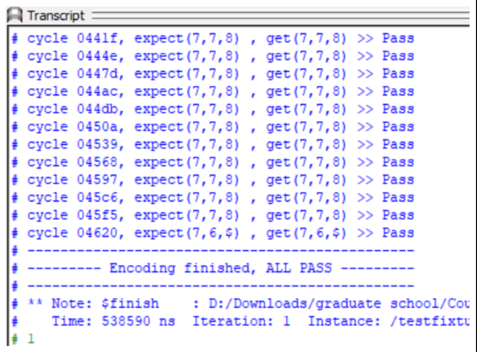
2022 Digital IC Design Homework 3

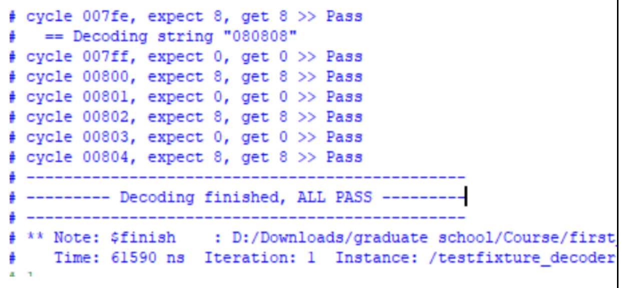
NAME	陳柏維
Student ID	P76101283

Simulation Result

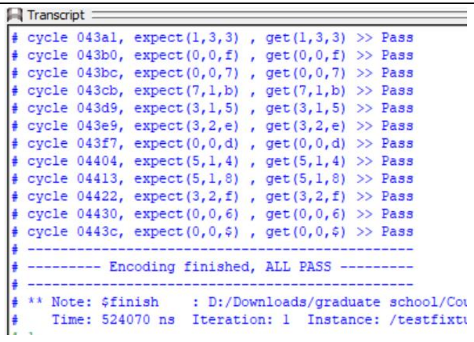
Functional simulation	[✓ Pass] / Fail (encoder)	[✓ Pass] / Fail (decoder)	Gate-level simulation	[✓ Pass] / Fail (encoder)	[✓ Pass] / Fail (decoder)



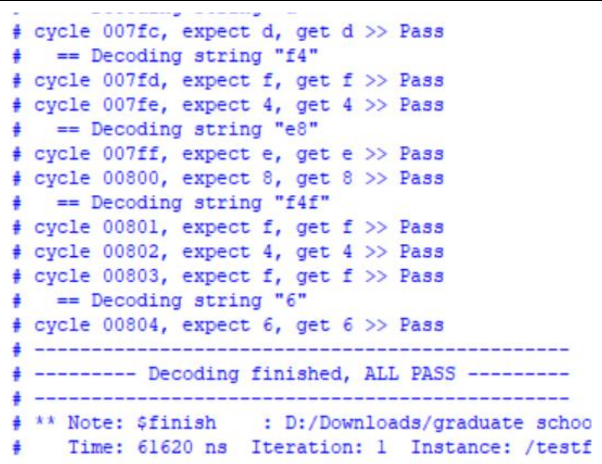
```
Transcript
# cycle 0441f, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 0444e, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 0447d, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 044ac, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 044db, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 0450a, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 04539, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 04568, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 04597, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 045c6, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 045f5, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 04620, expect(7,6,$) , get(7,6,$) >> Pass
# --------------------------------------------------
# --------- Encoding finished, ALL PASS ---------
# --------------------------------------------------
# ** Note: $finish    : D:/Downloads/graduate school/Cou
#    Time: 538590 ns  Iteration: 1  Instance: /testfixtu
# 1
```



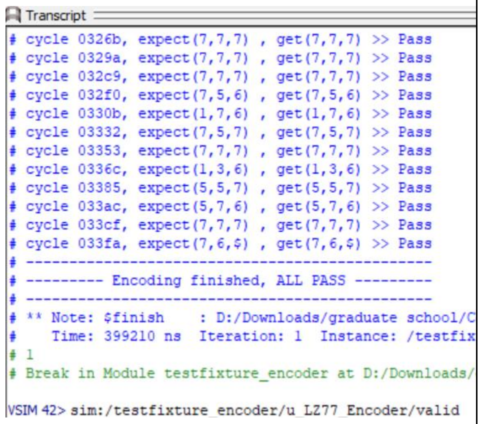
```
# cycle 007fe, expect 8, get 8 >> Pass
#   == Decoding string "080808"
# cycle 007ff, expect 0, get 0 >> Pass
# cycle 00800, expect 8, get 8 >> Pass
# cycle 00801, expect 0, get 0 >> Pass
# cycle 00802, expect 8, get 8 >> Pass
# cycle 00803, expect 0, get 0 >> Pass
# cycle 00804, expect 8, get 8 >> Pass
# --------------------------------------------------
# --------- Decoding finished, ALL PASS ---------
# --------------------------------------------------
# ** Note: $finish    : D:/Downloads/graduate school/Course/first
#    Time: 61590 ns  Iteration: 1  Instance: /testfixture_decoder
```



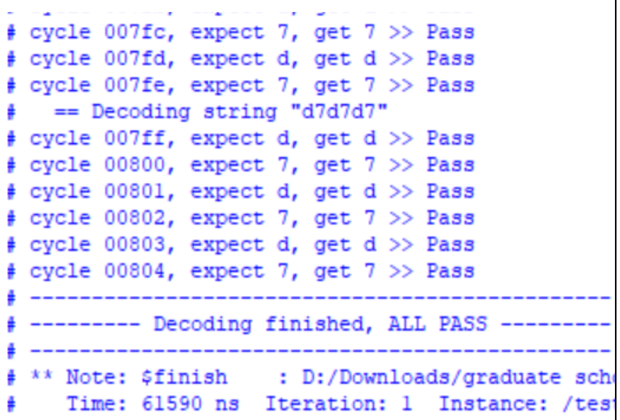
```
Transcript
# cycle 043a1, expect(1,3,3) , get(1,3,3) >> Pass
# cycle 043b0, expect(0,0,f) , get(0,0,f) >> Pass
# cycle 043bc, expect(0,0,7) , get(0,0,7) >> Pass
# cycle 043cb, expect(7,1,b) , get(7,1,b) >> Pass
# cycle 043d9, expect(3,1,5) , get(3,1,5) >> Pass
# cycle 043e9, expect(3,2,e) , get(3,2,e) >> Pass
# cycle 043f7, expect(0,0,d) , get(0,0,d) >> Pass
# cycle 04404, expect(5,1,4) , get(5,1,4) >> Pass
# cycle 04413, expect(5,1,8) , get(5,1,8) >> Pass
# cycle 04422, expect(3,2,f) , get(3,2,f) >> Pass
# cycle 04430, expect(0,0,6) , get(0,0,6) >> Pass
# cycle 0443c, expect(0,0,$) , get(0,0,$) >> Pass
# --------------------------------------------------
# --------- Encoding finished, ALL PASS ---------
# --------------------------------------------------
# ** Note: $finish    : D:/Downloads/graduate school/Cou
#    Time: 524070 ns  Iteration: 1  Instance: /testfixtu
```



```
# cycle 007fc, expect d, get d >> Pass
#   == Decoding string "f4"
# cycle 007fd, expect f, get f >> Pass
# cycle 007fe, expect 4, get 4 >> Pass
#   == Decoding string "e8"
# cycle 007ff, expect e, get e >> Pass
# cycle 00800, expect 8, get 8 >> Pass
#   == Decoding string "f4f"
# cycle 00801, expect f, get f >> Pass
# cycle 00802, expect 4, get 4 >> Pass
# cycle 00803, expect f, get f >> Pass
#   == Decoding string "6"
# cycle 00804, expect 6, get 6 >> Pass
# --------------------------------------------------
# --------- Decoding finished, ALL PASS ---------
# --------------------------------------------------
# ** Note: $finish    : D:/Downloads/graduate schoo
#    Time: 61620 ns  Iteration: 1  Instance: /testf
```



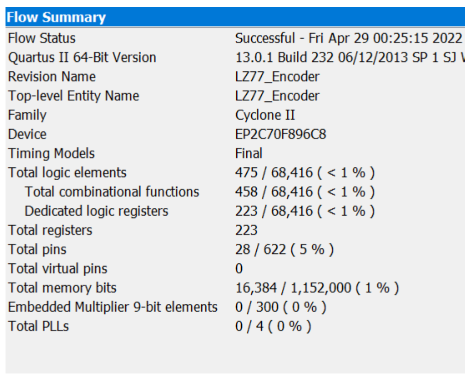
```
Transcript
# cycle 0326b, expect(7,7,7) , get(7,7,7) >> Pass
# cycle 0329a, expect(7,7,7) , get(7,7,7) >> Pass
# cycle 032c9, expect(7,7,7) , get(7,7,7) >> Pass
# cycle 032f0, expect(7,5,6) , get(7,5,6) >> Pass
# cycle 0330b, expect(1,7,6) , get(1,7,6) >> Pass
# cycle 03332, expect(7,5,7) , get(7,5,7) >> Pass
# cycle 03353, expect(7,7,7) , get(7,7,7) >> Pass
# cycle 0336c, expect(1,3,6) , get(1,3,6) >> Pass
# cycle 03385, expect(5,5,7) , get(5,5,7) >> Pass
# cycle 033ac, expect(5,7,6) , get(5,7,6) >> Pass
# cycle 033cf, expect(7,7,7) , get(7,7,7) >> Pass
# cycle 033fa, expect(7,6,$) , get(7,6,$) >> Pass
# --------------------------------------------------
# --------- Encoding finished, ALL PASS ---------
# --------------------------------------------------
# ** Note: $finish    : D:/Downloads/graduate school/C
#    Time: 399210 ns  Iteration: 1  Instance: /testfix
# 1
# Break in Module testfixture_encoder at D:/Downloads/

VSIM 42> sim:/testfixture_encoder/u_LZ77_Encoder/valid
```



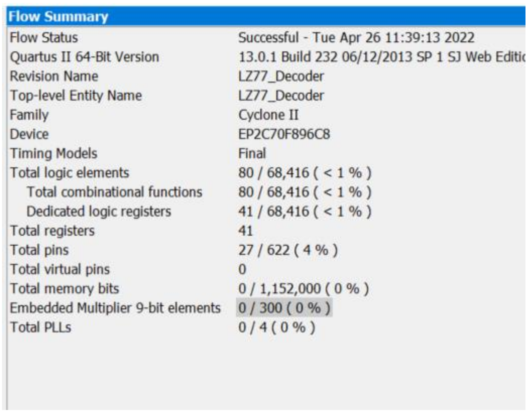
```
# cycle 007fc, expect 7, get 7 >> Pass
# cycle 007fd, expect d, get d >> Pass
# cycle 007fe, expect 7, get 7 >> Pass
#   == Decoding string "d7d7d7"
# cycle 007ff, expect d, get d >> Pass
# cycle 00800, expect 7, get 7 >> Pass
# cycle 00801, expect d, get d >> Pass
# cycle 00802, expect 7, get 7 >> Pass
# cycle 00803, expect d, get d >> Pass
# cycle 00804, expect 7, get 7 >> Pass
# --------------------------------------------------
# --------- Decoding finished, ALL PASS ---------
# --------------------------------------------------
# ** Note: $finish    : D:/Downloads/graduate sch
#    Time: 61590 ns  Iteration: 1  Instance: /tes
```

Synthesis Result	encoder	decoder
Total logic elements	475	80
Total memory bit	16384	0

Embedded multiplier 9-bit element	0	0
Simulation time img0	538590	61590
Simulation time img1	524070	61620
Simulation time img2	399210	61590



Flow Summary	
Flow Status	Successful - Fri Apr 29 00:25:15 2022
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ \
Revision Name	LZ77_Encoder
Top-level Entity Name	LZ77_Encoder
Family	Cyclone II
Device	EP2C70F896C8
Timing Models	Final
Total logic elements	475 / 68,416 (< 1 %)
Total combinational functions	458 / 68,416 (< 1 %)
Dedicated logic registers	223 / 68,416 (< 1 %)
Total registers	223
Total pins	28 / 622 (5 %)
Total virtual pins	0
Total memory bits	16,384 / 1,152,000 (1 %)
Embedded Multiplier 9-bit elements	0 / 300 (0 %)
Total PLLs	0 / 4 (0 %)



Flow Summary	
Flow Status	Successful - Tue Apr 26 11:39:13 2022
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Editic
Revision Name	LZ77_Decoder
Top-level Entity Name	LZ77_Decoder
Family	Cyclone II
Device	EP2C70F896C8
Timing Models	Final
Total logic elements	80 / 68,416 (< 1 %)
Total combinational functions	80 / 68,416 (< 1 %)
Dedicated logic registers	41 / 68,416 (< 1 %)
Total registers	41
Total pins	27 / 622 (4 %)
Total virtual pins	0
Total memory bits	0 / 1,152,000 (0 %)
Embedded Multiplier 9-bit elements	0 / 300 (0 %)
Total PLLs	0 / 4 (0 %)

Description of your design

Encoder:

就等到資料讀取完成之後，才開始做計算。主要 Stage 大概分成 4 個部分

1.Reading 2.Calculate 3.Output 4.Finish 。

Reading : 又分成一開始的資料讀取跟 look-ahead buffer 未滿之前的讀取。

Calculate: 原先是寫一個 function 但是後來不能合成就把它改成一個 clk 依據 sliding windows 的方式來偵測有沒有對到。

Output: 就把剛剛算出來的答案做輸出而已。

Finish: 就結束而已。

Decoder:

我認為沒有甚麼 stages 可言，所以直接用一個 sequential block 來輸出每一個 clk 所產生的 output。並且把提前半個 clk 來的 code_pos,code_len,用一個 wire 存起來，並在 clk 來的時候，針對剛剛所存的做判斷如果 index == code_len 則輸出 chardata[3:0] 否則 輸出之前的 search_buffer[code_pos] 並且 index ++。

*Scoring = (Total logic elements + total memory bit + 9*embedded multiplier 9-bit element)*