
JOINT INSTITUTE
交大密西根学院

Ve 270 Introduction to Logic Design

# Lab 7
# Design of a Digital Device

UM-SJTU Joint Institute
Shanghai Jiao Tong University
July 2021

## 1. Objective

To design a digital system that

- Rolls your SJTU student ID across the four SSDs on the FPGA board when the system starts, then
- Adds two 4-bit 2's complement numbers and displays the decimal results with the SSDs on the FPGA board.

## 2. Requirement

### *Part 1. Character Rolling*

The device is put in character rolling mode when the device starts functioning and a mode control switch M = 0. When the device is in character rolling mode, the device should display and roll the student ID of either lab partner selected by a toggle switch called P. For example, when P = 0, ID of the first student is displayed and rolled; when P = 1, ID of the second student is displayed and rolled. The selected student ID should be rolled through the four SSDs from left to right, or right to left, or back and forth.

The rolling speed is up to the designers, but should be reasonable so people are comfortable to see each character. In addition to the student ID, you may choose to display any other things or any pattern you want. Be creative!

The student ID should roll through the SSDs continuously and repeatedly until the rolling function is disabled when M = 1. As soon as the rolling mode is ended, the system should enter the calculator mode.

### *Part 2. Simple Calculator*

The calculator adds two 4-bit 2's complement numbers, and outputs one 4-bit 2's complement number which should be interpreted as a signed decimal number and displayed using one (for positive results) or two (for negative results) SSDs. The "–" sign must be displayed using one SSD for negative numbers. The calculator must also detect if there is any overflow in the results and indicate the overflowed results with an LED.

The two 4-bit operands of the calculator must be entered using the same four switches. Thus the operands have to be entered in turn, one after another. A push button must be used to function as the "enter" or "equal" key of the calculator. After the first number is formed using the four switches, the decimal equivalent should be displayed on the SSDs as soon as the "equal" button is pressed. After the second number is formed using the same four switches, the decimal result should be displayed on the SSDs as soon as the "equal" button is pressed. When the "equal" button is pushed multiple times, the calculator shall keep adding the second number to the results each time the button is pressed. The changing results should be updated on the SSDs accordingly. The following table simulates an example.

| Four Switches | Equal Button | SSD | Overflow |
|---|---|---|---|
| 1000 | push | -8 | 0 |
| 0011 | push | -5 | 0 |
| No change | push | -2 | 0 |

| | | | |
|---|---|---|---|
| No change | push | 1 | 0 |
| No change | push | 4 | 0 |
| No change | push | 7 | 0 |
| No change | push | -6 | 1 |
| No change | push | -3 | 0 |

The above paragraphs describe the basic functions for the device. Feel free to add more features and functions to the device.

**NOTE**: your design must be modeled with Verilog HDL.

## 3. Simulation, Synthesis, and FPGA Implementation

Simulate your circuit. It is a good idea to verify the functionality of all sub-circuits before you integrate them all together. Synthesize and implement your design on the Basys 3 FPGA board.

## 4. Deliverable

This is a 2-week lab. The full score for this lab is 300 points.

1) Demonstrate your circuits to the TAs before your lab session ends.
2) Upload source files on Canvas by **10pm, July 31, 2021**.