

# Домаћи задатак 1

Потребно је реализовати дигитални систем састављен од комбинационих компоненти по следећем упутству:

## 1. БЛОК-ДИЈАГРАМ ШЕМА: :

- На основу спецификације која је дата на крају упутства, нацртати блок-дијаграм шему система по узорку на слику сложеног система 4 из Лабораторијске вежбе 3.
- Шему је могуће цртати ручно па фотографисати или нацртати у неком софтверу.
- Сliku шеме сачувати под именом `block_diagram.jpg` у директоријум `LPRS1_Homework1_AA_008_2000_Solution`.

## 2. У датотеци

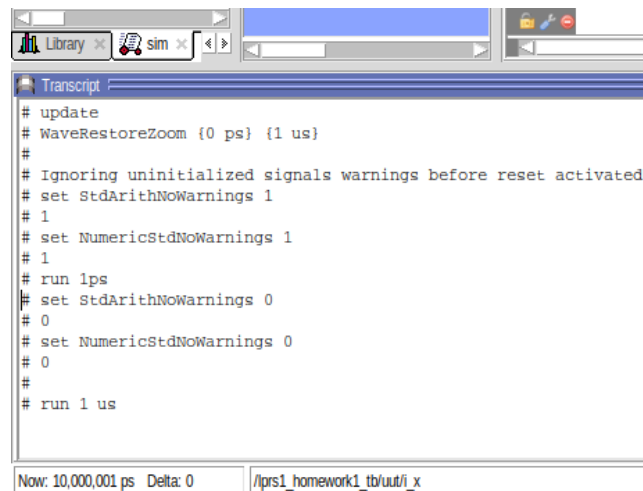
`LPRS1_Homework1_AA_008_2000_Solution/lprs1_homework1.vhd`  
реализовати ову блок шему у VHDL-у по задатој спецификацији.

## 3. НАПОМЕНЕ ЗА ПИСАЊЕ КОДА:

- Изнад дела кода који описује сваку појединачну компоненту у коментару написати назив описане компоненте.
- Сви интерни сигнали су ширине 4 бита.

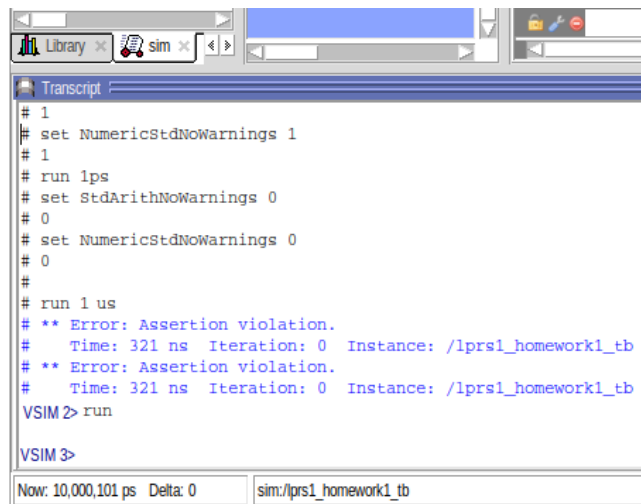
## 4. ПРОВЕРА ИСПРАВНОСТИ СИСТЕМА:

- Проверити исправност решења коришћењем симулатора ModelSim-Altera.
- У пројекту је дат тестбенч који аутоматски проверава исправност решења. Потребно је само покренути симулацију. Ако је дизајн исправан, Transcript панел у ModelSim-у ће бити без грешака, као на Слици 1:



Слика 1: Симулација без грешака

- Уколико дизајн није исправан, у Transcript панелу појавиће се грешке типа: `Error: Assertion violation` као што је приказано на Слици 2:



Слика 2: Симулација са грешакама

- Тестбенч није потребно мењати, нити ће исти бити прегледан. Он олакшава проверу и прегледање задатка.
- Такође, дозвољено је мењање тестбенча, ради дебаговања и вежбања.

## 5. НАПОМЕНЕ ЗА ПРЕДАЈУ РЕШЕЊА:

- Зиповати директоријум `LPRS1_Homework1_AA_008_2000_Solution` у зип датотеку `LPRS1_Homework1_AA_008_2000_Solution.zip`.
  - Не треба зиповати датотеке из `textttLPRS1_Homework1_AA_008_2000_Solution`, већ искључиво директоријум са датотекама.
  - Не треба зиповати директоријум целог пројеката, него само `LPRS1_Homework1_AA_008_2000_Solution`.
  - Уколико се пошаље нешто друго (нпр. цео пројекат), рад неће бити прегледан и резултоваће са оценом **D** односно **0** поена.
- Решење (.зип датотеку) поставити на **СОВУ** у делу **ЗАДАТАК 1**, у своју додељену групу.

# Спецификација

Потребно је реализовати следећи систем:

1. На сигнал `s_shl` довести `i_x` померен 2 бит(а) у лево логички.
2. На сигнал `s_shr` довести `i_y` померен 2 бит(а) у десно аритметички.
3. На сигналу `s_dec` поставити бит са редним бројем `i_z` на јединицу а остале бите на логичку нулу.
4. Сигналу `s_add` доделити збир `s_shl` и `s_shr` сигнала.
5. Од `s_dec` одузети `i_x` и разлику доделити `s_sub` сигналу.
6. На сигнал `s_const0` доделити 4.
7. На сигнал `s_const1` доделити 9.
8. На сигнал `s_mux` доделити:
  - `s_const0` ако је `i_sel` једнако 0
  - `s_sub` ако је `i_sel` једнако 1
  - `s_const1` ако је `i_sel` једнако 2
  - `s_add` ако је `i_sel` једнако 3
9. Сигналу `o_res` доделити сигнал `s_mux`.
10. На сигнал `o_cmp(0)` довести логичку јединицу ако је `s_mux` једнак 0.
11. На сигнал `o_cmp(1)` довести логичку јединицу ако је `s_mux` мањи од 6.
12. На сигнал `o_enc` довести индекс бита на логичкој јединици сигнала `s_mux`. Ако постоји више таквих бита, изабрати онај са најмањим индексом. Ако ни један бит није на логичкој јединици, резултат нека буде 3.

<b>ВАЖНА НАПОМЕНА:</b> Сви сигнали морају бити именовани како је тражено у спецификацији. Није дозвољено мењање назива сигнала! У супротном, задатак неће бити оцењен!
--