

Uputstvo za rad sa Xilinx Vivado programskim paketom

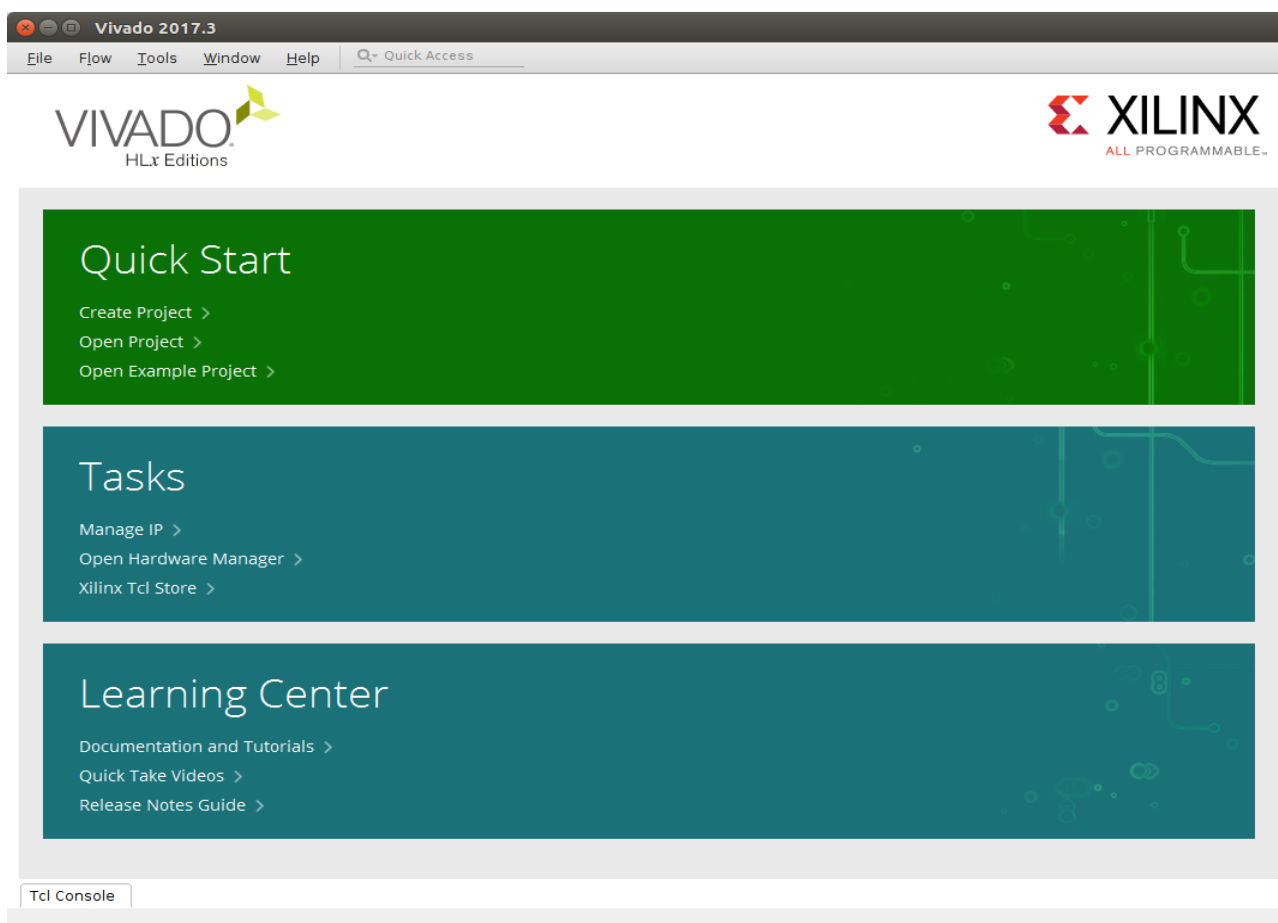
Novi Sad, februar 2018.

Startovanje programskog paketa

Xilinx Vivado 2017.3 programski paket se aktivira dvoklikom na odgovarajuću ikonicu na desktopu ili aktiviranjem u Start meniju Windowsa iste opcije.

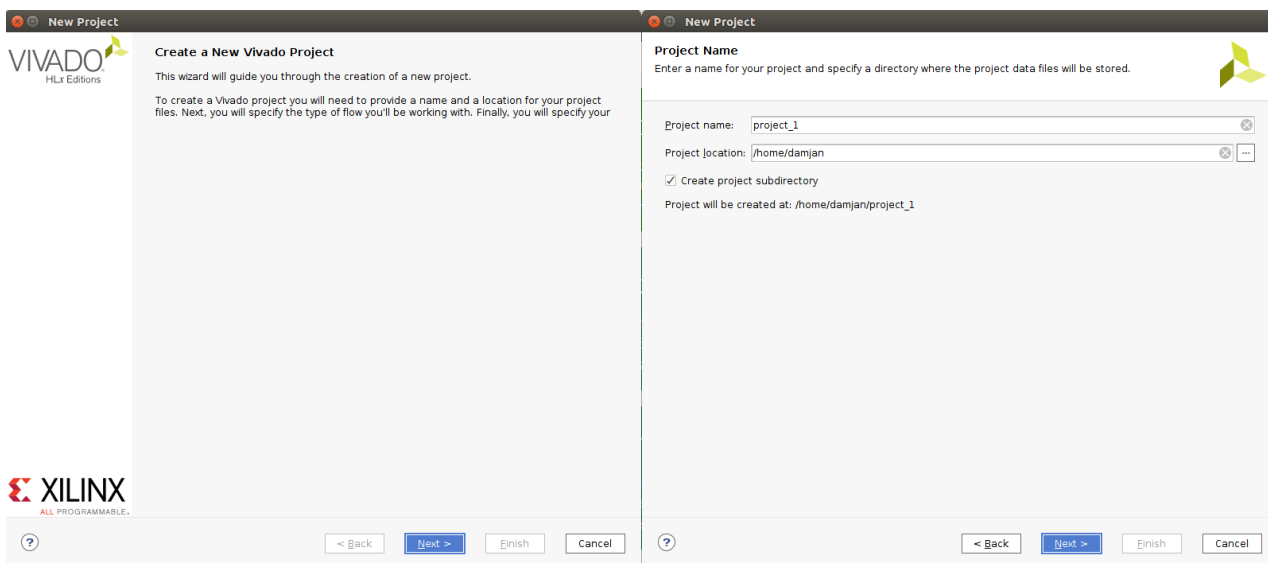
Kreiranje novog projekta

Kreiranje novog projekta počinje nakon odabira opcije *Create Project* u start meniju prikazanom na slici 1. Otvaraju se prozori (slika 2) od kojih je u prvom potrebno odabrati Next, a u drugom odrediti lokaciju gde će projekat biti snimljen i naziv projekta.

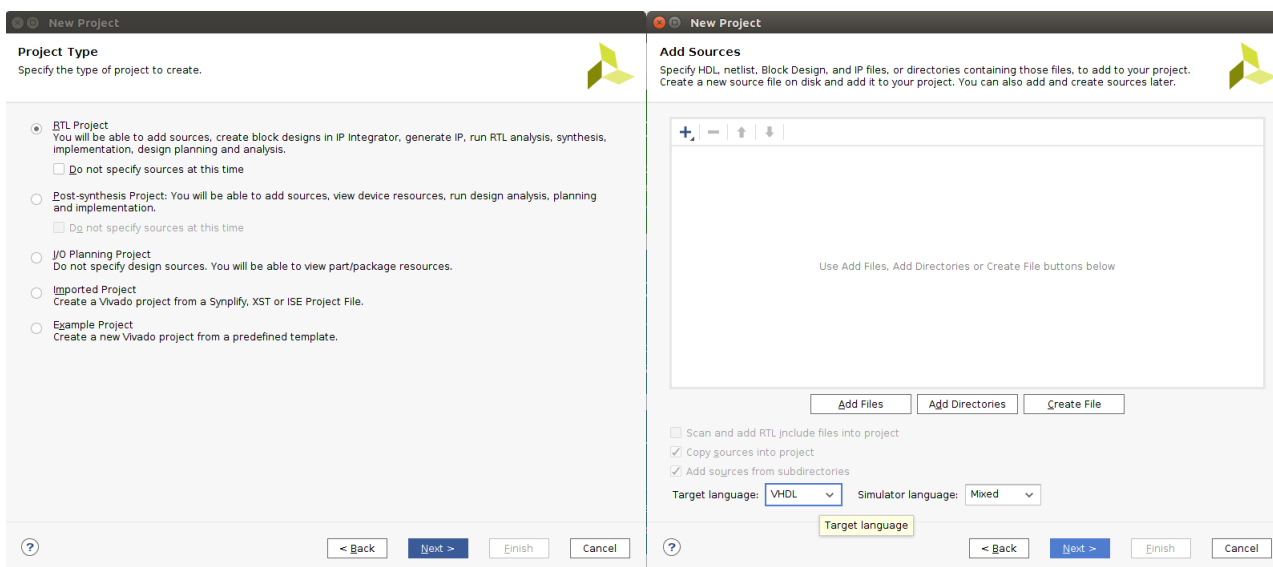


Slika 1: Kreiranje novog projekta

Nakon što smo uneli potrebne podatke (npr. *project_1*), pritiskom na *Next* se otvara novi prozor (slika 3 levo) u kome treba odabrati *RTL Project* i pritisnuti *Next*. Na sledećem prozoru je potrebno odabrati ciljani jezik koji je u našem slučaju *VHDL* i ponovo pritisnuti *Next*. Sledeći prozor koristimo u slučaju da hoćemo da dodamo fajlove sa ograničenjima projekta koji, za sada preskačemo i pritisnemo *Next*.



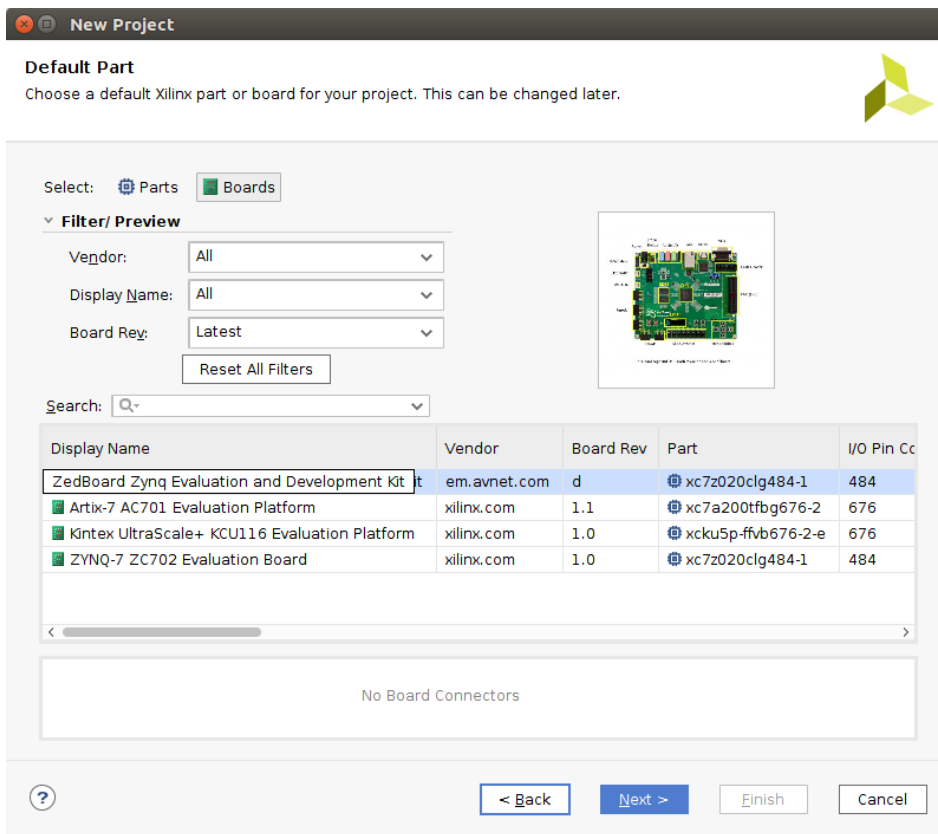
Slika 2: Odabir imena i lokacije čuvanja projekta



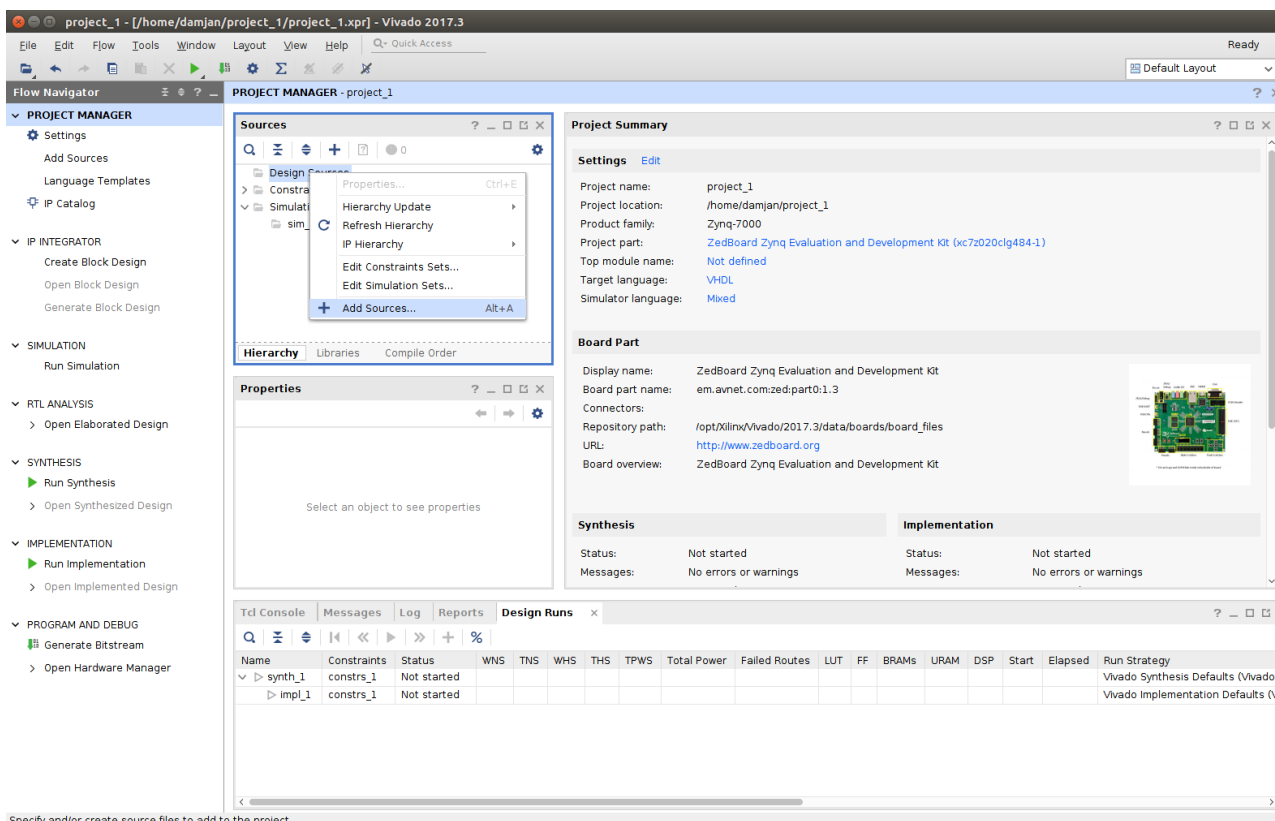
Slika 3: Određivanje tipa projekta i ciljanog jezika

Na sledećem prozoru je potrebno odabrati platformu koju ćemo koristiti prilikom implementacije modula koji ćemo projektovati. U prvih nekoliko vežbi nam to neće biti od velikog značaja pošto ćemo se implementacijom baviti u drugom delu kursa. Za sada ćemo odabrati *ZedBoard Zynq Evaluation and Development Kit* kao na slici 4. Sledeći prozor sumira odabrana podešavanja. Ukoliko smo zadovoljni, pritiskamo Next i otvara nam se kreirani projekat kao na slici 5.

Uvod u mikrorračunarsku elektroniku



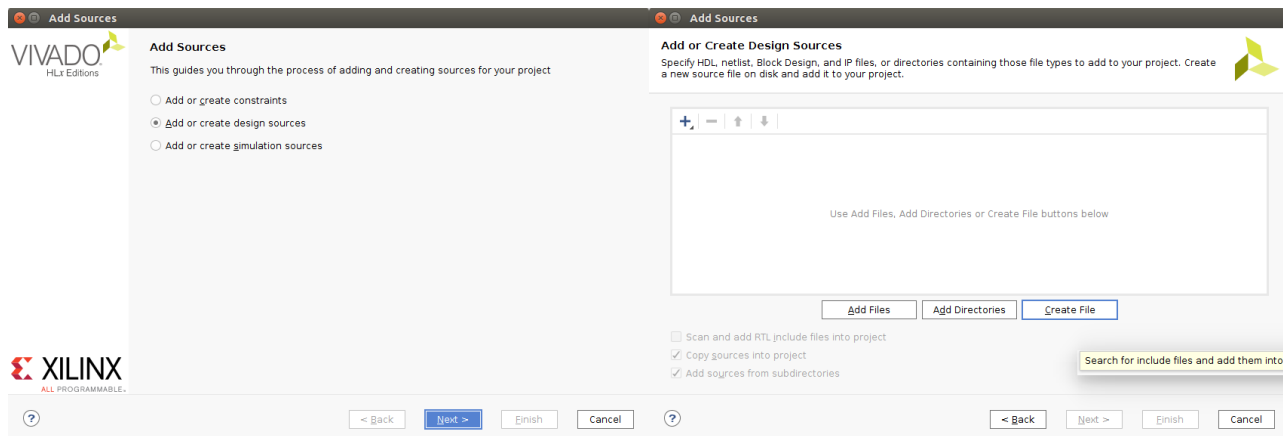
Slika 4: Odabir platforme



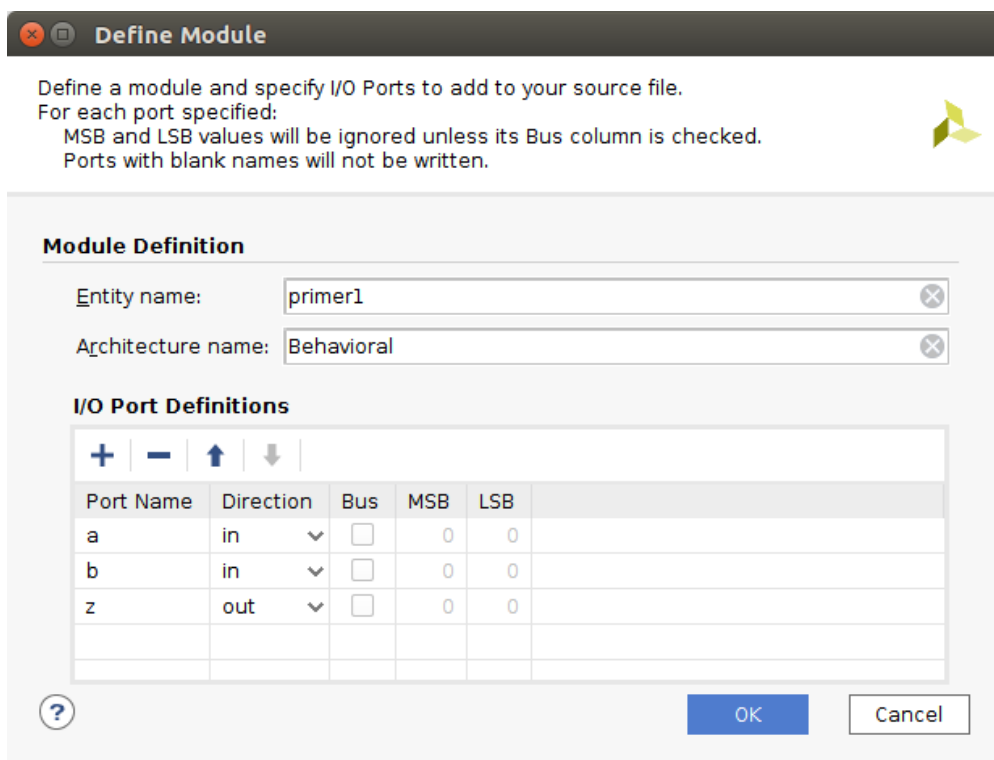
Slika 5: Kreirani projekat

Dodavanje novih modula

Pošto smo kreirali projekat, potrebno je dodati/napisati fajlove koji opisuju željeni modul. U *Sources* delu projekta pritisnite desni klik na *Design Sources* i odaberite *Add Sources* kao na slici 5. Otvoriće se prozor kao na slici 6 gde je potrebno odabrati *Add or create design sources* i pritisnuti *Next*, a zatim odabrati opciju *Create File* (slika 6 desno).



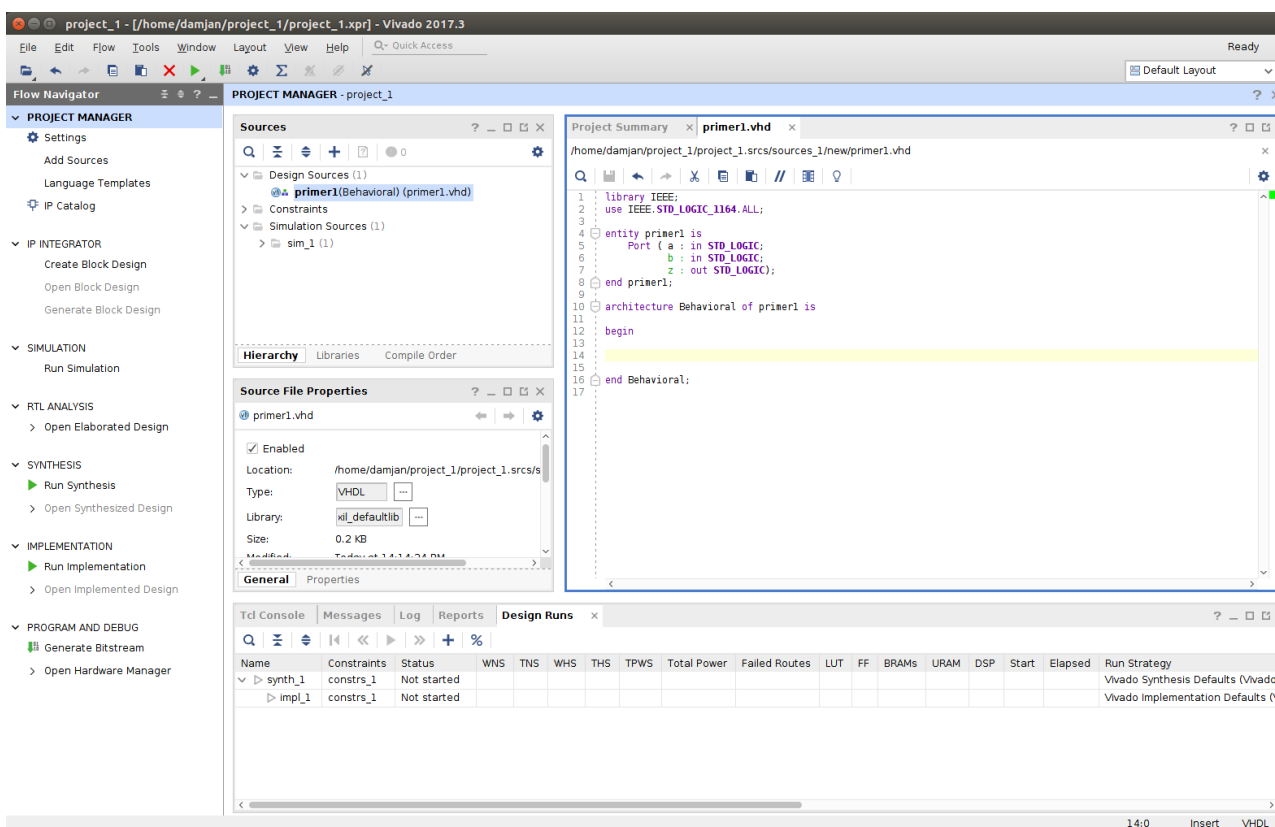
Slika 6: Dodavanje novog fajla u projekat



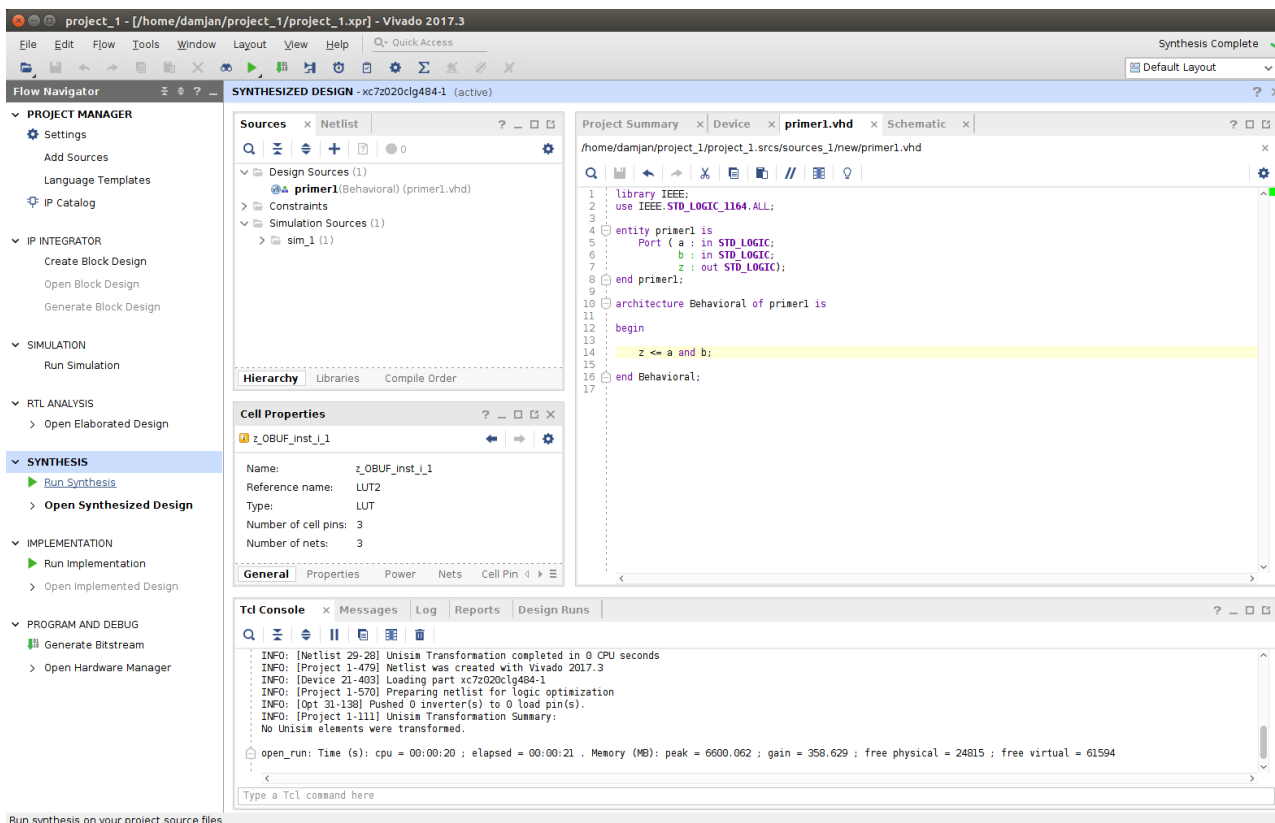
Slika 7: Definisanje portova modula

Pritiskom na *Create File* se otvara prozor u kome je potrebno upisati željeno ime fajla i pritisnuti *OK*. Ukoliko smo završili sa dodavanjem fajlova pritisnemo *Finish* u prozoru *Add Sources* (slika 6 desno). Pošto ove akcije otvara se prozor *Define Module* u kome možemo definisati nazive, tip (ulazni – *in*, izlazni – *out* ili ulazno-izlazni – *inout*) i širinu magistrale (slika 7). Nakon ovoga vraćamo se na glavni prozor u kome je dvoklikom na kreirani fajl *primer1* moguće otvoriti fajl u *Vivado* alatu kao na slici 8.

Uvod u mikrorračunarsku elektroniku



Slika 8: Automatski kreiran fajl posle definisanja portova



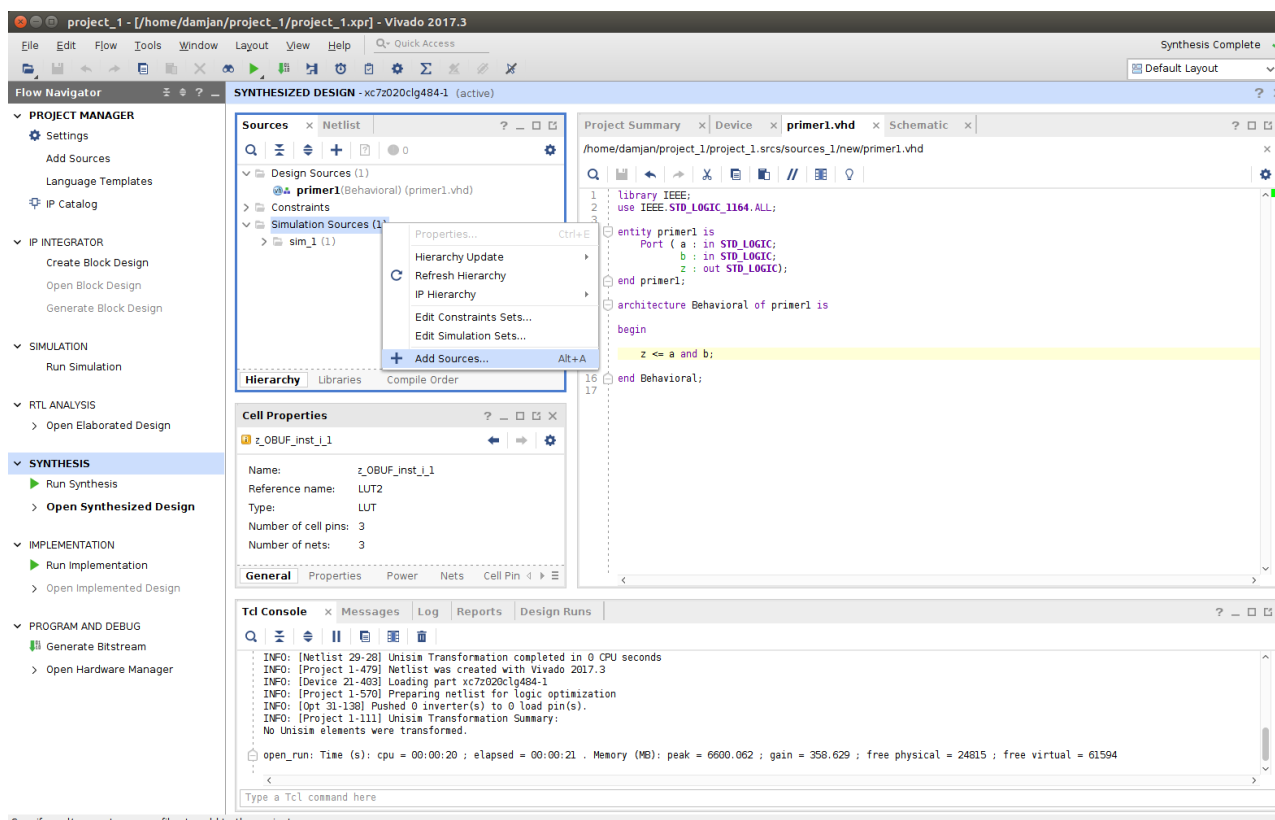
Slika 9: Izgled dopunjenog fajla koji opisuje dvoulazno I kolo

Pre bilo kakvog daljeg rada treba dopuniti VHDL program tako da radi nešto konkretno.

Kao najjednostavniji primer uzećemo da su *a* i *b* ulazi dvoulaznog i kola, a *z* izlaz. Dopunjeni kod izgleda kao na slici 9.

Simulacija

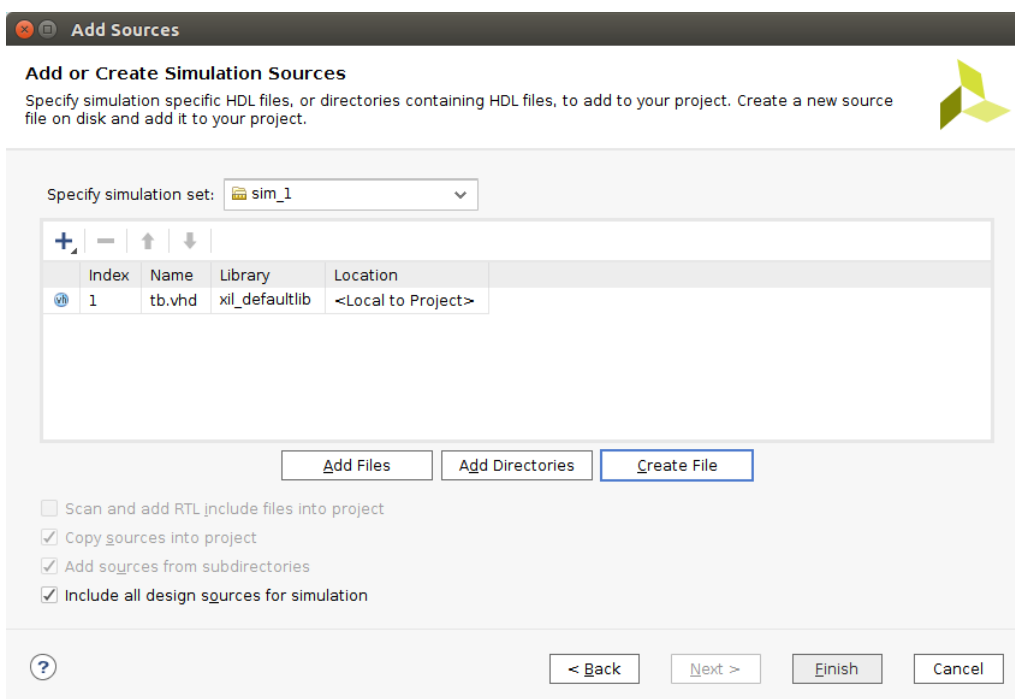
Ukoliko želimo da radimo simulaciju, potrebno je dodati u projekat VHDL *test bench* program (kraće *tb*). To je program u kome ćemo zadati ulazne test signale – test vektore. Da bi smo dodali *tb* program, treba kliknuti desnim na *Simulation Sources* u prozoru *Sources* i pritisnuti *Add Sources* (slika 10).



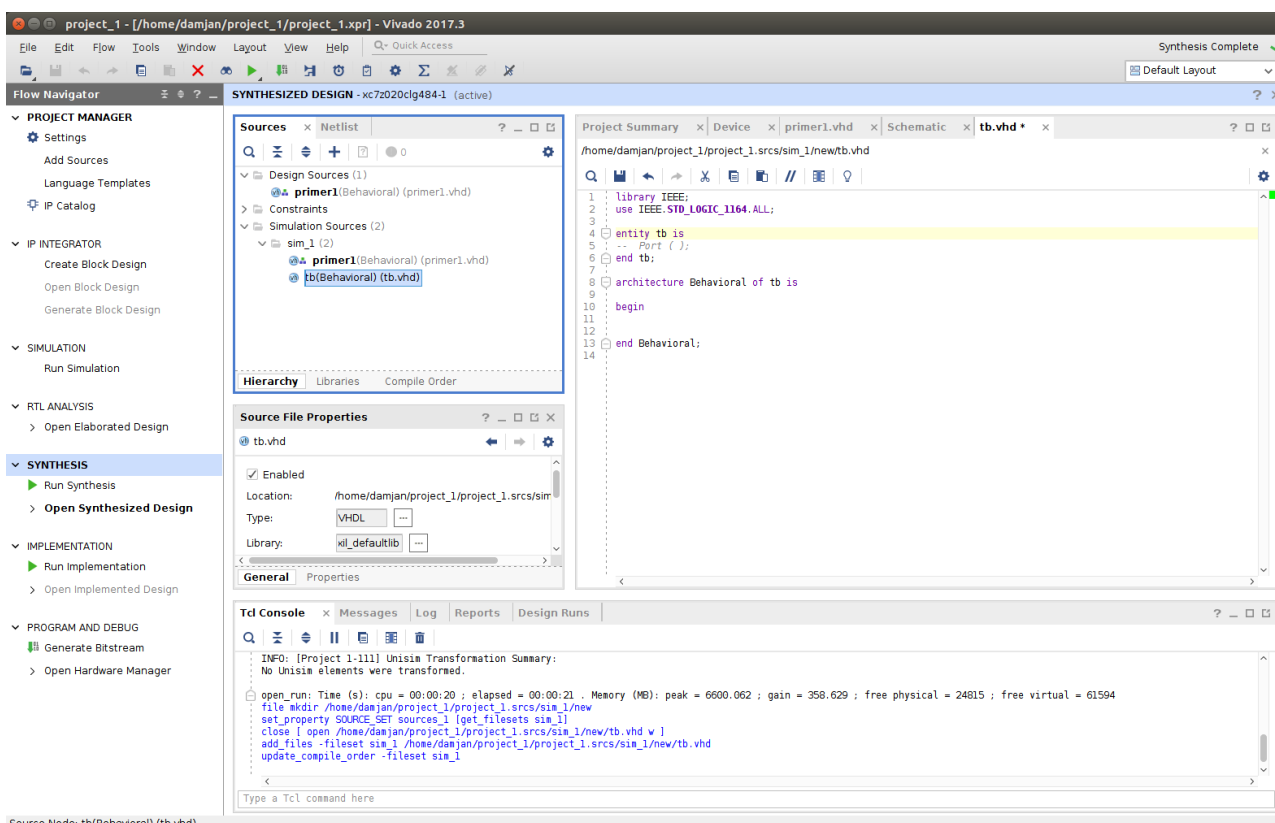
Slika 10: Dodavanje simulacionog fajla (*test bench-a*)

Slično kao i prilikom kreiranja *design* fajla, otvara se prozor u kome biramo opciju *Add or create simulation sources* i pritisakmo *Next*. Potom pritisakmo *Create File* i upisujemo ime fajla za simulaciju. Pritisak *OK* dugmeta nas vraća na predhodni prozor u kome je potrebno pritisnuti *Finish* (slika 11).

Kao i u slučaju pravljenja *design* fajla, i ovde se otvara prozor *Define Module* u kome možemo da definišemo portove modula. Pošto je *tb* fajl najviši u hijerarhiji nije potrebno dodavati portove već samo pritisnuti *OK* nakon čega se vraćamo na glavni prozor. Dvoklikom na *tb* otvara prozor kao na slici 12.



Slika 11: Dodavanje novog simulacionog fajla



Slika 12: Izgled novog tb fajla

Program je potrebno dopuniti instancom modula koji ćemo testirati (I kolo), signalima koji povezuju I kolo i stimulus i na kraju stimulusom. Po završetku program treba da izgleda kao na slici 13.


```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity tb is
-- Port ( );
end tb;

architecture Behavioral of tb is
    signal a : std_logic;
    signal b : std_logic;
    signal z : std_logic;
begin

    Ikolo:
    entity work.primer1(Behavioral)
    port map(a=>a,
            b=>b,
            z=>z);

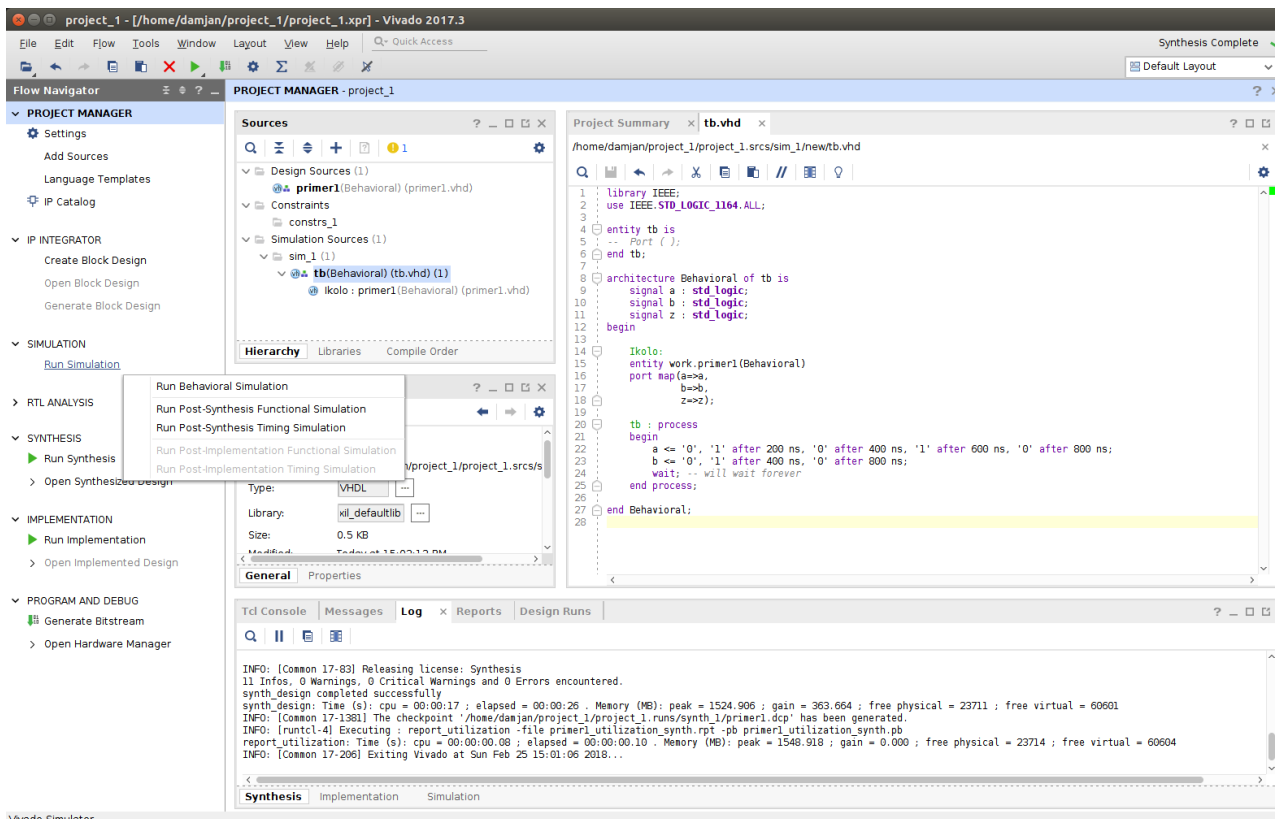
    tb : process
    begin
        a <= '0', '1' after 200 ns, '0' after 400 ns, '1' after
600 ns, '0' after 800 ns;
        b <= '0', '1' after 400 ns, '0' after 800 ns;
        wait; -- will wait forever
    end process;

end Behavioral;
```

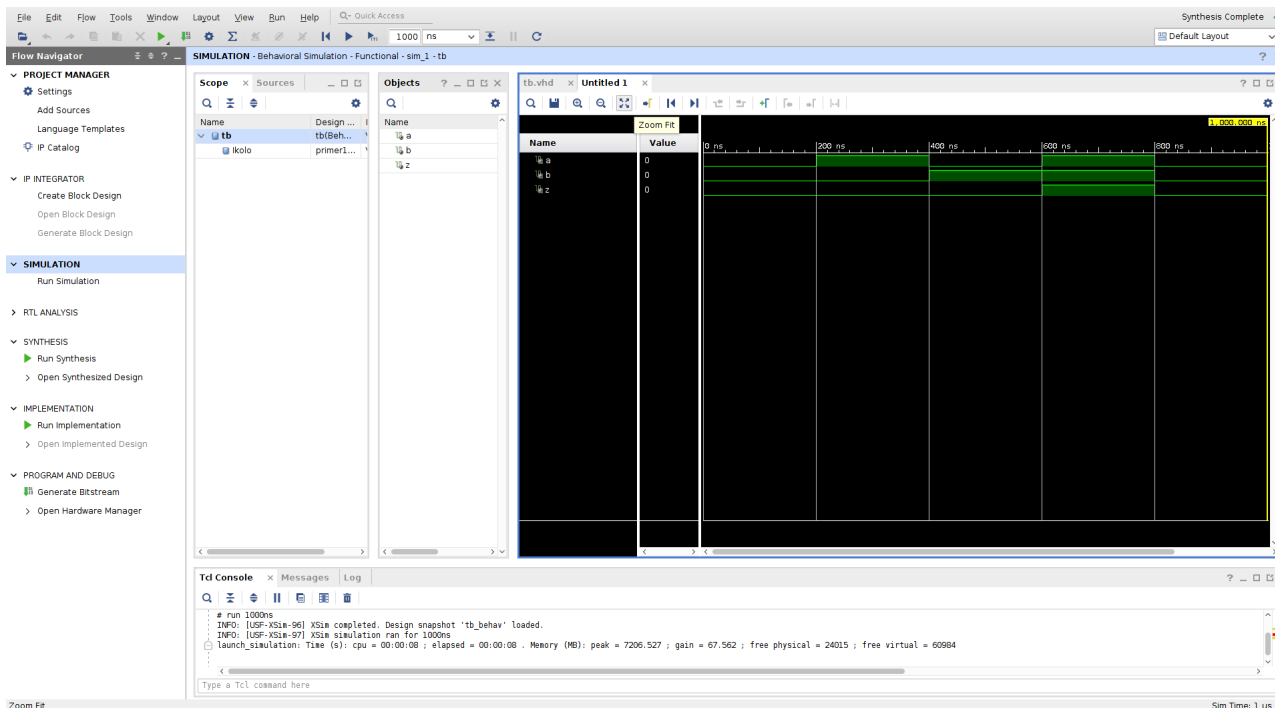
Da bismo pokrenuli simulaciju potrebno je u *Flow Navigator*-u kliknuti na *Run Simulation* polje koje se nalazi u *Simulation* delu navigatora. Potom biramo opciju *Run Behavioral Simulation* kao na slici 13.

Na slici 14 je prikazan rezultat simulacije u kome jasno vidimo da se model doulaznog I kola ponaša u skladu sa očekivanjima.

Uvod u mikroročunarsku elektroniku



Slika 13: Konačni izgled simulacionog fajla i pokretanje bihevoralne simulacije



Slika 14: Prikaz signala testiranog I kola