Cairo University **Seniors / 2023**
Faculty of Engineering
Electronics and Communication Department

Group Project

Project Title:

EDA Tool Design
Grade: 20 Grades
Due Date: 2 May 2022

Project Description:

Groups will be assigned one of the following EDA tools to build SW product fulfil the requirement as listed below.

Project#1 Verilog Lint

In this project the students will design static Verilog Design Checker, the checker will take Verilog (DUT) no Test Bench, and statically points to following List of Violations

- Arithmetic Overflow
- Unreachable Blocks
- Unreachable FSM State
- Un-initialized Register
- Multi-Driven Bus/Register
- Non Full/Parallel Case
- Infer Latch

Requirements

- Select a name for your Verilog Linter

• You should be able to parse Verilog code that can have different combinations of above constructs
• Verilog should be single Flatten Modules, No Hierarchy, No Instantiation
• Your Design should be only "Design Under Test" DUT no testbench is needed.

Your Lint EDA Product should have

o Parser
o Static Checker Engine
o Report Generator which produce a text file demonstrate
  ▪ Checks Name
  ▪ Line of Code it Happens in
  ▪ Signals/Variables Impacted

Evaluation

• Student will demonstrate 3 Verilog Cases, demonstrates combination of the above HDL Constructs, and demonstrate Lint results, reports.
• Their results will be compared with commercial linter as validation of results
• Presentation demonstrates the design of 3 main blocks (parser, engine, debugger)
• Teams will schedule a 20 minutes demo/slides using MS to deliver the project
• Using whatever programming language you prefer or scripting language (Python, Perl, TCL,..)
• You need to upload your executable, source code, document /slides describe the algorithm/approach used.

Project#2 Automatic Test Bench Generator

In this project the students will design Automatic Test Bench generator. The input will be Verilog DUT, the results will be Testbench that initialize variables using initial block, generate clock, instantiate the input DUT into the Testbench, and use always/initial with appropriate time delays to create test stimulus. You can use mix of directed tests and $random Verilog function, you should also add display and monitor in you generated. Verilog TB, to allow running this automatically created TB through simulation and validate the results.

Requirements

• Select a name for your Automatic TB generator
• You should be able to parse Verilog code of the DUT to extract Input Ports, Variables, If Condition Expression, Case Conditions, ..
  o Verilog Module
  o Input, Output Ports
  o Registers, Wires
  o Continuous Assignments
  o Always Block (Combinational, Clock Based)
  o If , Case
  o Non-Blocking Assignment
  o Logical Operator
• Your generator should be constructed from
  o Verilog Parser, that also extract control flow of your design (if/else, branches, cases,)
  o Stimulus Generator (Sequencer, Driver)
  o TB Writer: TB Module, Initial Block for Initialization, Clock Generation, Instantiation, Tests Generator, Monitor.

Evaluation

o Student will demonstrate 3 Verilog Cases, demonstrates creation of automatic 3 TB's for them
o Your Resulted TB should be simulated with original DUT using QuestaSim Results, and demonstrate coverage and waveform results
o Presentation demonstrates the design of 3 main blocks (parser, engine, TB generator) algorithms/approaches
o Teams will schedule a 20 minute demo/slides using MS to deliver the project
o Using whatever programming language you prefer or scripting language (Python, Perl, TCL,..)
o You need to upload your executable, source code, document /slides describe the algorithm/approach used.

Project#3 ORBDD based Equivalent Checker

The students here will design EC based on ORBDD, the input will be 2 Boolean functions and results will be ORBDD with Equivalence or Non-Equivalence Results Requirements

• Select a name for your RODBB Equivalence Checker
• Your work should include
    o Ability to enter 2 Boolean Functions to your Engine
    o Boolean Function Parser
    o ROBDD Constructor
    o ROBDD Comparator to result (Proof or Disproof)
    o Graphical Creator for results 2 Functions ROBDD and Formal Results about equality/inequality

Evaluation

o Student will demonstrate 3 Boolean Functions (Couples) and graphical presentation to their ROBDD and Prove / Dis-Prove of their equivalence
o Presentation demonstrates the design of 3 main blocks (Boolean Function parser, ROBDD Generator/optimization and graphical display) algorithms/approaches
o Teams will schedule a 20 minutes demo/slides using MS to deliver the project
o Using whatever programming language you prefer or scripting language (Python, Perl, TCL,..)
o You need to upload your executable, source code, document /slides describe the algorithm/approach used.