LSV-PA1 Report

2.

(a) See “alu.blif”.

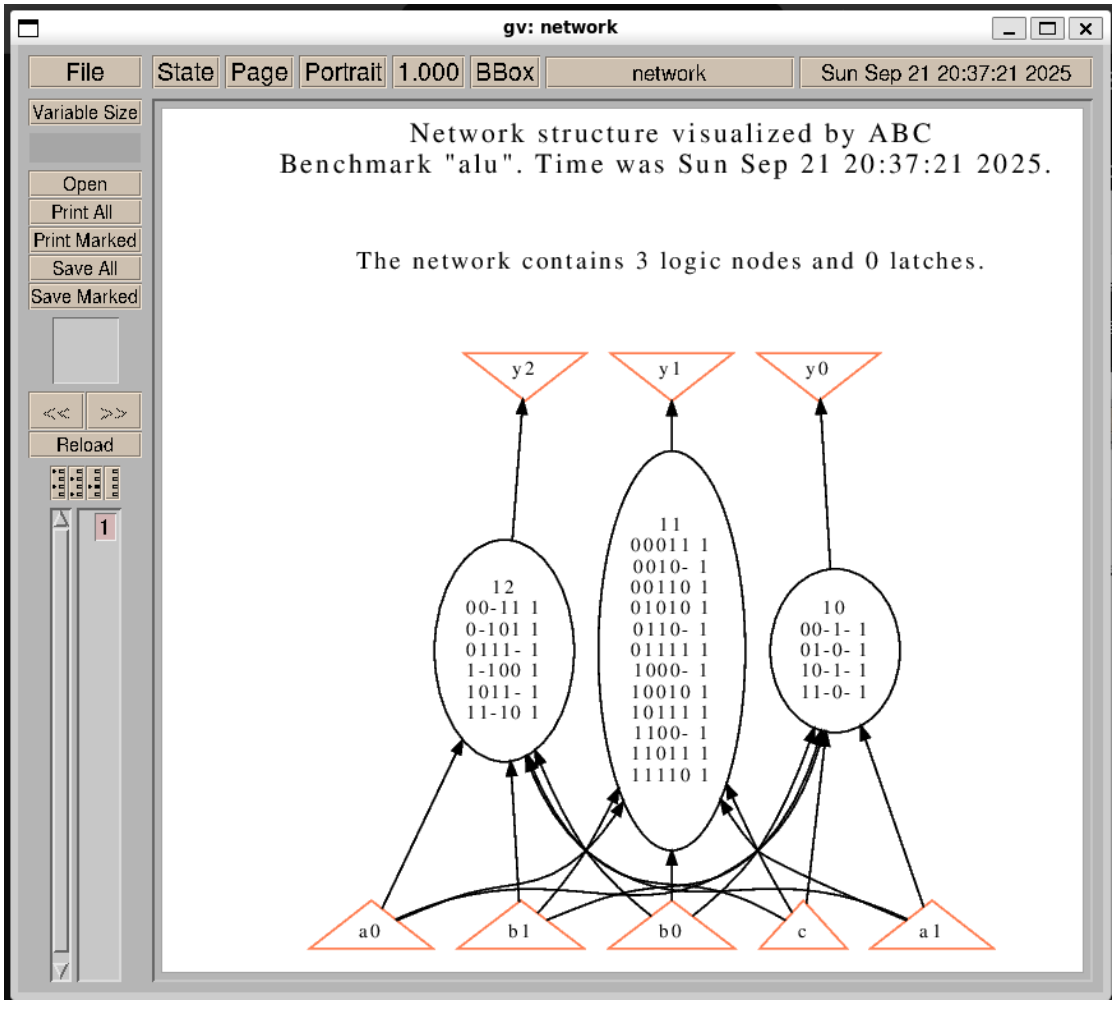
(b)

Commands:

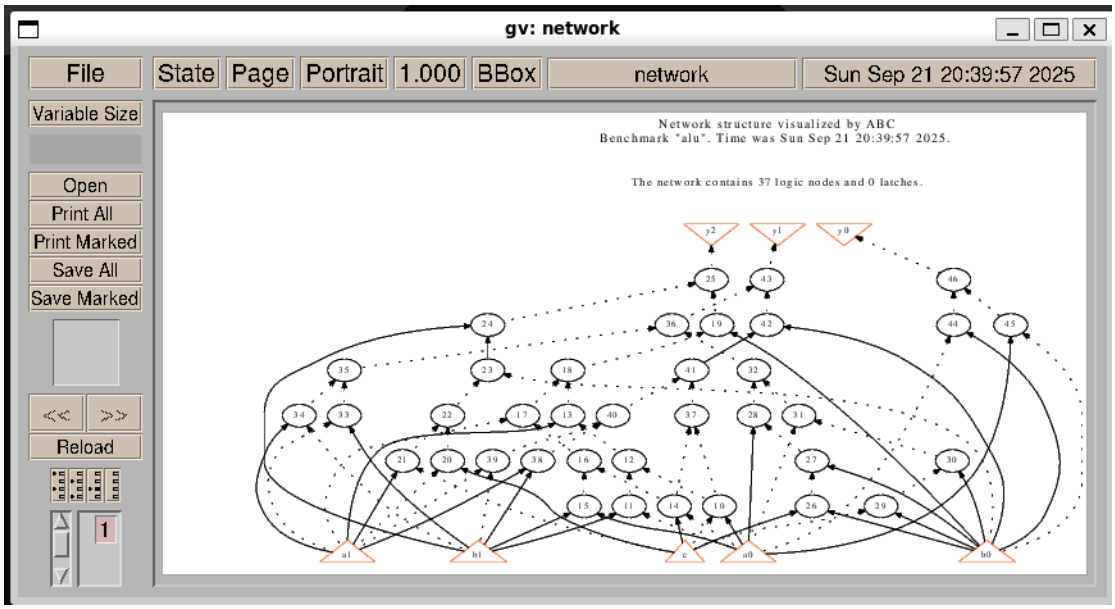
```
================================================
abc 01> read_blif alu.blif
abc 02> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =     3  edge =     15  cube =    22  lev = 1
abc 02> show
abc 02> Warning: Missing charsets in String to FontSet conversion
strash
abc 03> show
abc 03> Warning: Missing charsets in String to FontSet conversion

abc 03> collapse
abc 04> show_bdd -g
abc 04> Warning: Missing charsets in String to FontSet conversion
```

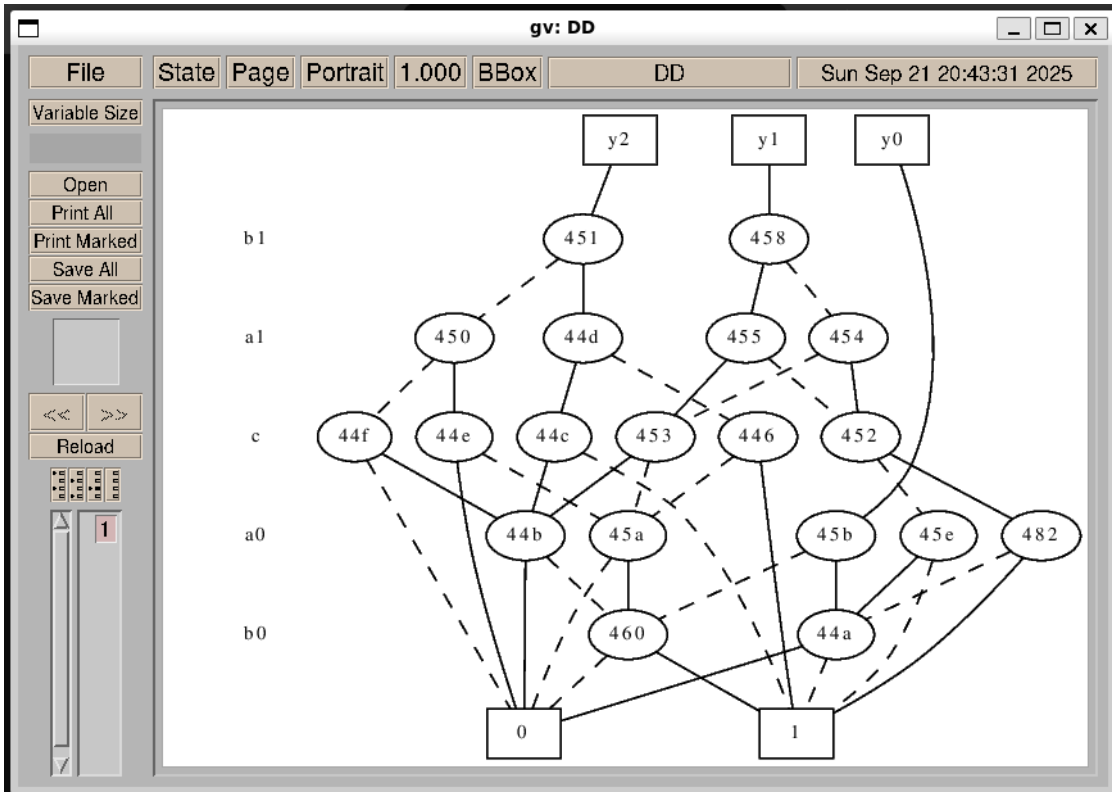
Network structure:



AIG:


gv: network
File
State
Page
Portrait
1.000
BBox
network
Sun Sep 21 20:39:57 2025
Variable Size
Open
Print All
Print Marked
Save All
Save Marked
<<
>>
Reload
1
Network structure visualized by ABC
Benchmark "alu". Time was Sun Sep 21 20:39:57 2025.
The network contains 37 logic nodes and 0 latches.

BDD:


gv: DD
File
State
Page
Portrait
1.000
BBox
DD
Sun Sep 21 20:43:31 2025
Variable Size
Open
Print All
Print Marked
Save All
Save Marked
<<
>>
Reload
1
y2
y1
y0
b1
a1
c
a0
b0
451
458
450
44d
455
454
44f
44e
44c
453
446
452
44b
45a
45b
45e
482
460
44a
0
1

3.

(a)

logic network in AIG:

```
abc 01> read_blif alu.blif
abc 02> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =     3  edge =     15  cube =    22  lev = 1
abc 02> aig
abc 02> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =     3  edge =     15  aig  =    39  lev = 1
```

structurally hashed AIG:

```
abc 01> read_blif alu.blif
abc 02> strash
abc 03> print_stats
alu                           : i/o =    5/    3  lat =    0  and =     37  lev =  6
```

The **strash** command optimizes the AIG, which generates the AIG with fewer nodes and edges.

logic network in BDD:

```
abc 01> read_blif alu.blif
abc 02> bdd
abc 02> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =     3  edge =     15  bdd  =    21  lev = 1
```

collapsed BDD:

```
abc 01> read_blif alu.blif
abc 02> collapse
abc 03> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =     3  edge =     12  bdd  =    18  lev = 1
```

The **collapse** command optimizes the BDD, which generates the BDD with fewer nodes and edges.

(b)

command sequence:

```
abc 01> read_blif alu.blif
abc 02> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =     3  edge =     15  cube =    22  lev = 1
abc 02> strash
abc 03> logic
abc 04> print_stats
alu                       : i/o =    5/    3  lat =    0  nd =    37  edge =     74  cube =    37  lev = 6
abc 04> show
abc 04> Warning: Missing charsets in String to FontSet conversion

abc 04> write_blif alu_sop.blif
```

The command **logic** transforms an AIG into a logic network with SOPs.

The resulting logic network is represented by “alu_sop.blif”.

The resulting logic network: