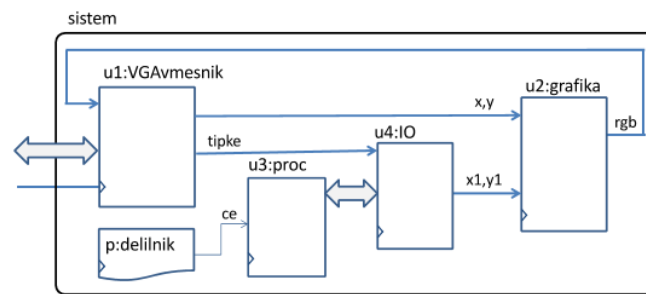


11. vaja: Grafični sistem s procesorjem



Digitalni sistem

Projekt [B25sistem.gar](#) vsebuje vzorčni digitalni sistem sestavljen iz VGA vmesnika, učnega procesorja, vhodno-izhodnega vmesnika in komponente za prikaz grafike. Glavne komponente so vključene v datoteki **sistem.vhd**, kjer je tudi delilnik ure, s katerim lahko upočasnimo delovanje procesorja.

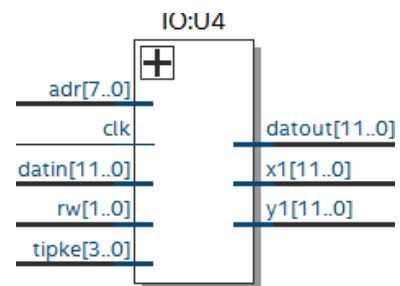
- **VGAvmesnik.vhd** vsebuje časovni generator signalov VGA z izhodnimi koordinatami (**x,y**) in vmesnik za periferne enote razvojne plošče: **tipke** in signale za LED matrice (**ledm** in **ledcnt**).
- **proc.vhd** vsebuje 16-bitni učni procesor **CPU16** in model programskega pomnilnika **program**. Procesor ima vhod **rst**, ki je vezan na tipko, vhod **ce** s katerim omejimo hitrost izvajanja programa in vhodno-izhodno vodilo (**rw**, **dataIO**, **dataProc**).
- **grafika.vhd** vsebuje vzorec komponente za prikaz sličice in logiko za izhodne barve. Komponenta prikaže kvadrat na koordinatah (**x1,y1**) in mrežo. Preveri, kako je narejen prikaz mreže!
- **io.vhd** je vzorec komponente za procesorski vhodno-izhodni vmesnik, ki ga je potrebno dopolniti.

Vhodno-izhodni vmesnik

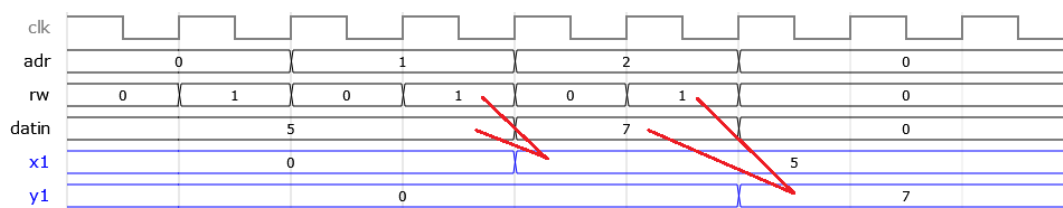
Vhodno-izhodni vmesnik vsebuje signale procesorskega vodila, vhod za stanje tipk ter izhodna registra **x1** in **y1**.

1. Naredi model vezja **io.vhd** s signali:

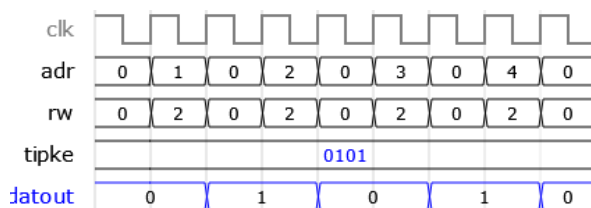
- **adr** je 8-bitno vhodno naslovno vodilo
- **datin** je 12-bitni vhod na katerega procesor piše
- **rw** je 2-bitni kontrolni vhod
- **tipke** so 4-bitni vhodni signal
- **datout** je 12-bitni izhodni register iz katerega procesor bere
- **x1** in **y1** sta 12-bitna izhodna registra



2. Naredi logiko za nastavljanje izhodnih registrov. Registra shranita vrednost iz 12-bitnega vhoda ob uri, pogoju **rw=1** in ustreznem naslovu: **x1** ob **adr=1** in **y1** ob **adr=2**.



3. Dodaj vhodni izbiralnik in register. Vektor s stanjem tipk razdeli na štiri enobitne signale, ki jih pošljemo na procesor ob ustreznem naslovu, kot prikazuje tabela. Ob uri, ustreznem naslovu in pogoju **rw=2** nastavi novo vrednost registra **datout**.



adr	datout	tipka
1	tipke(0)	S1
2	tipke(1)	S2
3	tipke(2)	S3
4	tipke(3)	S4

Preizkus na razvojni plošči

1. Dodaj opis vhodno-izhodnega vmesnika v Quartusov projekt in dopolni komponente za prikaz grafike. Za prikaz kroga manjka logika, ki določa barvo glede na vsebino pomnilnika ROM.
2. Prevedi sistem in preizkusi delovanje na razvojni plošči. Procesor izvaja program, ki nastavi **x1** in **y1**, nato pa ob aktivni tipki S4 poveča **x1**.

<pre> lda 5 sta x outp 1 lda 10 sta y outp 2 loop: inp S4 jze loop lda x add 1 sta x outp 1 jmp loop x db 0 y db 0 S4 di 4 </pre>	<pre> nastavi x na 5 pošlji na x1 (adr=1) nastavi y na 10 pošlji na y1 (adr=2) beri S4 (adr1) ponovi, če je S4==0 naloži x, če je S4!=0 povečaj x za 1 pošlji na x1 in ponavljaj </pre>
---	---

3. V sistemu je delilnik ure, ki upočasni izvajanje programa mikroprocesorja. Poskusi spremeniti faktor deljenja ure v datoteki sistem.vhd in poglej, kako vpliva na delovanje.
4. Dopolni program tako, da bo ustavil prištevanje, ko pride **x** do vrednosti 760. Najprej preveri delovanje na simulatorju procesorja <https://lniv.fe.uni-lj.si/cpe/>, nato pa še na razvojni plošči.
5. Spremeni program tako, da bo bral tudi tipko S1. Ob zaznani tipki naj zmanjšuje koordinato **x**.