CS 4341:

Digital Logic and Computer Design

Project, Part 3: Final Project

Due: November 22nd

Late: December 3rd,-15 points

Value: 80 points

Grading:

* The Updated Progress report will be checked to see that the project has made changes during the final month.   
  (5 points)
* The Circuit Diagrams and State Machine will be checked to match the System Design. The Code has to match, structurally, to the Circuit Diagram. The output has to match the State Machine, that is, show all operating states. Everything has to be complete, consistent, and correct. Every mismatch between code, document, output, or diagram, any conflict, is a point off. So, if you leave off a diagram with 20 parts, that is 20 points off. (75 points)

**Objective:**

Tempus Finita. The time allotted for the project has ended. As such, the final material has to be turned in. Updates of the previous two reports must be included, as well as the actual material for the project:

**Requirements:**

Updated Progress Report:

Has the project changed since the progress report? Any additions must be marked with **a bold blue font color**. Anything removed must be marked out with ~~strikethrough~~. The probability that nothing has changed between the progress report and the implementation is infinitesimal. Your Updated Progress Report should reflect changes and effort on the part of the cohort.

*Title Page:*

The first page should have the cohort name, the subtitle “Updated Report” and the date finished.

*Updated Description:*

What is the final vision for this project? What has been added or removed?

*Updated Member Tasks List:*

As development changed, tasks will have been completed or reassigned. Include an updated task list.

*Updated Software Discovery:*

What was the final version of the software used?

*Updated Participation Census:*

It has been some time since the progress report. Show an updated pie-chart showing the current participation of all cohort members.

System Design:

Now the project is complete, a list of parts, inputs, outputs, interfaces, modules, and modes will have been incorporated into the final system. The System Design should include lists or tables of the individual parts with descriptions. Included will also be a circuit diagram with datapath, and one or more state machines to show the modes of the system.

*Title Page:*

The first page should have the cohort name, the subtitle “System Design” and the date finished.

*Parts List:*

Computational Logic and Sequential Logic are made up of gates, components, registers and other datapaths. Make a list of each part and include a description of each.

*Input List:*

Both data and commands enter the system as feeds into various modules and components in the circuit. Make a list of each input line, and include a description of each.

*Output List:*

Various results and errors come out of the system for the users and entities to see and relate to. Make a list of each output line, and include a description of each.

*Interface List:*

Inside the system, control wires go between modules, and are neither inputs are outputs. Make a list of each interface wire, and include a description of each.

*Module List:* By the time the project is near completion, some modules are parts, and others are data paths. And some are included because they are needed for the hardware design language. Any component not in the Parts List should be included in the Module List. This includes the Testbench.

*Mode List (States):*

Any sequential circuit has more than one state. Some circuits can be created to have a single state machine with many states to control the system. Other systems can be comprised of multiple state machines working together. List out each mode, and include a description of each.

*Circuit Diagram:*

The project should include one circuit diagram of the overall system. No, it cannot be a single box with the label “Datapath” with twenty inputs and outputs. The diagram must follow the convention of inputs arriving from the left, and outputs going to the right, for each module. *Everything must be labelled and must match names in the system design. If the labels are inconsistent, then points will be deducted.*

*State Machines:*

Every system is controlled by one or more state machines. The states combine to create the “Current User View” or “Current System State” For each state machine created by the sequential logic in the system, a state machine diagram needs to be included. *Everything must be labelled and must match names in the system design. If the labels are inconsistent, then points will be deducted.*

Design Language Code and Output:

The final project will be implemented in the design language specified in the software discovery. After completing the design, the Design Language can be used to create the various components to run the system. The Design Language modules need to match the parts list in your specification. With the running program, the system can be tried for every possible combination, even setting error states. Given that language allows scripting, enormous truth tables can be created very quickly.

The Verilog code should have:

* A Module to match each component in the design
* Modules that combine components into the circuits
* A testbench that shows the key features, not all features
* A display to the screen that shows
* Each Input with a label
* Each operation with a label
* State information with a label
* Each output with a label.

Something like:

|  |
| --- |
| Num 1 Num 2 Operation Current Output Next State  10011001 (153) 10101010(341) 00 (Add) Running 111101110 (494) Running  10011001 (153) 00000000( 0) 11 (Divide) Running XXXXXXXXX (Nan) ERROR  … |

**Deliverables:**

A document in the form of <COHORTNAME>.UpdatedReport.pdf containing the final version of the progress report with all the changes marked.

A document in the form of <COHORTNAME>.SystemDesign.pdf containing the lists of parts, descriptions, and diagrams from the System Design section.

A zip file in the form of <COHORTNAME>.Code.zip containing all the files necessary to run the project.

A TEXT file in the form of <COHORTNAME>.Output.txt containing the output from the screen. This should be a text file, not a screen capture.

If and only if the software used to run this system includes a graphical output, then the output should be turned in as a single PNG file of the form <COHORTNAME>.Graphic.png

**Deliverables Checklist :**

* Updated Report
  + Title Page
  + Updated Report
  + Updated Description
  + Updated Member Tasks List
  + Updated Software Discovery
  + Updated Participation Census
* System Design
  + Title Page
  + Parts List
  + Input List
  + Output List
  + Interface List
  + Module List
  + Mode List
  + Circuit Diagram
  + State Machine Diagrams
* Code and Output
  + Code (Modules, Interfaces)
  + Output (Show the operations)