

4. vaja: V/I registri in program

V sistem bomo dodali vhodne in izhodne registre, preko katerih poteka povezava med CPU in grafičnimi komponentami VGA igre.

Napisali bomo program za preprosto igro katere cilj je spraviti žogico skozi odprtino v premikajočem se traku.

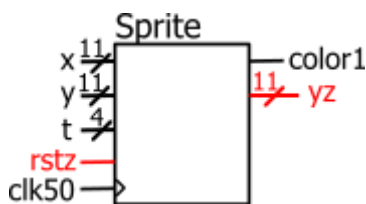
Povezava do prevajalnika cpu.html in eksperimentalnega prevajalnika za jezik C: cpuc.htm

Dogodki, ki jih obdeluje digitalni sistem se izvajajo pri zelo različnih hitrostih:

- pritisk tipke traja 100 ms ali več
- ukaz mikroprocesorja se izvede v 0.4 ms (signal **ce**)
- risanje grafike in zaznavanje trka pa v 0.00002 ms (signal **clk50**)

To pomeni, da bi program v procesorju ob vsakem pritisku na tipko zaznal več deset pritiskov, trk med dvema objektoma ($color='1'$ in $color1='1'$) pa verjetno sploh ne bi nikoli zaznal. V sistemu potrebujemo logiko za obvladovanje različno hitrih signalov - npr. ob zaznavi trka s hitro uro, nastavimo vrednosti registra (001), ki ga pobrišemo šele takrat ko procesor prebere ta register.

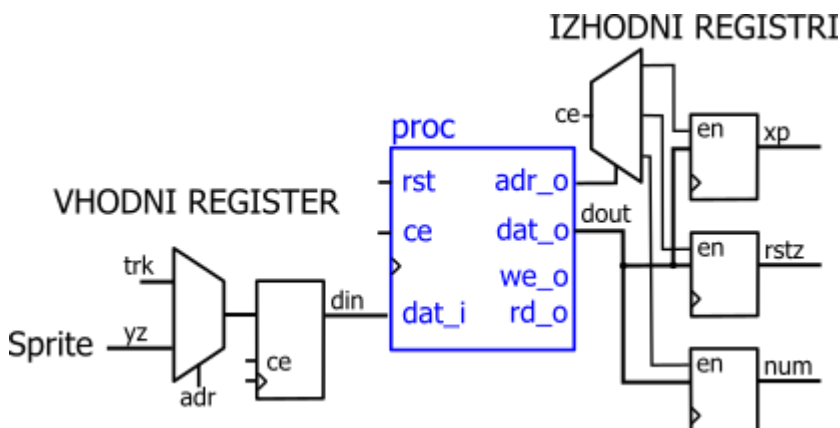
4.1 Komponenta Sprite



Popravi komponento za prikaz žogice, tako da bo imela dodaten vhod **rstz**, s katerim postavi žogico v začetno lego in 11-biten izhod **yz**, ki določa trenutni položaj žogice po y osi.

- Zapiši pogoj za resetiranje žogice: kadar je **rstz** na '1', naj se **y1** postavi na 0, **x1** pa ne bomo spreminjali, da bo igra bolj zanimiva.
- Izhod **yz** naj dobi vrednost notranjega signala **y1**.

4.2 V/I registri v sistemu



V datoteki sistem.vhd spremeni oz. dopolni logiko, ki povezuje vhodne in izhodne priključke procesorja.

- **din** predstavlja vhodni register s pogojem za vpis podatka ob **clk50** in **ce**. Deklariraj nov 12-bitni vektor **trk** in spremeni logiko za zaznavanje trka tako, da bo nastavljala ta signal (ob **rd='1'** naj se **trk** postavi na X"000", kadar sta **color** in **color1** na '1' pa naj se **trk** postavi na X"001")
- Na vhodu registra je izbiralnik, ki določa ali se na vhod procesorja prenese signal **trk** ali **yz**: ob naslovu **adr=0** se prenese **trk**, sicer pa **yz**.
- Napisana je že logika za izhodni register **xp**, dodali bomo pa še dva: 1-bitni register **rstz**, ki se nastavi ob naslovu **adr=1** in 4-bitni **num** ob naslovu **adr=2** (v opisu vezja sedaj odstrani stavek `num <= t`, ki je povezoval številčni prikaz s pritisnjeno tipko).

4.3 program

Dopolni še program, da bo izvajal logiko igrice.

