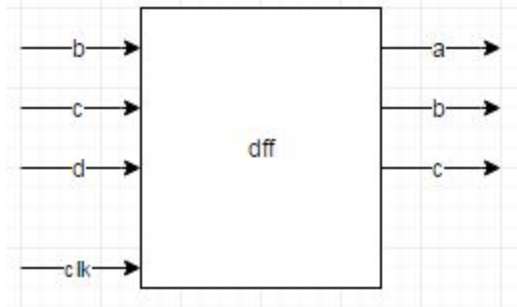


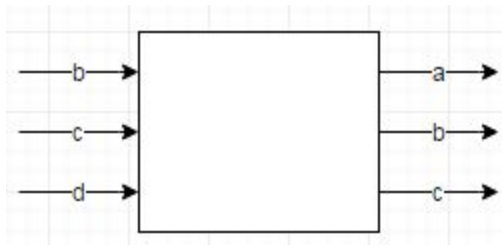
Exercise 7: Manual RTL-Synthesis

Task 1



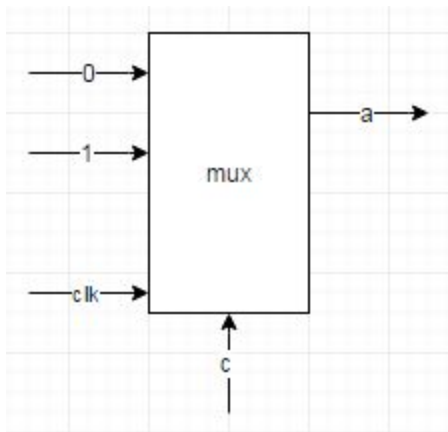
Signaalit a, b ja c päivittyvät nousevalla kellon reunalla. Koska määrittäminen tapahtuu synkronisesti prosessin sisällä, jokainen signaali päivittyy vuorotellen.

Task 2



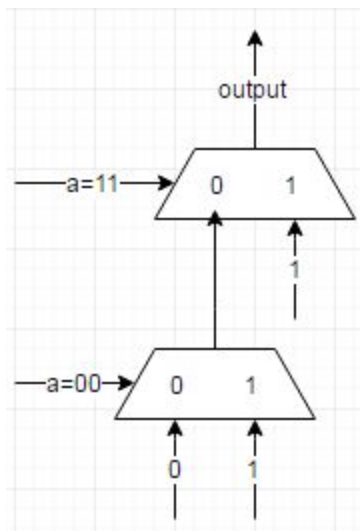
Koska kello-signaalia ei ole yhdistetty prosessiin, signaalit päivittyvät jatkuvasti.

Task 3



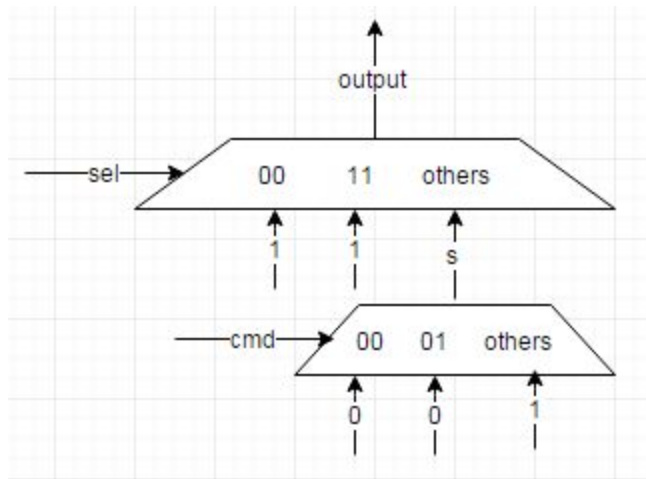
Perinteinen if-else-rakenne onnistuu 2:1 muxin avulla. Signaali c valitsee ulostulon (a) signaaliksi joko 0 (kun $c=0$) tai 1 (kun $c=1$). Tämä tapahtuu nousevalla kellon reunalla.

Task 4



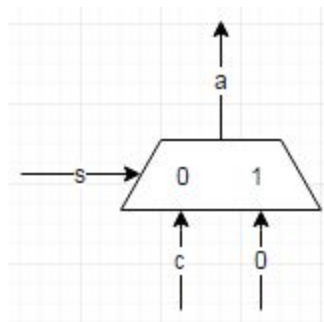
If-elsif-rakenne muodostuu ketjuttamalla aiempia 2:1 mux-komponentteja. A:n arvon ollessa 11 tai 00 outputin arvoksi tulee 1 ja muissa tilanteissa 0.

Task 5



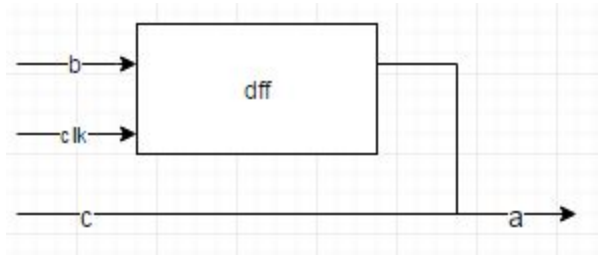
Case-rakenne muodostuu käyttämällä n -leveyden muxeja. Tässä tapauksessa $n=4$ (others käsittää käytännössä myös bittiparit 01 sekä 10). Lisäksi s :n arvo määräytyy toisen samanleveyden muxin perusteella.

Task 6



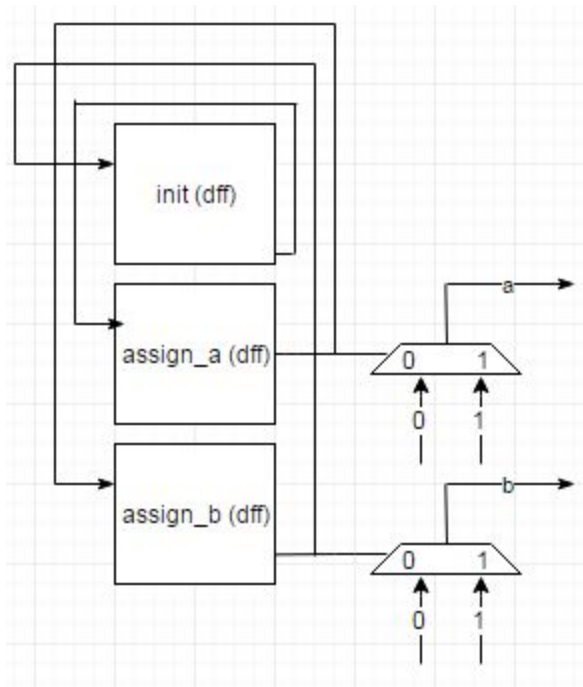
Ensimmäinen a :n määrittäminen ($a \leq b$) jätetään huomiotta, sillä sen jälkeen tapahtuu 1-2 muuta määrittäystä joista toinen jää voimaan. Muxi valitsee tässä a :n arvoksi joko c :n (kun $s=0$) tai 0 :n (kun $s=1$).

Task 7



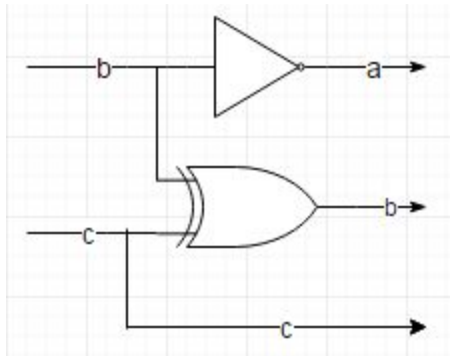
Käytännössä a:n signaali määrittyy c:n mukaan aina silloin, kun ei olla kellojakson nousevalla reunalla.

Task 8



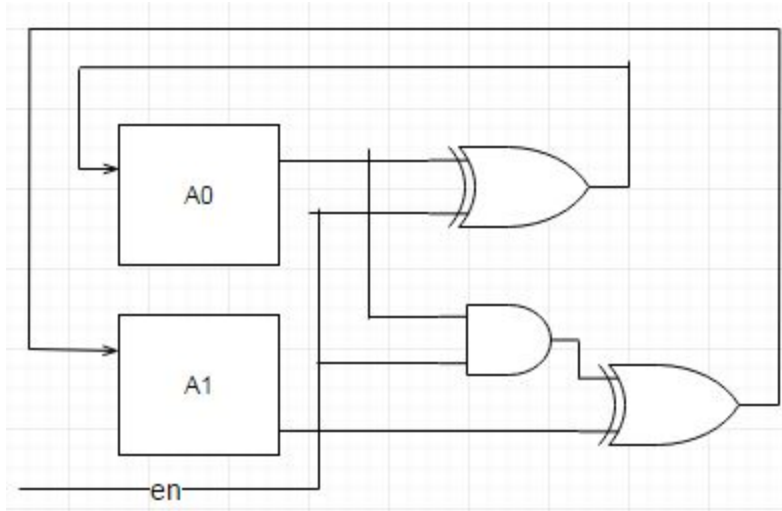
Kun tilakone on one-hot, niin se käytännössä tarkoittaa sitä, että jokaiselle tilalle luodaan oma kiikku. Tämän tilakoneen sykli on `init->assign_a->assign_b->init` ja se toistaa sitä. A:n ulostulo on yksi vain tilassa `assign_a`, kuten myös B:n ulostulo yksi vain tilassa `assign_b`.

Task 9



Task 10

Koska integerin range on 0..3, tarvitaan sen varastointiin vähintään kaksi bittiä dataa, eli käytännössä siis kaksi kiikkua.



Signaali *en* määrittää, tapahtuuko inkrementointia. Signaalit on takaisinkytketty kiikkuihin.