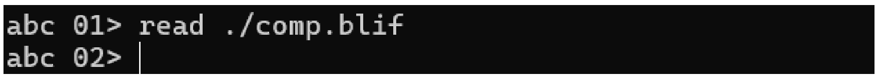
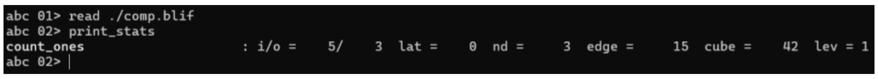
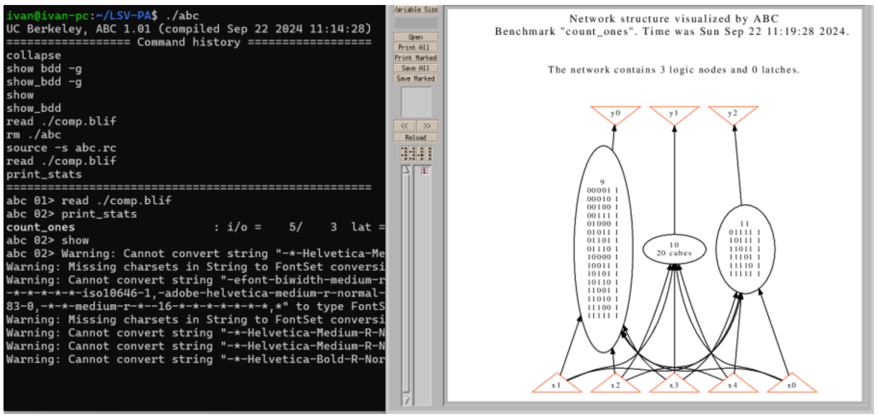
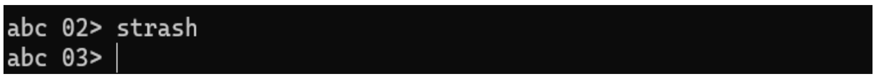
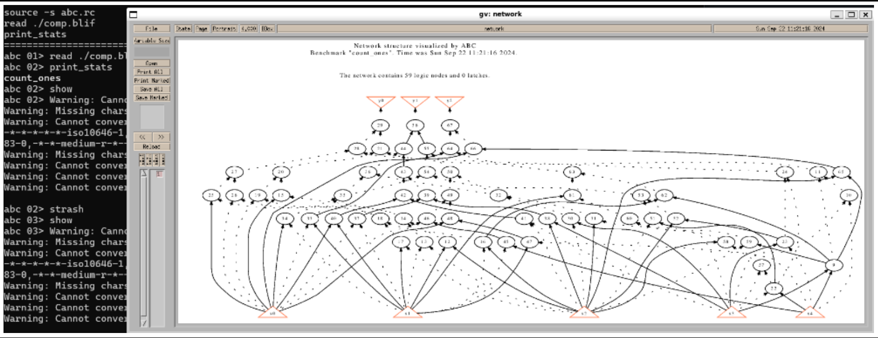
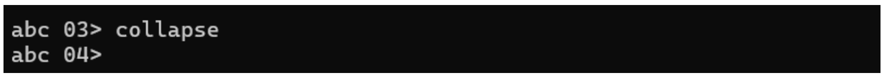
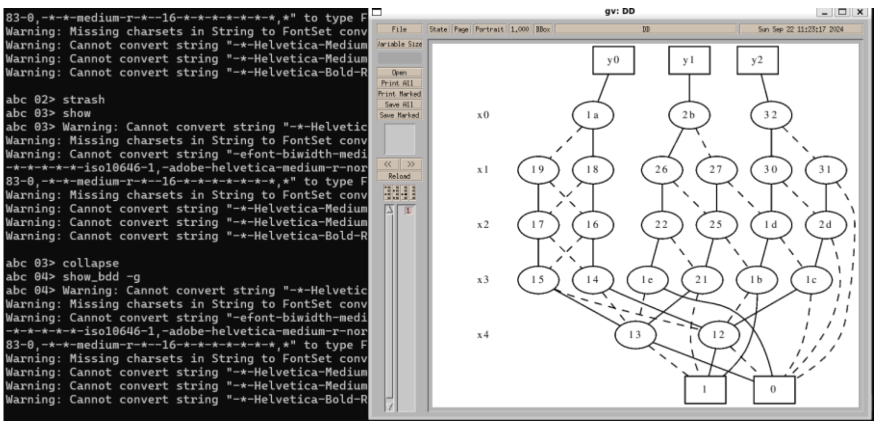
Logic Synthesis & Verification, Fall 2024

Programming Assignment 1

R13922153 林奕帆

1. Exercise 2

Steps	Screenshots
1. read the BLIF into ABC	
2. check statistics	
3. visualize the network structure	
4. convert to AIG	
5. visualize the AIG	
6. convert to BDD	
7. visualize the BDD	

```
abc 01> read ./comp.blif
abc 02>
```

```
abc 01> read ./comp.blif
abc 02> print_stats
count_ones                    : i/o =    5/    3  lat =    0  nd =     3  edge =     15  cube =     42  lev = 1
abc 02>
```

```
ivan@ivan-pc:~/LSV-PA$ ./abc
UC Berkeley, ABC 1.01 (compiled Sep 22 2024 11:14:28)
================== Command history ===================
collapse
show bdd -g
show_bdd -g
show
show_bdd
read ./comp.blif
rm ./abc
source -s abc.rc
read ./comp.blif
print_stats
=====================================================
abc 01> read ./comp.blif
abc 02> print_stats
count_ones                    : i/o =    5/    3  lat =
abc 02> show
abc 02> Warning: Cannot convert string "-*-Helvetica-Me
Warning: Missing charsets in String to FontSet conversi
Warning: Cannot convert string "-efont-biwidth-medium-r
-*-*-*-*-*-iso10646-1,-adobe-helvetica-medium-r-normal-
83-0,-*-*-medium-r-*--16-*-*-*-*-*-*-*,*" to type FontS
Warning: Missing charsets in String to FontSet conversi
Warning: Cannot convert string "-*-Helvetica-Medium-R-N
Warning: Cannot convert string "-*-Helvetica-Medium-R-N
Warning: Cannot convert string "-*-Helvetica-Bold-R-Nor
```

Network structure visualized by ABC
Benchmark "count_ones". Time was Sun Sep 22 11:19:28 2024.

The network contains 3 logic nodes and 0 latches.

```
abc 02> strash
abc 03>
```

```
source -s abc.rc
read ./comp.blif
print_stats
=======================
abc 01> read ./comp.bl
abc 02> print_stats
count_ones
abc 02> show
abc 02> Warning: Canno
Warning: Missing chars
Warning: Cannot conver
-*-*-*-*-*-iso10646-1,
83-0,-*-*-medium-r-*--
Warning: Missing chars
Warning: Cannot conver
Warning: Cannot conver
Warning: Cannot conver

abc 02> strash
abc 03> show
abc 03> Warning: Canno
Warning: Missing chars
Warning: Cannot conver
-*-*-*-*-*-iso10646-1,
83-0,-*-*-medium-r-*--
Warning: Missing chars
Warning: Cannot conver
Warning: Cannot conver
Warning: Cannot conver
```

Network structure visualized by ABC
Benchmark "count_ones". Time was Sun Sep 22 11:21:16 2024.

The network contains 59 logic nodes and 0 latches.

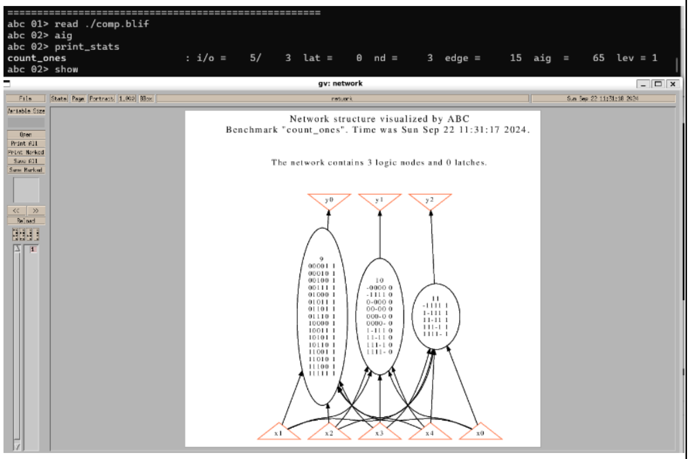
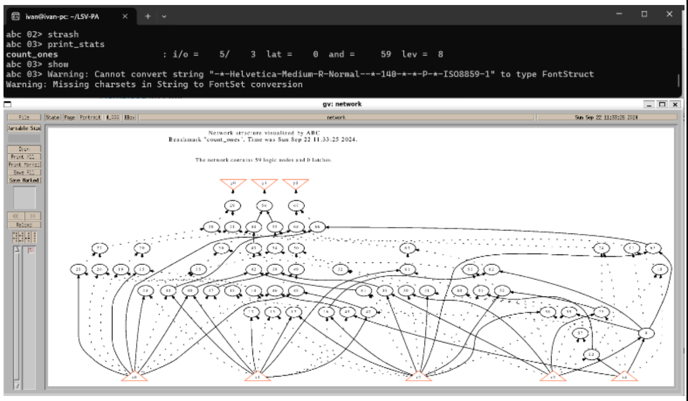
```
abc 03> collapse
abc 04>
```

```
83-0,-*-*-medium-r-*--16-*-*-*-*-*-*-*,*" to type F
Warning: Missing charsets in String to FontSet conv
Warning: Cannot convert string "-*-Helvetica-Medium
Warning: Cannot convert string "-*-Helvetica-Medium
Warning: Cannot convert string "-*-Helvetica-Bold-R

abc 02> strash
abc 03> show
abc 03> Warning: Cannot convert string "-*-Helvetic
Warning: Missing charsets in String to FontSet conv
Warning: Cannot convert string "-efont-biwidth-medi
-*-*-*-*-*-iso10646-1,-adobe-helvetica-medium-r-nor
83-0,-*-*-medium-r-*--16-*-*-*-*-*-*-*,*" to type F
Warning: Missing charsets in String to FontSet conv
Warning: Cannot convert string "-*-Helvetica-Medium
Warning: Cannot convert string "-*-Helvetica-Medium
Warning: Cannot convert string "-*-Helvetica-Bold-R

abc 03> collapse
abc 04> show_bdd -g
abc 04> Warning: Cannot convert string "-*-Helvetic
Warning: Missing charsets in String to FontSet conv
Warning: Cannot convert string "-efont-biwidth-medi
-*-*-*-*-*-iso10646-1,-adobe-helvetica-medium-r-nor
83-0,-*-*-medium-r-*--16-*-*-*-*-*-*-*,*" to type F
Warning: Missing charsets in String to FontSet conv
Warning: Cannot convert string "-*-Helvetica-Medium
Warning: Cannot convert string "-*-Helvetica-Medium
Warning: Cannot convert string "-*-Helvetica-Bold-R
```

2. Exercise 3

(a)

(1) The structurally hashed AIG has more levels(8 instead of 1), more edges(above 20 instead of 15), less gates(59 instead of 65) and same number of latches(both 0) compared to the origin logic network in AIG.

AIG	structurally hashed AIG
	

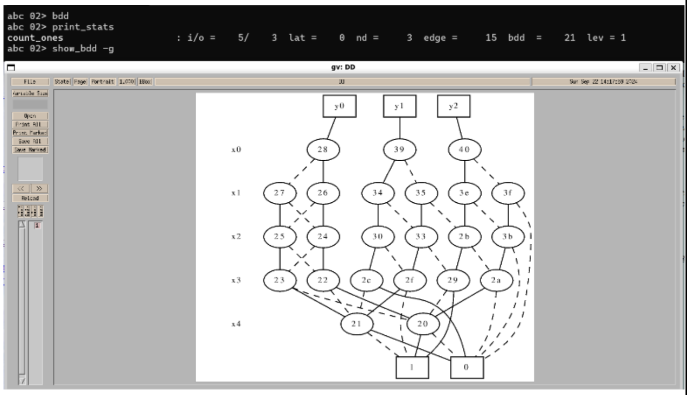
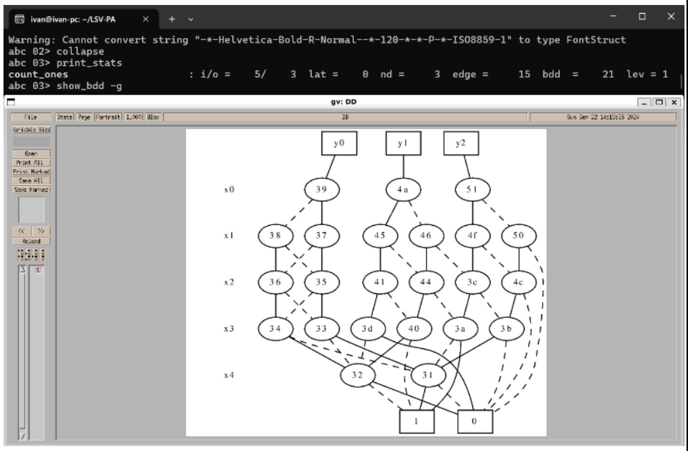
```
==================================================
abc 01> read ./comp.blif
abc 02> aig
abc 02> print_stats
count_ones                    : i/o =    5/    3  lat =    0  nd =     3  edge =     15  aig  =    65  lev = 1
abc 02> show
```

gv: network
Network structure visualized by ABC
Benchmark "count_ones". Time was Sun Sep 22 11:31:17 2024.
The network contains 3 logic nodes and 0 latches.

```
abc 02> strash
abc 03> print_stats
count_ones                    : i/o =    5/    3  lat =    0  and =     59  lev =  8
abc 03> show
abc 03> Warning: Cannot convert string "-*-Helvetica-Medium-R-Normal--*-140-*-*-P-*-ISO8859-1" to type FontStruct
Warning: Missing charsets in String to FontSet conversion
```

gv: network
Network structure visualized by ABC
Benchmark "count_ones". Time was Sun Sep 22 11:33:25 2024.
The network contains 59 logic nodes and 0 latches.

(2) The collapsed BDD has the same stats and also the same structure with the origin logic network in BDD, I think the reason is that the "collapse" command produce one BDD for each primary output, which coincides the way I implement my "comp.blif" file.

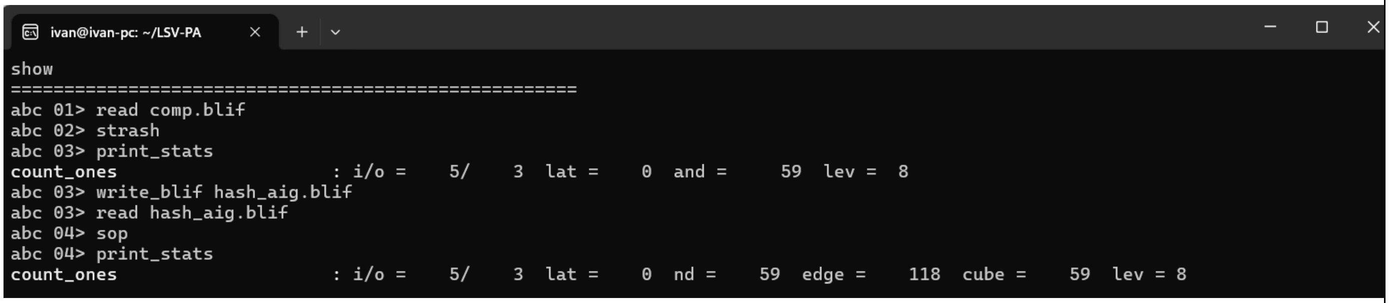
BDD	collapsed BDD
	

```
abc 02> bdd
abc 02> print_stats
count_ones                    : i/o =    5/    3  lat =    0  nd =     3  edge =     15  bdd  =    21  lev = 1
abc 02> show_bdd -g
```

```
Warning: Cannot convert string "-*-Helvetica-Bold-R-Normal--*-120-*-*-P-*-ISO8859-1" to type FontStruct
abc 02> collapse
abc 03> print_stats
count_ones                    : i/o =    5/    3  lat =    0  nd =     3  edge =     15  bdd  =    21  lev = 1
abc 03> show_bdd -g
```

(b) The commands I use: “write_blif [file]” => “read [file]” => sop
Since we cannot directly transform a hashed AIG to SOP, I first output the AIG file and read it as a logic network. Then, I can directly use “sop” command to convert the graph.

Commands



```
show
=====================================================
abc 01> read comp.blif
abc 02> strash
abc 03> print_stats
count_ones                    : i/o =    5/    3  lat =    0  and =     59  lev =  8
abc 03> write_blif hash_aig.blif
abc 03> read hash_aig.blif
abc 04> sop
abc 04> print_stats
count_ones                    : i/o =    5/    3  lat =    0  nd =    59  edge =    118  cube =    59  lev = 8
```

hashed AIG	SOP