

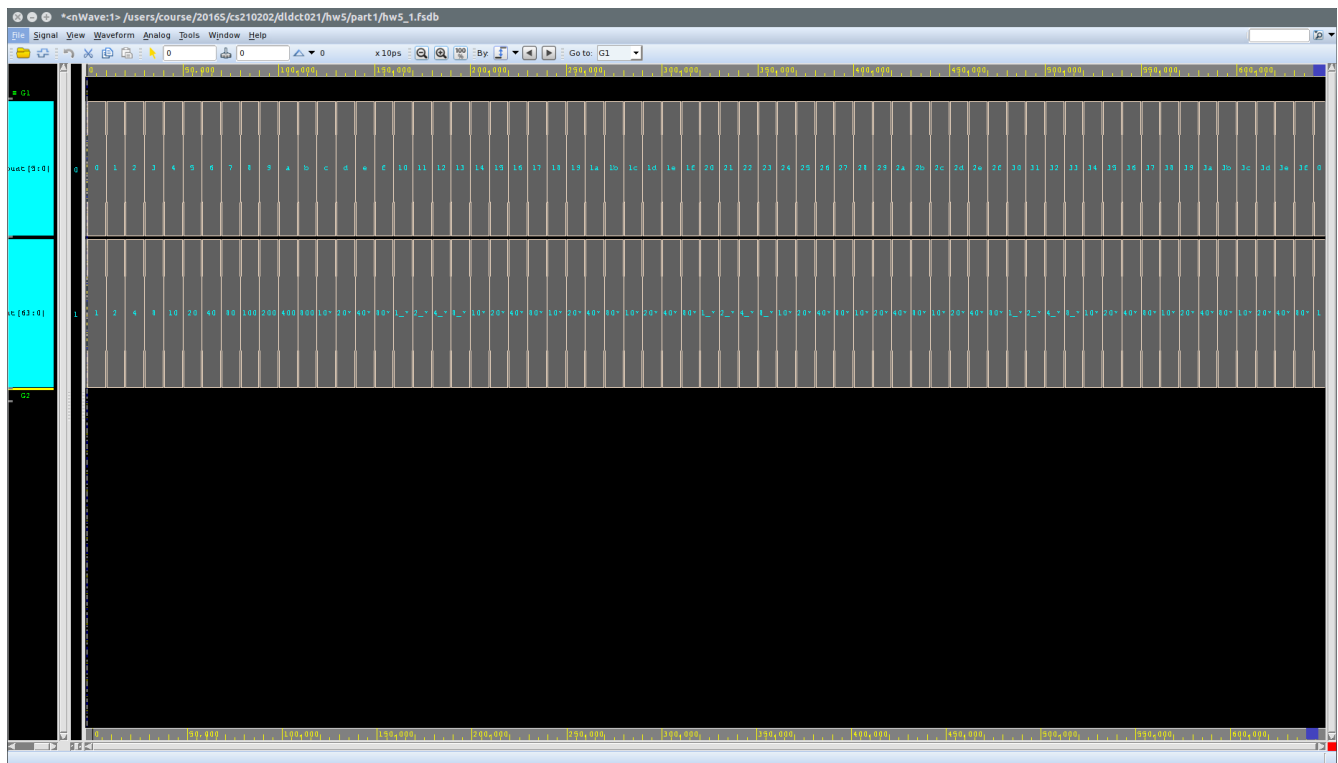
(a)

因為 decode 結果只有 64 個，而且又是 one-hot，所以 testbench 直接列出 64 種可能，而且非常容易檢查程式的正確性。主程式原本想要使用 for 迴圈，不過仔細想想可以利用 Verilog bitwise & 就可以快速組合出 output 了，其實不需要跑兩層迴圈。

Simulation result:

[illegible]

Waveform:



2.

(a)

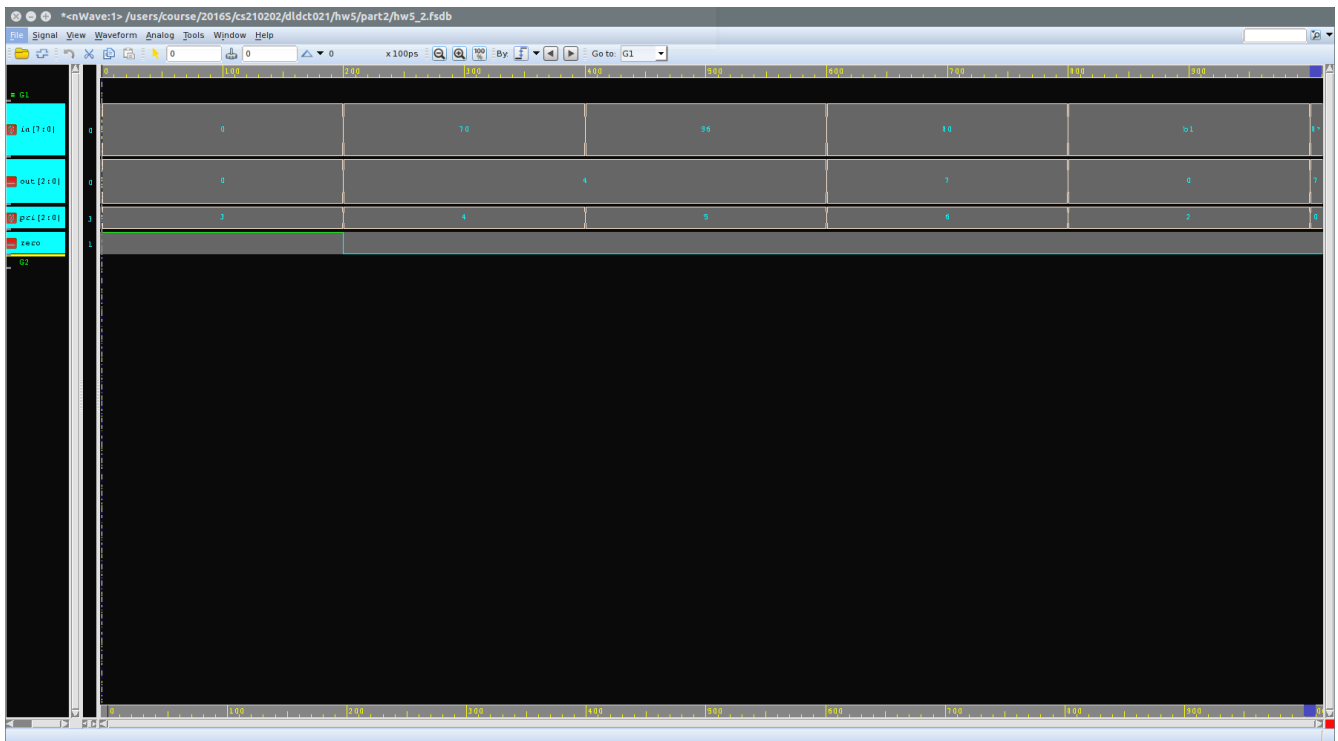
這題的 testbench 其實只需要測四種就行了：全為零， $pri == out$ ， $pri > out$ ， $pri < out$ ，因為主程式的輸出結果就只有以上那四種。至於其他組的 testcase，只是多放幾組來驗證程式輸出的穩定性。我的做法是仿照講義上的作法，是由優先序小的開始找，如果可以找到更高的優先序，則把輸出取代成新的。不過一開始在寫的時候有邏輯上的小錯誤，是因為初始條件要從  $pri+1$  開始，最後到  $pri+8 \pmod{8} = pri \pmod{8}$ ，才會讓  $pri$  位置優先序最高，而不能從  $pri$  開始。

(b)

Simulation result:

```
Terminal
[dldct021@ic28 part2]$ make
ncverilog hw5_2_test.v hw5_2.v +access+r
ncverilog: 14.10-s005; (c) Copyright 1995-2014 Cadence Design Systems, Inc.
Loading snapshot worklib.hw5_2_test.v ..... Done
*Verdi3* Loading libsscore_ius141.so
*Verdi3* : Enable Parallel Dumping.
ncsim> source /usr/cad/cadence/INCISIV/cur/tools/inca/files/ncsimrc
ncsim> run
FSDB Dumper for IUS, Release Verdi3_J-2014.12-SP3, Linux, 07/05/2015
(C) 1996 - 2015 by Synopsys, Inc.
*Verdi3* FSDB WARNING: The FSDB file already exists. Overwriting the FSDB file may crash the programs that are using this file.
*Verdi3* : Create FSDB file 'hw5_2.fsd'
*Verdi3* : Begin traversing the scopes, layer (0).
*Verdi3* : End of traversing.
in=00000000, pri=3; out=0, zero=1
in=01110000, pri=4; out=4, zero=0
in=10010110, pri=5; out=4, zero=0
in=10000000, pri=6; out=7, zero=0
in=10110001, pri=2; out=0, zero=0
in=10000100, pri=0; out=7, zero=0
ncsim: *W,RNQUIE: Simulation is complete.
ncsim> exit
[dldct021@ic28 part2]$
```

Wave form:



3.

(a)

這一題的 testbench 選法其實和上一題差不多，因為是由兩組 decoder 所組合的輸出，所以需要八組測試，而每個需要測到使用一個 decoder 輸出，和兩個 decoder 輸出的結果。其他組 testcase，也只是用來驗證程式的穩定性。主程式的部份，因為有 zero 的幫助，所以可以讓主程式判斷更簡潔。分配優先順序的方法就是，依照 decoder 有沒有 pri 的位置，若沒有，則優先序從 3'b111(=7)開始找，這樣可以保證輸出是從左向右搜尋的。再做一些邏輯判斷以後，就可以組合出輸出的答案了。

(b)

Simulation result:

```
Terminal
Building instance overlay tables: ..... Done
Building instance specific data structures.
Loading native compiled code: ..... Done
Design hierarchy summary:
      Instances  Unique
Modules:         4      3
Registers:       10     7
Scalar wires:     3     -
Vectored wires:  11     -
Always blocks:    3     2
Initial blocks:   3     3
Cont. assignments: 2     2
Pseudo assignments: 6     6
Simulation timescale: 100ps
Writing initial simulation snapshot: worklib.hw5_3_test.v
Loading snapshot worklib.hw5_3_test.v ..... Done
*Verdi3* Loading libsscore ius141.so
*Verdi3* : Enable Parallel Dumping.
ncsim> source /usr/cad/cadence/INCISIV/cur/tools/inca/files/ncsimrc
ncsim> run
FSDb Dumper for IUS, Release Verdi3_J-2014.12-SP3, Linux, 07/05/2015
(C) 1996 - 2015 by Synopsys, Inc.
*Verdi3* FSDb WARNING: The FSDb file already exists. Overwriting the FSDb file may crash the programs that are using this file.
*Verdi3* : Create FSDb file 'hw5_3.fsd'
*Verdi3* : Begin traversing the scopes, layer (0).
*Verdi3* : End of traversing.
in=0000000000000000, pri= 1; out= 0, zero=1
in=0000000000000000, pri=11; out= 0, zero=1
in=0000000110100001, pri= 8; out= 8, zero=0
in=1000000100000000, pri= 9; out= 8, zero=0
in=0100000000000000, pri=10; out=14, zero=0
in=1000000000000000, pri=14; out=15, zero=0
in=1001001001100100, pri= 2; out= 2, zero=0
in=1000010001010001, pri= 3; out= 0, zero=0
in=0010010111000000, pri= 4; out=13, zero=0
in=0000101001000000, pri= 5; out=11, zero=0
in=0000000010000000, pri= 6; out= 7, zero=0
ncsim: *W,RNQUIE: Simulation is complete.
ncsim> exit
[dldct021@ic28 part3]$
```

Wave form:

