ETE 101b Digital Design

Term Project

Murat Işık

Oğuzhan Topal - 190503001

Design Assumptions:

1. Single Operation at a Time:

The calculator assumes that only one operation (addition, subtraction, multiplication, or division) is performed at a time.

2. Numeric Input Limitations:

The calculator assumes that numeric input is limited to 4-digit numbers (0 to 9999).

3. Error Handling:

The calculator assumes basic error detection for division by zero and overflow in addition and multiplication.

4. Reset Behavior:

The calculator assumes that a reset signal initializes the system to a default state, clearing previous results and errors. It is also needed at the start of the program.

5. Sequential Operation:

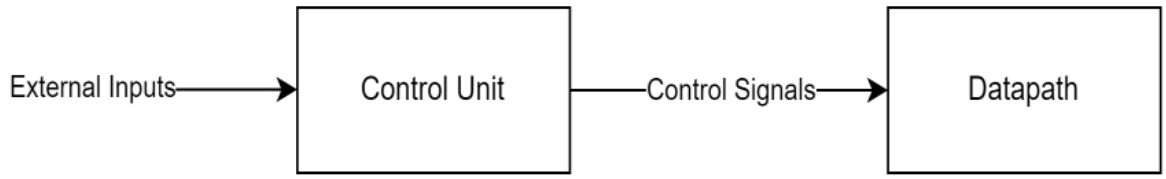
The calculator assumes a sequential operation where the user enters numbers and operations one at a time.

6. No Floating Point Arithmetic:

The calculator assumes integer arithmetic and does not handle floating-point numbers.

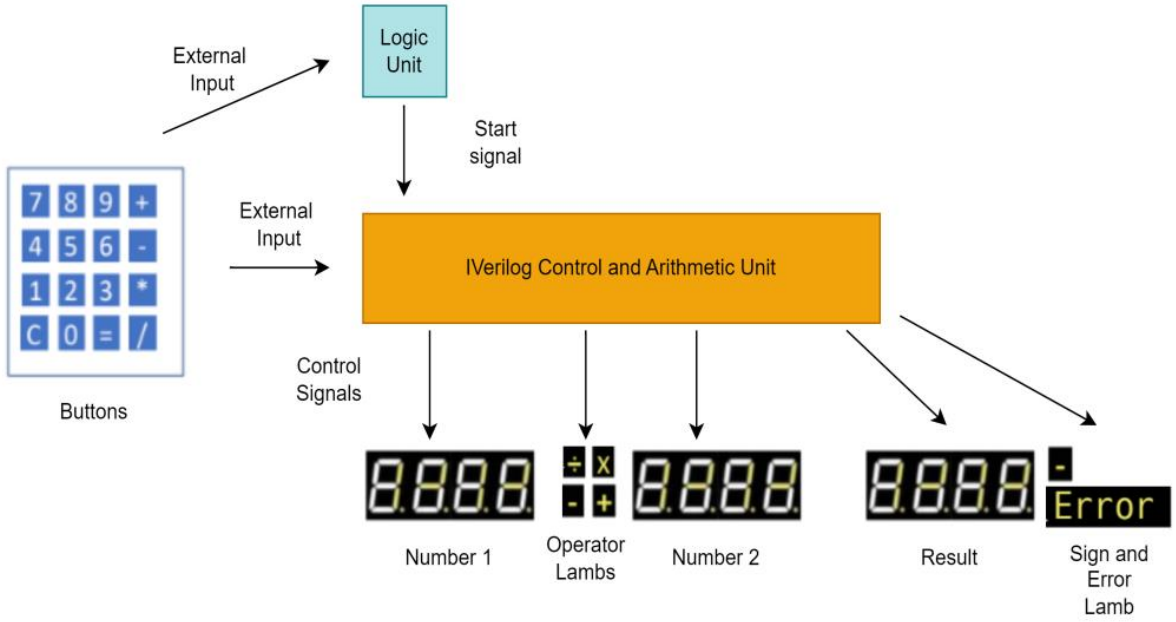
Design Approach:

I used only one control unit in the project. The only inputs are external inputs provided by the user and start signal. With these inputs, the control unit calculates the results and shows in the displayers.



Block Diagram:

Here is the detailed block diagram of Calculator.



Summary Of Results:

1. Project Overview:

The project aimed to design a calculator using Verilog, capable of performing basic arithmetic operations (addition, subtraction, multiplication, division) and error handling.

2. Functionalities Implemented:

Number Input: The calculator successfully captured and stored numbers on each button press and showed in displayers.

Operator Selection: The design accurately detected the selected operation based on the provided operator input.

3. Arithmetic Operations:

Addition and Multiplication: The calculator correctly performed addition and multiplication operations within the defined display size with error detection.

Subtraction: The subtraction operation was implemented and produced correct results for both negative and positive numbers.

Division: The division operation was implemented with error detection, addressing division by zero scenarios.

4. Error Handling:

The calculator project incorporated error detection mechanism, particularly for division by zero or overflow conditions.

5. Display Logic:

The logic for converting internal binary representations of numbers into 7-segment display formats was successfully implemented.

6. Room for Improvement:

The project summary acknowledges that the calculator has no clock input, so it is not synchronized, and also that the error handling mechanism only detects errors, not prevents them in advance.

7. Conclusion:

The calculator project demonstrated successful implementation of basic arithmetic operations in Verilog, with a modular and well-documented design. Identifying and addressing areas for improvement will contribute to refining the overall functionality.

Lessons Learned:

I learned how to properly use always blocks, if – else statements and case blocks in IVerilog and how to control 7 segment displayers with control unit.