⚠ You have previously submitted this assignment with David Rama Jimeno. Group can only
   change between different assignments.

This course has already ended.

Assignment description
(/comp.ce.240/spring-
2024/E12/tb_i2c_config/)

My submissions (10/1000) ▾

Exercise 12: I2C controller testbench

Note that parameter for dai_clk_mode is now given beforehand in E11 guide! This needs to
be taken care of in the test bench.

Hint: At least a basic mutation test (http://en.wikipedia.org/wiki/Mutation_testing) is always
recommended to be done with a test bench. So, when the design under test works even partially,
break it purposely and check if the test bench notices it. For example, comment out some
assignment statement, negate an if-statement's condition, change a compared value plus-minus 1,
etc. Make corrections to the test bench so that it notices the errors.

Essential requirements for the test bench:

- Sometimes returns NACK instead of ACK
- Verifies the correctness of device address and read/write bit
- Verifies that configuration values are received correctly by the test bench.
- Verifies that the register address and data values are correct as specified in the previous
  exercise.
- Contains at least 5 non-trivial asserts (2 given)
- Code coverage at least 90% of statements and branches, and 100% of states. The code
  coverage of the test bench itself is not important.

Download a template for the test bench (https://plus.tuni.fi/graderA/static/compce240-
f2021/E12/tb_i2c_config.vhd) and edit it. It already supports the checking of ACK.

Code coverage

The code coverage analysis tells among other things which statements, branches, conditions and states of a state machine are went through during the run of a test bench. For the analysis, the source code has to be compiled either with -cover or -coverAll parameter.
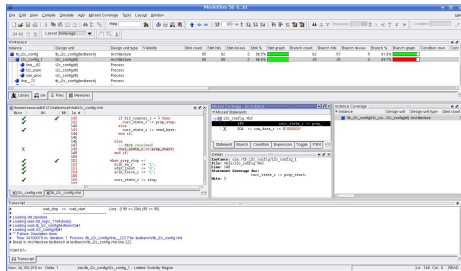
In this exercise use:

```
> vcom -check_synthesis -cover sbf files.vhd
```

After compiling the code, the simulator has to be started with the -coverage parameter.

```
> vsim -coverage work.tb_i2c_config
```

When simulating with the -coverage parameter, Modelsim shows statistics like in the example below:

(https://plus.tuni.fi/graderA/static/compce240-f2021/E12/modelsim_coverage.jpg)

See ModelSim User Manual (http://www.microsemi.com/document-portal/doc_view/131619-modelsim-user) for additional information.

Some of the measured things are hard to cover completely, like the state transitions of state machines, which requires testing the reset in all possible states in order to test all of the state transitions.

Return:

- Put your **own** VHDL-code under `E12` folder in your Git repository
- Return files:
  - `i2c_config.vhd`
  - `tb_i2c_config.vhd`
- Check that the files' header comments are valid, made according to instructions, and you have followed the coding rules
- Push the changes to your repository and submit (with your partner if you have a group!)
  - Use the **ssh variant** of the repository url in the submission. Otherwise the tests will fail 100% even with working design.
  - The url looks like *git@course-gitlab.tuni.fi:compce240-spring2024/<your_group_number>.git*.

Enter your Git repository address for grading

Did you remember git add - git commit - git push?

| Submit with David Rama Jimeno   ⌄ | **Submit** |

Earned points

6 / 6

Exercise info

Assignment category
VHDL exercises

Your submissions
10 / 1000

Points required to pass
6

Deadline
Sunday, 21 April 2024, 23:59

Late submission deadline
Friday, 31 May 2024, 23:59

Group size
1-2

Total number of submitters
55