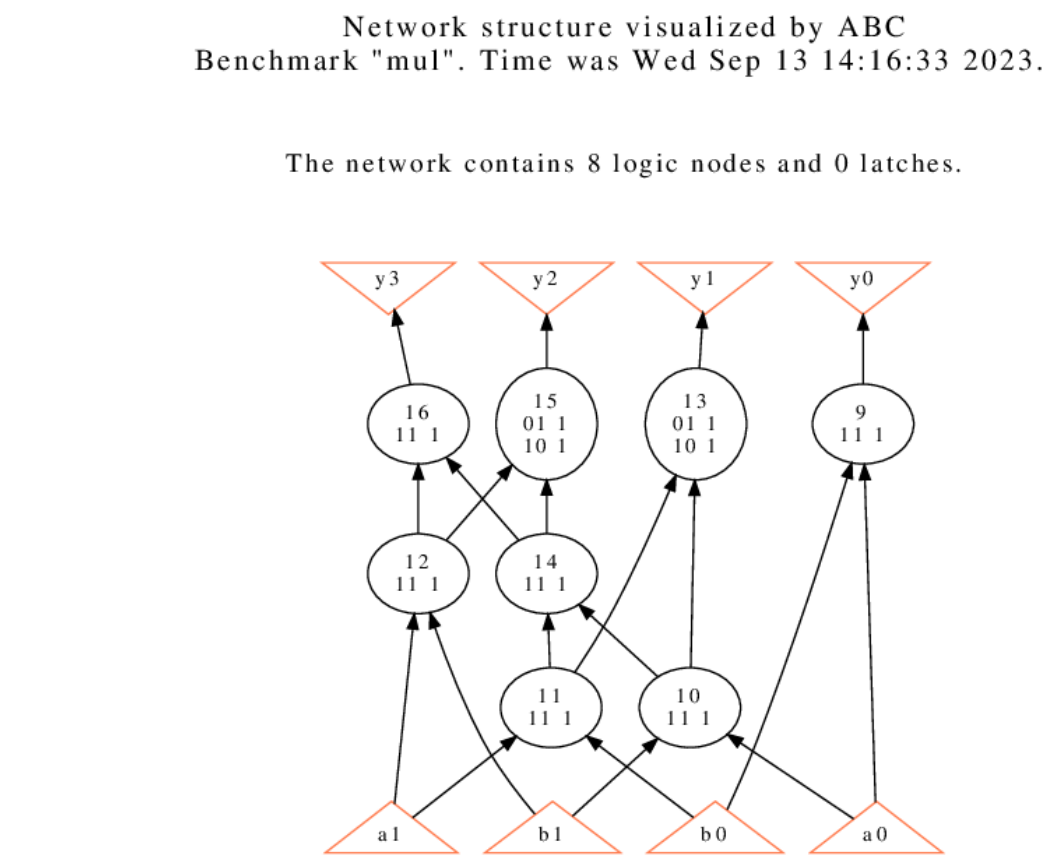
r11943098 電子所 EDA 組碩二 洪子祐 PA1

(二)

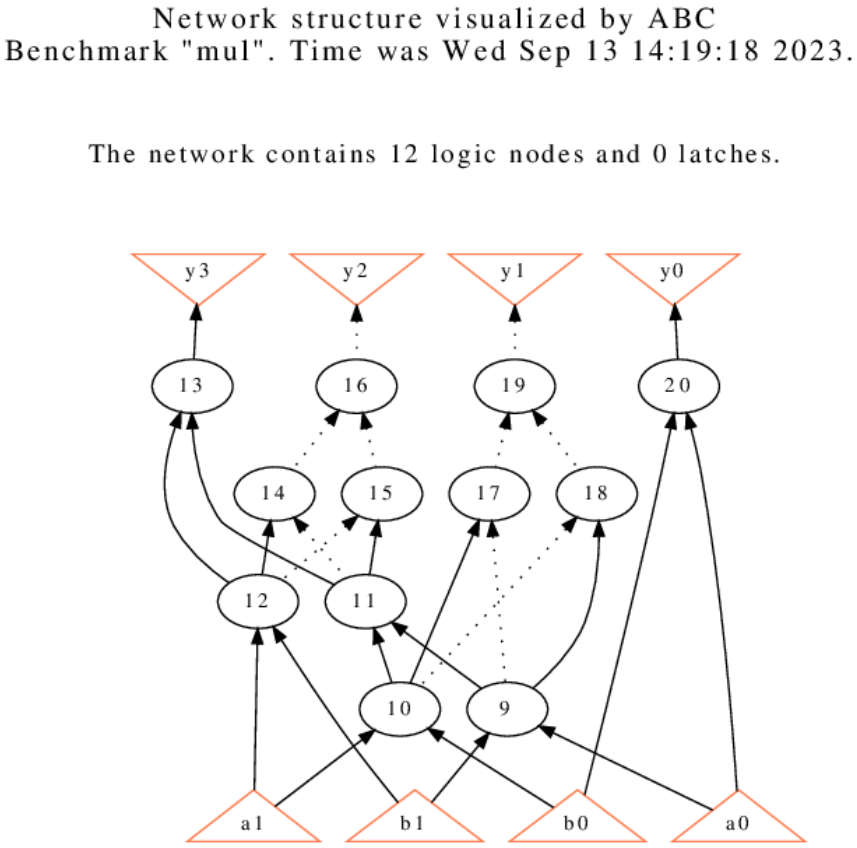
1.~3.

```
UC Berkeley, ABC 1.01 (compiled Sep 11 2023 02:15:39)
================== Command history ==================
show_bdd
readd lsv/pa1/mul.blif
source -s abc.rc
read lsv/pa1/mul.blif
print_stats
strash
show
collapse
show_bdd -g
clear
=====================================================
abc 01> read lsv/pa1/mul.blif
Hierarchy reader flattened 2 instances of logic boxes and left 0 black boxes.
abc 02> print_stats
mul                           : i/o =    4/    4  lat =    0  nd =     8  edge =     16  cube =    10  lev = 3
abc 02> show
abc 02> sh: 1: gv: not found

abc 02>
```



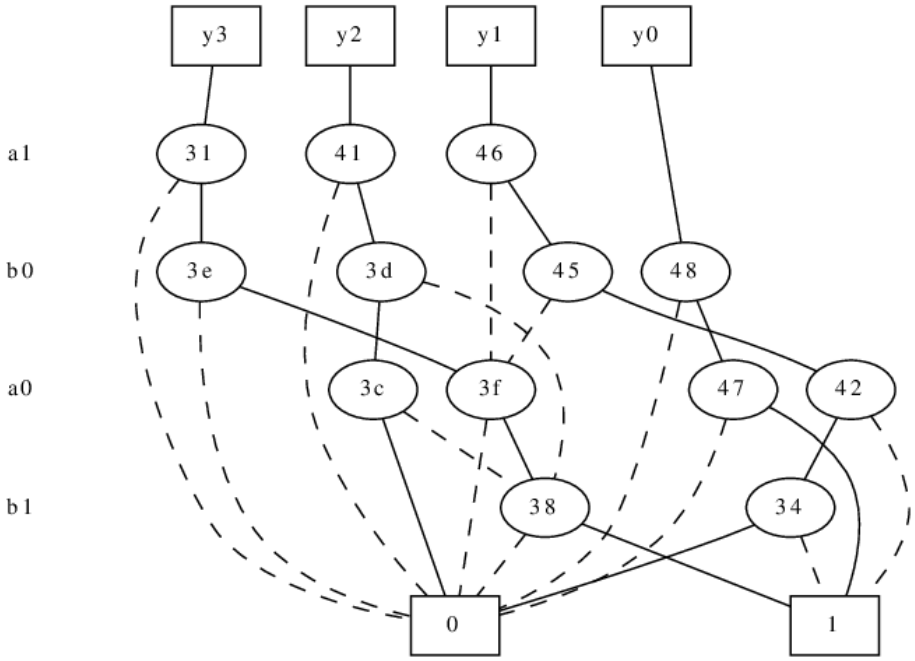
4.~5.

```
================================================
abc 01> read lsv/pa1/mul.blif
Hierarchy reader flattened 2 instances of logic boxes and left 0 black boxes.
abc 02> print_stats
mul                         : i/o =    4/    4  lat =    0  nd =     8  edge =     16  cube =    10  lev = 3
abc 02> show
abc 02> sh: 1: gv: not found

abc 02> strash
abc 03> show
abc 03> sh: 1: gv: not found

abc 03>
```



6.~7.

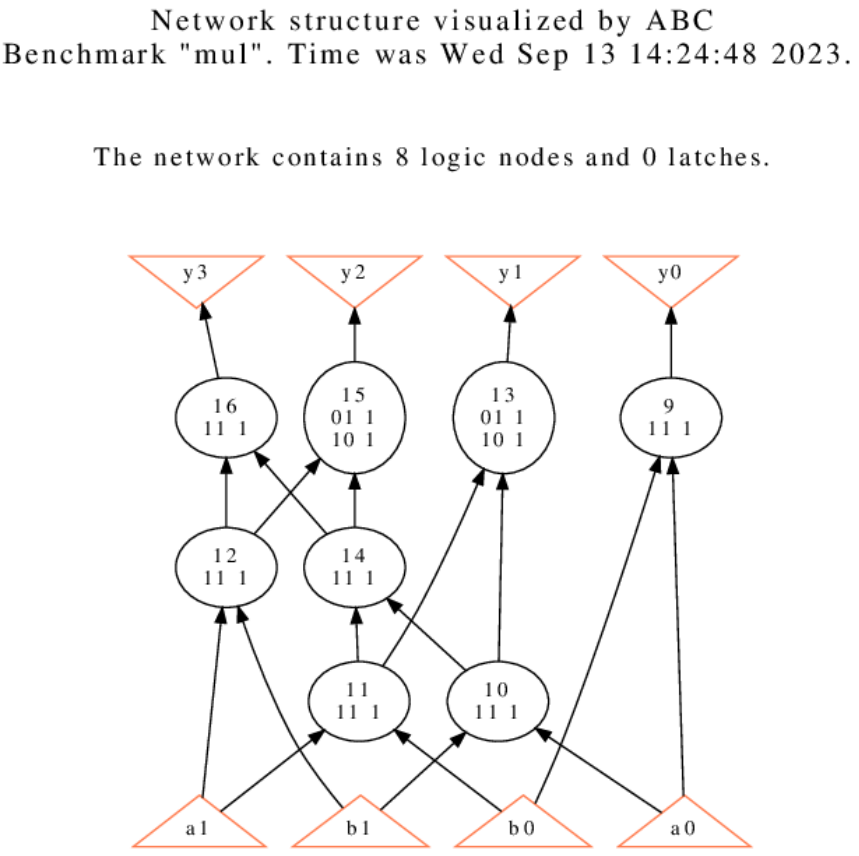
```
abc 03> show
abc 03> sh: 1: gv: not found

abc 03> collapse
abc 04> show_bdd -g
abc 04> sh: 1: gv: not found

abc 04>
```


y3
y2
y1
y0
a1
31
41
46
b0
3e
3d
45
48
a0
3c
3f
47
42
b1
38
34
0
1

(三)

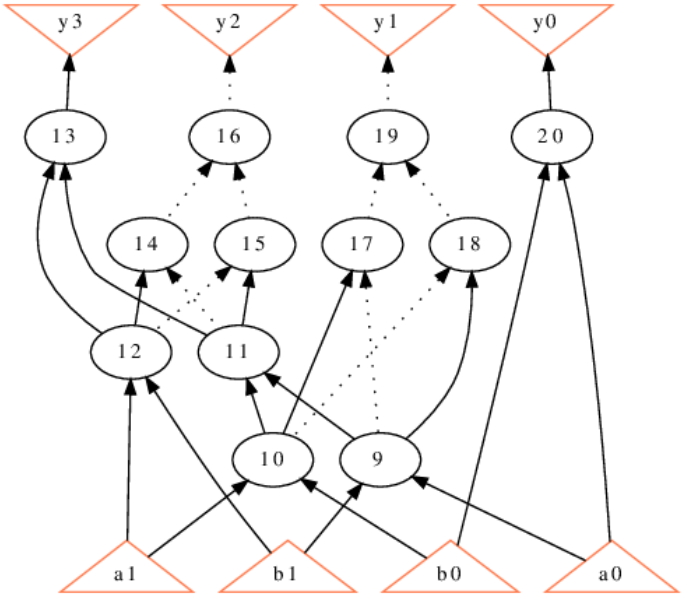
(a)

(1)


Network structure visualized by ABC
Benchmark "mul". Time was Wed Sep 13 14:24:48 2023.
The network contains 8 logic nodes and 0 latches.
y3
y2
y1
y0
16
11 1
15
01 1
10 1
13
01 1
10 1
9
11 1
12
11 1
14
11 1
11
11 1
10
11 1
a1
b1
b0
a0

Network structure visualized by ABC
Benchmark "mul". Time was Wed Sep 13 14:27:11 2023.

The network contains 12 logic nodes and 0 latches.

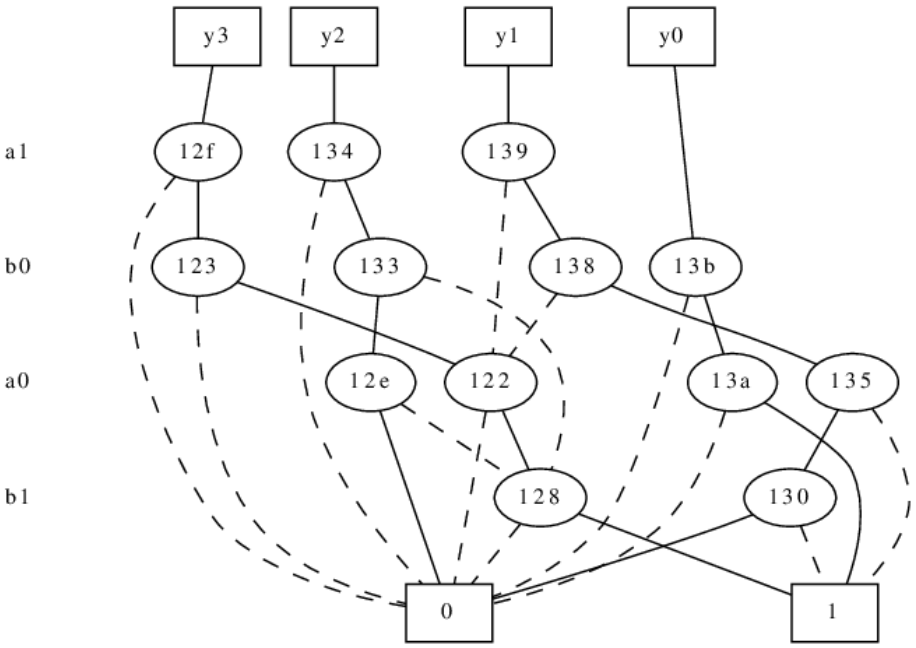


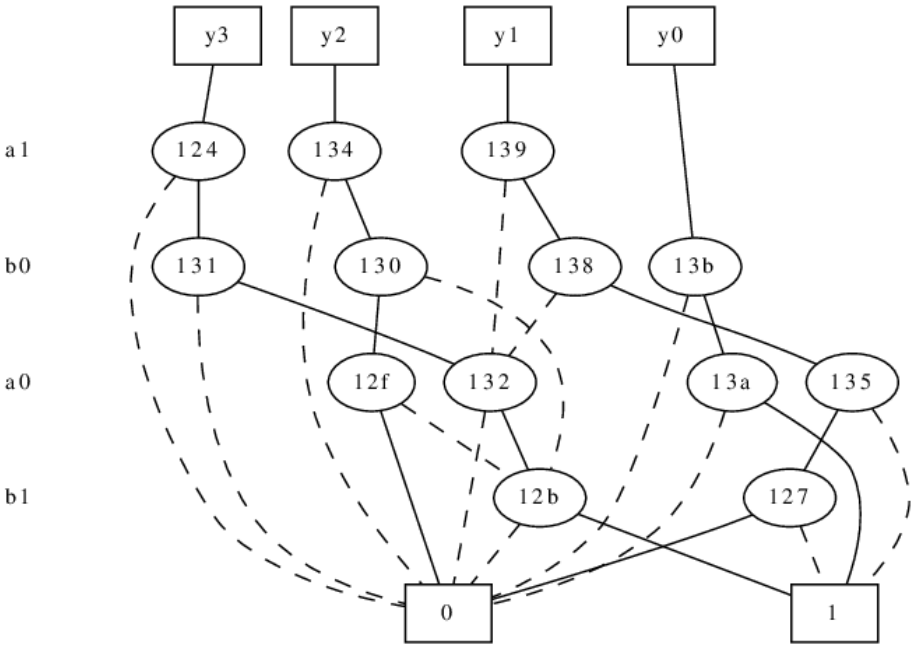

```
================================================
abc 01> read lsv/pa1/mul.blif
Hierarchy reader flattened 2 instances of logic boxes and left 0 black boxes.
abc 02> print_stats
mul                         : i/o =    4/    4  lat =    0  nd =     8  edge =     16  cube =    10  lev = 3
abc 02> aig
abc 02> print_stats
mul                         : i/o =    4/    4  lat =    0  nd =     8  edge =     16  aig  =    12  lev = 3
abc 02> show
abc 02> sh: 1: gv: not found

abc 02> strash
abc 03> print_stats
mul                         : i/o =    4/    4  lat =    0  and =     12  lev =  4
abc 03> show
abc 03> sh: 1: gv: not found

abc 03> aig
Error: Converting to AIG is possible only for logic networks.
abc 03>
```

Find:

剛 aig 完的時候，show 的結果不會改變，但 print_stats 會出現差異。代表實際上是同一種資料架構(LN)，但預設功能不同，故顯示資料時給出資訊會不同。另外，Strash 後 show 的結果會變成真正的 AIG，不能直接 aig 化。

(2)


y3
y2
y1
y0
a1
12f
134
139
b0
123
133
138
13b
a0
12e
122
13a
135
b1
128
130
0
1




```
abc 03> read lsv/pa1/mul.blif
Hierarchy reader flattened 2 instances of logic boxes and left 0 black boxes.
abc 04> print_stats
mul                       : i/o =    4/    4  lat =    0  nd =     8  edge =     16  cube =    10  lev = 3
abc 04> bdd
abc 04> print_stats
mul                       : i/o =    4/    4  lat =    0  nd =     8  edge =     16  bdd  =    16  lev = 3
abc 04> show_bdd -g
abc 04> sh: 1: gv: not found

abc 04> collapse
abc 05> print_stats
mul                       : i/o =    4/    4  lat =    0  nd =     4  edge =     14  bdd  =    14  lev = 1
abc 05> show_bdd -g
abc 05> sh: 1: gv: not found

abc 05> bdd
abc 05> print_stats
mul                       : i/o =    4/    4  lat =    0  nd =     4  edge =     14  bdd  =    14  lev = 1
abc 05> aig
abc 05> bdd
abc 05> print_stats
mul                       : i/o =    4/    4  lat =    0  nd =     4  edge =     14  bdd  =    14  lev = 1
abc 05>
```

Find:

可以看到，這次的 show_bdd -g 結果相同，而且還可以轉回去。

而從 print_stats 的結果可以發現，collapse 後 bdd 計數下降了，代表有實際去做了 BDD 圖的縮減，而這代表了邏輯簡化，因此哪怕 aig 再 bdd，也無法再次得到 bdd=16 的狀態了。

另外，雖然 show_bdd 的結果相同，但號碼不同，這也能代表說這些自己生成的 bdd node，本身並不是相同的。而是有重新計算並化撿得到的結果。

(b)

Strash -> SOP can done by :

collapse

sop