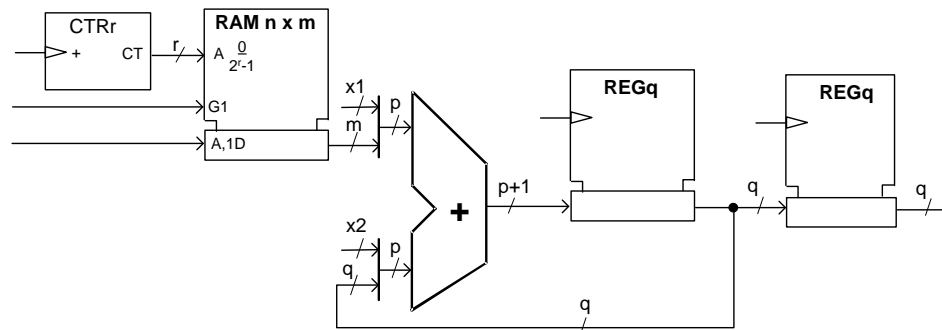


Fig. 2 Architecture template to implement computing of a cumulative sum.

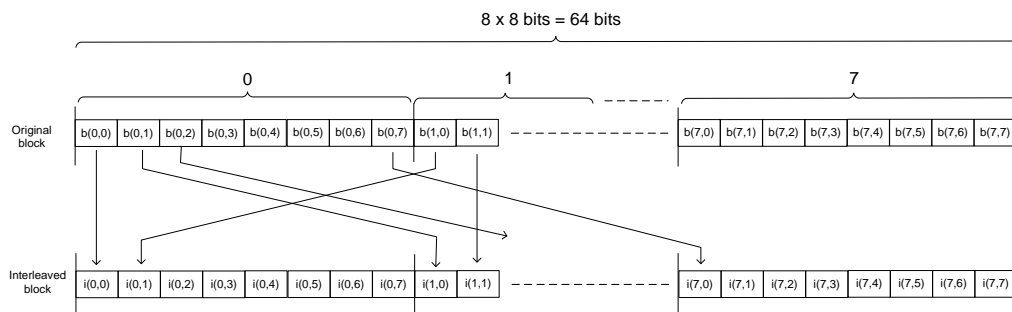
Tehtävä 4

Eräs lomituksen periaate on määritelty kuvassa 3. 64 bitin binäärivektori syötetään *Data_in*-tulon kautta sarjamuodossa lomittajalle. Saman vektorin on tarkoitus löytyä lomituksen jälkeen lomitettuna 64 bitin RAM:sta.

- a) Suunnittele kuvan 4 lakurit siten, että lomitus toimii.
- b) Arvioi toteutuksen looginen kompleksisuus.

Interleaving is done as presented in Fig. 3. Serial 64 bits bit vector to be inter-leaved is fed into *Data_in* input.

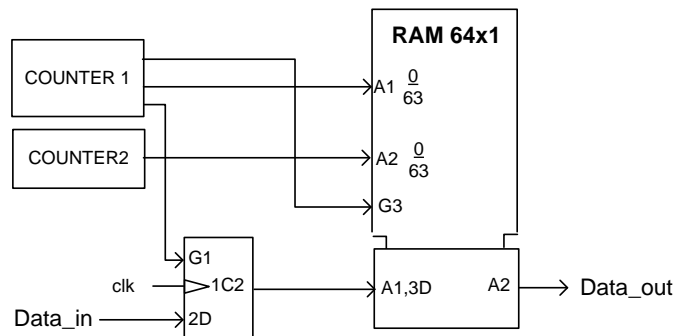
- a) Design the counters in Fig. 4 needed to implement the inter-leaver.
- b) Estimate the logic complexity of the design.



$$b(0,0) \rightarrow i(0,0), b(0,1) \rightarrow i(1,0), b(0,2) \rightarrow i(2,0), b(0,3) \rightarrow i(3,0) \dots b(0,7) \rightarrow i(7,0) \dots b(1,7) \rightarrow i(7,1)$$

Lomituksen periaate on tässä se, että tavun sisältämien bittien osoite ja tavuosoite vaihdetaan keskenään. Indexes are swapped.

Kuva 3 Lomituksen (interleaving) periaate.



Kuva 4