COE 203: Digital Logic Design Lab
Project Term-062

Vending Machine

Project Requirements

Design a simple ***vending machine*** on the Spartan-3 board. The goals of this project are to:

1. Design a digital circuit according to given specifications.
2. Gain experience with modern CAD techniques used to design digital systems.
3. Learn how to get information from professionally written manuals (e.g., 7-Segment)
4. Learn to test projects with simulation.
5. Learn to download projects to an FPGA board and perform physical testing.
6. Learn how to work in a team.
7. Learn how to meet the deadlines.
8. Learn the communication skills.
9. Learn how to write a project report.

1. Project Behavioral Specification

You have to design and construct a circuit for a cookie vending machine using switches, lamps (LEDs) and seven-segment displays on the Spartan-3 board. This machine sells cookies for 75 Halala each. Assume that there are only 25, 50, and 100 Halala coins. Three different types of coins can be denoted by three different switches on the board. Cookie can be denoted by an LED and in case when 100 Halala coin or two 50 Halala coins are inserted in the slot; machine will return 25 Halala coin along with the cookie which can be denoted by another LED.

Note: Button presses are only recognized on the rising edge of the clock. As a result all input buttons must be held until the next clock rising edge.

2. Project Organization

As a minimum, the following blocks should be designed:

SEVSEG:
A seven segment decoder, which can display a BCD digit (amount of the coins inserted) on the 7-segment display. Note: Its easier to design it in Verilog.

CONTROL:
A state machine for controlling the inputs to different blocks of the circuit. Design the machine with as few states as possible.

3. Project Implementation

Each of the modules above should be implemented as either a schematic or a Verilog module. Create a symbol for that schematic or module to use in other, higher level schematics. **The top-level design must be a schematic.**
Clock can be by a push button. You can use the code available at my website for debouncing. Each of the above modules should be tested separately using Modelsim. You can refer to the Digilent board documentation to find out how to implement SEVSEG.

4. Guidelines

To be able to complete the project before the deadline, divide the work properly among the team members. Set a deadline to complete the design of each of the above blocks. Include all the files in one project directory. To be able to integrate properly, use the same module definitions in all sub-blocks.

5. Project Report

This section gives a detailed description of the format that you have to use in preparing project report. It is expected that you will follow this format closely. Departures from it may result in points deduction from your grade.

General Report Organization

The final project report must contain the following sections arranged in the order given below:

1. A title page
2. A written description of the project
3. Breakdown of Tasks
4. A section for all the main modules like SEVSEG and CONTROL. For each of these sections, include the following in the order given.

a. Symbols for the module and any submodules you created to design the module.
b. All Verilog code for the module.
c. All schematics for the module
d. A representative set of simulation waveforms used to test the module.
e. A summary of resource usage and timing parameters

All reports must be typed and all documentation must be computer generated.

The remainder of this document will give the detailed requirements for each of these sections.

Title Page

The title page should be a single page with only the following information:

1. Course number with section
2. Project title
3. Your names and IDs
4. The date the project is submitted

Project Description

This section should describe your design. It should provide the reader with information he would need to understand how the circuits work and how you designed them. It should contain the following clearly labeled subsections:

1. Introduction: This is a brief (usually one or two paragraphs) statement of the purpose of the project. It should be written in general terms without giving details. It should be written as if the circuits you design were to be a product or part of a larger product. Do not give any implementation details (e.g., how many gates were used) or go into the details of how you designed the circuits. Just talk about what the circuit does in general terms. Note that this section can be very similar to the first section of this document, (Project Behavioral Specification), but it should be written as a finished product description, not as an assignment of something to be done.

2. Theory of operation: Explain how your circuit works, but do not give implementation details. This should be an expanded version of the introduction. That is, give a high level description of what your circuits do and how they do it. For example, you could explain any algorithms you implemented, any conditions or restrictions the user must observe to use the circuits, and the high level structure of your circuits at the block diagram level.

3. Design details: This subsection is where you can go into the details of your design. It should contain any logical expressions you use, any Karnaugh maps or algebraic simplifications you performed, and any tables or state diagrams for sequential circuits. It should explain these design techniques if they are not self-explanatory. It should refer to the detailed documentation (such as schematic diagrams and Verilog programs) explicitly. This section should also contain a description of any unusual problems you had and how you solved them and any aspect of your design that is novel.

4. Testing details: Most circuits are too large to test by applying all inputs. You have to choose a subset that will do a good job of verifying that the circuits work. This subsection should explain why you chose the test sequences you used and why you think they are sufficient.

Breakdown of Tasks

This section should include a table specifying which task was done by whom. An even breakdown of task is expected.

Verilog Code

This section should contain a listing of all the Verilog code in this project. The code should be commented and easy to read. Put the following information as comments at the beginning of each Verilog description:

- Project title
- Title for the module
- Your name and your partner's name
- Date on which the code was printed

Schematic Diagrams

All schematic diagrams should be included in the report according to the following rules:

1. Make sure all input and output connectors are labeled with the proper signal name. Add labels for any interior signals that appear in the written description of the circuit, especially those that appear in logical expressions.
2. The exact placement of parts and wires on the diagrams is not critical. However, you should ensure that the circuit inputs are on the left side of the page and circuit outputs on the right side so that information flow is from left to right across the page.

Testing results

This section contains a representative sample of the results of the testing you did to confirm that your circuit works properly. This primarily consists of the waveforms generated by the simulation of the Verilog code and schematic diagrams. These tests should be explained in the project description section, which should explicitly refer to the test data in this section. For that purpose, the data from each test must be labeled or numbered and the project descriptions should refer to these labels. To this end, put the following on each item in this section:

- Title or label for the test (e.g., Test 1: Test of seven segment module)
- Your name and your partner's name
- Date when the test was run.

Put this information in a box in the lower right corner of the waveform diagrams. You do not need to include all of your test results. Only include a representative sample of test waveforms. These samples should give an example of all of the operations the module is supposed to perform, but you do not need to include all of the tests you performed to confirm.

Resources used and timing parameters

The resource usage should list the number of slices, LUTs, IOBs, and Flip-flops used in the design. The timing parameters should include all relevant parameters such as minimum clock period, global setup time, global hold time, global delay, and I/O delay (delay from input to output terminals).

6. Project Procedures

For each section, the instructor will assign lab groups consisting of two students. If the number of students is odd, there will be one group of three students. Each group must give a presentation to the instructor in the last lab along with a copy of the Project Report. Attach the grading sheet given below at the end of the report.

Grading Sheet	
Name:	**ID # :**
Student A:	
Student B:	
Project Demonstration (70 points)	
Simulation: **(35 points)**	
Worst Case Speed Parameters, Number of LUTs & IOBs **(5 points)**	tsu_______ ns (clock setup time) th_______ ns (clock hold time) tco_______ ns (clock to output delay) Maximum combinational delay_______ ns #LUTs: _______, #IOBs _______
Hardware works fully? **(30 points)**	
Project Report (30 points)	
Clarity of documentation **(5 points)** Design Description **(10 points)**, Breakdown of Tasks **(5 points)** Simulation Waveforms **(5 points)** Diagrams (Verilog code, state diagrams, schematics) **(5 points)**	
Total points 100	