

\*\*\*بسمه الله الرحمن الرحيم\*\*\*

موضوع :

آموزش نوشتن Test bench در VHDL همراه مثال

تهیه و تنظیم : فرحناز صادقی درجزینی

## فهرست

۳	.....مقدمه
۴	.....Test bench چیست؟
۴	.....نوشتن کد به زبان VHDL
۵	.....طراحی برنامه به روش Test bench
۸	.....اجرای برنامه simulate با Modelsim
۱۴	.....ساخت فایل Test bench و شبیه سازی برنامه
۱۹	.....منابع

## مقدمه

نرم افزار Quartuse 2 دارای محیط ها و قابلیت های متفاوت است , Test bench یکی از قابلیت های این نرم افزار است که در شبیه سازی برنامه VHDL و شماتیک استفاده میشود . در این پروژه نحوه نوشتن برنامه Test bench در محیط VHDL , ایجاد فایل Modelsim و شبیه سازی Test bench بررسی شده است .

## Test bench چیست ؟

Test bench شبیه wave form است اما چون در HDL نوشته میشود امکان انعطاف پذیری بیشتری دارد و همچنین فضا برای پیچیدگی در Test bench بیشتر است , بدین صورت که با آن می توان یک قطعه سخت افزاری ایجاد کرد. این عوامل موجب مطلوب بودن Test bench شده است. جهت ایجاد فایل Test bench ابتدا باید برنامه نویسی vhdl خود را با توجه به توضیحات زیر تغییر دهیم , برای مثال برنامه گیت and را در نظر بگیرید مراحل کار به صورت زیر است :

- ۱- نوشتن کد به زبان vhdl
- ۲- طراحی برنامه به روش test bench
- ۳- اجرای برنامه به روش simulate یا modelsim
- ۴- ساخت فایل شبیه سازی به کمک test bench

## نوشتن کد به زبان VHDL

```
library ieee;

use ieee.std_logic_1164.all;

Entity AND_GATE is
Port ( in1 : in std_logic ;
      in2 : in std_logic ;
      out1: out std_logic
);

End AND_GATE ;

Architecture behavioral of AND_GATE is

Begin

Out1<= in1 and in2 ;

End behavioral ;
```

## طراحی برنامه به روش Test bench

توجه داشته باشید که برای نوشتن برنامه test bench باید چند تغییر انجام دهیم .

۱- تعریف قطعه در معماری : برای طراحی برنامه به این روش باید قطعه تعریف نماییم برای این عمل از دستور component استفاده می نماییم . ورودی و خروجی های قطعه مانند ورودی و خروجی های entity تعریف میشود . و دستور component را باید در معماری تعریف نماییم . فرم کلی این دستور به صورت زیر است :

Component name

Port(

ورودی ها و خروجی ها

);

End component;

۲- نمونه گیری از یک قطعه : برای آنکه از قطعه تعریف شده نمونه گیری کنیم از دستور port map استفاده می نماییم , این دستور در معماری تعریف میشود و فرم کلی آن به صورت زیر است :

فرم کلی : L1: name port map ( in1=> in1 ,

in2=>in2 ,

Out1=>out1

);

توجه داشته باشید نام port map و component باید یکی باشد .

۳- در صفحه test bench میتوانیم گزارش بنویسیم که در انتهای simulate نمایش داده شود .

با کمک دستور report این کار را انجام میدهیم :

Assert false

Report "message";

Severity failure ;

۴- توجه داشته باشید که در متن اصلی برنامه بعد Port map باید در خطوط برنامه از تاخیر استفاده کنیم .  
از

برای تاخیر از دستور WAIT استفاده میشود این دستور داخل PROCESS تعریف میشود .

برای مثال میخواهیم یک تاخیر 10ms ایجاد کنیم :

Wait for 10ms ;

برای برنامه گیت And داریم  
:

--and\_test\_bench

library ieee;

use ieee.std\_logic\_1164.all;

entity sadeghi\_test\_bench is

Architecture a of sadeghi\_test\_bench is

Component s\_test\_bench

port(

in1: in std\_logic;

in2: in std\_logic;

out1: out std\_logic

);

End component;

Signal in1: std\_logic := '0' ;

Signal in2: std\_logic := '0' ;

Signal out2: std\_logic := '0' ;

U1: s\_test\_bench port map( in1=>in1 ,

in2=>in2,

Out1=>out1

);

PROCESS

Begin

Wait for 10ms ;

In1 <='1' ;

Wait for 10ms ;

In2<='0' ;

In1<='1' ;

Wait for 10ms ;

In1<='1' ;

Wait for 10ms ;

Assert false

Report "simulation over" ;

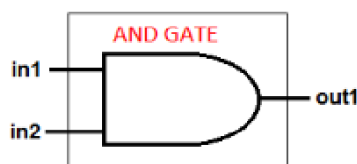
Severity failure ;

End process ;

End a ;

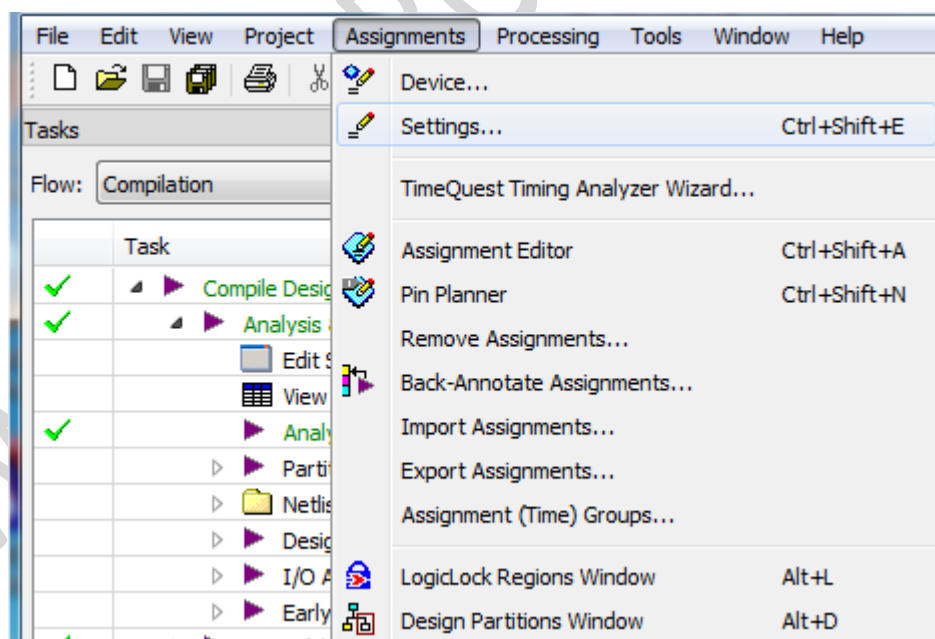
## اجرای برنامه simulate با Modelsim

ابتدا گیت and را در محیط bhd رسم مینماییم .



گزینه start compilation را میزنیم تا برنامه کامپایل شود , سپس فایل native link را طبق روند زیر عمل ایجاد میکنیم :

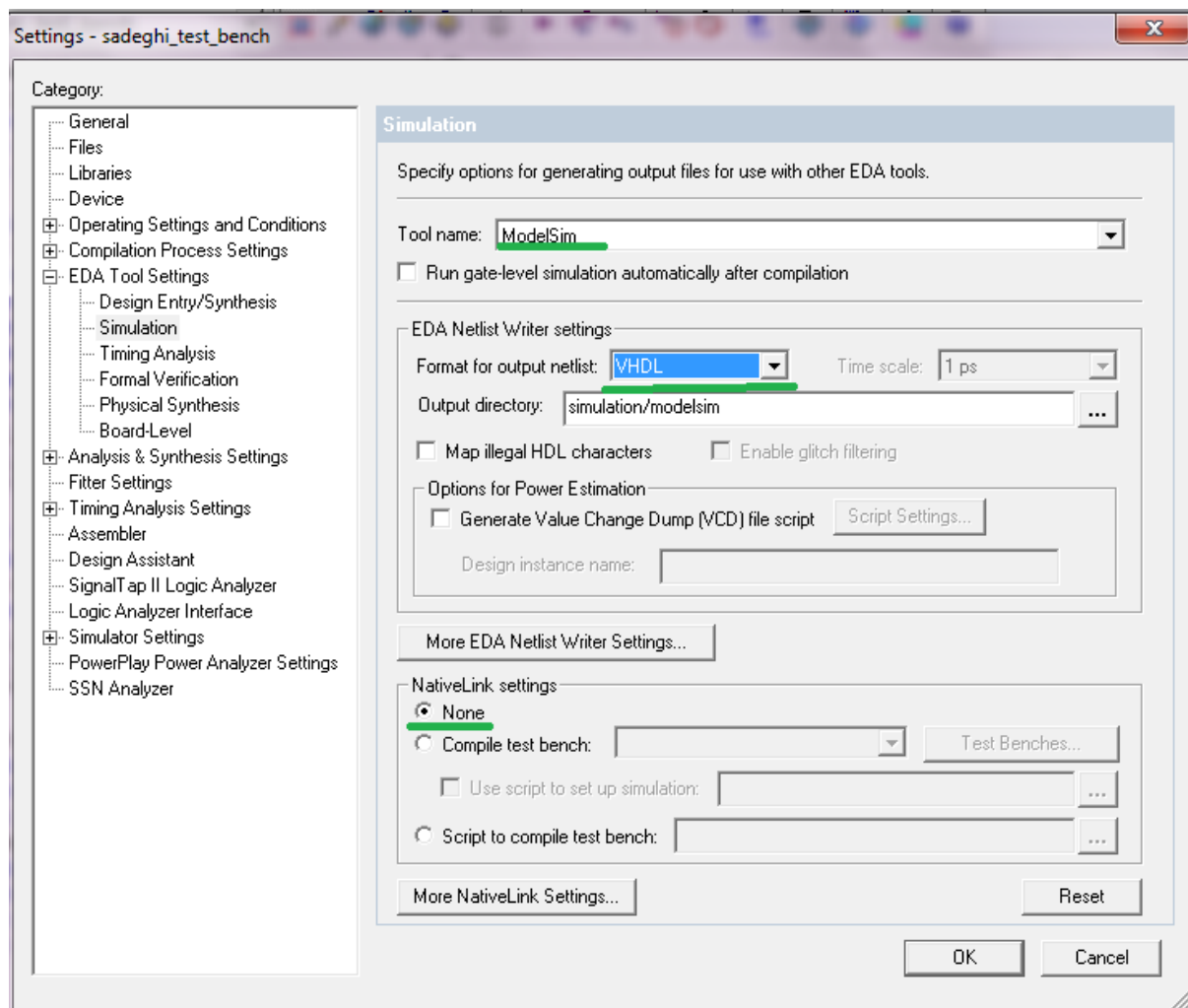
Assignments → settings → simulation



صفحه زیر باز میشود :

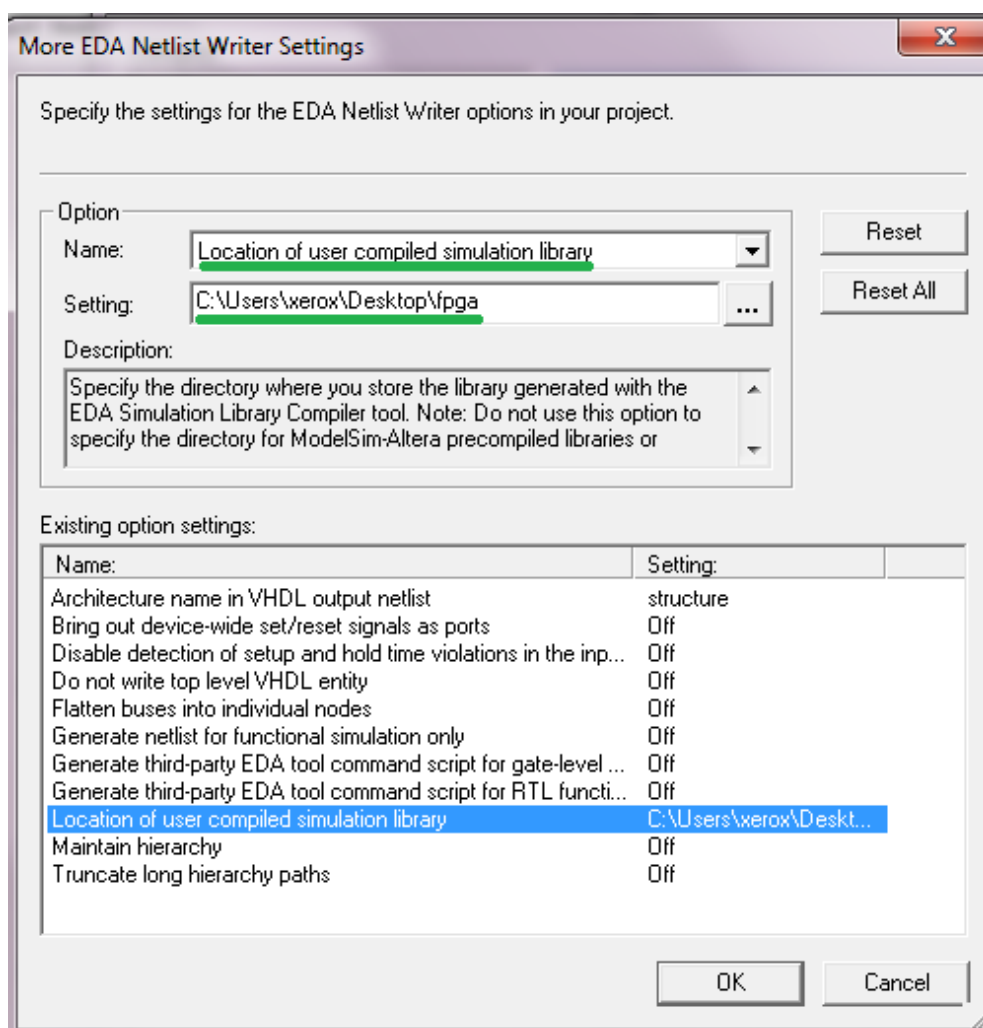
در این صفحه در قسمت tool name گزینه modelsim , در قسمت format for out put netlist گزینه vhdl و در قسمت nativelylink setting گزینه none را انتخاب میکنیم .





سپس روی گزینه more EDA netlist writer setting کلیک میکنیم .

صفحه زیر باز میشود :

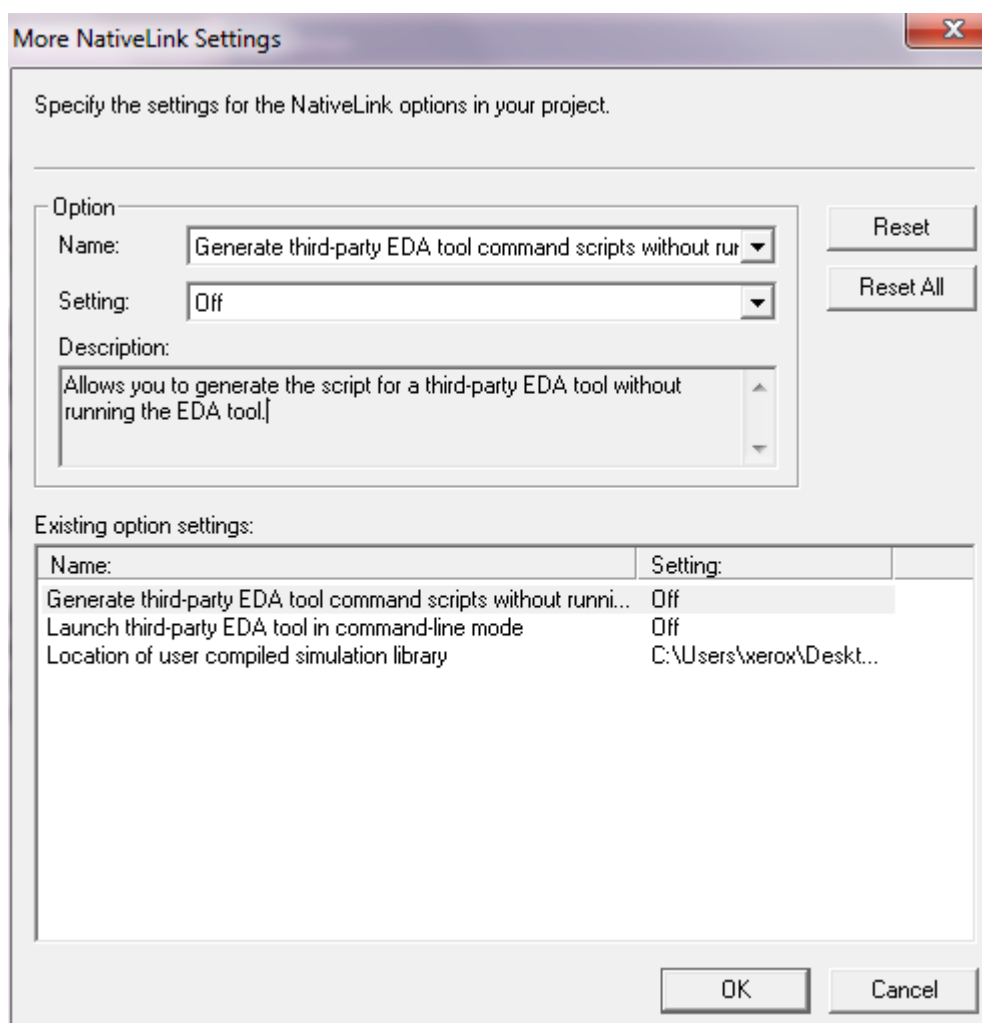


Name

و setting را مطابق شکل انتخاب کرده و در کادر پایین صفحه روی نام انتخابی در بالا کلیک میکنیم , سپس ok را انتخاب میکنیم . ( setting مسیر save فایل برنامه است . )

پس از بسته شدن این صفحه در صفحه قبل (sttting sadeghi\_test\_bench) روی گزینه more nativlink setting کلیک میکنیم .

صفحه زیر باز میشود:



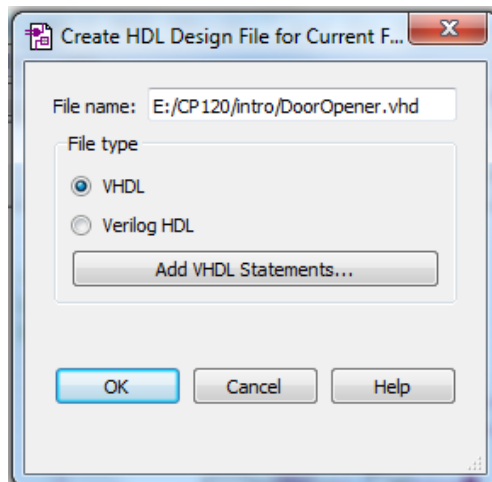
در این صفحه نیز name و setting را مطابق آنچه در شکل نشان داده شده است انتخاب میکنیم و در کادر پایین صفحه اولین گزینه را که شامل name و setting بالای صفحه است را انتخاب مینماییم، ok میکنیم، در صفحه قبل نیز ok میکنیم.

چون modelsim به زبان سخت افزاری HDL اختصاص دارد ما برای فایل VHDL خود باید یک فایل HDL بسازیم برای این منظور روند زیر را دنبال میکنیم:

File → creat /update → creat HDL design file for current file

VHDL → OK

صفحه زیر باز میشود :



حال open file را زده وبه همان آدرسی بروید که فایل بذنامه را ذخیره کرده اید مشاهده میکنید که در آنجا یک فایل جدید به نام فایل اصلی همراه پسوند vhd. ایجاد شده است. آن را انتخاب و open میکنیم.

سپس روند زیر را طی میکنیم :

Project → add current file to project

سپس

Project → add/remove files in project

صفحه setting دوباره باز میشود :

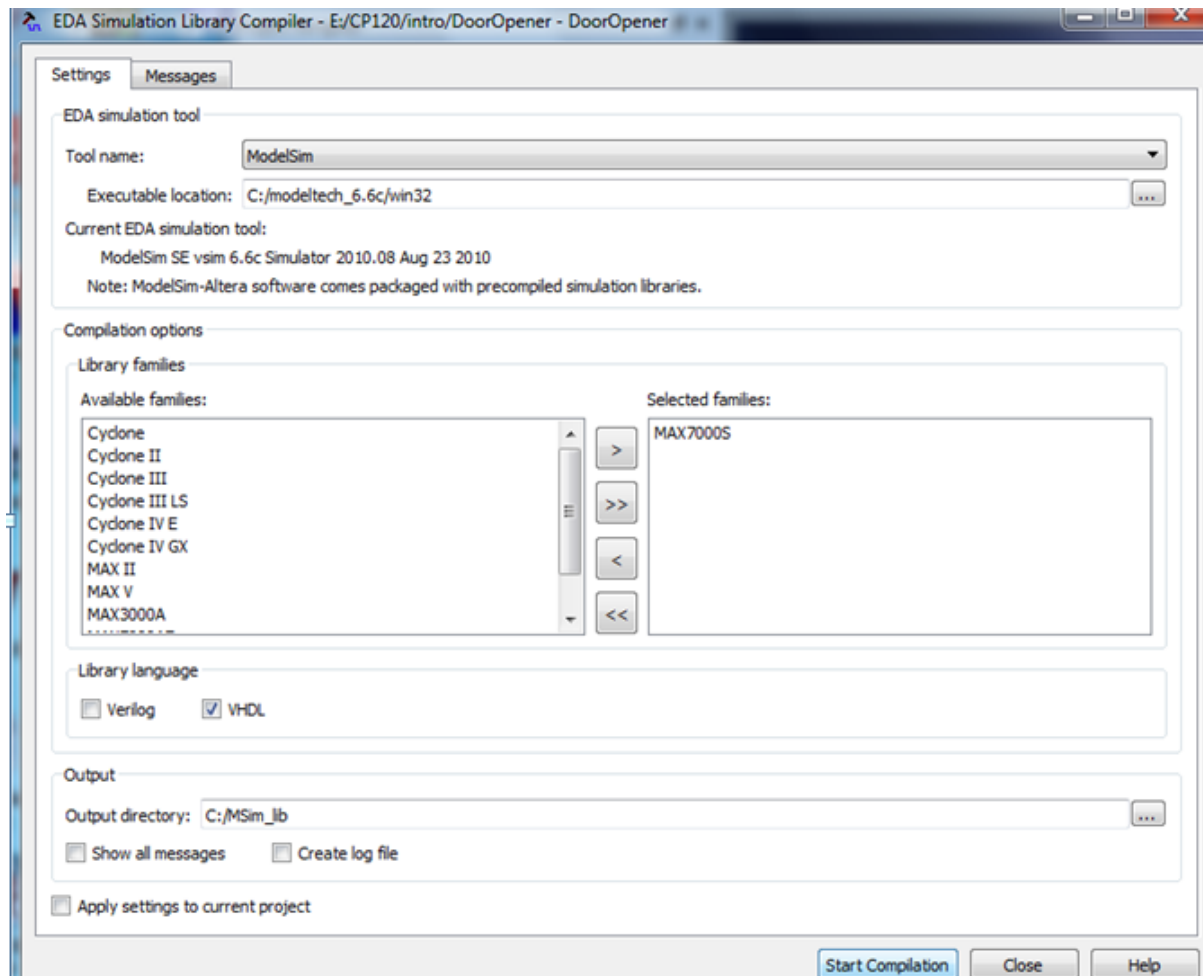
این بار روی گزینه file کلیک کرده و در منوی باز شده دو فایل وجود دارد یکی با پسوند vhd و دیگری با پسوند bdf. فایل با نام bdf را انتخاب میکنیم و ok میکنیم. حال روند زیر را طی میکنیم :

Processing → start → start simulation

سپس

Tools → launch EDA simulation library compiler

صفحه زیر باز میشود :



MUX7000S→VHDL→ start complition

این روند را در این صفحه اجرا میکنیم :

## ساخت فایل شبیه سازی به کمک Test bench :

در قسمت قبل یاد گرفتیم چگونه قسمت modelsim را در HDL بسازیم , در این بخش یاد میگیریم چگونه برنامه را بوسیله test bench در modelsim شبیه سازی کنیم . حال برای ساخت test bench این برنامه روند زیر را اجرا میکنیم :

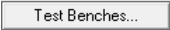
Processing → start → start test bench template writer

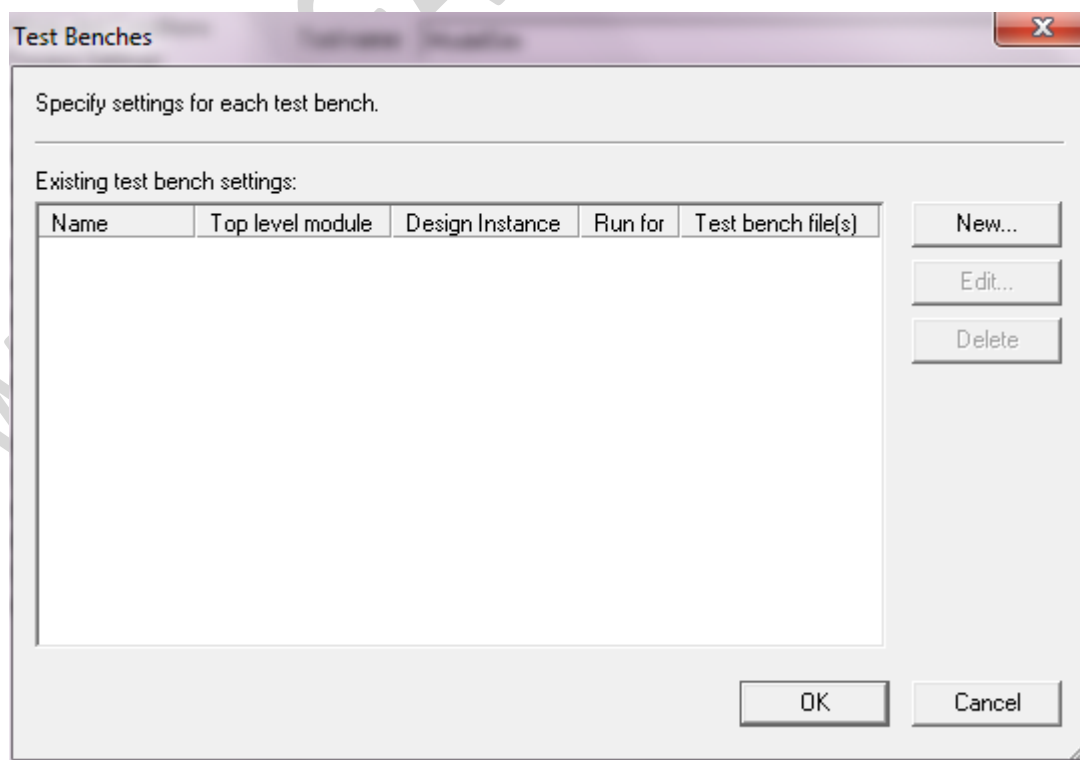
سپس فایل برنامه را در قسمت open file پیدا کرده و open میکنیم .

در اینجا باید برنامه را مطابق test bench تغییر دهیم .

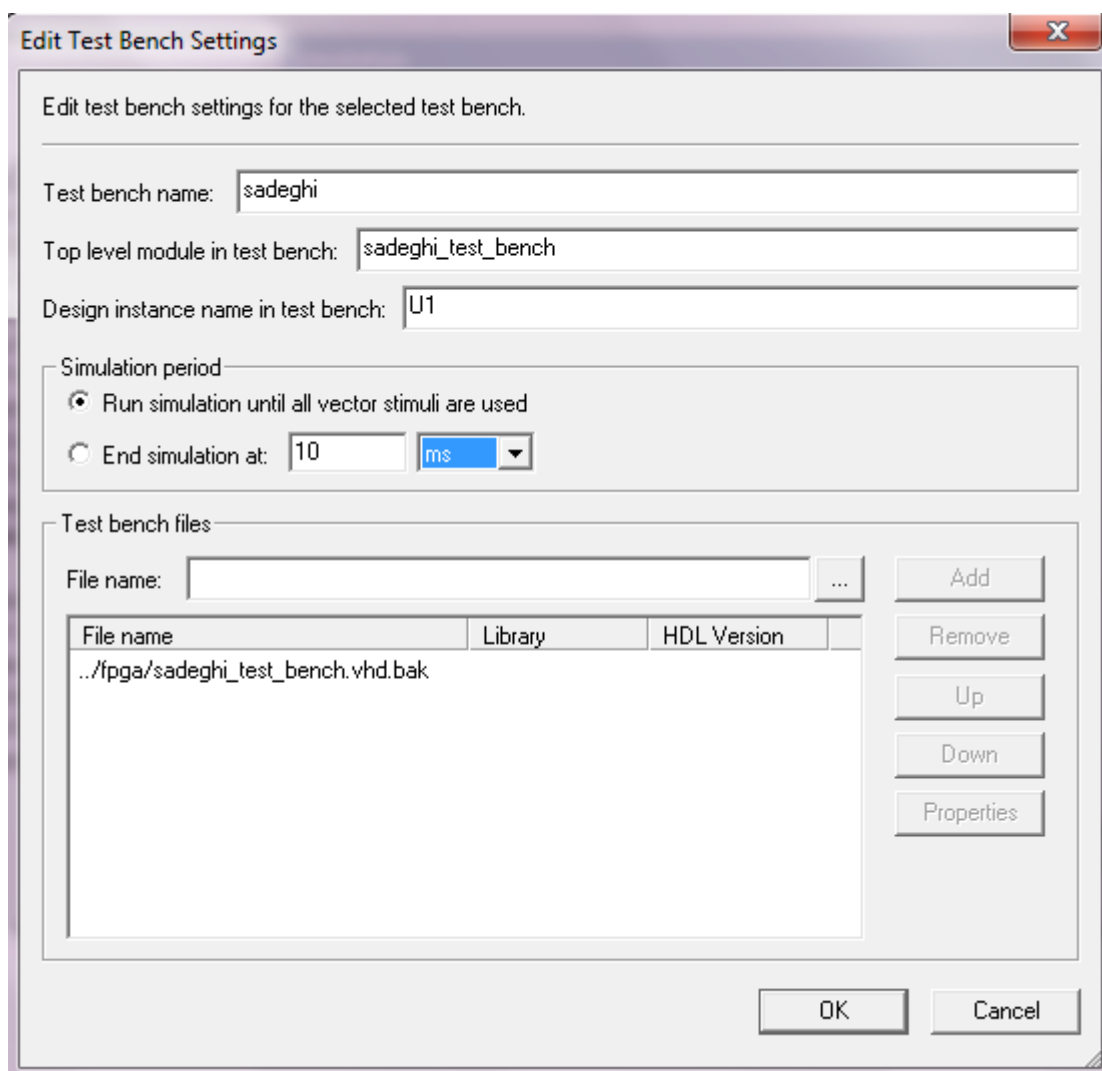
روند زیر را اجرا میکنیم :


Assignments → settings → simulation

در صفحه setting در قسمت nativlink گزینه compile test bench را انتخاب و روی آیکن  کلیک میکنیم , صفحه زیر باز میشود :



روی new کلیک میکنیم یک صفحه جدید باز میشود :



در این پنجره در قسمت Test bench file یک اسم دلخواه مینویسیم ، در قسمت top level اسمی که در برنامه برای entity تعریف کرده ایم را مینویسیم ، در design instance نام ، label ، port map را که U1 است را مینویسیم . در File name فایل برنامه را پیدا کرده و روی آیکون  کلیک میکنیم ، یک فایل به نامی که در بالا نوشتیم ایجاد میشود ، ok می نماییم . در دو پنجره test bench و setting ، ok میکنیم .

سپس روند زیر :

Project → add current file to project

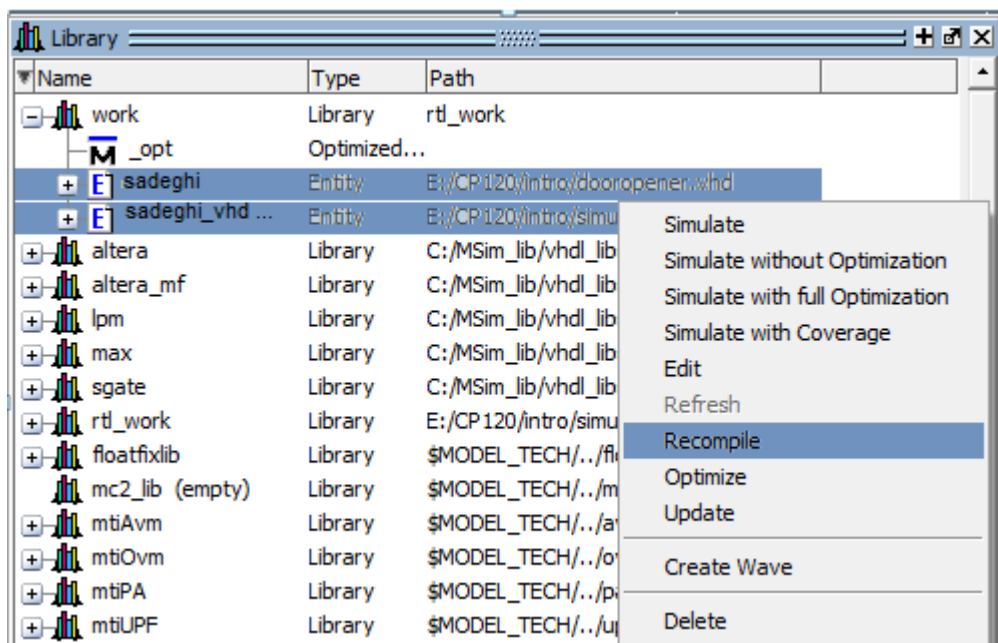
سپس

Processing → start → start analysis & elaboration

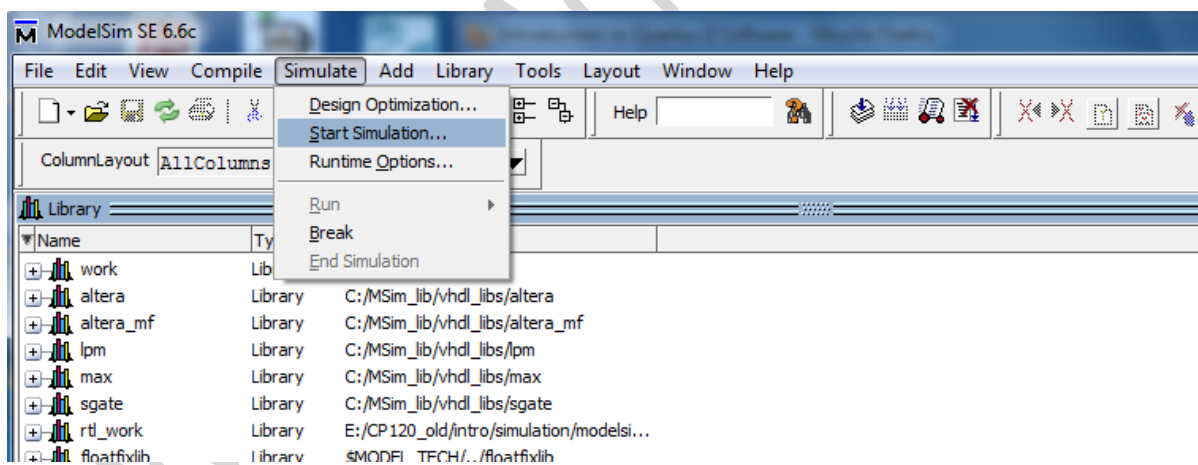
سپس

Tool → run EDA simulation tool → EDA RTL simulation

پنجره modelsim باز میشود در library دو فایل test bench وجود دارد هر دو فایل را با هم انتخاب کرده و کلیک راست مینماییم و گزینه recompile را میزنیم :

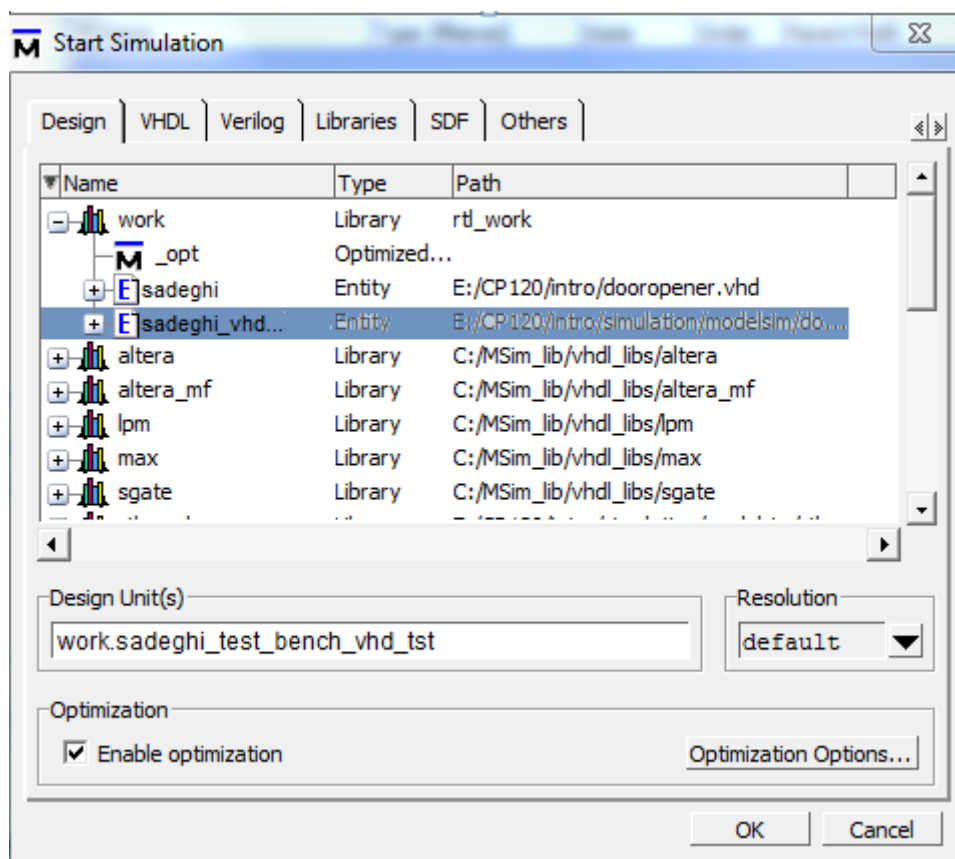


در نوار ابزار بالای پنجره modelsim گزینه start simulation → simulate را انتخاب میکنیم .

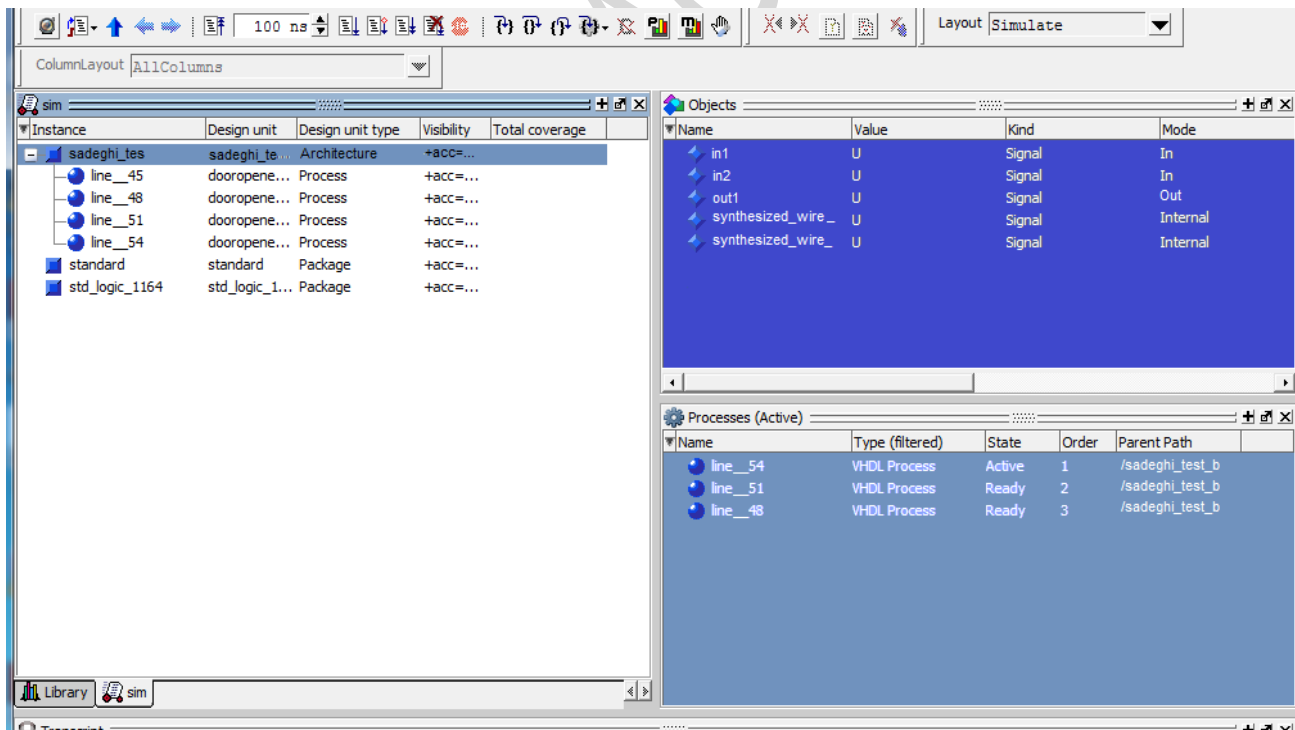


در پنجره start simulation دومین فایل را که شامل پسوند vhd است را انتخاب و ok میکنیم .

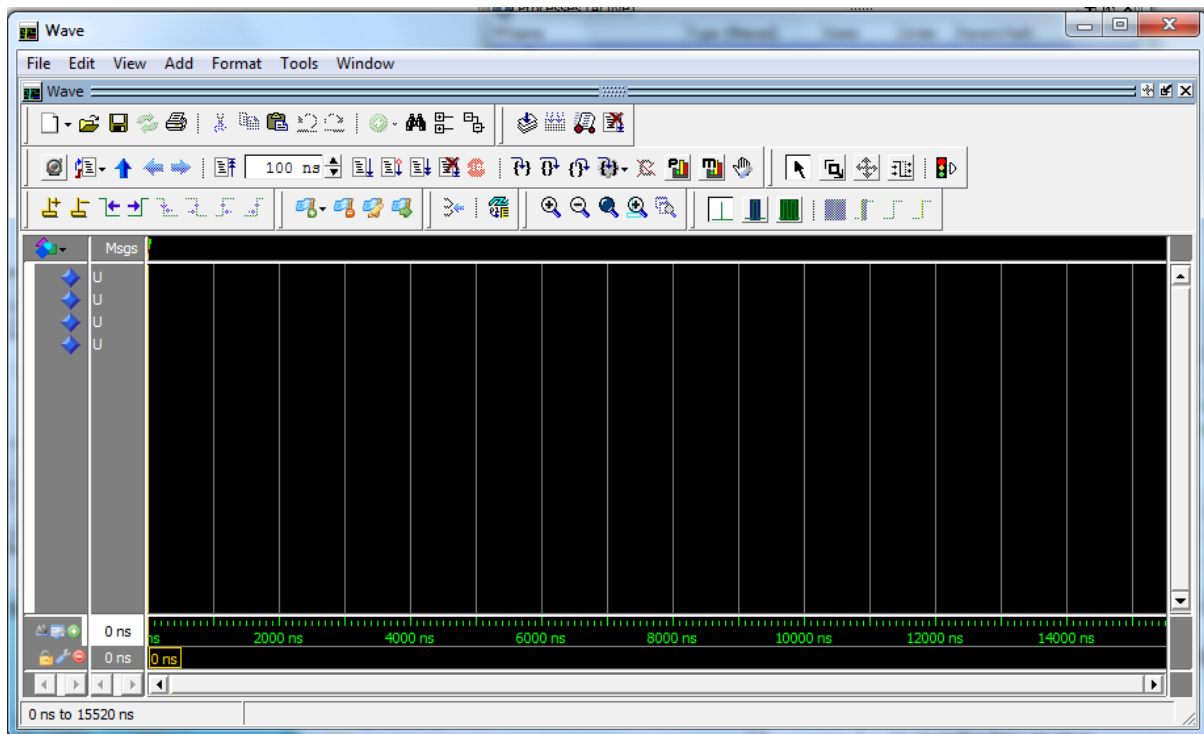




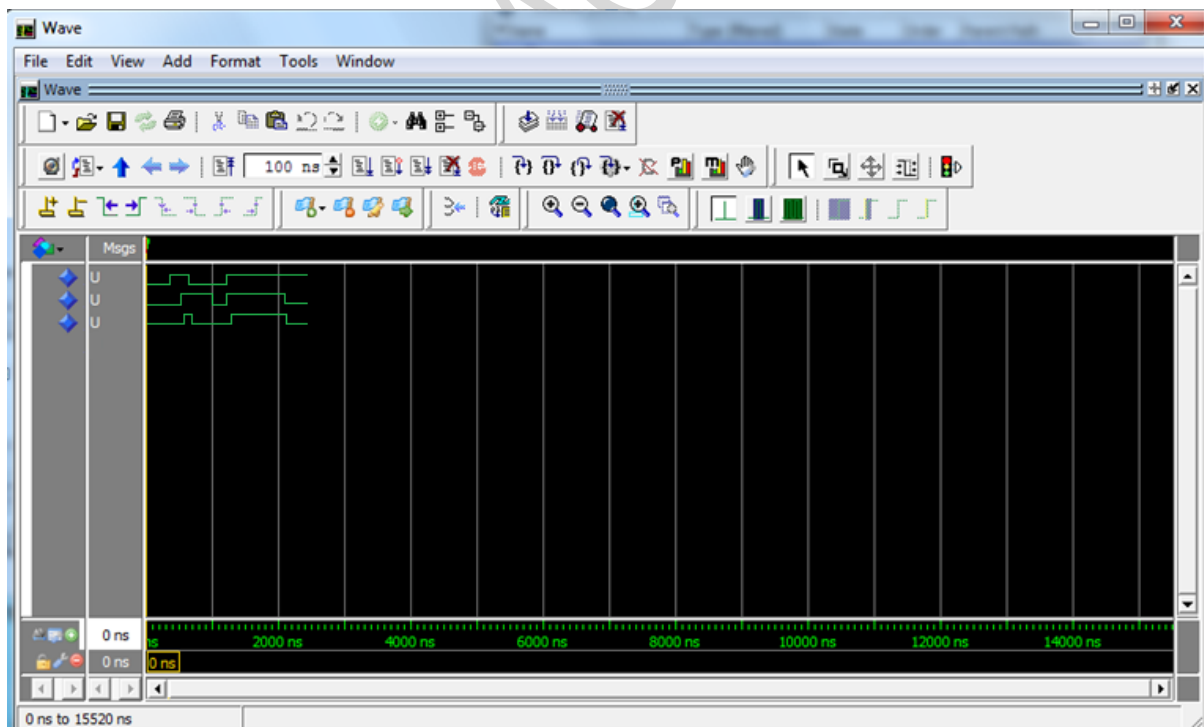
در پنجره modehsim سه قسمت ظاهر میشود که شامل لیست سیگنالها، لیست sim و process است :



روی لیست سیگنالها (object) کلیک راست کرده و Add→wave→selected signals را انتخاب میکنیم ، صفحه شبیه سازی test bench به نام wave باز میشود :



جهت اجرای شبیه سازی روی آیکون run simulation کلیک نمایید .



جهت بزرگ نمایی سیگنالها روی آیکون کلیک نمایید .

## منابع

فارسی :

مجموعه دستورات زبان توصیف سخت افزار سرعت بالا VHDL / مرتضی شعبان زاده .

English:

Quartus II Setup / Jan keneth bekkeng , UI.

VHDL ( programming by example ) / Douglas L.Perry

Vushiraz.blogfa.com