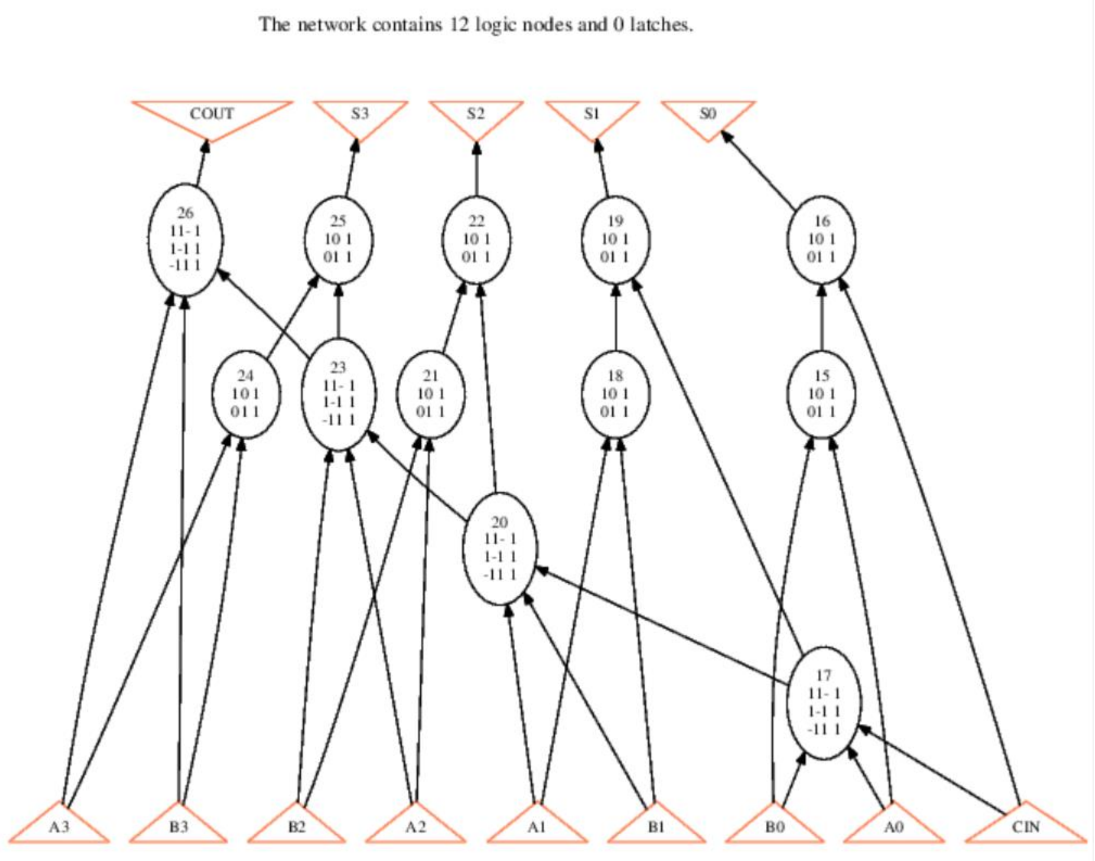
LSV-PA1 r09943146 王星蘋

PART 1

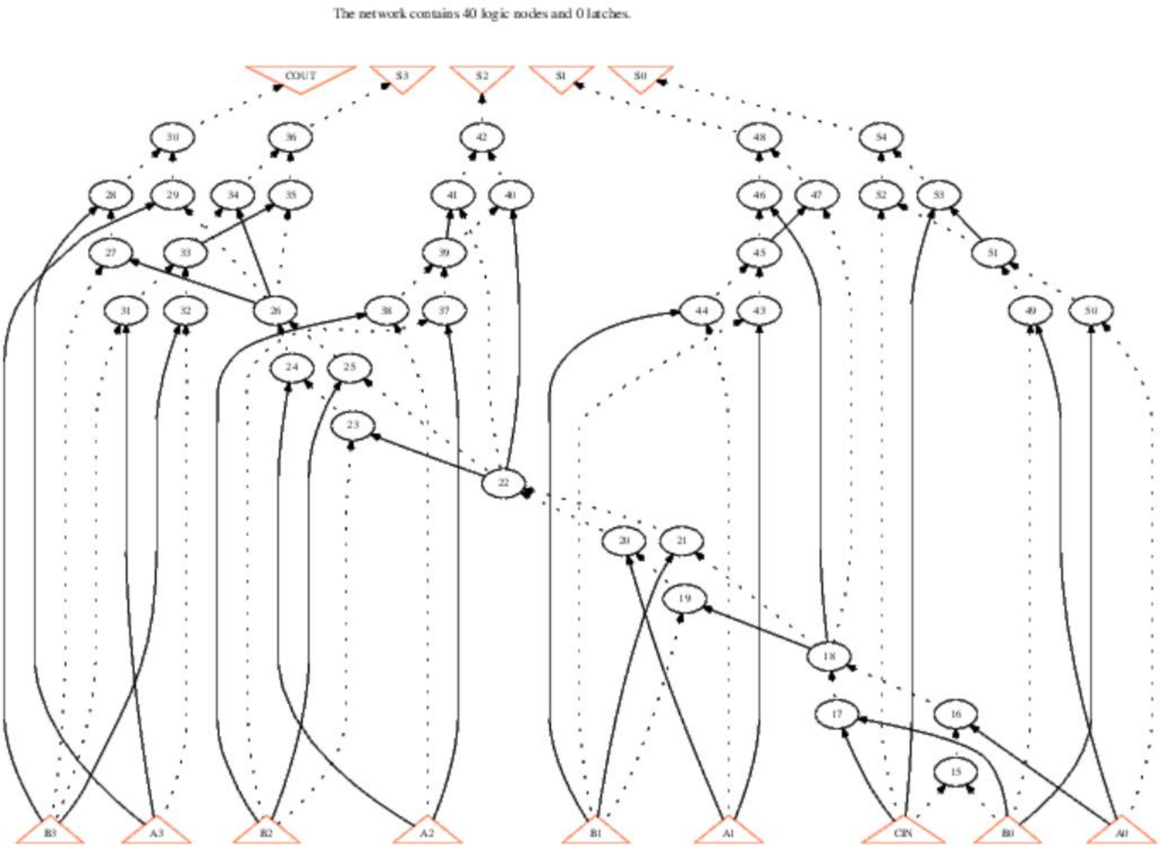
(a) The blif file is named 4bitadder.blif and put in the same directory.

(b) Results of show and show_bdd:

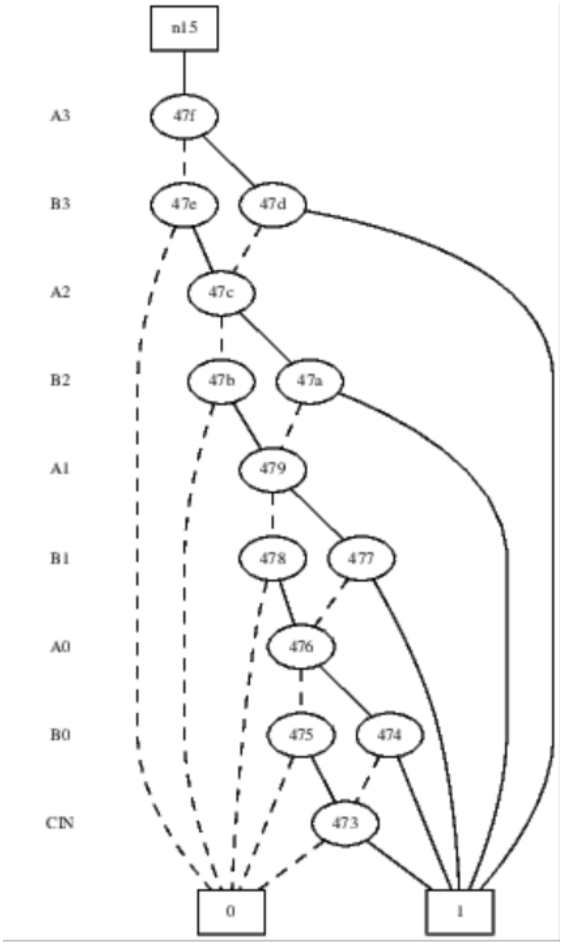
Step 3. (show)



Step 5.(show, after strash)



Step 7.(show_bdd, it’s the result of the first PO, COUT.)



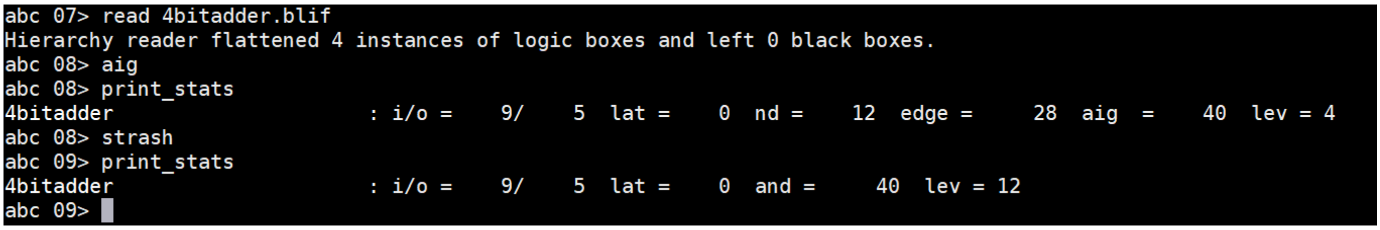
PART 2

(a1) logic network in AIG vs. structurally hashed AIG

command “aig” : converts node functions to AIG.

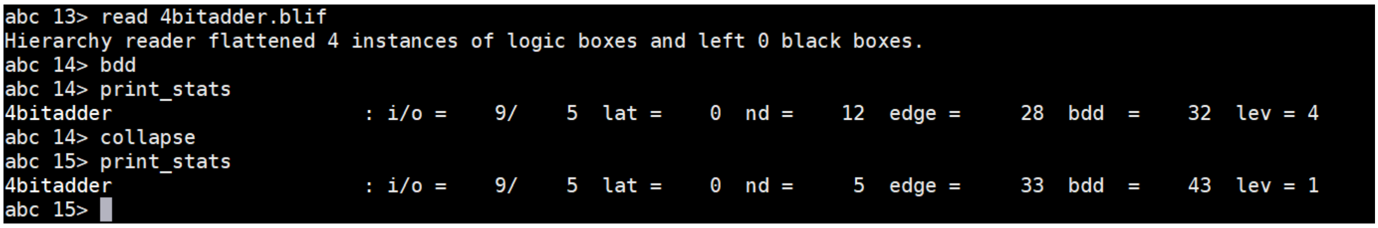
command “strash”: Do structural hashing and transform combinational logic into an AIG.

Take 4-bit full adder as an example, show the status by command “print_stats”:



```
abc 07> read 4bitadder.blif
Hierarchy reader flattened 4 instances of logic boxes and left 0 black boxes.
abc 08> aig
abc 08> print_stats
4bitadder                     : i/o =    9/    5  lat =    0  nd =    12  edge =     28  aig  =    40  lev = 4
abc 08> strash
abc 09> print_stats
4bitadder                     : i/o =    9/    5  lat =    0  and =     40  lev = 12
abc 09> 
```

(a2)



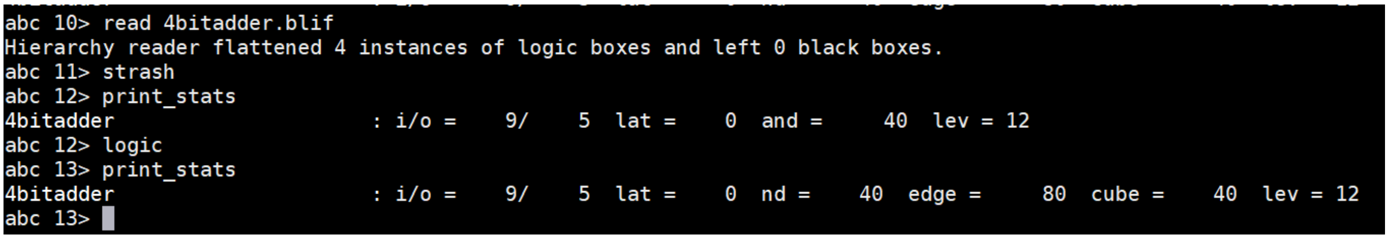
```
abc 13> read 4bitadder.blif
Hierarchy reader flattened 4 instances of logic boxes and left 0 black boxes.
abc 14> bdd
abc 14> print_stats
4bitadder                     : i/o =    9/    5  lat =    0  nd =    12  edge =     28  bdd  =    32  lev = 4
abc 14> collapse
abc 15> print_stats
4bitadder                     : i/o =    9/    5  lat =    0  nd =     5  edge =     33  bdd  =    43  lev = 1
abc 15> 
```

command “bdd”: converts node functions to bdd.

command “collapse”: collapses the network by constructing global BDD.

Take 4-bit full adder as an example, command “collapse” constructs global function to BDD. (global function: inputs=A0,A1,A2,A3,B0,B1,B2,B3,CIN. output=COUT,s0,s1,s2,s3) While command “bdd” constructs local function to BDD. (local function: inputs=a, b, cin. output=s, cout)

(b) command “logic” can transform an AIG into logic network with SOPs



```
abc 10> read 4bitadder.blif
Hierarchy reader flattened 4 instances of logic boxes and left 0 black boxes.
abc 11> strash
abc 12> print_stats
4bitadder                    : i/o =    9/    5  lat =    0  and =     40  lev = 12
abc 12> logic
abc 13> print_stats
4bitadder                    : i/o =    9/    5  lat =    0  nd =    40  edge =     80  cube =    40  lev = 12
abc 13> 
```