

1. in 2. vaja: Kombinacijska vezja

Reši naloge in oddaj datoteke z modelom vezja skupaj z odgovori na vprašanja (tekstovna datoteka) v spletno učilnico. Naloga označena z * je neobvezna naloga, ki jo reši, če imaš dovolj časa.

Maksimum

Naredi model vezja, ki primerja dve vhodni 16-bitni vrednosti in pošlje na izhod največjo vrednost. Vezje bomo opisali s pogojnim prireditvenim stavkom: **when ... else**.

Namig za VHDL

Večbitne signale v jeziku VHDL določa standardni podatkovni tip, npr. 16-bitni vektorje deklariramo: `std_logic_vector(15 downto 0)`

V standardni knjižnici niso definirane vse računske operacije z vektorji, pri primerjavi se obravnavajo kot nepredznačena števila. Če želimo podpreti vse operacije in eksplicitno določiti vrsto podatkov v vektorju, uporabimo vektorski podatkovni tip: `unsigned` ali `signed`.

1. Naredi model vezja, ki deluje z nepredznačenimi števili in preveri delovanje s simulacijo.
 - a) Kako deluje vezje, če obravnavamo vrednosti vhoda kot predznačene? (npr. primerjava vektorja s samimi enicami in samimi ničlami.)
 - b) V orodju Quartus preglej shemo vezja (Tools, Netlist, RTL) in zasedenost FPGA (Total logic elements). Ali je kaj drugače, če spremenimo vhode v predznačene vrednosti?
2. Dodaj v vezje enobitni izhod, ki se postavi na 1, kadar sta vhodni vrednosti enaki in preveri delovanje s simulacijo.
3. Dodaj v vezje izhod zero, ki se postavi na 1, kadar je maksimalna vrednost 0.
 - a) Kaj je potrebno storiti v VHDL, če želimo za izračun izhoda zero uporabiti vrednosti, ki jo peljemo na izhod za maksimum?

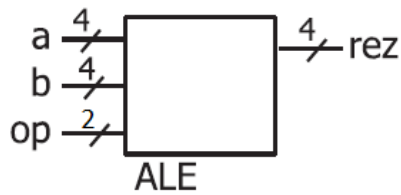
*Srednja vrednost

Naredi vezje s tremi 16-bitnimi vhodi (a,b in c), ki vsebujejo nepredznačene vrednosti. Vhodne vrednosti razvrsti po velikosti in pošlji na izhod tisto, ki je na sredini seznama. Primer:

- izhod=5 pri a=2,b=7,c=5
- izhod=3 pri a=4,b=3,c=2
- izhod=4 pri a=4,b=3,c=4

1. Naredi visokonivojski model vezja s primerjavami. Npr. izhod = a, kadar je $b \leq a \leq c$ ali $c \leq a \leq b$. Koliko primerjalnikov potrebujemo za opis vezja?
2. Ugotovi, kako bi opisal vezje s čim manj primerjalniki (namig: uporabi mrežo za sortiranje treh vrednosti, angl. [sorting network](#)).

4-bitna aritmetično logična enota



Opiši vezje 4-bitne aritmetično logične enote, ki izvaja štiri operacije nad predznačenima vhodnima signaloma **a** in **b** (signed) ter shrani rezultat v 4-bitni izhod **rez**. Operacije določa 2-bitni nepredznačen vhod **op**: 0 določa seštevanje, 1 odštevanje, 2 logično IN operacijo (**and**) in 3 logično ALI operacijo (**or**).

Namig za VHDL

Signal **op** izbira izraz, ki bo povezan z izhodom. Visokonivojski model ALE zato temelji na izbiralniku, ki ga lahko opišemo na več načinov:

- funkcijsko, s pogojnim prireditvenim stavkom. Poskusi najprej le za vsoto in razliko, potem pa dodaj logični operaciji (razširjen staven **when...else**).
- postopkovno s pogojnim stavkom in procesom.
- postopkovno z izbirnim stavkom **case**.

Pri računskih operacijah mora biti na levi in desni strani prireditve enako število bitov. Če želimo izračunati vsoto s prenosom, pred seštevanjem razširimo operanda na ustrezno število mest: `resize(a,5) + resize(b,5)`

1. Naredi model ALE in preizkusi delovanje s simulacijo. S katerim stavkom je najboljše opisana ALE v jeziku VHDL?
2. Preveri delovanje računskih operacij na simulaciji z različnimi kombinacijami na vhodu.
 - a) V orodju ModelSim nastavi prikaz predznačenih vrednosti signalov (desni klik, Radix, Decimal).
 - b) Spremeni deklaracijo vhodnih vektorjev v nepredznačene vrednosti (unsigned). Ali je sedaj rezultat simulacije drugačen? Poskusi razložiti.
 - c) Pri katerih vrednostih vhoda dobimo napačen rezultat seštevanja ali odštevanja, ne glede na deklaracijo?
3. Povečaj izhod **rez** na 5 bitov, da bo na najvišje mesto prišel prenos pri seštevanju in odštevanju in preveri delovanje na primerih, kjer smo prej dobili napačen rezultat.

***Množilnik**

Naredimo vezje za izračun produkta 4-bitne in 2-bitne vrednosti. Poglejmo najprej kako bi naredili računanje s svinčnikom in papirjem. Predpostavljamo, da so vrednosti nepredznačene.

1101×10 (desetiško 13×2) ----- 0000 1101 ----- 11010 (desetiško 26)	1101×11 (desetiško 13×3) ----- 1101 1101 ----- 100111 (desetiško 39)
---	--

Množenje z dvobitno vrednostjo naredimo v dveh korakih. Najprej množimo z najbolj desnim bitom in dobimo 0 ali pa vrednost množenca (če je množitelj 1). Nato množimo z naslednjim bitom množitelja, zapišemo rezultat pomaknjen za eno mesto v levo in seštejemo oba delna rezultata.

- Naredi opis množilnika in preveri delovanje s simulacijo. Vhoda sta 4-bitni **a**, 2-bitni **b**, izhod pa 6-bitni **m**. Deklariraj notranje signale za delne produkte.
 - izračun delnih produktov opišemo s pogojnim stavkom:
 $p1 \leq a \text{ when } b(0)='1' \text{ else } "0000";$
 - pri vsoti en delni produkt zamaknemo v levo, kar naredimo tako da dodamo z operatorjem & ničle na ustrezni strani:
 $m \leq ("00" \& p1) + ('0' \& p2 \& '0');$
- Poskusi sedaj opisati množilnik z dvema 4-bitnima vhodoma in 8-bitnim izhodom.