

POKAZNA VEŽBA 5

Sekvencijalne mreže – složeni sistemi

Potrebno predznanje

- Urađene pokazne vežbe: 1, 2, 3 i 4
- Standardne sekvencijalne mreže – registri, brojači, pomerački registri

Šta će biti naučeno tokom izrade vežbe?

Nakon urađene vežbe, bićete u mogućnosti da:

- Razumete primenu memorijskih elemenata u digitalnim sistemima
- Projektujete digitalne sisteme sa memorijskim elementima
- Opišete sekvencijalne mreže pomoću VHDL jezika za opis digitalnih sistema
- Napišete test bench za verifikaciju sekvencijalnih mreža
- Projektujete sisteme sa više sekvencijalnih i kombinacionih mreža

Apstrakt i motivacija

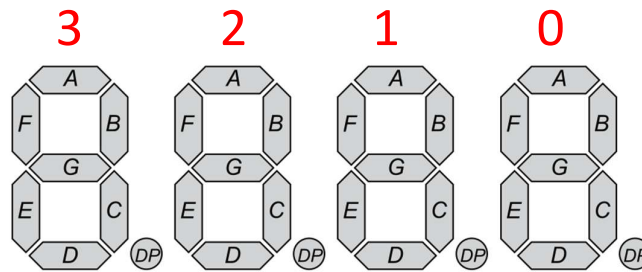
Složeni sekvencijalni sistemi igraju ključnu ulogu u mnogim modernim elektronskim uređajima. Ova vežba prikazuje koncepte složenih sekvencijalnih sistema i vremenskog multipleksiranja i pokazuje mogućnosti njihove primene. Efikasnim upravljanjem resursima i komunikacijom između različitih komponenti elektronskih uređaja, postiže se optimizacija performansi. Ova vežba takođe pokazuje prednosti vremenskog multipleksiranja kao tehnike za deljenje resursa i efikasnu upotrebu ograničenih hardverskih resursa.

TEORIJSKE OSNOVE

1. Vremensko multipleksiranje

Kako bi više različitih korisnika koristilo isti medijum za prenos podataka bez ometanja, koriste se razne metode za višekorisnički pristup. Jedna od najčešće korišćenih metoda je multipleksiranje, koja omogućava da se podaci iz više izvora koriste istovremeno. Postoji više načina multipleksiranja, a za potrebe vežbi ćemo koristiti **vremensko multipleksiranje** koje podrazumeva da se različitim korisnicima dodeljuju različiti intervali (slotovi) koje mogu koristiti za komunikaciju.

Vremensko multipleksiranje (engl. *Time-Division Multiplexing* ili TDM) je tehnika koja se koristi za kontrolu više 7-segmentnih displeja (ili drugih uređaja) povezanih na iste pinove mikrokontrolera, koja omogućava da se naizmenično prikazuje informacija na svakom od displeja, tako da se korisniku čini da svi displeji rade istovremeno.



Slika 1-1. 7-segmentni displeji na MAX1000 platformi

MAX 1000 platforma ima četiri 7-segmentna displeja koji koriste **ISTE PINOVE** za aktiviranje segmenata **a-g**, tako da je u određenom trenutku **uključen samo jedan displej**. Uz pomoć vremenskog multipleksiranja i nesavršenosti ljudskog oka da primeti toliko brzo osvežavanje prikaza, mi nemamo utisak da se zapravo displeji uključuju i isključuju naizmenično. Kako bismo aktivirali određeni displej potrebno je podesiti pinove za ***o_mux_row_or_digit_0*** i ***o_mux_row_or_digit_1***, kao u Tabeli 1-1. Displeji su označeni zdesna nalevo brojevima 0-3. Ukoliko se ove vrednosti ne podese, podrazumevano se uključuje displej 3.

Tabela 1-1: Aktiviranje više 7-segmentnih displeja

7-segmentni displej	<i>o_mux_row_or_digit_1</i>	<i>o_mux_row_or_digit_0</i>
0	0	0
1	0	1
2	1	0
3	1	1

Za naizmenično korišćenje displeja za prikazivanje može se koristiti takt generator ili programski brojač kako bi se odredila brzina prebacivanja između displeja. Nakon što je prvi displej prikazao svoj sadržaj, on se isključuje, a drugi displej se uključuje i postavlja se za prikazivanje svog sadržaja. Ovaj proces se ponavlja za svaki displej u sekvenci. Brzina prelaska između displeja je obično toliko brza da ljudsko oko doživljava kontinuiran prikaz svih displeja kao da su istovremeno aktivni.

Ljudsko oko može da primeti promene u osvežavanju displeja otpilike na **25Hz** što je oko **40 ms**, što bi na platformi koja radi na **12 MHz** značilo da promene nisu vidljive otprilike na 15. bitu brojača koji broji do 1 sekunde. Kako bismo omogućili vremensko multipleksiranje najbolje je koristiti 14. i 15. bit brojača sekunde i na osnovu njih menjati vrednosti ***o_mux_row_digit***.

ZADACI

1. Brojač – primena

1.1. 24-bitni brojač na led diodama

U VHDL-u implementirati 24-bitni brojač **scnt** koji uvećava vrednost za 1 na svaku rastuću ivicu takt signala ukoliko je signal dozvole brojanja (**iEN**) aktivan. Uzeti da je aktivna vrednost dozvole 1. Uraditi implementaciju sistema na MAX 10 platformi i prikazati najviše bite brojača (23 downto 16) na led diodama. Obratiti pažnju na ponašanje led dioda i razliku intenziteta osvetljenja.

Tabela 1-2. Pinovi za povezivanje komponente brojač

Prolazi	Smer	Komponenta	Location	I/O standard
iCLK	input	I_CLK	PIN_H6	3.3-V LVTTTL
iRST	input	I_RST	PIN_F1	3.3 V Schmitt Trigger
iEN	input	I_SW_0	PIN_M3	3.3 V Schmitt Trigger
oCNT [23:16]	output	oLED[7:0]	D8, C10, C9, B10, A10, A11, A9, A8	3.3-V LVTTTL

1.2. Brojač sekundi na 7-segmentnom displeju

Prethodno implementirani 24-bitni brojač **scnt** izmeniti tako da broji po modulu **N**, tačnije dok na platformi MAX10 ne prođe 1 sekunda. Vrednost N ćemo odrediti pomoću radnog takta platforme. Ako znamo da ploča MAX10 radi na 12MHz, do koliko brojač treba da broji da bi u sistemu prošla jedna sekunda?

Kada brojač izbroji do 1s, potrebno je aktivirati interni signal **stc**, koji pokazuje da je u sistemu prošla jedna sekunda.

Napraviti još jedan 4-bitni brojač **scnt_seg**, koji će da broji sekunde u sistemu. Brojač sekundi treba da radi po modulu 10 i da povećava svoju vrednost za 1 kada je aktivan signal **stc**.

Dodati komponentu za prikaz 4-bitne vrednosti brojača sekundi **scnt_seg** na 7-segmentnom displeju **o7SEGM**.

Za simulaciju sistema podesiti maksimalnu vrednost 24-bitnog brojača na **12**, a zatim proveriti ispravnost sistema.

Nakon simulacije sistema, proveriti ispravnost na MAX10 platformi postavljanjem maksimalne vrednosti 24-bitnog brojača na **12 000 000** i odgovarajućih pinova kao u Tabeli 1-3.

Tabela 1-3. Pinovi za povezivanje komponente brojač

Prolazi	Smer	Komponenta	Location	I/O standard
iCLK	input	I_CLK	PIN_H6	3.3-V LVTTTL
iRST	input	I_RST	PIN_F1	3.3 V Schmitt Trigger
iEN	input	I_SW_0	PIN_M3	3.3 V Schmitt Trigger
oCNT [23:16]	output	oLED[7:0]	D8, C10, C9, B10, A10, A11, A9, A8	3.3-V LVTTTL
o7SEGM[6:0]	output	o_n_col_0_or_7segm_a ... o_n_col_0_or_7segm_g	H8, K10, H5, H4, J1, J2, L12	3.3-V LVTTTL

2. Složeni sekvencijalni sistem

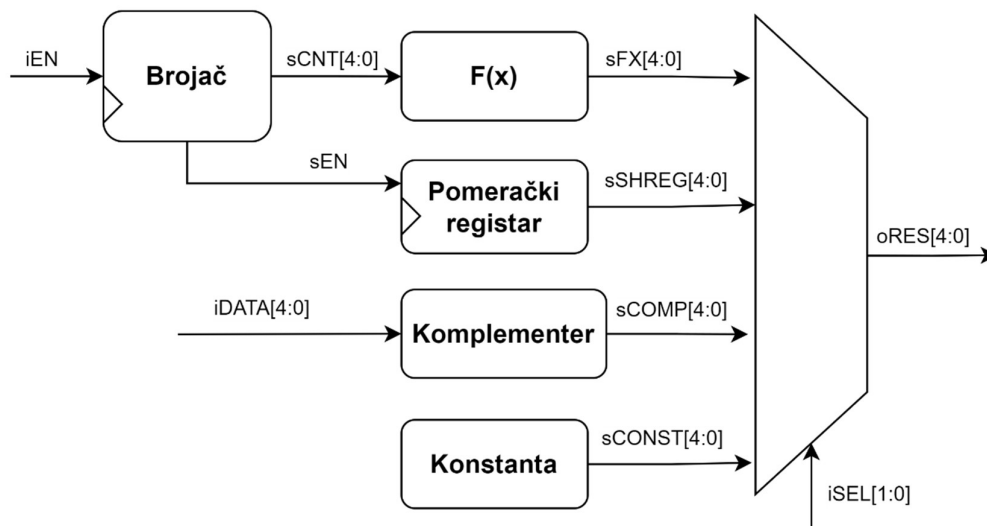
U VHDL jeziku za opis digitalnih sistema opisati i simulirati digitalni sistem prikazan na slici.

Ulazi i izlazi digitalnog sistema:

- **iCLK** – signal takta, podrazumevan u svim sekvencijalnim mrežama
- **iRST** – signal reseta, aktivan na log. 1, asinhron, podrazumevan u svim sekvencijalnim mrežama
- **iEN** – signal dozvole rada brojača
- **iSEL [1:0]** – signal selekcije multipleksera
- **iDATA [4:0]** – ulazni vektor podataka koji prolazi kroz komplementer
- **oRESULT [4:0]** – izlaz iz multipleksera

Opis sistema:

- Brojač broji po modulu 32. Ukoliko ima dozvolu rada **iEN** treba da uvećava svoju vrednost za 1 na svaku rastuću ivicu. U zavisnosti od vrednosti brojača će pomerački registar rotirati svoj sadržaj.
- Pomerački registar **rotira** svoju vrednost udesno kada je vrednost brojača parna. Bit koji bi „ispao“ pomeranjem udesno dolazi na mesto najznačajnijeg bita. Početna vrednost pomeračkog registra je 10.
- Komplementer obavlja operaciju komplementiranja ulaza **iDATA**, odnosno na izlazu treba da se nađe vrednost $\sim iDATA$ (posmatrati da je **iDATA** označen u II komplementu).
- $F(x)$ je blok čija je funkcionalnost opisana jednačinom $F(x) = x+3$. Vrednost brojača je ulaz za blok $F(x)$.
- Multiplexer u zavisnosti od vrednosti ulaza **iSEL** na izlaz prosleđuje vrednost iz $F(x)$ ako je **iSEL** = 0, iz pomeračkog registra za vrednost **iSEL** = 1, komplementera za **iSEL** = 2 ili konstantu 17 za **iSEL** = 3. Broj bita koji predstavlja ovu konstantu prilagoditi ostatku sistema.



Slika 1 – Arhitektura sistema

Sistem simulirati na sledeći način:

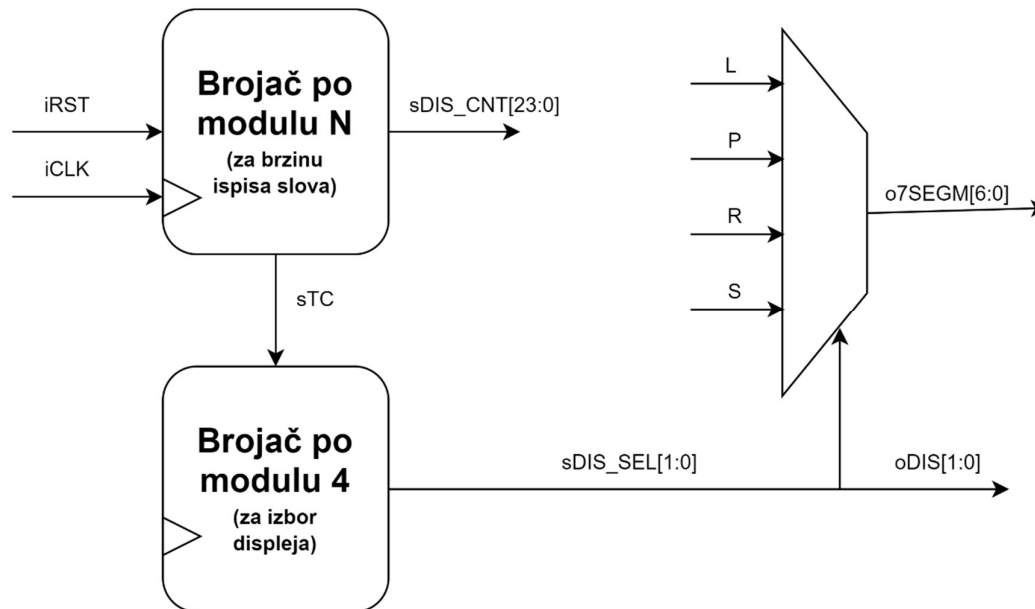
- Resetovati sistem 2.25 perioda takta
- 10 perioda takta prikazivati izlaz iz $F(x)$
- 20 perioda takta prikazivati izlaz iz pomeračkog registra ali tako da barem 5 uzastopnih perioda takta registar ne promeni svoju vrednost
- Prikazati kao izlaz iz komplementera vrednost -4 i zadržati na izlaz 2 periode takta
- 5 perioda takta prikazivati na izlazu konstantu 17

3. Ispis LPRS na 4 displeja

U VHDL jeziku za opis digitalnih sistema opisati i implementirati digitalni sistem za ispisivanje skraćenice **LPRS** na četiri 7-segmenta displeja.

Ulazi i izlazi digitalnog sistema:

- **iCLK** – signal takta, podrazumevan u svim sekvencijalnim mrežama
- **iRST** – signal reseta, aktivan na log. 1, asinhron, podrazumevan u svim sekvencijalnim mrežama
- **oDIS[1:0]** – izlazni signal za podešavanje ispisa na jedan od četiri 7-segmentna displeja
- **o7SEGM [6:0]** – izlazni signal za segmente a - g



Slika 2 – Arhitektura sistema

Sistem se sastoji iz 24-bitnog brojača **sDIS_CNT** koji broji po modulu N i uvećava svoju vrednost za 1 na svaku rastuću ivicu takt signala. U zavisnosti od modula brojača slova se prikazuju istovremeno ili jedno za drugim:

- Manji moduo (N ima jedinicu na 14. ili 15. bitu) rezultuje bržim izmenama koje oko ne registruje, pa deluje da su sva 4 displeja aktivna istovremeno.
- Veći moduo (N ima jedinicu na 23. ili 22. bitu) rezultuje sporijim izmenama i tada se slova prikazuju jedno za drugim.

Kada je brojač po modulu N izbrojao do maksimalne vrednosti, aktivira se terminalni signal **sTC**, koji treba da omogući promenu vrednosti kod drugog brojača po modulu 4.

Brojač **sDIS_SEL** je širine 2 bita i broji **UNAZAD** po modulu 4. Ovaj brojač ima početnu vrednost 3 i umanjuje svoju vrednost za 1 na svaku rastuću ivicu takt signala i ukoliko je signal **sTC** aktivan. Na ovaj način je omogućen prikaz na četiri 7-segmentna displeja jedan za drugim ili istovremeno.

Multiplexer preuzima vrednost **sDIS_SEL** i na izlaz 7-segmentnog displeja **o7SEGM** prosleđuje jedno od slova: L, P, R ili S u formatu za uključivanje segmenata a-g (Tabela 1).

Vrednost <i>sDIS_SEL</i>	<i>oDIS</i>	<i>o7SEGM</i>
3	„11“	"1110001" -- slovo L
2	„10“	"0011000" -- slovo P
1	„01“	"0001000" -- slovo R
0	„00“	0100100" -- slovo S

Simulirati ispravnost sistema, ali uz smanjenu vrednost modula N = 10, a zatim proveriti ispravnost na platformi MAX 10 korišćenjem različitih vrednosti modula N i povezivanjem sledećih pinova:

Prolazi	Smer	Komponenta	Location	I/O standard
iCLK	input	I_CLK	PIN_H6	3.3-V LVTTTL
iRST	input	I_RST	PIN_F1	3.3 V Schmitt Trigger
oDES [1:0]	output	o_mux_row_or_digit_0 o_mux_row_or_digit_1	PIN_J13, PIN_K11	3.3-V LVTTTL
o7SEGM[6:0]	output	o_n_col_0_or_7segm_a ... o_n_col_0_or_7segm_g	H8, K10, H5, H4, J1, J2, L12	3.3-V LVTTTL

ZAKLJUČAK

Ova vežba je pokazala primer složenog sistema koji koristi kombinacione i sekvencijalne komponente, kao i način opisivanja ovakvih sistema u VHDL jeziku. Sada poznajete oba glavna tipa digitalnih sistema – kombinacione i sekvencijalne mreže. Njihovom kombinacijom se projektuju svi složeniji digitalni sistemi. U nastavku predmeta neće biti naučeno ništa konceptualno novo, svaki digitalni sistem je sastavljen od ova dva tipa komponenti. U ostatku vežbi naučićete kako da primenite ova dva osnovna tipa u projektovanju sistema koji rade određenu funkciju, kao što su automati, aritmetička kola, upravljačka kola i kako da povezujete više jednostavnijih sistema u jedan složen koristeći modularnost u opisu. No, osnovno znanje ste upravo stekli i koristeći do sada naučeno možete napraviti proizvoljno složen digitalni sistem.