National Tsing Hua University
Department of Electrical Engineering
EE429200 IC Design Laboratory, Fall 2021

Lab 03: Testbench Debugging and Writing (1%)

Assigned on Sep 30, 2021

Due day on Oct 7, 2021

Objective

In this lab, you will learn:

1. Understand the basic operation of Function Unit (FU).
2. Write test bench to verify the function of Function Unit and debug.

Demo checklist:

- ☐ Show TA how do you verify FU, and the pattern coverage of your test pattern.
- ☐ Answer the following question.
  1. Why do we declare input drivers as *reg* and output loads as wire variable in testbench?
  2. We use *#delay* to simulate the waveform of input pattern, but can we use it to describe the circuit?
  3. Is it useful to verify your circuit if the operation commands in circuit and testbench are the same? (ex. Y = A+B;).

Environment Setup

Copy lab file packages from ee4292. Decompress the package and enter it. You can check the file list in Appendix.

```
$ cp ~ee4292/iclab2021/lab03.zip .
$ unzip lab03.zip
$ cd lab03
```

Description

Functional Unit is one of the most critical designs in most of the microprocessor or DSP (Digital Signal Processor). In lab2, we have a simple ALU to do some simple arithmetic functions. A Functional Unit based on the ALU in lab2 and with furthermore functions, is provided in this lab. ***lab3_fu.v*** describes the Functional Unit (shown in Figure 1.) with two 16-bit input signals and 5-bit instructions. You can input signals and select the function you want to execute by instructions. The instruction is defined in Table 1. In this lab, you will write a test bench to test the provided FU and correct the bugs in it.

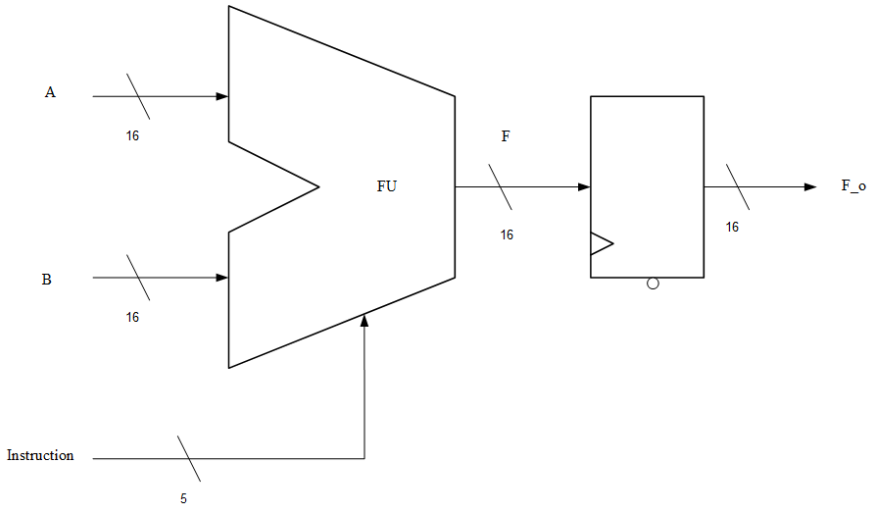

Figure 1. ALU architecture.

Function	Instruction[4:0]	Operation
Arithmetic	5'b00000	F = A
Arithmetic	5'b00001	F = A + 1
Arithmetic	5'b00010	F = A + ~B
Arithmetic	5'b00011	F = A + ~B + 1
Arithmetic	5'b00100	F = A + B
Arithmetic	5'b00101	F = A + B + 1
Arithmetic	5'b00110	F = B
Arithmetic	5'b00111	F = A - 1
Logic	5'b01000	F = A and B
Logic	5'b01001	F = A or B
Logic	5'b01010	F = A xor B
Logic	5'b01011	F = ~A
Shifter	5'b10000	F = shift-right B by 1 bit
Shifter	5'b10001	F = shift-left B by 1 bit
Shifter	5'b10010	F = rotate-right B by 1 bit
Shifter	5'b10011	F = rotate-left B by 1 bit

Table 1. ALU function table.

Action Items

I. Complete the FU.

Add the rotator function into FU. You can reference table to complete this lab.

II. Write your own testbench to test FU.

1. In this part, you have to verify whether FU (lab3_fu.v) is correct or not.
2. You need to design the testbench and the test patterns to test all functions of your FU design in testbench (**start from lab3*fu*test.v**).
   A. Please at least verify the case **A= 100, B= (-201)** for each function in Table 1. A reference result of a. is provided in answer.txt. *(Hint: There are 4 errors in original lab3_fu.v)*
   B. Generate other test patterns with generate*golden.py. Please modify your test bench and compare your results with golden2.dat (generated by generate*golden.py).
3. You need to display your results on the screen by *$display* command and write them to a file (output.txt) by *$fdisplay* command in test bench *lab3*_fu_*test.v*.
4. Run simulation to debug and complete all the functions of FU.
   $ ncverilog –f lab3_run.f

Appendix

Filename	Description
lab3_fu.v	RTL code for FU
lab3_fu_test.v	Test bench for FU
lab3_run.f	Filelist
golden.dat	Golden pattern
generate_golden.py	Python script used to generate golden patterns
answer.txt	Answer for FU