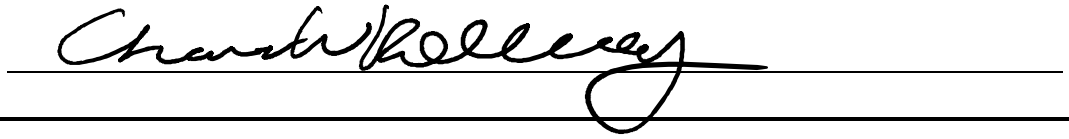
ECE 3544: Digital Design I
Project 1 (Part A): Introduction to the ModelSim Environment

Student Name: Charles Kelley

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 applicable is the design process you followed to more general design and larger-scale 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)
- *Waveforms showing correct operation of your design.* (15 points)

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

_____ **Project Grade**

Project 1A

Purpose and Objective

The purpose of Project 1A is an introduction to the processes relevant for Digital Design, ECE 3544, including an understanding of the basic design and simulation tools of ModelSim. This project specifies two parts: an introduction to ModelSim test benches and an introduction to designing and testing a circuit from specifications. Specifically, three modules will be simulated and one will be designed to meet the project specifications.

Project Specification Questions

Based on the structure of the 74138-decoder module, produce a neat gate-level schematic of this module and include it in your report.

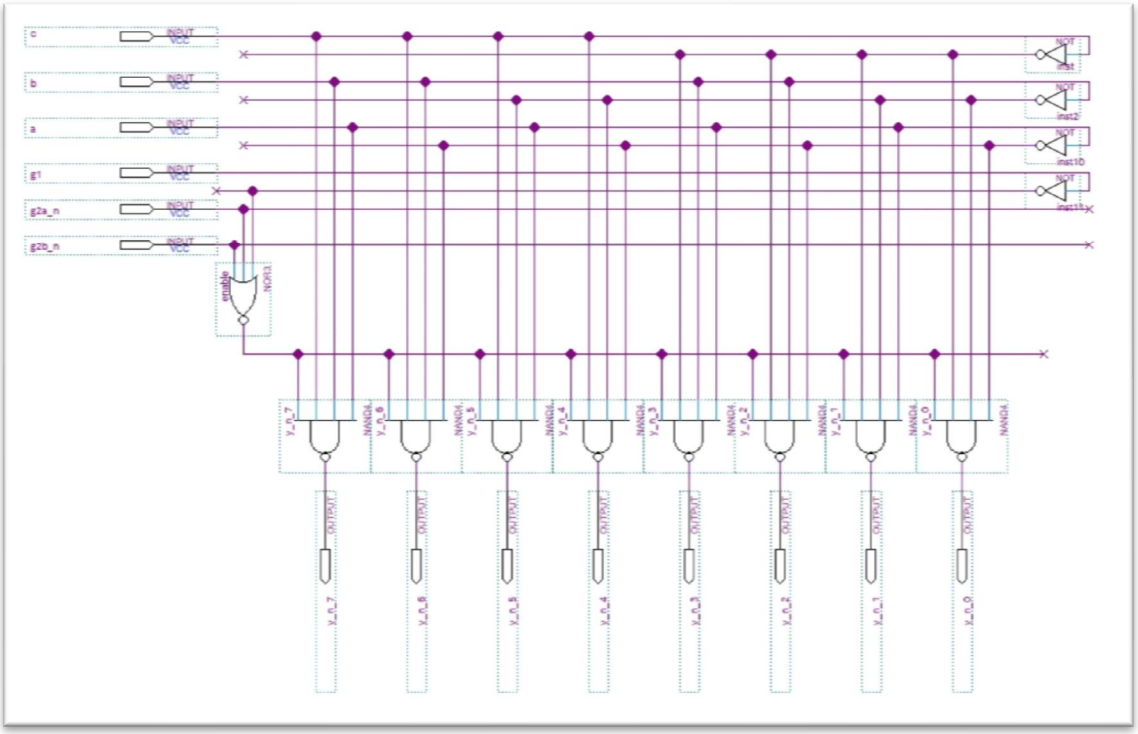


Figure 1. Gate-level Schematic of 74138-decoder

Based on the structure of the 4-to-16 decoder module, produce a neat block diagram of this module and include it in your report.

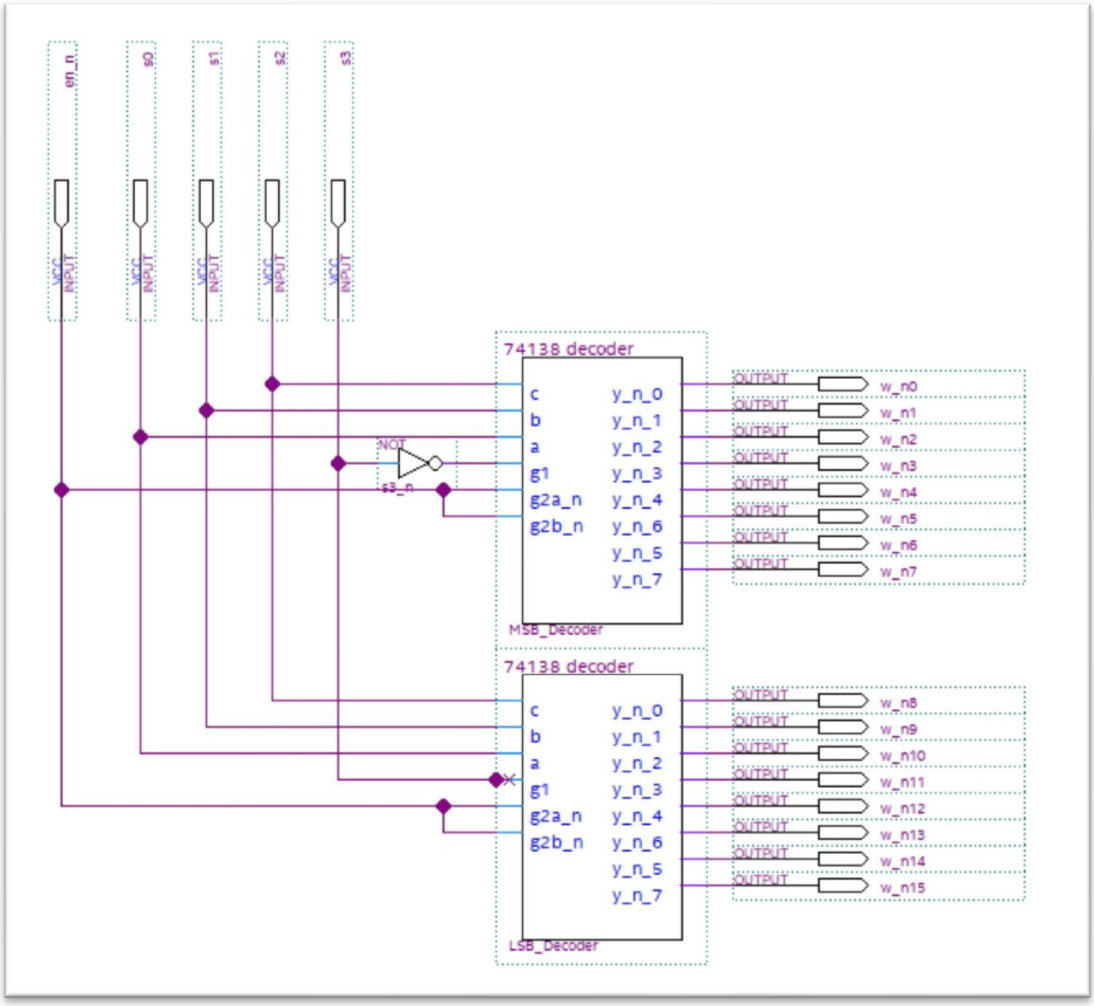


Figure 2. Block Diagram of the 4-to-16 Decoder

What qualities do the test benches have in common? (Focus on the fact that a test bench is a Verilog procedure.) In what significant ways do the test benches differ?

The two test benches provided in part 1 of this project have many similarities. From the start, both test benches start with a module declaration that includes neither inputs nor outputs. Within the module, both testbenches declare a single instance of the module they are testing. The inputs for the device under testing are declared as with the keyword “reg” because the testbench modifies these values. The initial blocks in the test benches are another similarity. Each bench uses an initial block with incremental timesteps to simulate several regs moving through different values.

Design Process

The following section describes the design process for the Rock, Paper, & Scissors Judge, RPS Judge, module implemented in Verilog code. The design process started wit an evaluation of the RPS Judge specifications. With three bits for five possible hands in the game, a design choice had to be made from the initial begin to how these options would be chosen. A system where bits were chosen in ascending order by three-bit grey code was used first. This ran into a

problem that simplification of the circuit could only be driven to a certain degree. The second and final method for choosing bits was to match bits based on which hand beat another. Both losing hands for an opponent's given hand shared a single bit in order to encourage simplification. On a smaller scale this allowed for a very custom design that could also accommodate Verilog code using sum of products. On a larger scale, the by-hand evaluation of the game outcomes would become unreasonable; however, a more general approach could be used to group inputs or outputs based on convenience and order could be formulated. This general method would design circuit outputs to cluster in a manner like how k-maps are grouped in grey code.

Once the bits were chosen for each hand of the RPS Judge, a sum of products was derived for the hands that resulted in player one winning. This sum of products was then transformed to use only nand gates. The next outcome that was determined was whether a tie occurred. This outcome occurs only when the two hands are identical bitwise; therefore, the three bits were compared, using xnor gates, and then checked if they were all similar. From this point, finding whether player two won was a simple as checking if neither player one had won nor there was a tie.

This solution to the RPS Judge provided a sensible design that was implemented using only primitive gates in Verilog. The module was then simulated in a test bench and verified. Included below in Table 1. are the bits corresponding to each hand.

Player's Hand	**Scissors**	**Paper**	**Rock**	**Lizard**	**Spock**	**Unused**
Input Value(s)	010	100	011	000	111	001, 101, 10

Table 1. Player hands and input values

Simulation Results – Decoders and Game

A test bench was used to simulate and test the Rock, Paper, & Scissors Judge module. The test bench was designed using a combination of regs and wire to simulate the two players' hands and the winning output respectively. The players' hands were implemented using regs because the test bench asserted values over 500ns. The smaller sample size of possible game variations allowed a reasonably size bench to test each game possibility. Included below, in Figure 3, is the waveform for the RPS Judge showing the 25 games that were simulated.

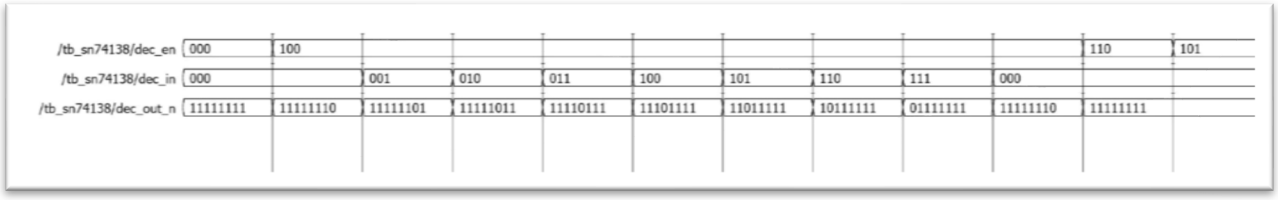


Figure 3. Simulation waveform of sn34138 decoder

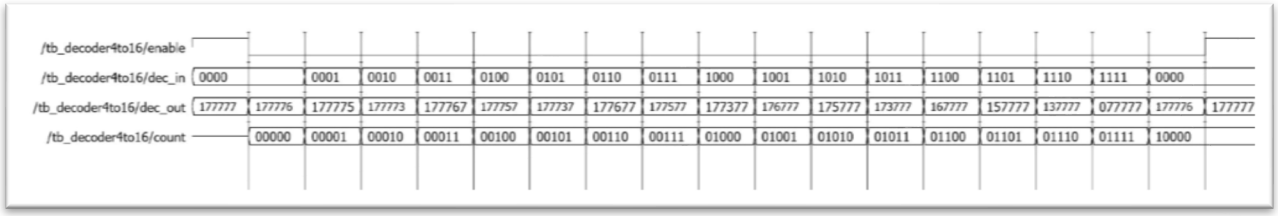


Figure 4. Simulation waveform of 4-to-16 decoder

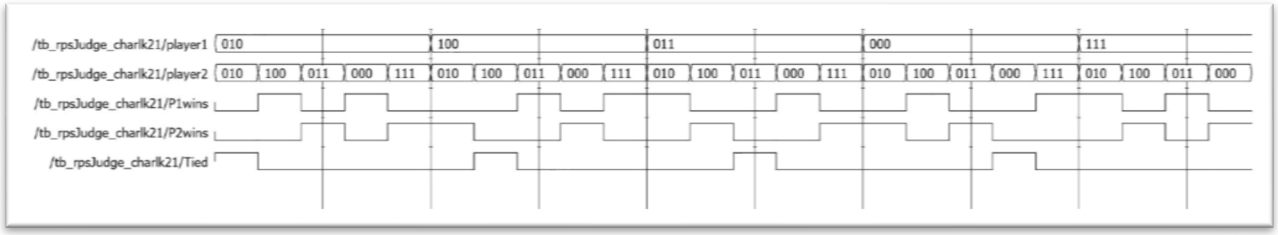


Figure 5. Simulation waveform of Rock, Paper, & Scissors Judge module

Conclusion

In conclusion, a module was designed to implement a Rock, Paper, and Scissors game in Verilog. The process of designing and testing the RPS Judge module highlighted the skills and functions necessary for Digital Design 1. An understanding of both Quartus and ModelSim as well as a review of Boolean algebra and primitive logic gates were achieved through the project. This project also developed problem solving skills where no singular solution can be achieved, but rather choices had to be made to shape a specific solution. While having both a functioning module to play a game of rock, paper, scissor, lizard, and Spock with a friend or enemy, this project provided an introduction into the digital design.