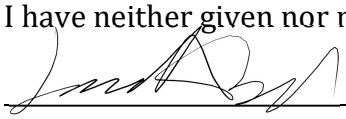
ECE 3544: Digital Design I
Project 1 (Part B): Simulation in the ModelSim Environment; Continuous Assignment Models

Student Name: Jacob Abel

Honor Code Pledge: I have neither given nor received unauthorized assistance on this assignment.



Grading: The design project will be graded on a 100 point basis, as shown below:

Manner of Presentation (30 points)

_____ Completed cover sheet included with report (5 points)

_____ Organization: Clear, concise presentation of content; Use of appropriate, well-organized sections (15 points)

_____ Mechanics: Spelling and grammar (10 points)

Technical Merit (70 points)

_____ General discussion: *Did you describe the objectives in your own words? Did you address the questions posed in the project specification? Did you discuss your conclusions and the lessons you learned from the assignment?* (10 points)

_____ Design discussion: *What was your approach to deriving the circuit you had to design? How does the design process that resulted in your continuous assignment model compare to the one you used to obtain a structural model that used primitive gates? How applicable is the design process you followed in this assignment to larger and more general designs?* (10 points)

_____ Testing discussion: *What was your approach to formulating your test benches, and how did you verify the correctness of your design and implementation?* (10 points)

_____ Supporting figures (20 points total)
- *Correct waveforms showing simulation of both decoder modules.* (5 points)
- *Correct waveforms showing correct operation of your design.* (15 points)

_____ Supporting files: *Do the submitted files produce the correct response?* (20 points)

_____ **Project Grade**

Project 1-B

ECE3544 CRN:82989

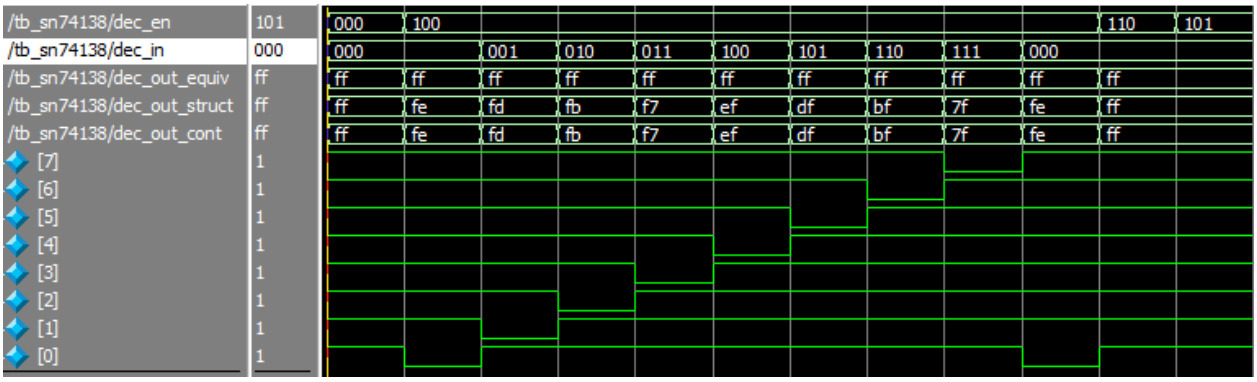
Jacob Abel

September 26, 2018

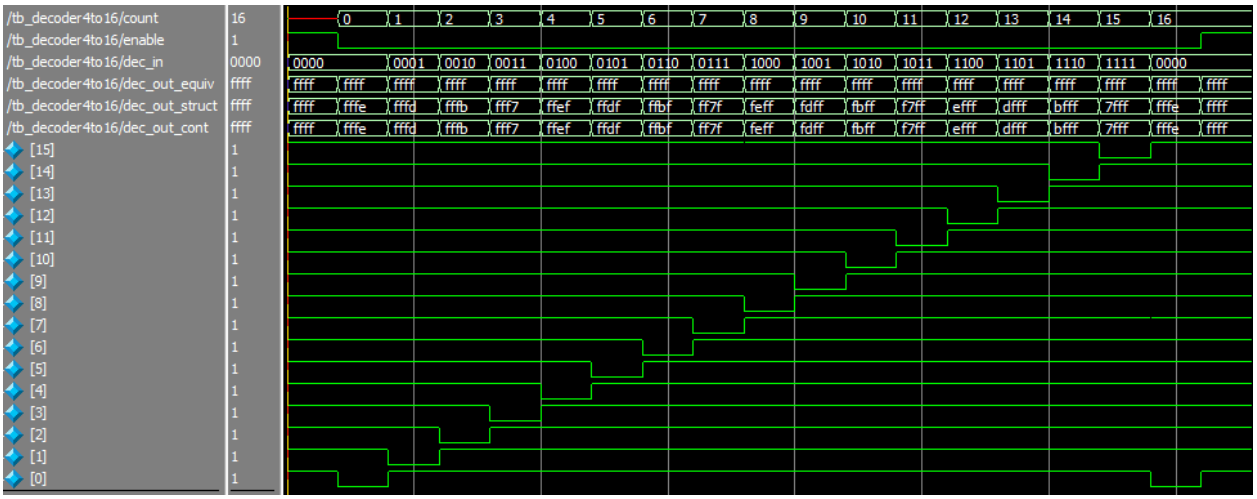
Objective

The objective of this project is to demonstrate basic competence with the Continuous Assignment Verilog Paradigm. Additionally this project demonstrates the functional equivalence between Continuous Assignment and Structural Verilog Paradigms.

Decoder Simulations



74138 3-to-8 Decoder Test-Bench Waveform



4-to-16 Decoder Test-Bench Waveform

The test-benches were adapted to XNOR the outputs of the structural and continuous assignment implementation results. As evidenced by the simulation waveforms, this Continuous Assignment implementation is functionally equivalent to the Structural Verilog implementation of Project 1-A.

Rock Paper Scissors Design

The following project description is identical to that of Project 1-A. This implementation of the project must be implemented in Continuous Assignment logic unlike the Structural Verilog of Project 1-A.

The project requires that a combinatorial circuit be designed that computes the results of a ”Rock, Paper, Scissors” game. The circuit accepts two inputs: `player_a[1:0]` and `player_b[1:0]` and outputs three 1-bit outputs: `player_a_wins`, `player_b_wins`, and `tie_game`. The standard rules of the game apply:

- Rock beats Scissors
- Scissors beat Paper
- Paper beats Rock

The values of the input moves are as follows.

Move	Rock	Paper	Scissors
Input Value	11	10	00

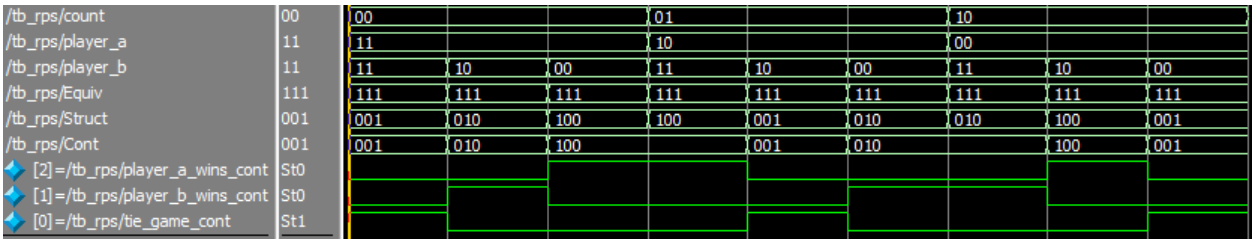
Design Process Reflection

The design process for this project was far better adapted to a mid to large scale design project than the process from Project 1-A. This process focused around adapting the test bench to easily demonstrate equivalence to an already proven functional design such as the previously implemented Structural Verilog ”Rock, Paper, Scissors” module. Once an equivalence test could be established it became trivial to validate functionality.

The strategy for the design of the actual implementation module was taking the implementation from Project 1-A and condensing it into Continuous Assignment logic. The fundamental structure of the module was roughly the same as in Project 1-A however all the logic was condensed into 5 assign statements. The first two established the hands of the players and the remaining assigns handle the actual game itself. At this point debugging was remarkably easy as any deviation showed up glaringly obvious in the equivalence waveform. Both through conversion from Structural Verilog and in the creation of new modules, Continuous Assignment is significantly more efficient both from a development, readability, and maintainability standpoint.

This design process can work well for large scale designs as it is more or less an adaptation of unit tests from software development. With some additional boilerplate, this design process can easily be extended from simply proving equivalence to serving as a proper unit test suite. Additionally it could be extended to sum all test results and output whether all tests passed.

Rock Paper Scissors Simulation



Rock Paper Scissors Simulation

The simulation demonstrates that the Continuous Assignment Implementation(`Cont`) matches the intended behaviour and is equivalent to the Structural Verilog Implementation(`Struct`). The test bench was designed to cover all possible inputs while minimising repetition. The test bench is the same as the test bench from Project 1-A with an XNOR comparison waveform (`Equiv`) added to demonstrate equivalence between the Structural Verilog and Continuous Assignment waveforms.

Conclusion

This project demonstrated that equivalence between varying module implementations can be easily proven. Additionally it demonstrated that structural Verilog can be easily converted to continuous assignment logic and in doing so greatly improves both readability and conciseness. This project overall can be considered a success. Unlike the prior project, sufficient time was allotted for completion and the decreased stress due to a less impending deadline resulted in both less errors and more accurate debugging.