Computer-Aided VLSI System Design, Fall 2019

Verilog Homework 1 – ALU Design

TA: 董子維 (f07943011@ntu.edu.tw)

1. Homework Goal

From the example of the ALU basic arithmetic unit, let you know how to complete the RTL level circuit description, and then learn to use the simulator (NC-verilog) and testbench to run RTL-level simulation on the workstation. Pay attention to the Notes and Appendix to design the hardware and Verilog Coding (Coding Style).

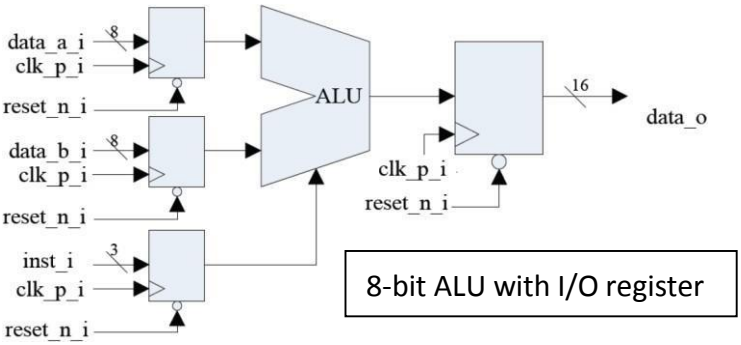
2. ALU : Introduction

The Arithmetic logic unit (ALU) is one of the most important components of a computer processor. The ALU has math, logic, and some designed operations in the computer. Based on LAB1, some instructions need to be designed and completed for image processing and also to complete ALU with I/O register. Please state the below diagram 8-bit ALU with I/O register, 3-bit instruction set circuit.

Inst_i [2:0]	Operation	Notes
000	Signed Addition	data_o = data_a_i + data_b_i
001	Signed Subtraction	data_o = data_b_i - data_a_i
010	Signed Multiplication	data_o = data_a_i * data_b_i
011	AND	data_o = data_a_i & data_b_i
100	XOR	data_o = data_a_i⊕data_b_i
101	Absolute Value	data_o = | data_a_i |
110	Addition & Divide by 2	data_o = (data_b_i + data_a_i) / 2
111	Max	data_o=max(data_a_i, data_b_i)

[Notes] Considerations between digital systems and signals:

- Output signal “data_o” has 8bits and it is shown in 2’s complement。
- Input data(a & b) are signed for all instructions
- For these operations, if the operation overflows, that means: overflow_o = 1.。
  - overflow (definition): The result of the operation is greater than the maximum range that can be represented by the number of bits, resulting in an incorrect output result.
- Before and after the operation, if the total number of bits of the variable (reg or wire) are not equal, the system will fill the bites with zeros. When it is necessary please use sign extension for the signals.
- For inst = 110, ??????when you are dividing by 2 you have to drop down divide by 2 to unconditionally drop down to the integer bit.
- If it is a fix point operation, you need to pay attention to the precision and the ¿area?, that is, the total bit number and the position of the decimal point (shift bit)¿has to pick the place?, which can be in the case where the integer part does not overflow and the decimal part is accurate enough. Make the number of bits the least (area). (No need to pay attention to this operation)



1. The homework provides an uncompleted RTL template program. Please follow the above spec. table complete the program. In order to speed up the debugging process, it is recommended to simulate each time an instruction is completed. In the testbench:
   - If assign “test_all_ins” = 1, all instructions will be tested (default setting)
   - If assign “test_all_ins” = 0, specify a specific "test_instruction" to test the corresponding instruction

   (ex: test_all_ins = 0, test_instruction = 000 : only test the signed addition function)

Filename	Description
HW1_alu.v	Unfinished Verilog RTL template
HW1_test_alu.v	Testbench for ALU design

2. Please use the provided testbench to verify your ALU with I/O registers design, we will rate your design according to testbench.

3. Upload request

Please follow the ceiba instructions

4. Due Date: 10/10 13:59

5. Grading

The ALU has a functional test score of 12.5 points, and all eight functions are worth 100 points. It should be noted that when testing the operation, additional testing will be used when testing each function.。

5. Appendix

[Naming Rules]

Using Naming Rules will greatly improve the maintainability and readability of Code. The most important thing is that it can explain precissely the hardware architecture. We recommend naming the variables in the design using the naming rules provided below.

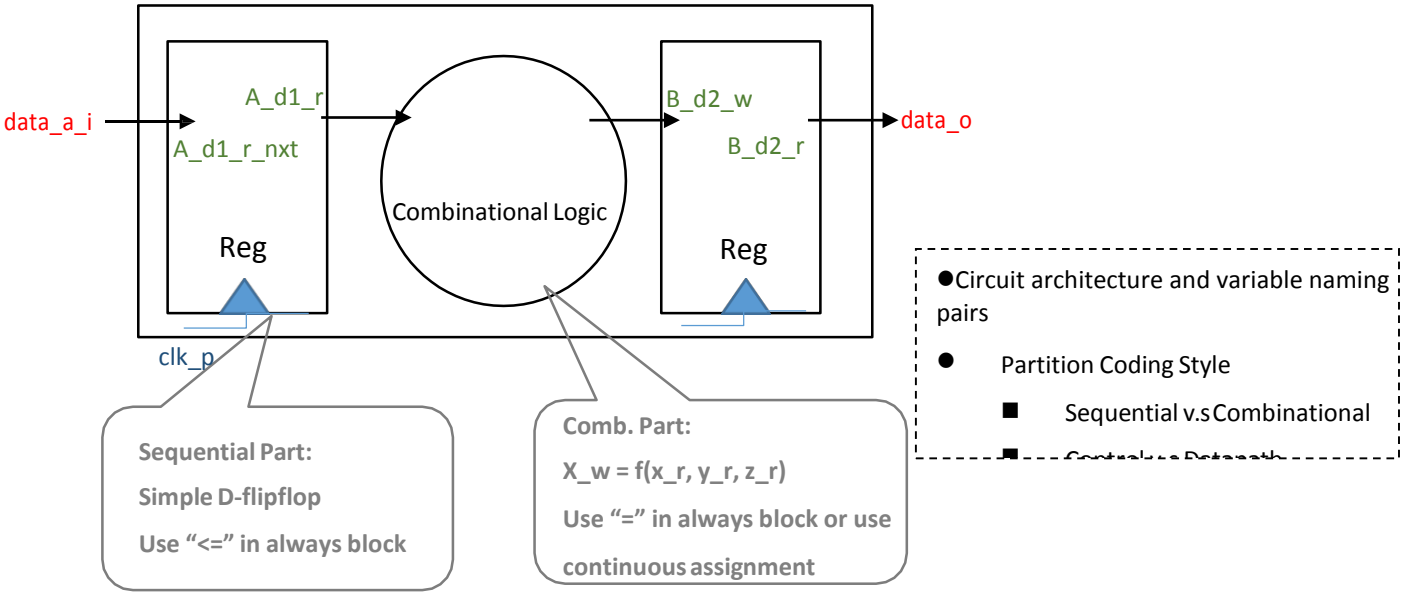
1. "_i" and "_o" should be added to the end of the input and output variables in a module
2. If it is clk or reset signal, "_p" or "_n" can be used to indicate whether he is a positive edge trigger or a negative edge.
3. If the variable is a flipflop in the circuit, then after the reg variable name:
   - For Flipflop inputs add "_w" to indicate write, for flipflop output add "_r", which means read.
   - Flipflop input: a_nxt, Flipflop output: a

   (Note that the emphasis here is that flipflop is not reg, because as long as the variables declared in the block are reg type, it is possible to actually pull the line or declare it as reg, but the variables outside the block must be declared as wire and must also be a zip???)
4. With the number of layers in the pipeline in the system, the signal will be delayed by x cycles, after which the "_dx" can be expressed as the signal at the xth layer. Ex: a_d1_w (flipflop input of the first layer))
5. If it is necessary to use truncate (such as fixpoint operation) for a_w intermediate operation, the intermediate wire in the combinational circuit can be named tmp and then connected to a_w because the syntax limit cannot be written in one line.

Ex:

a_tmp = b_r + c_r;

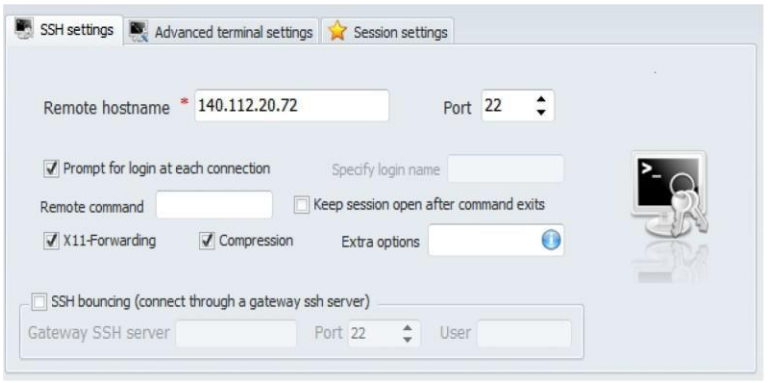
a_w = a_tmp[5:3]



[Run NC-Verilog Simulation on GIEE's Workstation]

(1) Connect to GIEE's Workstation: [Take MobaXterm for illustration]

Available workstation list: http://cad.ee.ntu.edu.tw/ws_list.htm



(2) Environment Setup: [type following commands in the console]

1. Making Directory:

   mkdir HW1

2. Enter the Directory:

   cd HW1

3. Source the default file:

   source /usr/cadence/cshrc

(3) Check Verilog Code by NC-Verilog

1. Type this command:

   ncverilog [`design_filename]

   ex: ncverilog HW1_alu.v

(4) Run Simulation with a test bench

1. Type this command:

   ncverilog [`testbench_filename] [`design_filename]

   ex: ncverilog HW1_test_alu.v HW1_alu.v

PS: From the big module to the small module, if there is tsmc13.v in the future, it should be added at the end to let the simulator convert the foundry standard circuit module into a understandable basic behavior block (ex:and), or include memory , IO pad extra non-standard cell module's external-IP behavior model。

(5) If you want to check the waveform, (In principle, this operation can be completed without looking at the waveform.)

1. Type the command:

   ncverilog [`testbench_filename] [`design_filename] +access+r

For more GIEE's workstation information, please refer to: http://cad.ee.ntu.edu.tw/

For more workstation commands, please refer to:
http://linux.vbird.org/linux_basic/redhat6.1/linux_06command.php