

LABORATORIJSKI ZADATAK 3

Sekvencijalne mreže

Potrebno predznanje

- Urađena pokazna vežba 4
- Standardne sekvencijalne mreže – registar, brojač

Šta će biti naučeno tokom izrade vežbe?

Nakon urađene vežbe:

- Naučićete da kombinujete kombinacione i sekvencijalne mreže u složen sistem,
- Razumećete merenje vremena u digitalnim sistemima,
- Poboľjšaćete vaše razumevanje projektovanja sekvencijalnih digitalnih sistema.

Apstrakt i motivacija

Jedan od prvih digitalnih sistema sa kojima ste se susreli u životu je verovatno bio digitalni sat (ili je bar bio nešto starijim generacijama koje nisu odrastale uz mobilne telefone :-)). Da li ste se ikada zapitali kako on radi? U ovoj vežbi ćete otkriti osnove funkcionisanja digitalnih časovnika i napraviti sistem koji predstavlja pojednostavljenu verziju digitalnog časovnika – štopericu koja broji sekunde. Za početak napravićete brojač koji broji jednu sekundu, a nakon toga istog iskoristiti da brojite koliko je sekundi prošlo od početka rada sistema. Krajnji proizvod će biti sistem koji broji sekunde i ima mogućnost kontrole rada – pokretanje i zaustavljanje. Koristeći iste principe kao u ovoj vežbi, sistem se može proširiti da broji minute, sate, dane i postane pravi digitalni časovnik.

Šta treba doneti na termin laboratorijske vežbe?

- Logičku i/ili blok šemu sistema na papiru ili računaru (za **krajnji izgled sistema**).
- VHDL opis **krajnjeg koraka** u realizaciji sistema (međukoraci nisu potrebni) – krajnji korak je poslednji korak koji uspete da realizujete.
- Testbench za krajnji korak sistema.
- Generisanu *.bit* datoteku za konfigurisanje E2LP platforme za krajnji korak sistema.

ZADACI

1. Brojač jedne sekunde

E2LP platforma ima ugrađeni oscilator koji generiše signal takta frekvencije 24 MHz. Bilo koja sekvencijalna mreža koju bi povezali na taj takt i koja bi menjala izlaze prema tom taktu, ne bi bila proverljiva na E2LP platformi pošto ljudsko oko nema mogućnost registrovanja promena signala pri toj frekvenciji. Kako bi provera sistema bila moguća, a i kada god je potrebno smanjiti ulazni takt, potrebno je u sistem ubaciti **delitelj takta**.

Brojač se može iskoristiti kako bi se izvršilo deljenje takta, odn. od takta veće frekvencije napraviti takt manje frekvencije. Brojač modula N ima mogućnost da podeli takt N puta, odn. ako izlaz brojača postavimo na logičku jedinicu samo u jednom od N stanja, izlazni takt će imati period N puta veći, odn. frekvenciju N puta manju od ulaznog takta. Izlaz ovog brojača treba da bude jednobitan i predstavlja signal kada je brojač završio ciklus brojanja, tj. treba da ima vrednost 1 u jednom od N stanja. Izbor stanja u kome će ovaj signal imati vrednost 1 je proizvoljan, ali najčešće se uzimaju prvo stanje (0) ili poslednje stanje (N-1). Ovaj signal se naziva **terminal count**.

Kao vaš prvi zadatak u ovoj vežbi, napravite brojač koji će da se uvećava za 1 na svakoj ivici takta. Jednobitni izlaz brojača, terminal count, treba da ima vrednost 1 na kraju jedne sekunde, u poslednjem od N stanja ovog brojača (gde je N broj taktova u jednoj sekundi).

Brojač nema kontrolnih ulaza, tj. jedini ulazi su takt i reset. Izlaz brojača je jednobitni i predstavlja terminal count signal.

Pinove dodeliti prema tabeli 1-1 (ukoliko je ovo poslednji korak vežbe koji ćete uraditi).

Tabela 1-1. Dodela pinova

Prolaz	Smer	Komponenta na E2LP platformi
iCLK	in	CLOCK
inRST	in	RESET
oTC	out	LED0

Provera brojača u simulaciji je obavezna pre nego što nastavite sa sledećim korakom, a sinteza nije neophodna dok ne završite vežbu. Tabelu 1-1 zanemarite ukoliko ćete raditi naredne zadatke.

2. Štoperica bez kontrole

Dodati još jedan 8-bitni brojač koji broji sekunde koristeći signal od brojača napisanog u prethodnom zadatku. Za ovaj brojač koristiti takođe ulazni takt od 24 MHz. Izlaz brojača treba da se uvećava za 1 svake sekunde. Ovaj brojač broji sekunde, a ne taktove, tako da ne treba da se povećava u svakom taktu. Povećanje vrednosti treba da se desi samo u onom taktu u kome prethodni brojač završi sa brojanjem jedne sekunde.

Pinove dodeliti prema tabeli 2-1.

Tabela 2-1. Dodela pinova

Prolaz	Smer	Komponenta na E2LP platformi
iCLK	in	CLOCK
inRST	in	RESET
oSEC [7:0]	out	LED7 – LED0

Ukoliko bi simulirali ovaj sistem, simulacija bi trajala dugo jer treba da prođe 24,000,000 taktova da bi prošla jedna sekunda. Kako bi simulacija bila dovoljno brza, za njene potrebe smanjiti period brojanja prvog brojača da on broji intervale od 1 μ sec, umesto 1 sec. Ali nemojte zaboraviti da ovo vratite na brojanje sekundi prilikom sinteze!

3. Kontrola štoperice

Nakon što smo napravili štopericu koja broji sekunde (bez prestanka), vreme je da uvedemo kontrolu štoperice. Dodaćemo tri kontrolna signala:

- inSTART – signal pokretanja štoperice,
- inSTOP – signal zaustavljanja štoperice,
- inCONTINUE – signal nastavka rada štoperice.

Signal inSTART treba da pokrene štopericu da broji sekunde od nule, ukoliko je ona bila ugašena. Ukoliko je bila upaljena, ovaj signal vraća štopericu u stanje 0 i ona nastavlja da broji.

Signal inSTOP treba da zaustavi štopericu.

Signal inCONTINUE treba da pusti štopericu da nastavi brojanje od stanja u kome je stala prilikom poslednjeg pritiska signala inSTOP. Kako bi rad sistema bio što precizniji, omogućiti da ovaj signal nastavi brojanje u istom delu sekunde u kome ga je inSTOP zaustavio. Npr, ukoliko je štoperica zaustavljena nakon 2.9 sec, do naredne promene stanja štoperice treba da prođe 0.1 sec nakon pritiska na inCONTINUE (a ne čitava sekunda). Ovo lako možete omogućiti pametnim izborom koji od brojača ćete kontrolisati ovim kontrolnim signalima!

Vodite računa da stanje rada štoperice (radi ili ne radi) mora negde da stoji zapamćeno i da za ovu informaciju nije dovoljna samo kombinaciona logika kontrolnih signala. Dodeliti pinove prema tabeli 3-1.

Tabela 3-1. Dodela pinova

Prolaz	Smer	Komponenta na E2LP platformi
iCLK	in	CLOCK
inRST	in	RESET
inSTART	in	JOY0
inSTOP	in	JOY2
inCONTINUE	in	JOY1
oSEC [7:0]	out	LED7 – LED0

OPŠTE NAPOMENE

Prilikom crtanja šeme na papiru ili računaru, koristiti blokove za standardne kombinacione i sekvencijalne mreže – nema potrebe da poznate komponente crtate na nivou logičkih kola. Bitno je da logička šema ispravno opiše logiku sistema. Sve nestandardne komponente koje koristite morate na neki način definisati – bilo istinitosnom tablicom ili opisom na nivou logičke funkcije.

Implementaciju sistema izvršiti za poslednji urađen korak zadatka. Nije neophodno imati urađenu implementaciju za svaki korak. Simulaciju treba raditi posle svakog koraka, jer na sledeći korak treba preći jedino ukoliko je prethodni funkcionalno proveren.

Na termin vežbe doneti **jedan** VHDL opis i **jedan** testbench (nije neophodno donositi ni opis ni testbench međukoraka). Međukoraci su tu da bi vama olakšali put ka kompletnom sistemu i omogućili da vaš sistem projektujete inkrementalno – počev od jednostavnijih komponentata ka složenijim.

ZAKLJUČAK

U ovoj vežbi videli ste kako možemo meriti vreme unutar digitalnog sistema. Primenjivost sistema koji ste napravili je mnogo šira od običnog brojača sekundi – kada god u vašem digitalnom sistemu imate potrebu za praćenjem vremena, metode koje ste primenili u ovoj vežbi su primenjive i pri zadovoljavanju te potrebe. Ova vežba vas je upoznala i sa deliteljima takta koji su neizostavni deo složenijih digitalnih sistema. Na ovom predmetu se nećemo baviti ovom problematikom, odn. naši sistemi će raditi na jednom taktu, onom koji dolazi sa oscilatora na platformi. Složeniji sistemi često koriste više taktova i sastoje se iz više domena takta, a sinhronizacija signala između različitih domena takta predstavlja izazov koji prevazilazi naš predmet. Bez obzira, vi ste u ovoj vežbi upoznati sa najjednostavnijim načinom smanjenja takta u sistemu – jednostavnim brojanjem! Više o podeli takta i takt domenima možete pročitati u literaturi iz projektovanja digitalnih sistema, no mi se nećemo baviti tom temom u nastavku ovih vežbi.