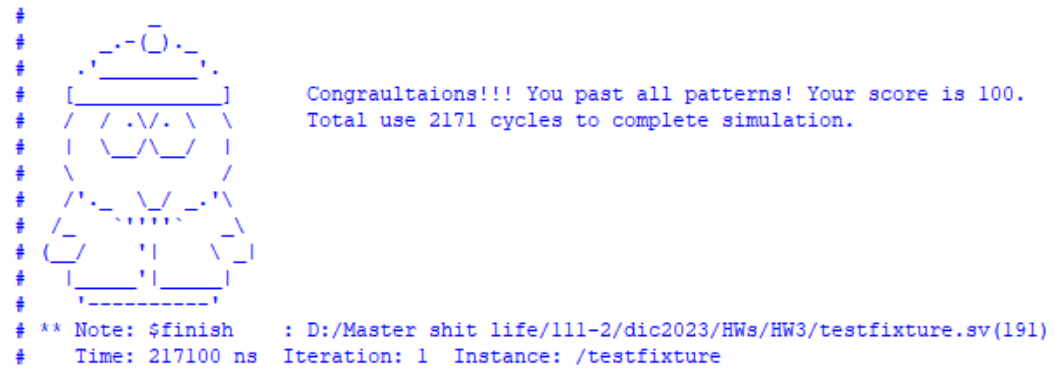
2023 Digital IC Design Homework 3

NAME	何坤霖
Student ID	N26112445

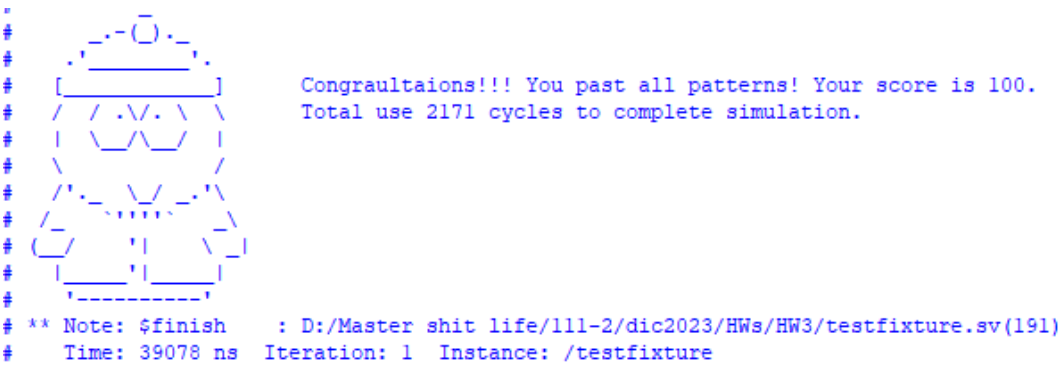
Simulation Result

Functional simulation	100	Gate-level simulation	100

Functional



Gate-level



Synthesis Result

Total logic elements	934
Total memory bits	0
Embedded multiplier 9-bit elements	1
Total cycle used	2171
Clock width	18

Flow Summary

<<Filter>>

Flow Status	Successful - Wed Apr 19 11:23:46 2023
Quartus Prime Version	20.1.1 Build 720 11/11/2020 SJ Lite Edition
Revision Name	AEC
Top-level Entity Name	AEC
Family	Cyclone IV E
Device	EP4CE55F23A7
Timing Models	Final
Total logic elements	934 / 55,856 (2 %)
Total registers	533
Total pins	19 / 325 (6 %)
Total virtual pins	0
Total memory bits	0 / 2,396,160 (0 %)
Embedded Multiplier 9-bit elements	1 / 308 (< 1 %)
Total PLLs	0 / 4 (0 %)

Description of your design

我主要拆分成 5 個 state : DATA_IN, IN2POST, CAL, OUT, IDLE。

Sequential

DATA_IN :

跟 clk 把個個 token 存在 infix_string 直到遇到 8'd61 ('=')

同時累計 infix_id，當狀態切換時(輸入 '=')，把 idx 給 len。

IN2POST :

把存放在 infix_string 的值一筆筆拿出來，遇到數字 token 就直接存放到 postfix_str，其他運算子根據類別區分 priority。

8'd40 ('('): 直接存放在 op_stack，更新 in_idx 以再下一個 clk 讀取 infix_string 數值。

8'd41 ('('): 檢查 op_stack top 位置有沒有 ')' 沒有就繼續 pop，同時更新 post_str_idx ，來求下一個存放位置。

8'd42 ('*'): 檢查 op_stack top 位置有沒有 '*'，沒有就 push 進去 op_stack，有就 pop 出來到 postfix_string。

8'd43 ('+'), 8'd45 ('-'): 除了遇到 '(' 其他都要 pop 出來到 post_str_idx，遇到就把當前 token push 進 op_stack。

也檢查 op_idx 有沒有為 0(op_stack 是否為 empty)，沒有就跟著 clk 一個個 pop 到 postfix_string。

最後 op_idx 為 0 時把 postfix 長度更新上到 post_str_len。

CAL :
取出 postfix_string 的值，遇到數字就轉換再 push 到 cal_stack，如果遇到 operator 就 pop 出上面兩個數值再執行對應的運算，再 push 結果回去，值到 post_str_idx 等於 post_str_len。
OUT :
取出 cal_stack 最下面的值就是運算結果，同時把 valid 設 1 等待 tb 檢查。
IDLE:
把 valid 設回 0，同時 reset 一些數值(result, infix_string，postfix_string, cal_stack, op_stack, inf_str_len, post_str_len, in_idx, op_idx, post_str_idx, cal_idx)。

Next State Logic

初始狀態 DATA_IN，當遇到 8'd61 ('=') 轉成 IN2POST，其他都 DATA_IN。
在 IN2POST 當 (in_idx < in_str_len)，就切到 CAL state，其餘繼續轉。
在 CAL 當 (post_str_idx < post_str_len) 繼續算，其他到 OUT。
OUT 等待 1 cycle，就切到 IDLE, 最後回到 DATA_IN。

*Scoring = Area cost * Timing cost*

*Area cost = Total logic elements + Total memory bits + 9*Embedded multipliers 9-bit elements*

*Timing cost = Total cycle used * Clock width*

*** Total logic elements must not exceed 1500.**