

Logic Synthesis & Verification, Fall 2024

National Taiwan University

Report of Programming Assignment 1

電子所碩一 R13943074 楊珩

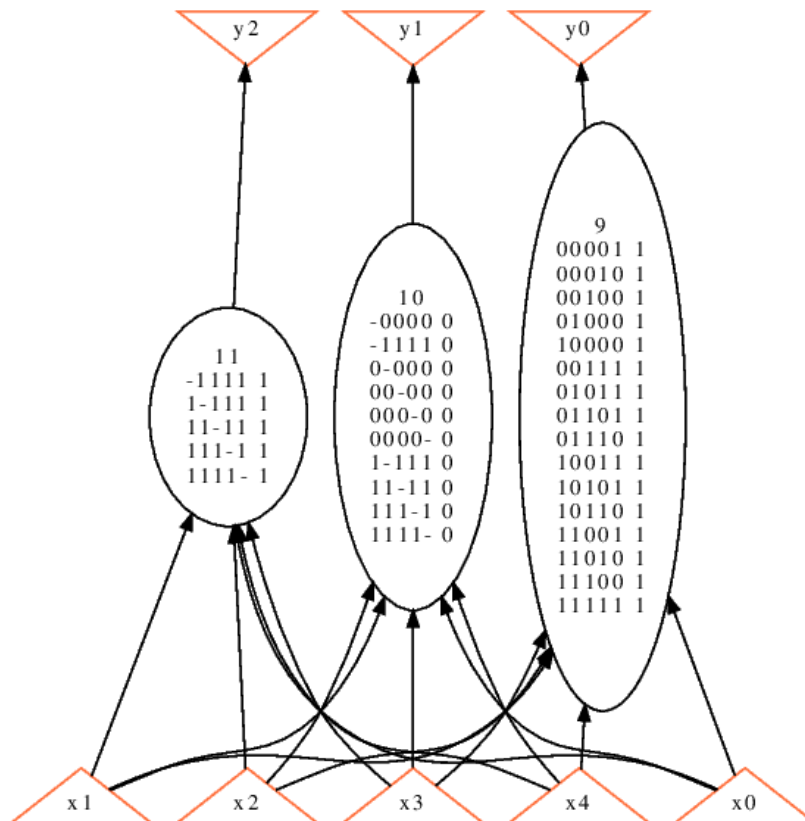
2 [Using ABC]

1. read the BLIF file into ABC (command “read”)
2. check statistics (command “print stats”)

```
abc 01> read ./lsv/pa1/comp.blif
abc 02> print_stats
comp          : i/o =   5/   3  lat =   0  nd =   3  edge =   15  cube =   31  lev =   1
```

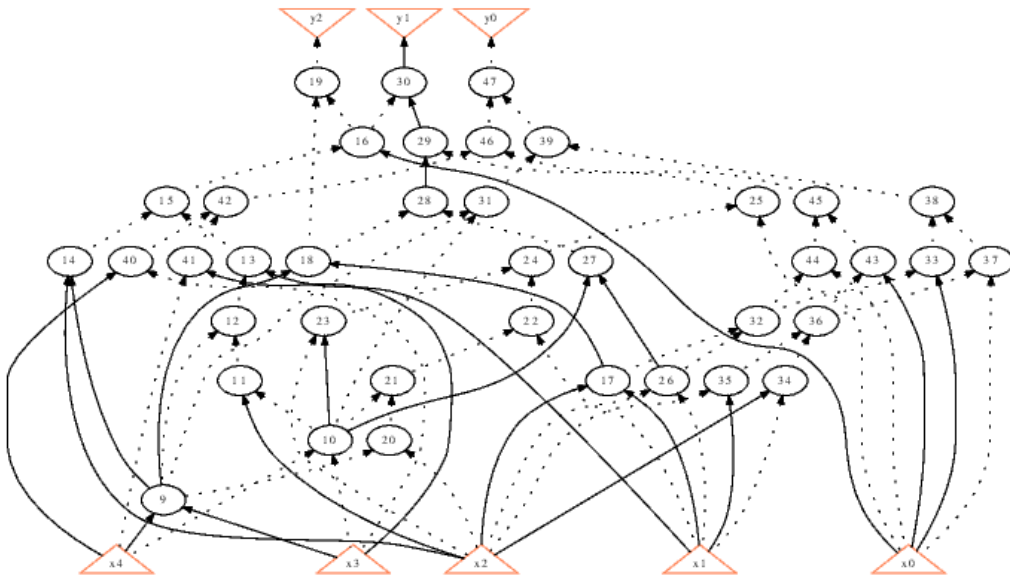
3. visualize the network structure (command “show”)

The network contains 3 logic nodes and 0 latches.

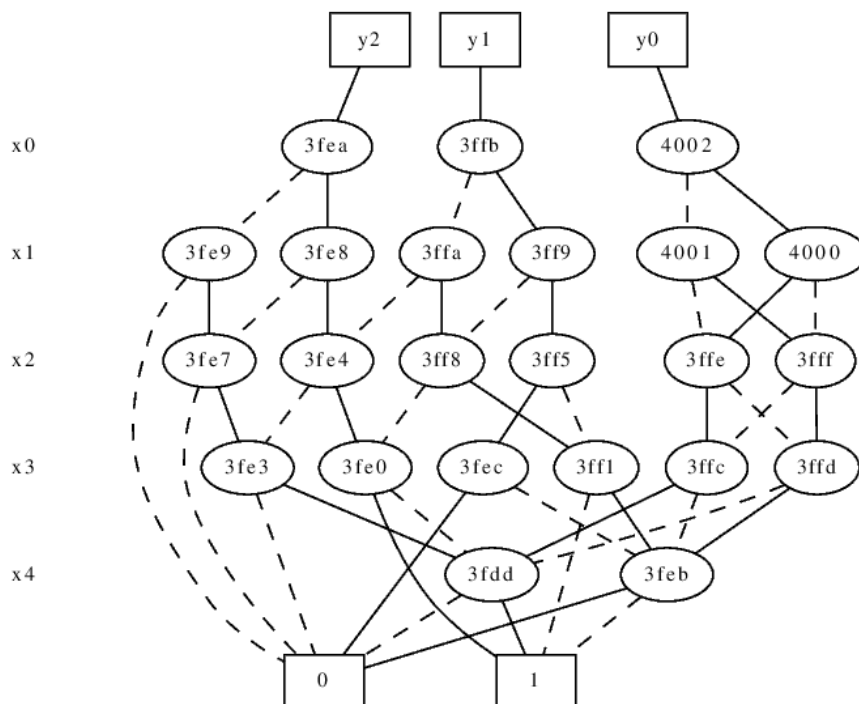


4. convert to AIG (command “strash”)
5. visualize the AIG (command “show”)

The network contains 39 logic nodes and 0 latches.



6. convert to BDD (command “collapse”)
7. visualize the BDD (command “show bdd -g”; note that “show bdd” only shows the first PO; option “-g” can be applied to show all POs)

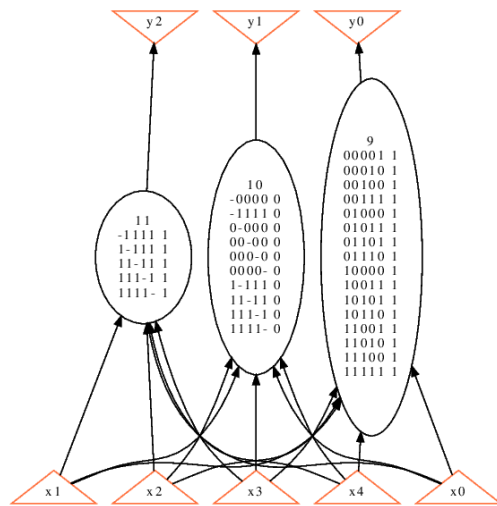


3 [ABC Boolean Function Representations] (a)

1. logic network in AIG (by command “aig”) vs. structurally hashed AIG (by command “strash”)

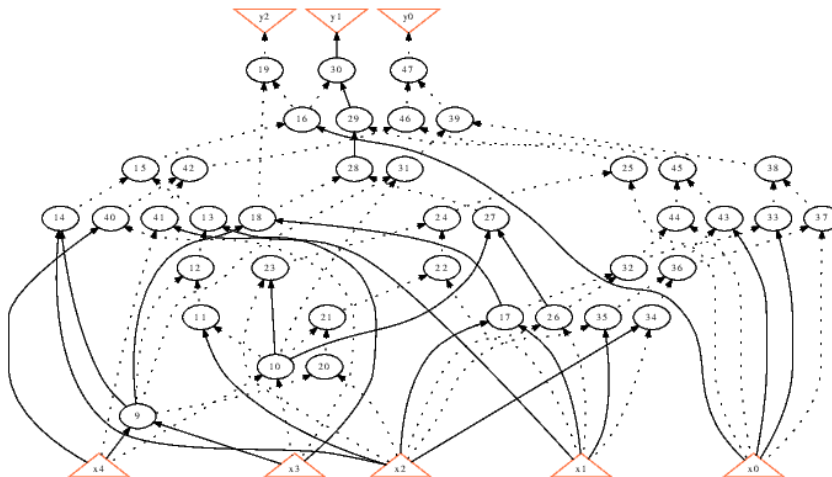
```
abc 05> aig
abc 05> ps
comp : i/o = 5/ 3 lat = 0 nd = 3 edge = 15 aig = 53 lev = 1
```

The network contains 3 logic nodes and 0 latches.



```
abc 05> strash
abc 06> ps
comp : i/o = 5/ 3 lat = 0 and = 39 lev = 8
```

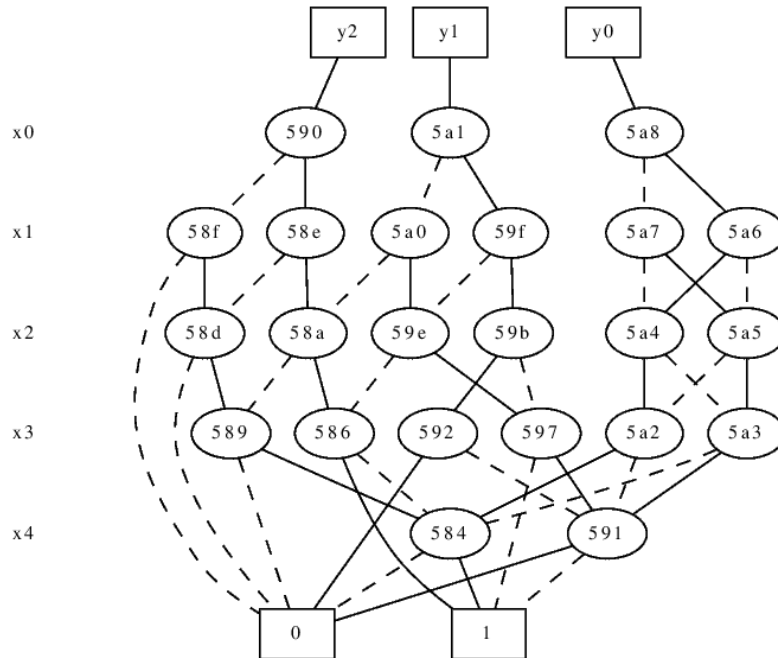
The network contains 39 logic nodes and 0 latches.



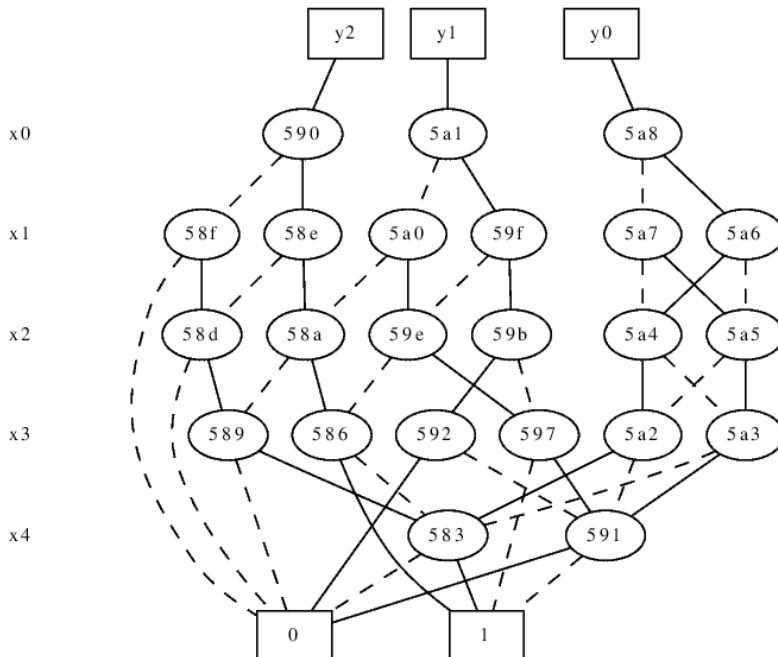
Thoughts: The number of aig is 53 after command “aig”, which means there are 53 AND gates in total. However, there are extra AND gates in node 9, 10, 11 and they don’t show how they interpret in each node. Instead, the command “strash” (global structural hash) convert the whole circuit to aig, which has 39 AND gates in total.

2. logic network in BDD (by command “bdd”) vs. collapsed BDD (by command “collapse”)

```
abc 09> bdd
abc 09> ps
comp      : i/o = 5/ 3 lat = 0 nd = 3 edge = 15 bdd = 21 lev = 1
```



```
abc 11> collapse
abc 12> ps
comp      : i/o = 5/ 3 lat = 0 nd = 3 edge = 15 bdd = 21 lev = 1
```



Thoughts: The collapsed BDD simplifies the representation of the Boolean function, making the network more efficient. However, for the comp.blif they look the same.

3 [ABC Boolean Function Representations] (b)

Read file -> strash -> logic

```
abc 12> read ./lsv/pa1/comp.blif
abc 13> strash
abc 14> logic
abc 15> ps
comp : i/o = 5/ 3 lat = 0 nd = 39 edge = 78 cube = 39 lev = 8
```

The network contains 39 logic nodes and 0 latches.

