



Out: September 13, 2022

Due: September 19, 2022

The goal of this practical session is to become familiar with Mentor's Questa SIM which will be used in CME 435 for constructing and testing our testbenches. Questa is an integrated environment for design and verification. In addition to compilation and simulation, Questa also provides additional features for advanced functional verification.

All the CME 435 practical sessions are to be completed on Linux platform. As such, you will first set up your Linux environment for accessing Questa. The set-up procedure will be provided in detail by the instructor during the first practical session.

Part I: Getting Familiar with Questa SIM Commands

Similar to ModelSim, Questa SIM compiles and simulates SystemVerilog codes. In this section, you will learn how to carry out the basic functions using Questa SIM Commands.

1. Prepare the folder for this pre-lab exercise.
 - 1.1. Download cme435_ex1.zip from Canvas.
 - 1.2. Unzip the file. You should cme435_ex1 folder. Under this folder, there are script, dut, and verification folders. Do not make any change to the folders. The same file structure and naming conventions must be used for subsequent lab exercises and the class project.
 - 1.3. The script folder is a placeholder for .f, .do, and .csh files.
 - 1.4. Browse dut folder and verification folder. All files are encrypted. The three files in dut folder are related to the design to be verified, where ex1_dut.svp is the top-level design file. There is one single testbench file stored in verification folder.
2. Create a working library:
 - 2.1. Open a terminal.
 - 2.2. Change directory to cme435_ex1 folder.
 - 2.3. Create a library and map the work library to the newly created library.

```
vlib work  
vmap work work
```

NOTE: Questa SIM creates a directory called work and writes a specially-formatted file named _info into that directory. The _info file must remain in the directory to distinguish it as a Questa SIM library. **Do not edit the folder contents from your operating system**; all changes should be made from within Questa SIM. Questa SIM also adds the library to the Library window and records the library mapping for future reference in the Questa SIM initialization file (modelsim.ini).

3. Compile Verilog files.
 - 3.1. Make sure you are in cme435_ex1 folder.
 - 3.2. Compile all three files in the dut folder using vlog command. For example,

```
vlog dut/ex1_slave.svp
```
 - 3.3. Compile the testbench file in the verification folder.

```
vlog verification/tbench_top.svp
```



4. Run simulation in command-line mode.
 - 4.1. Start QuestaSim in command-line mode with `-c` option, where `tbench_top` is the testbench module name.

```
vsim -c tbench_top
```
 - 4.2. Run simulation.

```
VSIM> run -all
```
5. Exit from QuestaSim.

```
VSIM> exit
```

Part II: Automating Simulation Using C Shell Script

In this part, you will create a simple C shell script to automate the compilation and simulation process. **All your submitted and completed assignments for CME 435 must be able to run from Linux command prompt.** To this end, GUI and interactive mode, although supported by Questa, may only be used as the last resort during debugging.

1. In the script folder, there is an `ex1.csh` C shell template.
 - 1.1. Open the file with any preferred text editor.
 - 1.2. Examine the file and make sure you understand the purpose of each line.
 - 1.3. Add all the `vlog` commands that were used in the previous part. For example,

```
vlog dut/ex1_dut.svp
vlog verification/tbench_top.svp
```
- 1.4. Add `vsim` commands into the script.

```
vsim tbench_top <<eof
run -all
exit
eof
```
- 1.5. In `cme435_ex1` folder, run and test the script. Make sure everything works as expected.

```
script/ex1.csh
```
2. QuestaSim allows all SystemVerilog files to be compiled in orderly manner with one `vlog` command. Although this can be done at the command prompt, the `.f` option is more commonly used.
 - 2.1. Create a script/`ex1.f` file.
 - 2.2. List all `svp` files in `ex1.f` including the testbench `svp` file. For example,

```
dut/ex1_dut.svp
dut/ex1_master.svp
dut/ex1_slave.svp
verification/tbench_top.svp
```
 - 2.3. Open `script/ex1.csh`, replace `vlog` commands with only one `vlog` to compile all `svp` files with the `-f` option.

```
vlog -f script/ex1.f
```
 - 2.4. In `cme435_ex1` folder, run and test the script. Make sure everything works as expected.
3. Variables can be used in a C shell script.
 - 3.1. Define a variable for the testbench module.

```
set tbench_top = tbench_top
```



- 3.2. Replace tbench_top related to vsim simulation with \$tbench_top. That is,

```
vsim $tbench_top <<eof
```

- 3.3. Run and test the script.

Part III: Automating Simulation Using Tcl Script

Tcl is the de facto standard command language for Electronic Design Automation (EDA) applications. You can automate repetitive behavior, extend the functionality, and control multiple tools with a single script. In Part III of this exercise, you will learn how to use Tcl scripts together with a shell script.

1. Create a simple Tcl script for automating QuestaSim simulation.

- 1.1. Create a script/ex1.do file.

- 1.2. In script/ex1.do Tcl file, add two vsim simulation commands that were used before.

```
run -all  
exit
```

2. Modify the C shell script by replacing vsim with the following line.

```
vsim -c $tbench_top -do script/ex1.do
```

3. Run the C shell script from the command line. Make sure everything works as expected.

4. Create a Tcl script that can compile all svp files and run simulation in batch mode. Below is a simple Tcl example.

Filename: script/ex1_all.do

```
onbreak {resume}

set tbench_top tbench_top

vlog top.v           # compile all source files
vsim $tbench_top      # load simulator with optimized design
run -all              # run simulation
quit -f
```

- 4.1. Open script/ex1_all.do Tcl file with any preferred text editor.

- 4.2. Examine the file and make sure you understand the purpose of each line.

- 4.3. Modify the Tcl file to compile all svp files and run simulation in batch mode. This means that script/ex1.csh should contains one one vsim command that will compile all svp files and run simulations.

```
vsim -c -do script/ex1_all.do
```

- 4.4. Run the script from the command line. Make sure everything works as expected.

Deliverable

You are not required to submit this tutorial exercise. However, it will be presumed that you know how to run Questa Sim in batch mode by using shell and Tcl script programming in the subsequent lab exercises.