NCTU-EE IC LAB – Fall 2018

Lab01 Exercise

Design: Code Calculator

Data Preparation

1. Extract files from TA's directory:
   % tar xvf ~iclabta01/Lab01.tar
2. The extracted LAB directory contains:
   a. Practice/ : example code
   b. Exercise/ : your design

Design Description and Examples

"Win or Go Home!". At the final stage of NCTU Millionaire, you are asked to answer a question based on a series of simple mathematic operations. The only challenge is the remaining time, 20 ns. If you answer it in time correctly, you will win a prize of a million dollar. "Ready… Start…".

First, you will receive a sequence with 4 numbers {**in_n0, in_n1, in_n2, in_n3**} and a 4-bit **opt** signal. Then the 2 possible operations are given in the following order:

- **Sort**:
  Sort the sequence if **opt[0]** is 1, and the order depends on **opt[1]**
  For example, {2, 1, 3, 5} becomes {5, 3, 2, 1}, if **opt[1] = 0**.
- **Reduce Mean**:
  If **opt[1]** is 1, find the mean value of the sequence with/without sorting (**round-down**) and subtract it from all the numbers in the sequence.
  For example, {4, 2, 7, 9} has mean value 5 (round down from 5.5), after reduce mean operation, the sequence will be {-1, -3, 2, 4}.
  For example, {-1, -2, -3, -4} has mean value -2 (round down from -2.5), after reduce mean operation, the sequence will be {1, 0, -1, -2}.

After operations with/without sorting and/or reducing mean, you will get a sequence **n0, n1, n2, n3**}. Finally, the output answer can be obtained by one of the following equations (**remember all the numbers are signed**):

- Eq0 : ((n3 + n2*4) * n1) / 3 **(round-down the answer if it is not integer)**
- Eq1 : (2 * n1 * n0) + n3

(**Hint: Try to use behavior modeling description instead of gate level description**)

The summary of the description and specifications are as followings:

Input Signal	Bit Width	Description
in_n0	4	The first number of code. Ranged from **-8**~**7**.
in_n1	4	The second number of code. Ranged from **-8**~**7**.
in_n2	4	The third number of code. Ranged from **-8**~**7**.
in_n3	4	The forth number of code. Ranged from **-8**~**7**.
opt	4	Operator for different mode. The operation will be encode as following: **opt[0]:** 1: Sort. 0: Don't sort. **opt[1]:** 1: Ascending sort. 0: Descending sort. **opt[2]:** 1: Reduce mean. 0: Don't reduce mean. **opt[3]:** 1: Eq1. 0: Eq0.

Output Signal	Bit Width	Description
out_n	9	The answer. Ranged from -256~255

Examples:

1. **Initial numbers {-4, -5, -7, 1} with opt = 4'b0101:**

   {-4, -5, -7, 1} $-(sort\ ,descending)->$ {1, -4, -5, -7} $-(reduce\ mean)->$ {4, -1, -2, -4} $-(Eq0)->$ 4 // In this case, average = -3.75, round down to -3

2. **Initial numbers {-3, 0, 7, 7} with opt = 4'b1110:**

   {-3, 0, 7, 7} $-(reduce\ mean)->$ {-5, -2, 5, 5} $-(Eq1)->$ 25

3. **Initial numbers {2, -3, -7, -3} with opt = 4'b0100:**

   {2, -3, -7, -3} $-(reduce\ mean)->$ {4, -1, -5, -1} $-(Eq0)->$ 7

4. **Initial numbers {-8, -2, 0, -3} with opt = 4'b0000:**

   {-8, -2, 0, -3} $-(Eq0)->$ 2

Inputs

1. The input signals **in_n0, in_n1, in_n2,** and **in_n3** are **signed** 4-bit inputs ranged from -8 to 7.
2. The input signal opt is a 4-bit inputs indicated whether to do the operations and

which equation to use to get the final result.

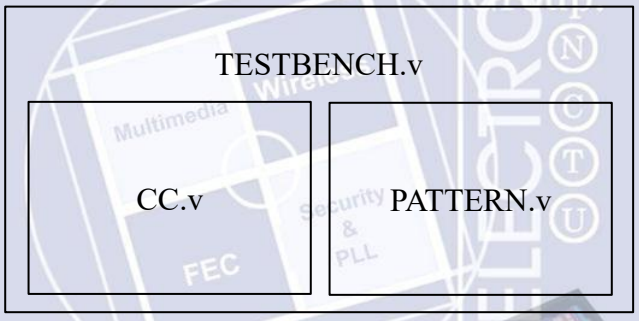
Outputs

The output signal **out_n** is a signed number ranged from -256 to 255. This represents the correct password.

Specifications

1. Top module name : CC (File name: CC.v)
2. Input pins : in_n0, in_n1, in_n2, in_n3, opt
3. Output pins : out_n0
4. After synthesis, check the "CC.area" and "CC.timing" in the folder "Report". The area report is valid only when the slack in the end of "CC.timing" is non-negative.
5. The synthesis result **cannot** contain any **latch**.
   Note: You can check if there is a latch by searching the keyword "**Latch**" in 02_SYN/**syn.log**

Block Diagram



Grading Policy

The performance is determined by the area and delay of your design. The less cost your design has, the higher grade you get.

Function Validity: 70%

Performance: 30% (area: 30%)

Note

1. Please upload the following file on e3 platform before **13:20 at noon** on **Sep. 25**:
   CC_iclab??.v (?? is your account no.)
   Ex: CC_iclab99.v

2. Template folders and reference commands:

In RTL simulation, the name of template folder and reference commands is:

01_RTL:

"**./01_run"**

02_SYN/ (Synthesis):

./01_run_dc

(Check **latch** by searching the keyword "**Latch**" in 02_SYN/**syn.log**)

(Check the design's timing in /Report/CC.timing)

03_GATE_SIM/:

./01_run

You can key in **./09_clean_up** to clear all log files and dump files in each folder

Example Waveform

Input and output signal: