Flip-flop Reset Design

Objective:

The lab tries to show through simulation the difference between FFs with asynchronous and synchronous reset and also with or without a data-enable.

Introduction:

We have provided a project folder, where you can find a project file (*.mpf*), two Verilog source files (*.v*), two script files (*.do*), and this lab manual. You can invoke this project in Modelsim as follows. Make sure that the path of target *.mpf* file is correctly selected.

File → Open → ff_reset_verilog.mpf

The first source file (*ff_reset_verilog.v*) is the main design file, where different versions of Flipflop outs (represented as *Q_**) have been generated for different purposes. These *Q_** can be found in Table 1. The second source file (*ff_reset_verilog_tb.v*) is the testbench to test the design file. A typical testbench file usually generates clock signal, apply stimulus/input vectors, etc.

*Q_**	How they are generated?
Q_bad_r	BAD coding results in treating the RESET as a Data enable
Q_async_r	FF with asynchronous reset
Q_sync_r	FF with synchronous reset
Q_no_r	FF with no reset at all
Q_async_r_de	FF with asynchronous reset with Data Enable
Q_sync_r_de	FF with synchronous reset with Data Enable
Q_no_r_de	FF with Data Enable but no reset at all

Procedures:

1. Complete *TODOs* by setting a default *Q* value when reset in *ff_reset_verilog.v*, either using Notepad++ or Modelsim as editor.

2. In Modelsim, compile the design file and testbench file and make sure your design has no syntax errors. Errors will be displayed in the *transcript* window.

3. Run command: *do ff_reset_verilog.do* in the *transcript* window in Modelsim, which invokes another *.do* file. These two scripts simulate your design for 600ns and display the waveforms.

4. Answer questions in *ff_reset_verilog.v* and discuss with your TA.