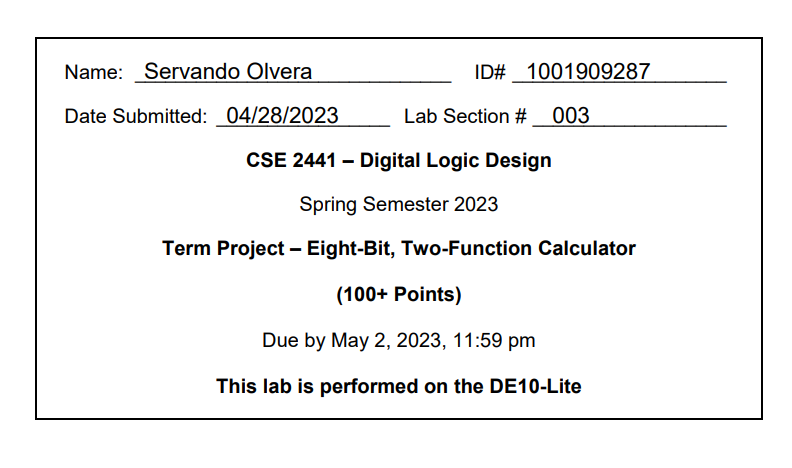
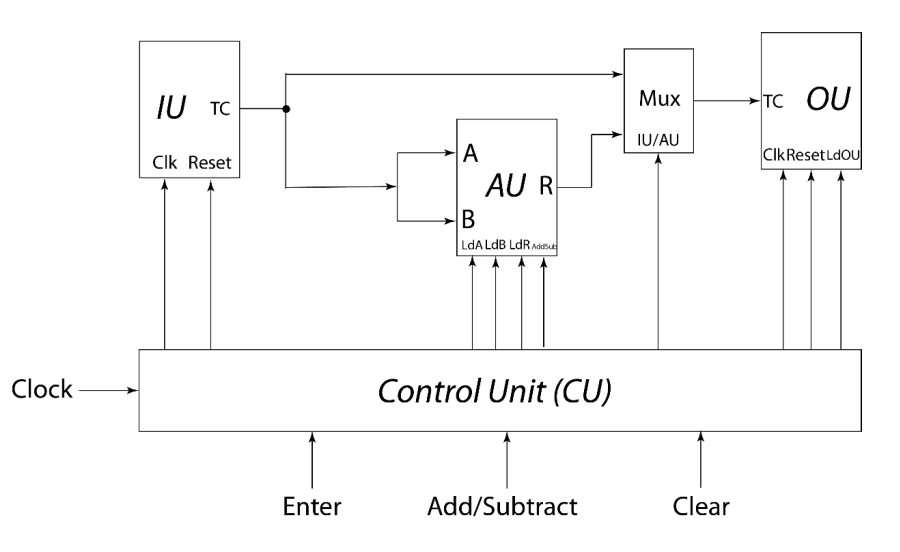
TERM PROJECT



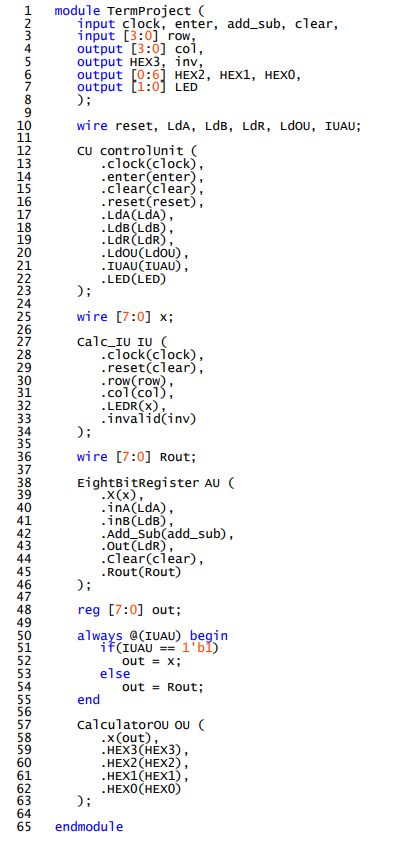
Calculator Top-Level Functional Diagram:



Added Features:

* Invalid Input Recognizer

Top-Level Verilog Code:



Control Unit

State Diagram:

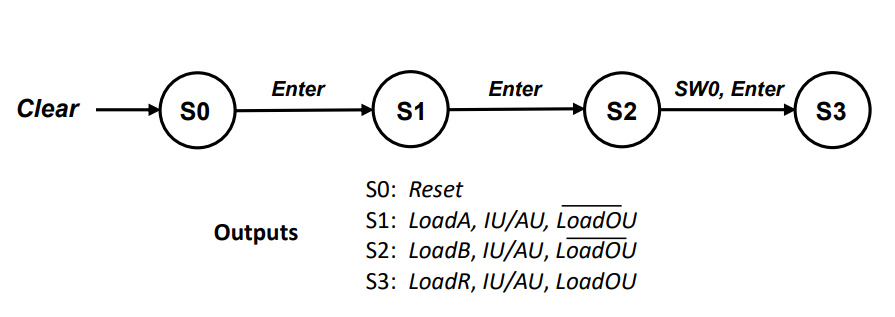
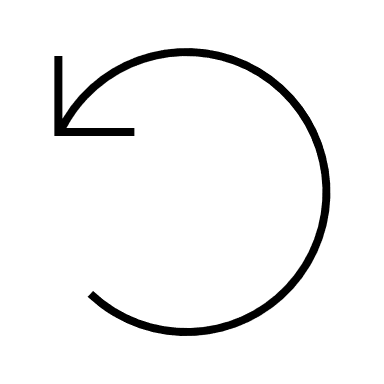
Shape

Description automatically generated with medium confidence

***CLEAR***

Shape

Description automatically generated with medium confidenceShape

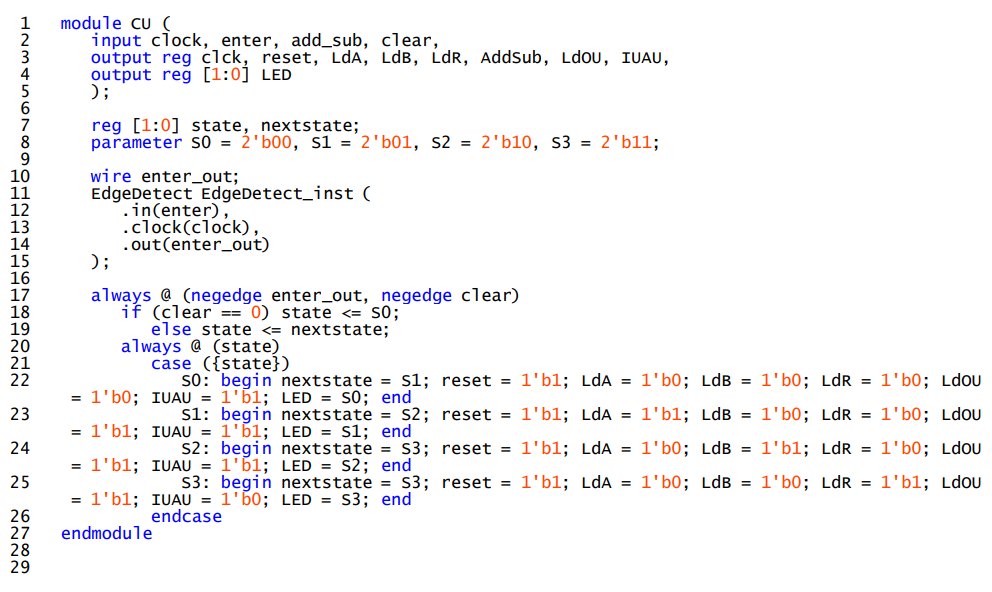
Description automatically generated with medium confidence

***ENTER***

***CLEAR***

***CLEAR***

Verilog Code:



Test Demo:

Demoed in person by \_\_Andrew\_\_\_

Table of Results:

|  |  |
| --- | --- |
| Test Input | Output |
| 74 + 35 | 109 |
| 74 – 35 | 39 |
| -74 + 35 | -39 |
| -74 – 35 | -109 |
| 127 + 6 | -123 |
| 127 – 6 | 121 |
| 9 + 10 | 19 |
| 9 – 10 | -1 |
| 88 – 125 | -37 |
| 88 + 125 | -43 |

Unresolved Problems:

N/A

Lesson Learned:

Breaking down a rather large project into smaller components greatly facilitates the completion of said project. For instance, at the beginning of this course we learnt about the basics of Digital Logic, from there we started to build up into more complex topics. We went from basic Logic Gates to High-level Logic Devices, Synchronous Sequential Circuits, and Finite State Machines.

In a likely manner, with this Term Project its assembly was done quite easily since it had already been broken down for us into smaller chunks of work, which were in essence the foundation of the project. Labs 8, 9, and 10 took care of the Calculator’s Arithmetic Unit, Output Unit, and Input Unit, respectively, and all this work amounted to about 80% of the Term Project. All there was left to do, was to merely implement the Control Unit, a MUX, and the Top-Level Module, where everything could connect. Having already the knowledge of how these components communicate with one another, the creation of Control Unit and the simple instantiations of the Arithmetic Unit, Output Unit, and Input Unit along with a simple if else statement for the MUX, the project is then completed.

Taking all of this in mind, if one carefully analyzing the requirements for a project, breaks it down into smaller chunks of work, and organizes how it all will fit together, it allows for a more efficient workflow. ­­­­­