**National Cheng Kung University**

**Department of Electrical Engineering**

***Introduction to VLSI CAD (Spring 2022)***

**Lab Session 3**

**Design of ALU and Fixed point Multiplication**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Student ID | | |
| 鄭智宇 | E94096110 | | |
| Practical Sections: | | Points | Marks |
| Prob A | | 20 |  |
| Prob B | | 30 |  |
| Prob C | | 30 |  |
| Report | | 15 |  |
| File hierarchy, naming…etc. | | 5 |  |
| Notes | | | |

**Due Date: 15:00, March 08, 2023 @ moodle**

**Deliverables**

1. All Verilog codes including testbenches for each problem should be uploaded.

NOTE: Please **DO NOT** include source code in the paper report!

1. All homework requirements should be uploaded in this file hierarchy or you will not get the full credit.

NOTE: Please **DO NOT** upload waveforms!

1. Important! TA will use the command in Appendix A to check your design under SoC Lab environment, if your code can not be recompiled by TA successfully using the commands, you will not get the full credit.
2. **If you upload a dead body which we can’t even compile you will get NO credit!**
3. **All Verilog file should get at least 90% superLint Coverage.**
4. **File hierarchy should not be changed; it may cause** your code can not be recompiled by TA successfully using the autograding commands

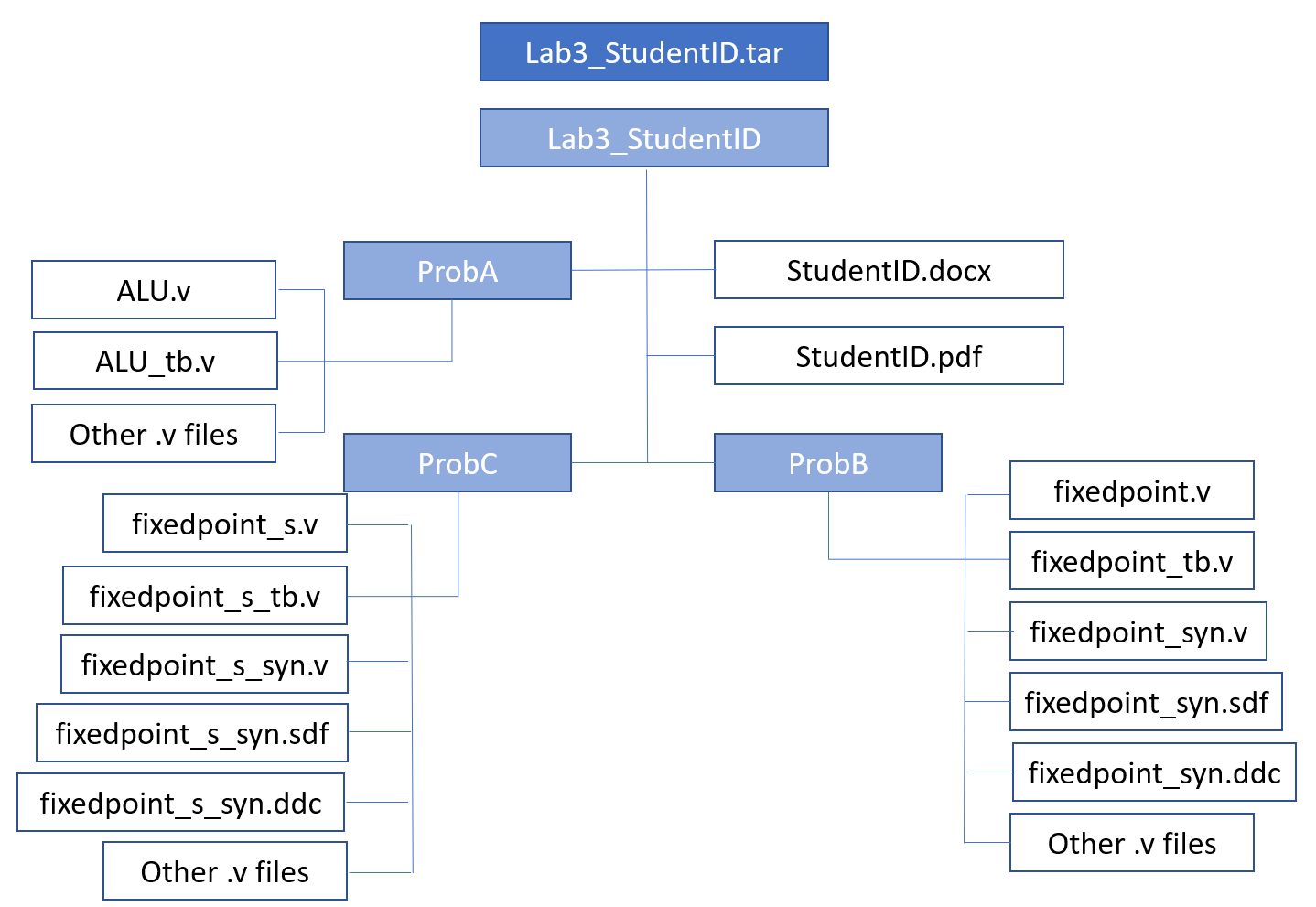
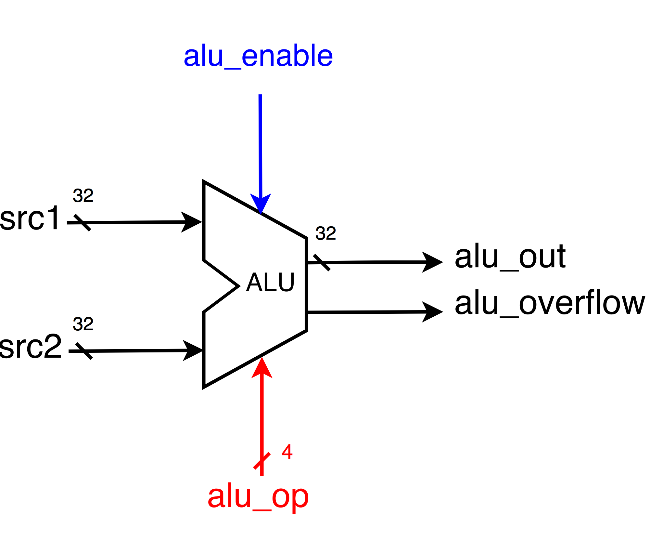


Fig.1 File hierarchy for Homework submission

**Objectives:**

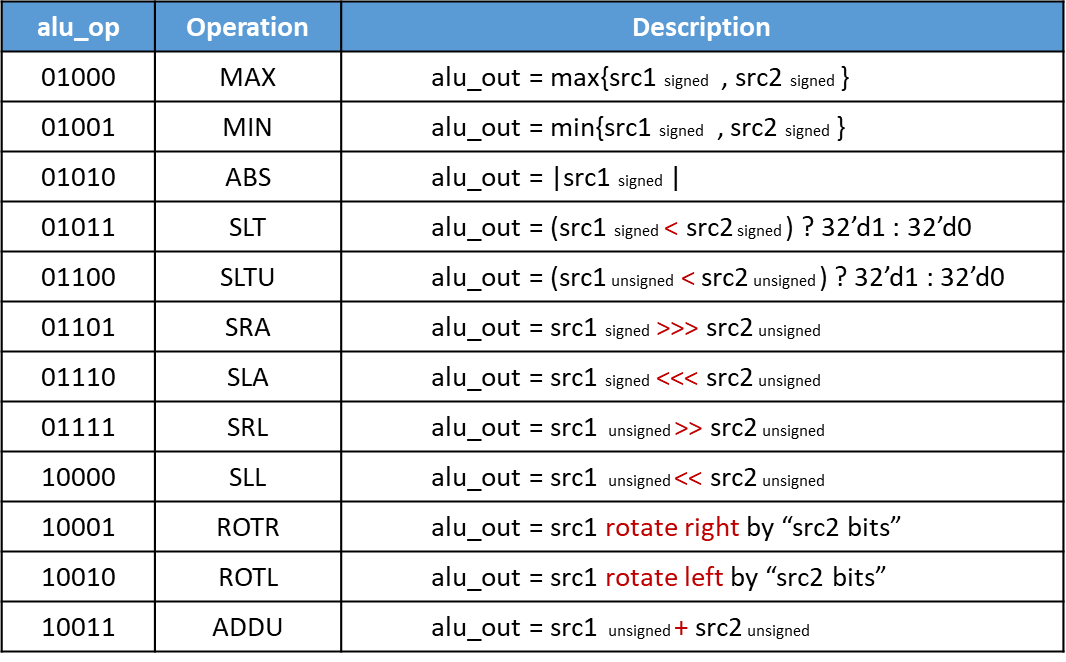
**Learn how to design an ALU.**

Prob A: Arithmetic Logic Unit



1. Based on the reference code, please implement the following operations.





* + **Please attach your design waveforms.**

|  |
| --- |
| Your simulation result on the terminal. |
|  |
| Your waveform : |
| Step1 : ssrc1 = src1 的signed 型態  ssrc2 = src2 的signed 型態  Step2 : 利用ssrc1, src1, ssrc2, src2, alu\_op判斷alu\_overflow, alu\_out |
| SuperLint Coverage |
| 0 warning  Coverage : 100% |

Prob B: Practice fixed point

**Design your Verilog code with the following specifications:** Number format: unsigned numbers.

* 1. The frame code and testbench are given. Follow the frame code to finish this homework. The decimal part should be rounded.
  2. Follow the PPT file to synthesize your code.

**After you synthesize your design, you may have some information about the circuit. Fill in the following form.**

|  |  |  |
| --- | --- | --- |
| **Timing (slack)** | **Area (total cell area)** | **Power (total)** |
| **19.16** | **79.031998** | **23.4755 uW** |

**Please attach your design waveforms.**

|  |
| --- |
| Your simulation result on the terminal. |
|  |
| Your waveform (RTL & Synthesis) : |
| **RTL:**    Step1 : all\_out (16bits) = in1(8bits)\*in2(8bits)  Step2 : out(8bits) = all\_out[15:8] + 4'd1 (須進位)  = all\_out[15:8] (不須進位)  **Synthesis:**      合成完後，在每個step間會出現延遲 |
| SuperLint Coverage |
| 1 warning  Coverage 約等於95% |

Prob C: Practice fixed point (signed)

**Design your Verilog code with the following specifications:** Number format: signed numbers.

1. The frame code and testbench are given. Follow the frame code to finish this homework. The decimal part should be rounded.
2. Follow the PPT file to synthesize your code.

**After you synthesize your design, you may have some information about the circuit. Fill in the following form**

|  |  |  |
| --- | --- | --- |
| **Timing (slack)** | **Area (total cell area)** | **Power (total)** |
| **18.96** | **107.492398** | **32.1012 uW** |

**Please attach your design waveforms.**

|  |
| --- |
| Your simulation result on the terminal. |
|  |
| Your waveform (RTL & Synthesis) : |
| RTL:    Step1 : 統一將in1 , in2換成正數p\_in1 , p\_in2  Step2 : max\_out (16bits) = p\_in1(8bits)\*p\_in2(8bits)  Step3 : t\_out(8bit) = max\_out[14:7] ,取出sign[7], integer[6:1], decimal[0]  Step4 : roundf(8bits) = t\_out(8bits) + 8'd1 (須進位)  = t\_out(8bits) (不須進位)  Step5 :out(8bits) = 將t\_out 轉為原本的sign  Synthesis:    合成完後，在每個step間會出現延遲 |
| SuperLint Coverage |
| 1 warning  Coverage 大約等於 98% |

**At last, please write the lessons learned from this lab session, or some suggestions for this lab session. Thank you.**

1. **學習到合成相關知識**
2. **reg、wire的區別**
3. **小數的binary表示方法**

Appendix A : Commands we will use to check your homework

|  |  |  |
| --- | --- | --- |
| **Problem** |  | **Command** |
| **ProbA** | Compile | % ncverilog ALU.v |
| Simulate | % ncverilog ALU\_tb.v +define+FSDB +access+r |
| **ProbB** | Compile | % ncverilog fixedpoint.v |
| Simulate | % ncverilog fixedpoint\_tb.v +define+FSDB +access+r |
| Synthesis | % ncverilog fixedpoint\_tb.v +define+FSDB+syn +access+r |
| ProbC | Compile | % ncverilog fixedpoint\_s.v |
| Simulate | % ncverilog fixedpoint\_s\_tb.v +define+FSDB +access+r |
| Synthesis | % ncverilog fixedpoint\_s\_tb.v +define+FSDB+syn +access+r |