2022 Digital IC Design Homework 3

```
劉品宏
NAME
                   P76101699
Student ID
                                            Simulation Result
Functional
                                            Pass
                                                           Gate-level
                        Pass
                                                                                   Pass
                                                                                                       Pass
 simulation
                    (encoder)
                                        (decoder)
                                                           simulation
                                                                                (encoder)
                                                                                                    (decoder)
  (your pre-sim result) encoder/decoder
                                                           (your post-sim result) encoder/decoder
                                                           # cycle 03c12, expect(7,7,8) , get(7,7,8) >> Pass
# cycle 03c23, expect(7,7,8) , get(7,7,8) >> Pass
 # cycle 03c12, expect(7,7,8) , get(7,7,8) >> Pass
 # cycle 03c23, expect(7,7,8) , get(7,7,8) >> Pass
 # cycle 03c4b, expect(7,6,$) , get(7,6,$) >> Pass
                                                            # cycle 03c4b, expect(7,6,$) , get(7,6,$) >> Pass
  ----- Encoding finished, ALL PASS -----
                                                             ----- Encoding finished, ALL PASS -----
                                                           # ** Note: $finish : D:/altera/13.0spl/modelsim
# Time: 463080 ns Iteration: 1 Instance: /tes
 # ** Note: $finish : D:/Downloads/HW3/DIC HW3/t
    Time: 463080 ns Iteration: 1 Instance: /tes
  | cycle 00802, expect 8, get 8 >> Pass
                                                            # cycle 00802, expect 8, get 8 >> Pass
| cycle 00803, expect 0, get 0 >> Pass
                                                           # cycle 00803, expect 0, get 0 >> Pass
t cycle 00804, expect 8, get 8 >> Pass
                                                           # cycle 00804, expect 8, get 8 >> Pass
  ----- Decoding finished, ALL PASS -
                                                              ----- Decoding finished, ALL PASS
** Note: $finish
                          : D:/Downloads/HW3,
                                                             ** Note: $finish : D:/altera/13.0sp
      Time: 63060 ns Iteration: 1 Insta:
                                                                  Time: 63060 ns Iteration: 1 Insta
 (your pre-sim result) encoder/decoder
                                                            (your post-sim result) encoder/decoder
                       img1
                                                                                  img1
   # cycle 03c40, expect(3,2,f) , get(3,2,f) >> Pass
                                                          \# cycle 03c40, expect(3,2,f) , get(3,2,f) >> Pass
   # cycle 03c4a, expect(0,0,6) , get(0,0,6) >> Pass
# cycle 03c54, expect(0,0,$) , get(0,0,$) >> Pass
                                                          # cycle 03c4a, expect(0,0,6) , get(0,0,6) >> Pass # cycle 03c54, expect(0,0,$) , get(0,0,$) >> Pass
    ----- Encoding finished, ALL PASS ------
                                                            ----- Encoding finished, ALL PASS ------
    ** Note: $finish : D:/Downloads/HW3/DIC_HW3/t
Time: 463350 ns Iteration: 1 Instance: /tes
                                                          # ** Note: $finish : D:/altera/13.0spl/modelsim
# Time: 463350 ns Iteration: 1 Instance: /tes
                                                          # cycle 00802, expect 4, get 4 >> Pass
         cycle 00802, expect 4, get 4 >> Pass
                                                            cycle 00803, expect f, get f >> Pass
        : cycle 00803, expect f, get f >> Pass
: == Decoding string "6"
             Decoding string
                                                              == Decoding string "6
         cycle 00804, expect 6, get 6 >> Pass
                                                          # cycle 00804, expect 6, get 6 >> Pass
          ----- Decoding finished, ALL PASS
                                                            ----- Decoding finished, ALL PASS
        ** Note: $finish : D:/Downloads/HW3
                                                            ** Note: $finish : D:/altera/13.0sg
            Time: 61620 ns Iteration: 1 Insta-
 (your pre-sim result) encoder/decoder
                                                            (your post-sim result) encoder/decoder
  # cycle 024c5, expect(5,7,6) , get(5,7,6) >> Pass
                                                             # cycle 024c5, expect(5,7,6) , get(5,7,6) >> Pass
  # cycle 024d6, expect(7,7,7) , get(7,7,7) >> Pass
# cycle 024fe, expect(7,6,$) , get(7,6,$) >> Pass
                                                               cycle 024d6, expect(7,7,7) , get(7,7,7) >> Pass cycle 024d6, expect(7,7,7) , get(7,7,7) >> Pass cycle 024fe, expect(7,6,$) , get(7,6,$) >> Pass
  # ----- Encoding finished, ALL PASS ------
                                                               ----- Encoding finished, ALL PASS ---
  # ** Note: $finish : D:/Downloads/HW3/DIC_HW3/t
# Time: 284130 ns Iteration: 1 Instance: /tes
                                                             # ** Note: $finish : D:/altera/13.0spl/modelsim
# Time: 284130 ns Iteration: 1 Instance: /tes
     # cycle 00802, expect 7, get 7 >> Pass
                                                              # cycle 00802, expect 7, get 7 >> Pass
     # cycle 00803, expect d, get d >> Pass
                                                               # cycle 00803, expect d, get d >> Pass
     # cycle 00804, expect 7, get 7 >> Pass
                                                               # cycle 00804, expect 7, get 7 >> Pass
            ---- Decoding finished, ALL PASS
                                                               # ----- Decoding finished, ALL PASS
       ** Note: $finish
                             : D:/Downloads/HW3
                                                               # ** Note: $finish : D:/altera/13.0sp
          Time: 63060 ns Iteration: 1 Insta
                                                                    Time: 63060 ns Iteration: 1 Insta
```

Synthesis Result	encoder	decoder
Total logic elements	4412	87
Total memory bit	12228	0
Embedded multiplier 9-bit element	0	0
Simulation time img0	463080 (ns)	63060 (ns)
Simulation time img1	463350 (ns)	61620 (ns)
Simulation time img2	284130 (ns)	63060 (ns)

(your flow summary) encoder

Flow Status Successful - Thu Apr 21 15:32:12 2022 Ouartus II 64-Bit Version 13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition Revision Name LZ77 Encoder Top-level Entity Name LZ77_Encoder Cyclone II Device EP2C70F896C8 Timing Models Final 4,412 / 68,416 (6 %) Total logic elements Total combinational functions 318 / 68,416 (< 1 %) Dedicated logic registers 4,273 / 68,416 (6 %) Total registers Total pins 28 / 622 (5%) Total virtual pins Total memory bits 12,228 / 1,152,000 (1 %) Embedded Multiplier 9-bit elements 0/300(0%) Total PLLs 0/4(0%)

(your flow summary) decoder

Successful - Thu Apr 21 21:09:12 2022 Flow Status Quartus II 64-Bit Version 13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition Revision Name 1777 Decoder Top-level Entity Name LZ77 Decoder Device EP2C70F896C8 Timing Models Total logic elements 87 / 68,416 (< 1 %) Total combinational functions 54 / 68,416 (< 1 %) Dedicated logic registers 76 / 68,416 (< 1 %) Total registers 27 / 622 (4 %) Total pins Total virtual pins 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

在我的 encoder 設計中,分成 4 個狀態,第一個是接收資料,再來的狀態是找到第一個符合的字元,如果有找到,那接著的狀態就是比對接下來的字元是否有符合,最後才是輸出結果,那在我 function 的測試過了之後,進行合成時,發生了不少問題,因為最主要的問題是面積太大,大於 68416,然後去請教實驗室的學長,學長有指出我的 code 有哪些問題,然後有教我如何改正,所以就下修到 5 萬多,之後我又改了一些架構,雖然造成的 cycle 數更多了,但是我的面積又縮小更多了。

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