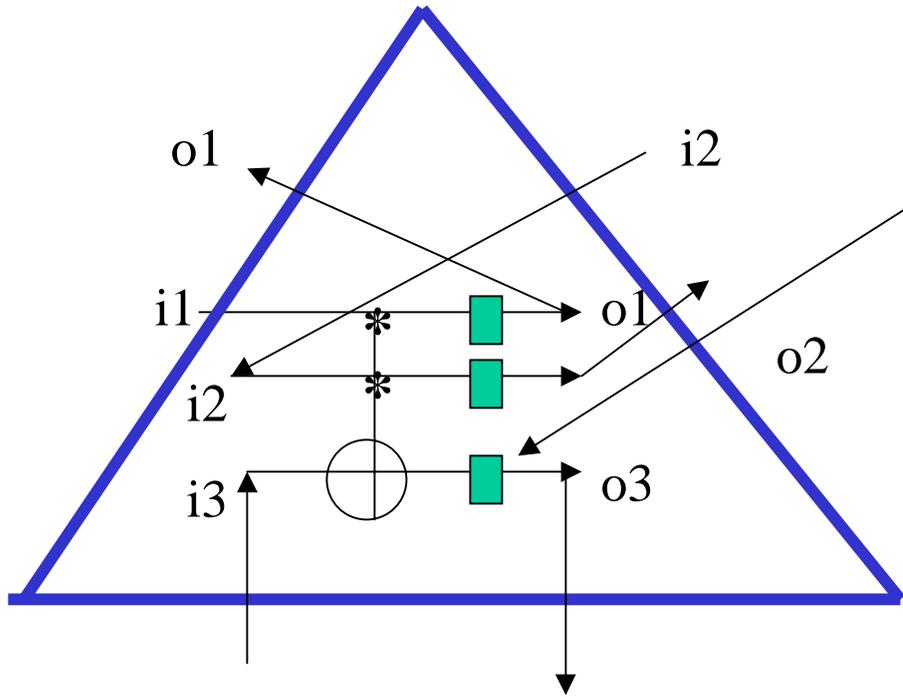


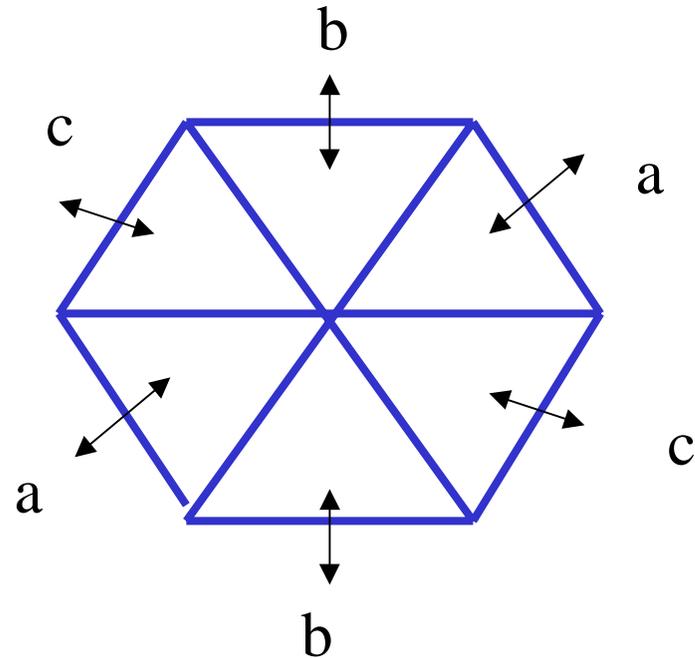
# Problem 1

Given is the cell of a cellular automaton



Grey rectangles are D flip-flops, clocks not shown.

The structure of cell connections is shown below. There are three cells as in left and three cells rotated clockwise by 180 degree



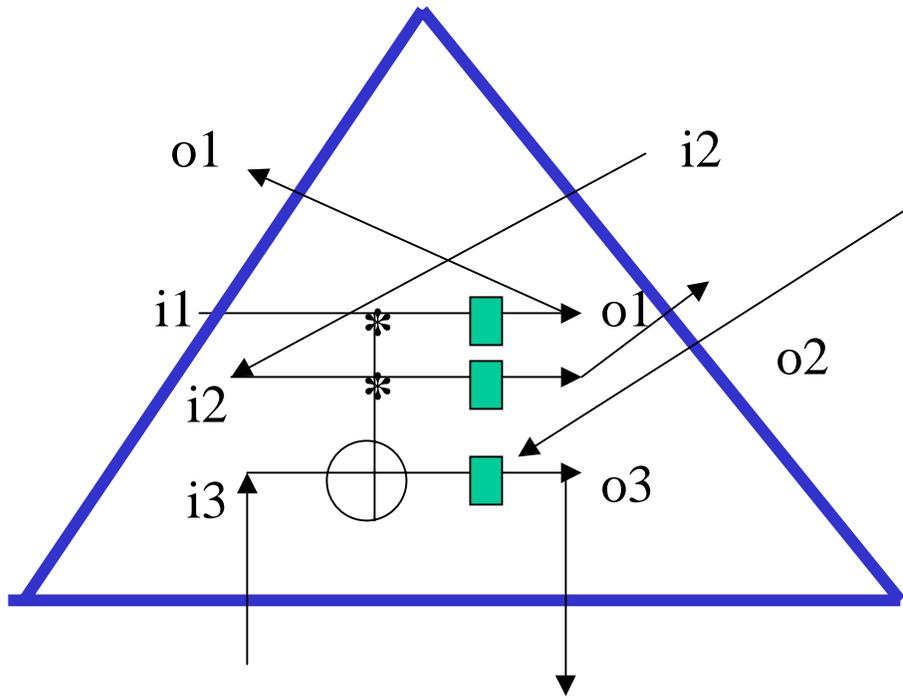
A). Analyze behavior of the system in which all cells are in state 0. The state is a natural number corresponding to binary signals in order  $o1, o2, o3$ , with  $o3$  as the least significant bit.

B). Analyze first two transitions of the system in which all cells are in state 7.

C). Try to guess the initial state that will lead to the longest cycle. Show your work.

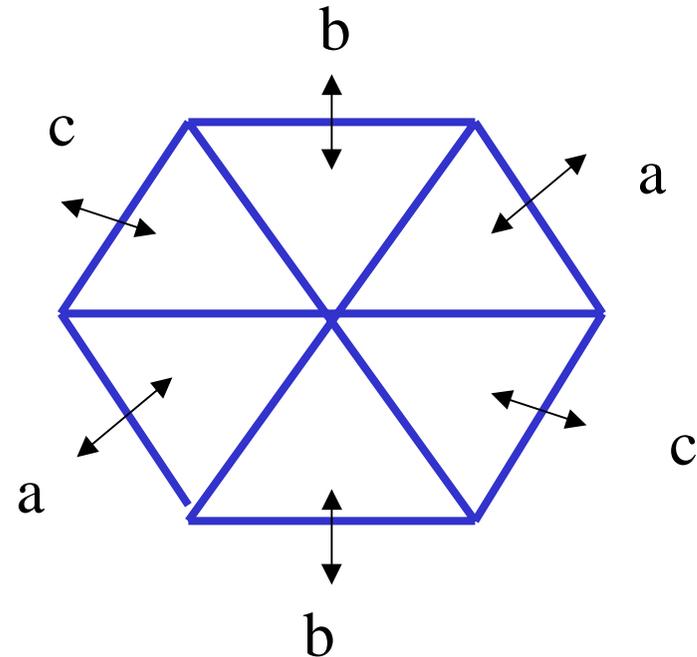
D) Can you generalize these results to arbitrary size CA of this type of cells and connection structure?

# Solution to Problem 1



Grey rectangles are D flip-flops, clocks not shown.

The structure of cell connections is shown below. There are three cells as in left and three cells rotated clockwise by 180 degree



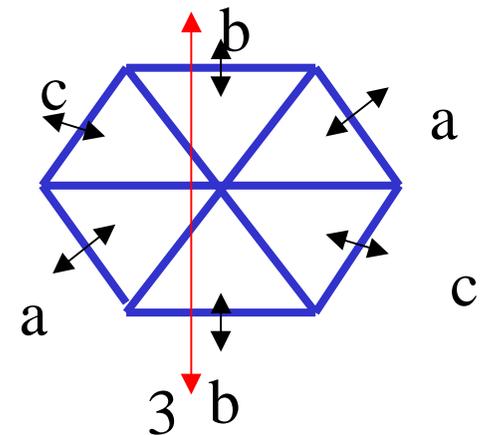
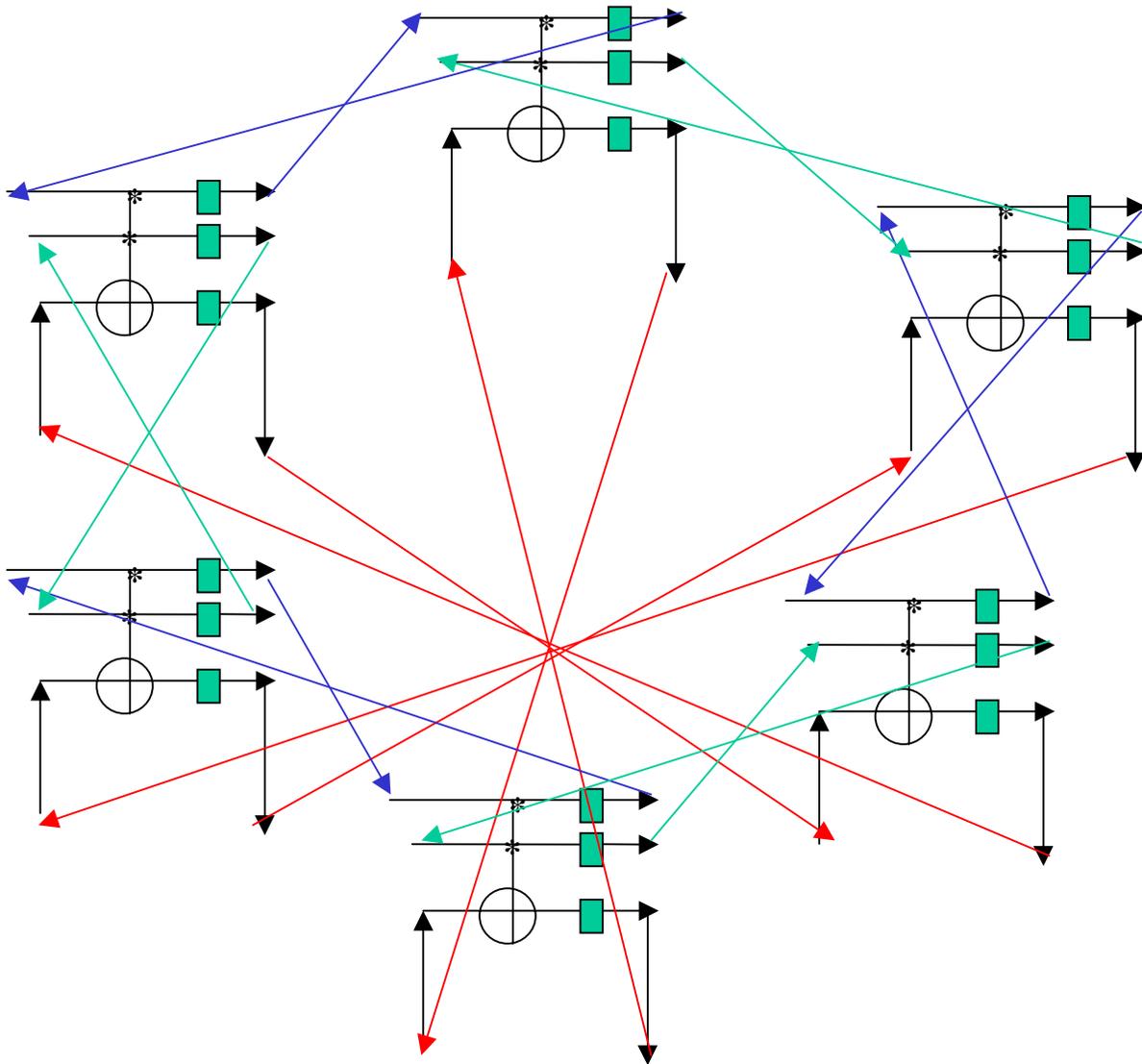
A). Analyze behavior of the system in which all cells are in state 0. The state is a natural number corresponding to binary signals in order  $o1, o2, o3$ , with  $o3$  as the least significant bit.

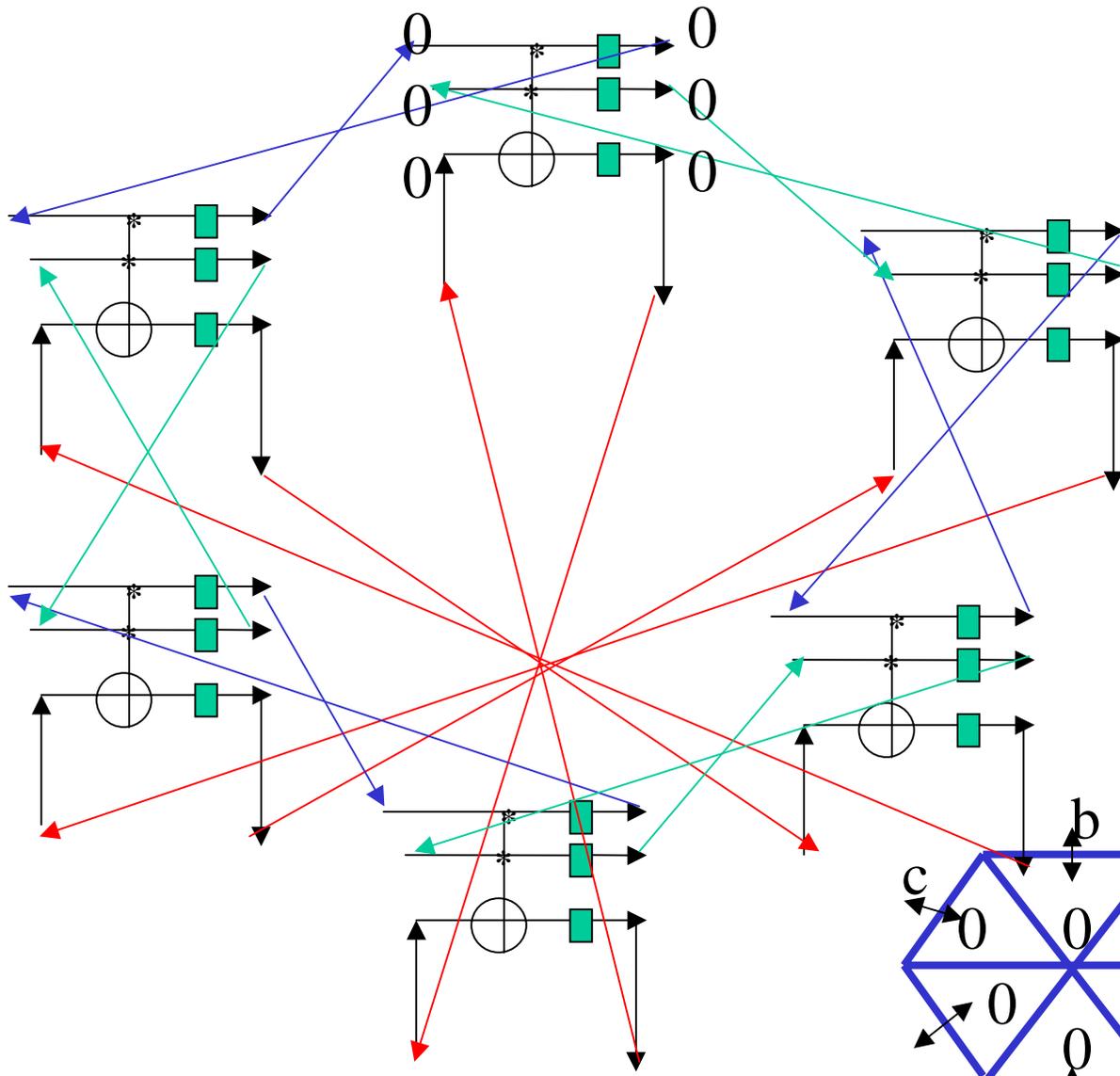
B). Analyze first two transitions of the system in which all cells are in state 7.

C). Try to guess the initial state that will lead to the longest cycle. Show your work.

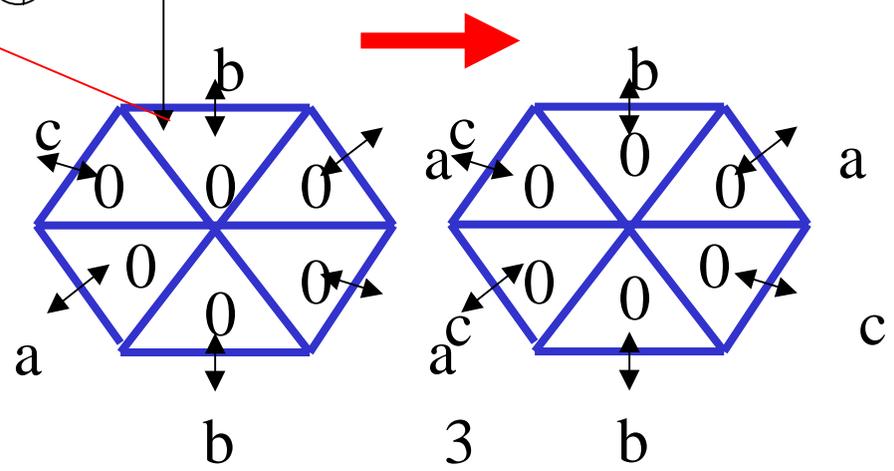
D) Can you generalize these results to arbitrary size CA of this type of cells and connection structure?

First we have to understand how the cells are connected to be able to analyze their behavior. Here is the network.

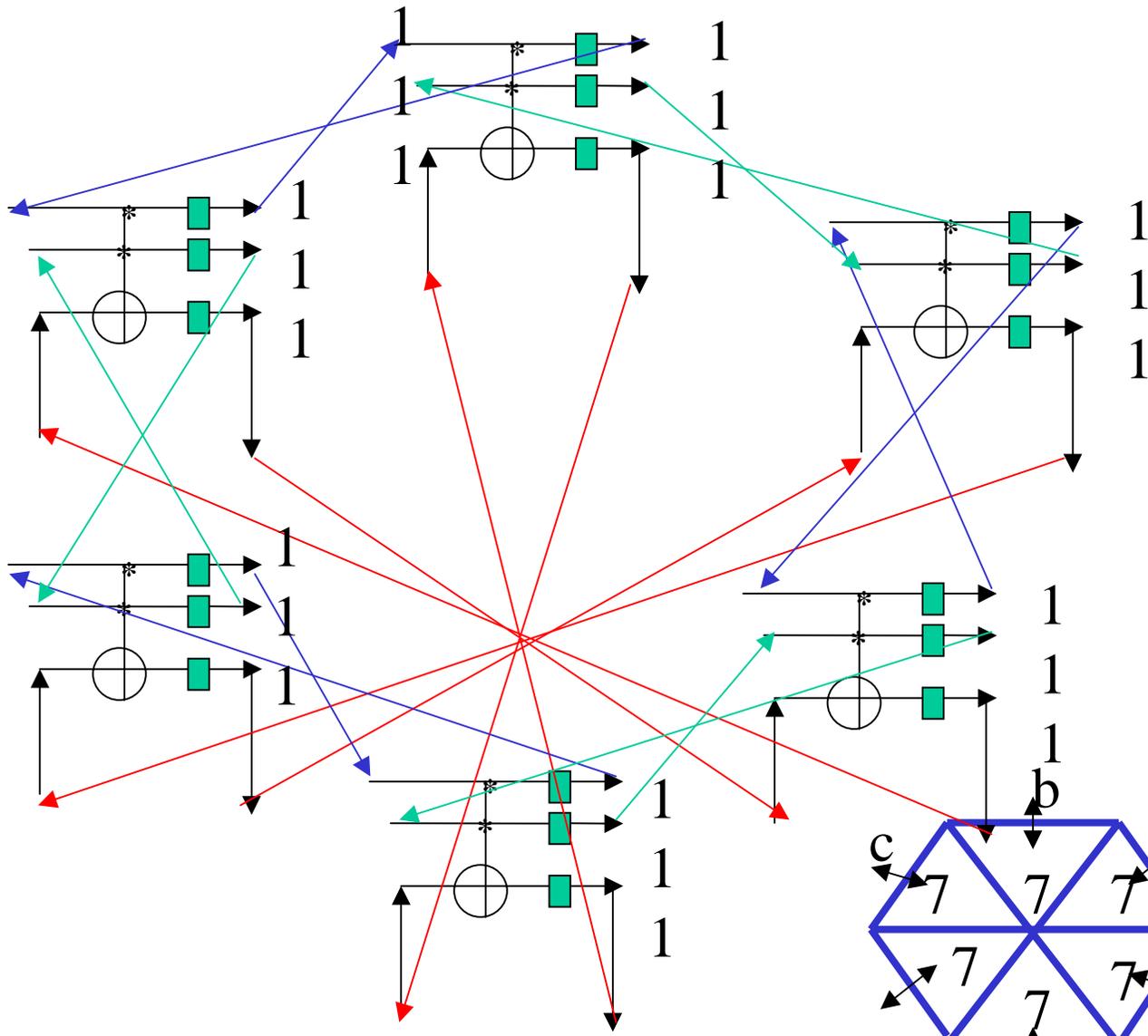




0--> 0 in  
 Toffoli gate  
 so the circuit  
 remains in  
 state 00000  
 as shown

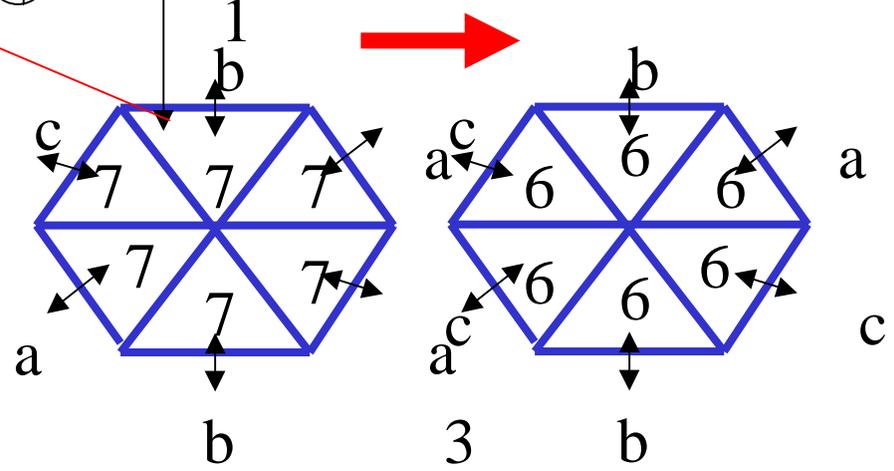


111-->11 0 in  
Toffoli gate.  
We denote it  
7-->6.

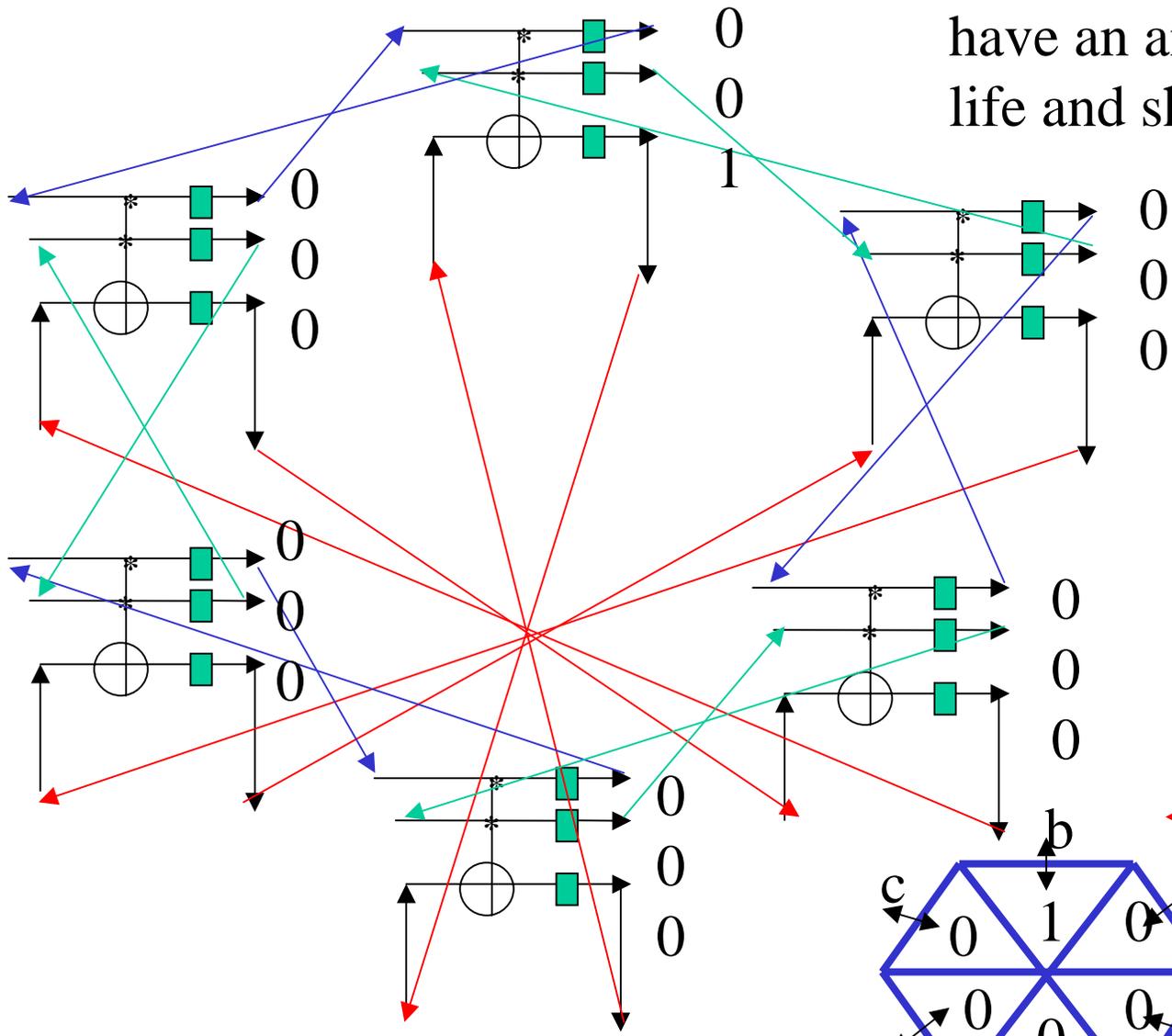


1  
1  
1

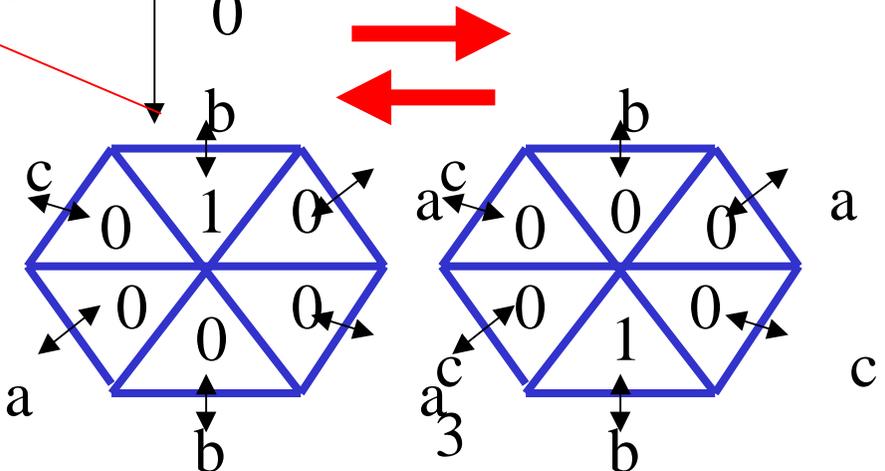
Now it should be obvious that if the cells are initially in the same state, nothing interesting happens. We have to try various states of cells, like in game of life.



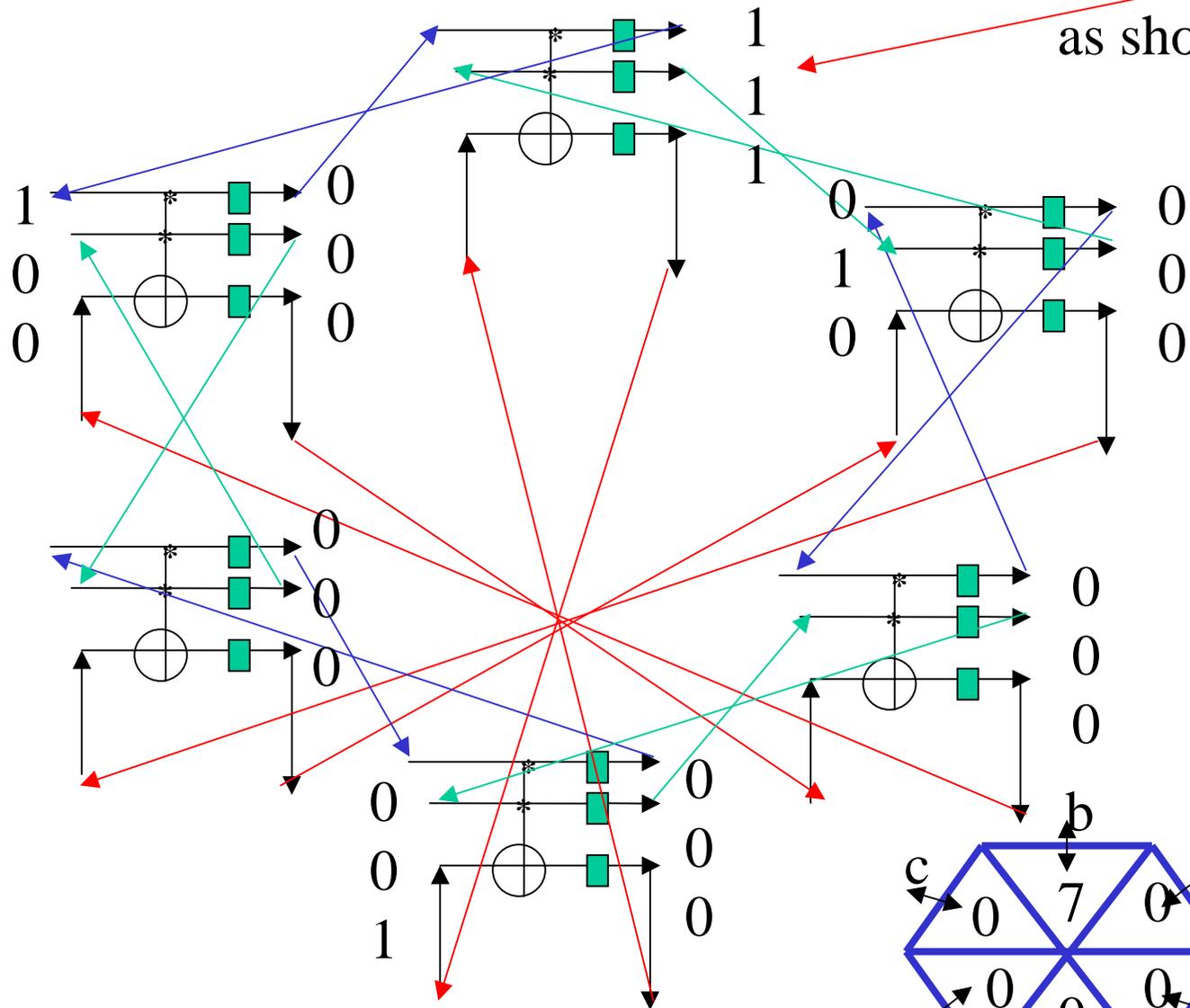
Let us assume that 001 is alive and 000 is dead and let us have an analogy with game of life and shift register.



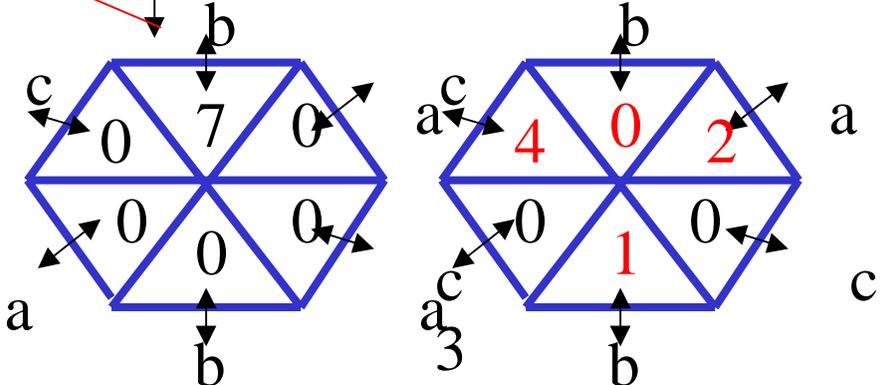
Now we obtained a system with two states, but it is still not interesting. So perhaps we should try a state in Toffoli gate that is in a longer cycle.



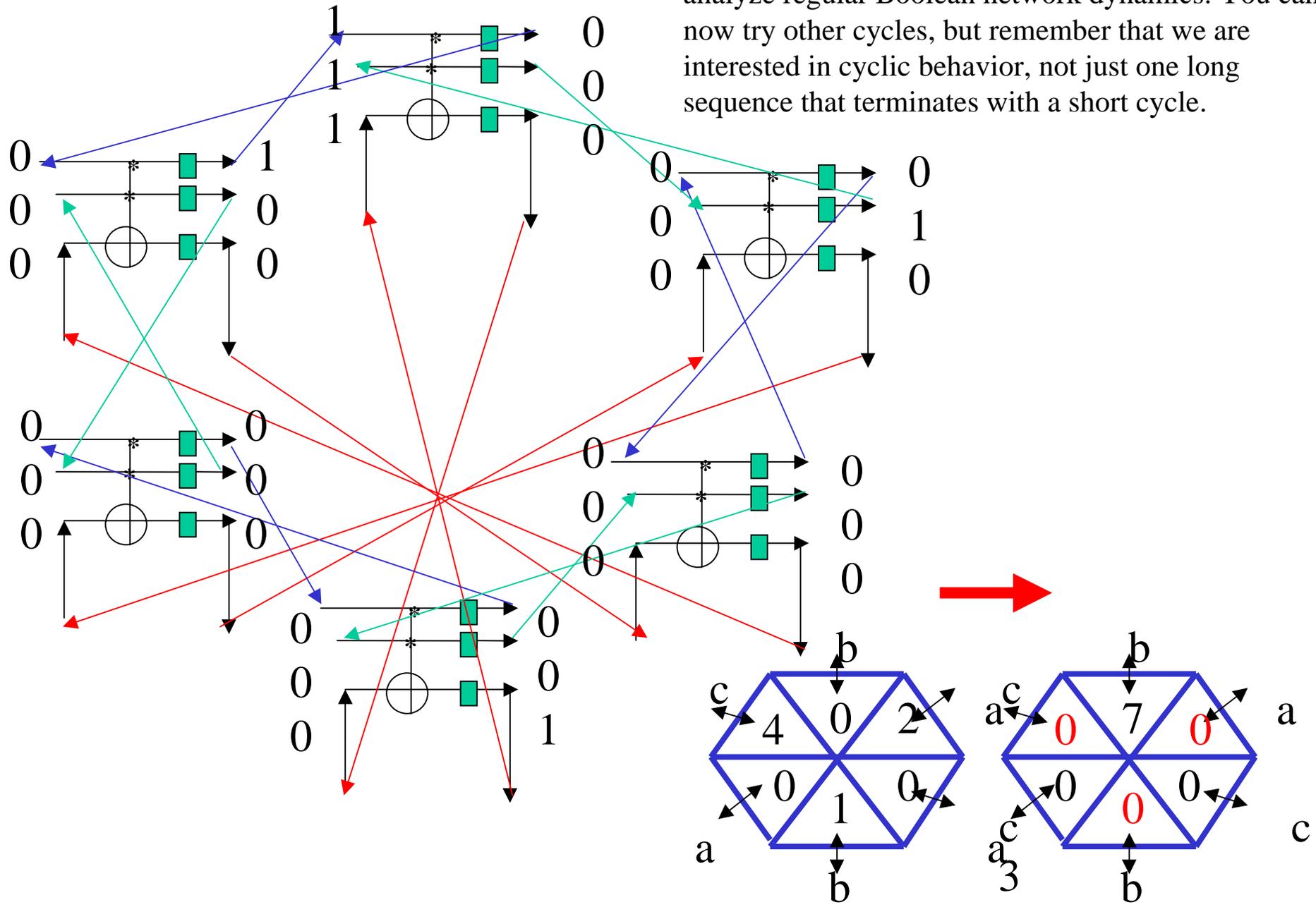
Now three ones from this cell went to different cells, as shown.



In next slide we will analyze next state



Thus we return to the initial state and we generated a cycle of length three. This is just an example how to analyze regular Boolean network dynamics. You can now try other cycles, but remember that we are interested in cyclic behavior, not just one long sequence that terminates with a short cycle.

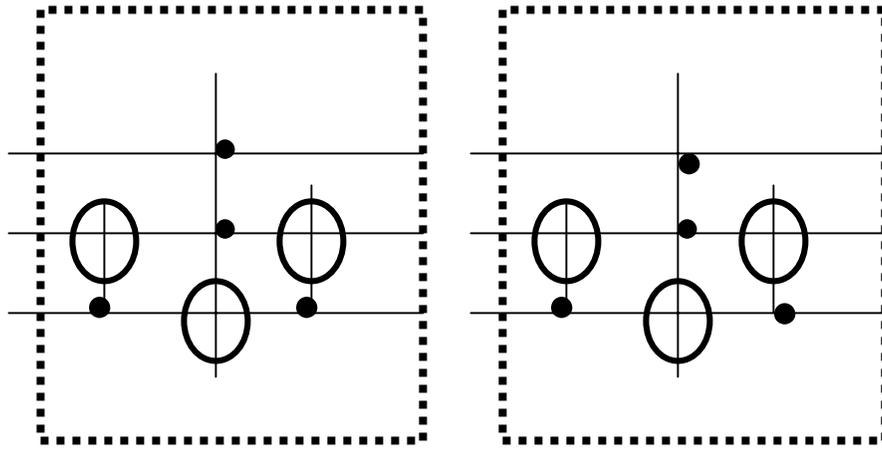


## Problem 2

- A) What is the inverse gate of the Fredkin Gate?
- B) Prove that the gate that you found is inverse to Fredkin gate using the method of graphical transformation in quantum notation.
- C) Do the same using truth tables of the gates.

# Solution to Problem 2

- A) What is the inverse gate of the Fredkin Gate?
  - Fredkin Gate itself
- B) Prove that the gate that you found is inverse to Fredkin gate using the method of graphical transformation in quantum notation.



Fredkin  
from  
Toffoli  
and  
Feynman

Fredkin  
repeated

First stage of graphic method is to draw a sequence of gates.

Next we cancel two gates that are mirrors.

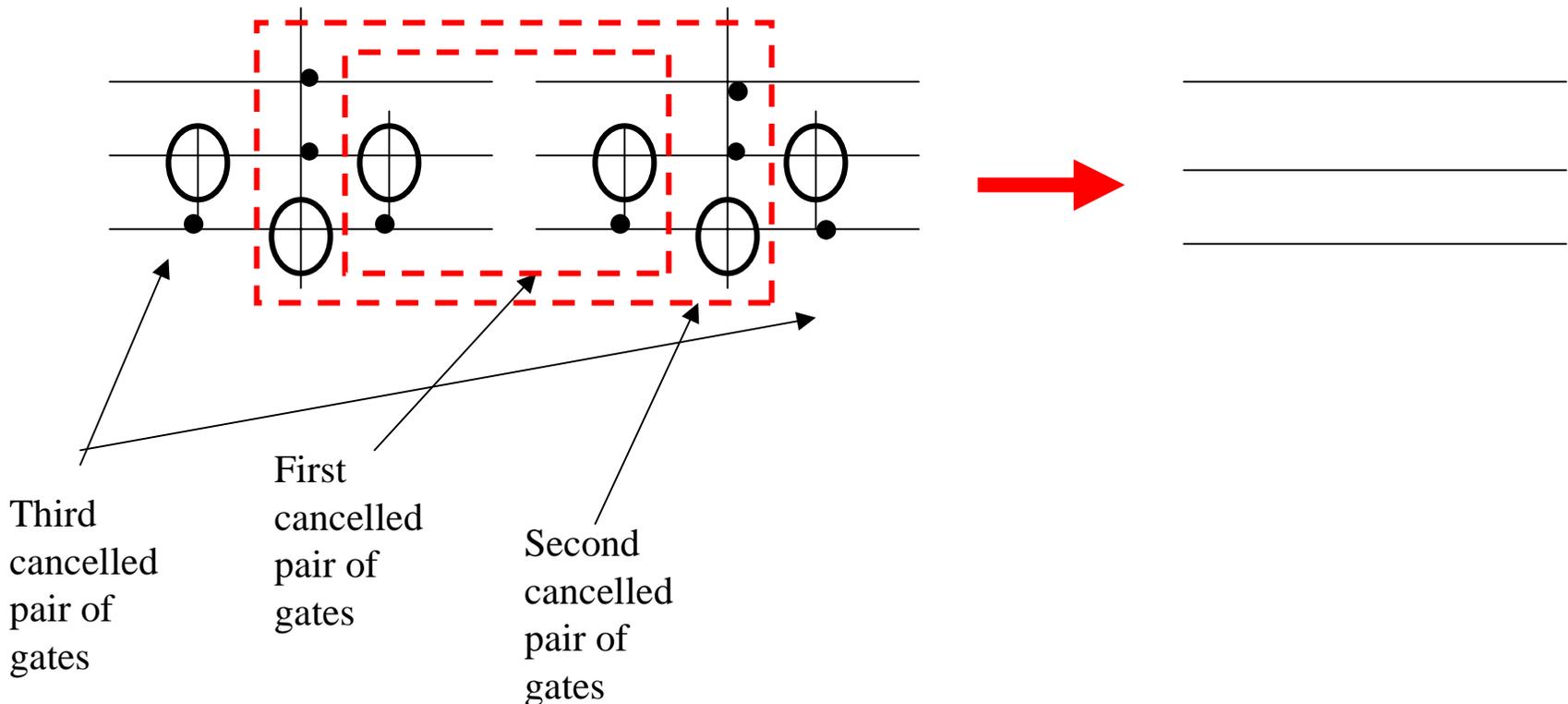
We repeat this three times

# Solution to Problem 2

First stage of graphic method is to draw a sequence of gates.

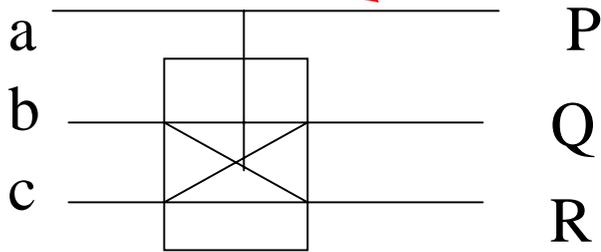
Next we cancel two gates that are mirrors.

We repeat this three times. After the third removal three wires remain which means identity. So the Fredkin gate is its own reverse.

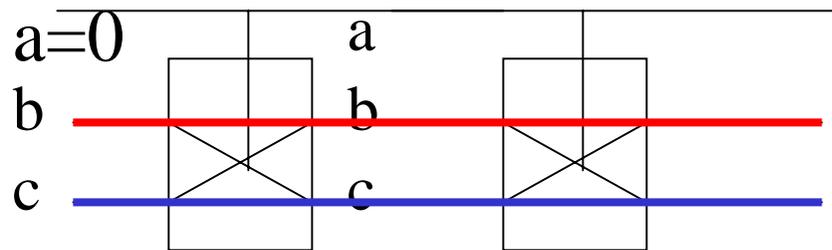


# Solution to Problem 2

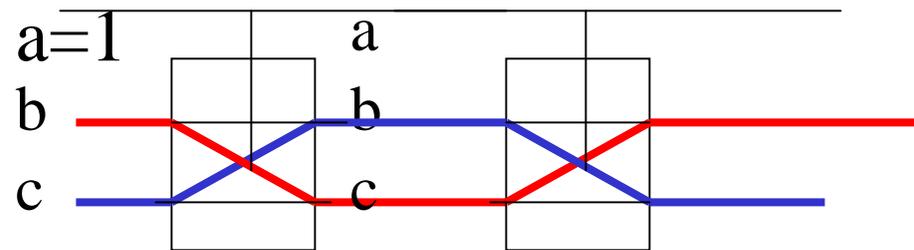
- Another graphical method is based on the notation that **Fredkin is a controlled swap**.



The two figures below prove that both for  $a=0$  and  $a=1$  the circuit is equivalent to three horizontal wires



For  $a=0$



For  $a=1$

# Solution to Problem 2

- C) Do the same using truth tables of the gates.

a b c	P Q R	P Q R	a b c
0 0 0	0 0 0	0 0 0	0 0 0
0 0 1	0 0 1	0 0 1	0 0 1
0 1 0	0 1 0	0 1 0	0 1 0
0 1 1	0 1 1	0 1 1	0 1 1
1 0 0	1 0 0	1 0 0	1 0 0
1 0 1	1 1 0	1 1 0	1 0 1
1 1 0	1 0 1	1 0 1	1 1 0
1 1 1	1 1 1	1 1 1	1 1 1

The truth tables of function  $F$  and  $F^{-1}$  are the same.  
So Fredkin is its own inverse.

# Problem 3

- A. Give an example of a gate that is reversible and conservative
- B. Give an example of a gate that is reversible and not conservative.
- C. Give an example of a gate that is conservative but not reversible.

Prove all your results. Verify.

# Solution to Problem 3

- A. Give an example of a gate that is reversible and conservative
- Fredkin gate is reversible and is conservative as it can be showed on previous slide since for every row of table it preserves the number of symbols 1 in input and output vectors.
- B. Give an example of a gate that is reversible and not conservative.
- Toffoli and Feynman gates are reversible and not conservative.
- C. Give an example of a gate that is conservative but not reversible.

ab	PQ
00	00
01	10
11	11
10	10

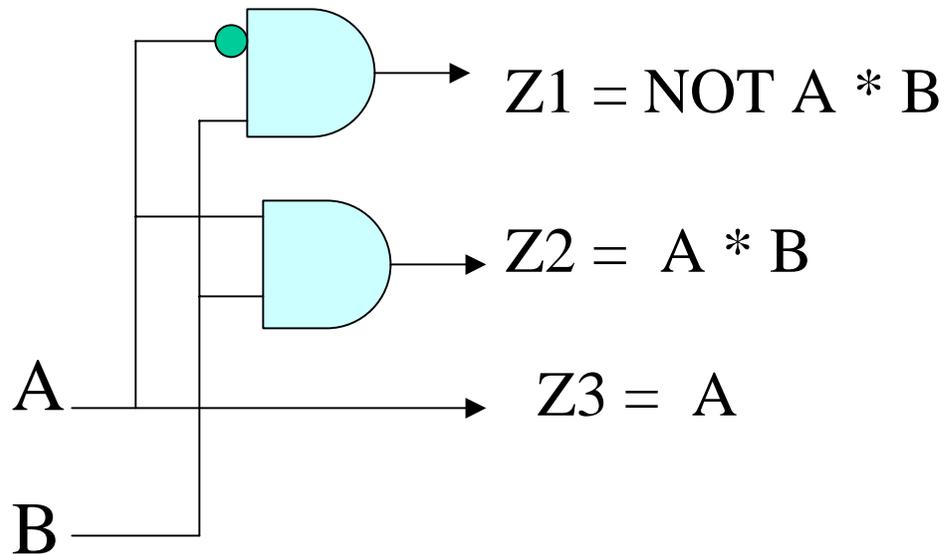
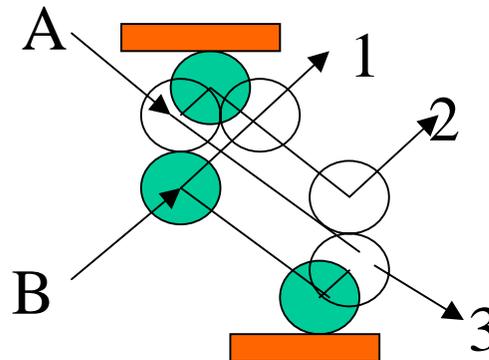
# Problem 4

- A) Realize the Fredkin Gate using Billiard Ball Model.
- B) Realize the Swap Gate using the Billiard Ball Model

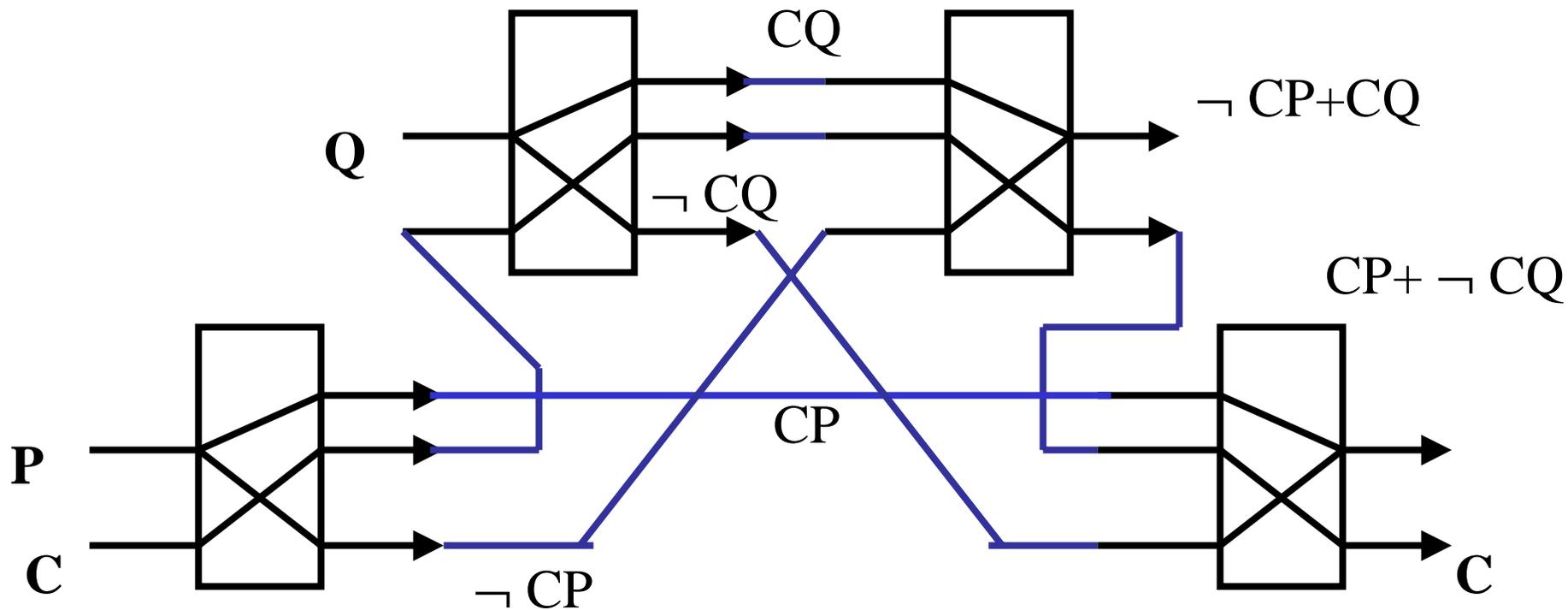
# Solution to Problem 4

- A) Realize the Fredkin Gate using Billiard Ball Model.
- B) Realize the Swap Gate using the Billiard Ball Model

Input		output		
A	B	z1	z2	z3
0	0	0	0	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	1



# Fredkin Gate from Switch Gates



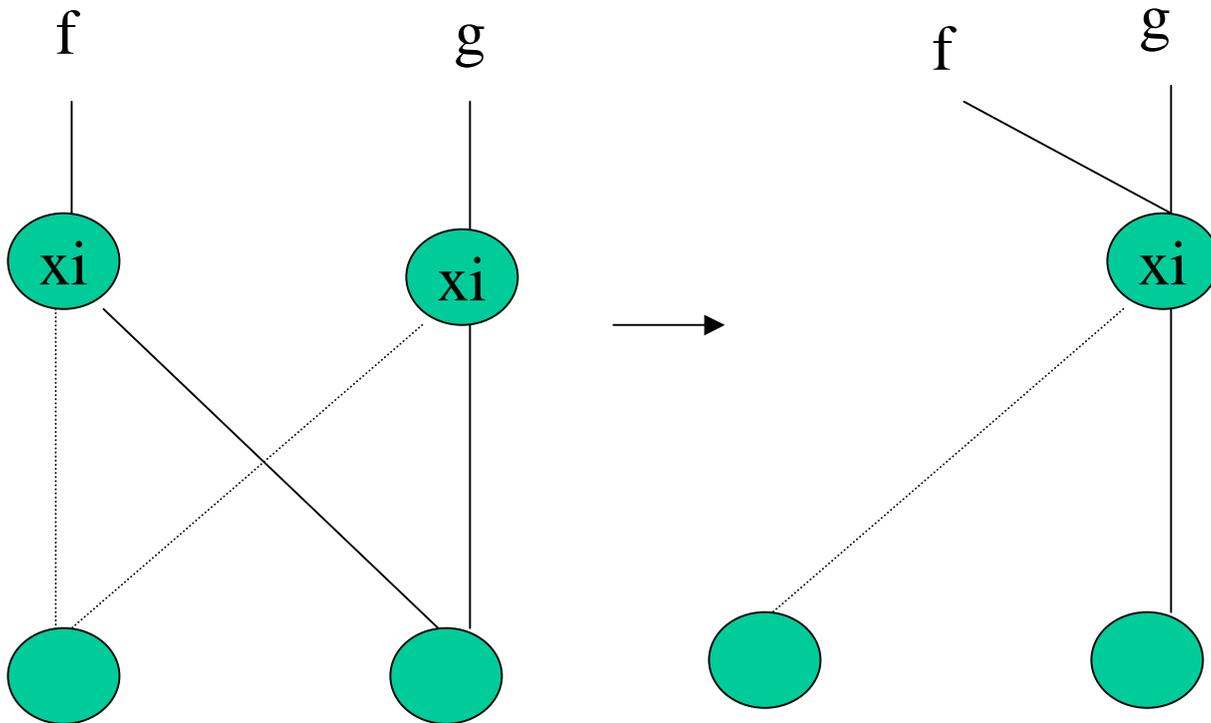
When input  $C$  is set to 1 the Fredkin gate realizes the swap gate.

# Problem 5

- A) Define what is Kronecker (Functional) Decision Diagram
- B) Derive such a diagram for functions  $F=A(C\oplus B) \oplus CD$ ,  $G= C\oplus B$  sharing as much of the diagram for both functions together.
- C) Convert the diagram to a circuit with Inverter, Toffoli and Feynman gates. Show all constant (if any) and garbage (if any) signals.
- D) Add the mirror and spy circuits in a standard way. Discuss the garbage in the new circuit.

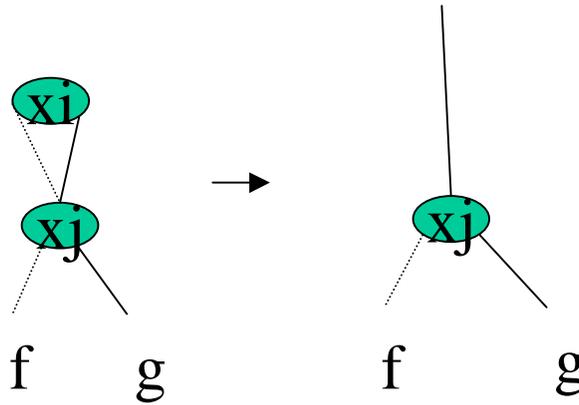
# Solution to Problem 5

- A) Define what is Kronecker (Functional) Decision Diagram
  - This is an ordered decision diagram similar to BDD in which for every variable (level) you can have only one type of expansion: Shannon (as in BDD), Positive Davio or Negative Davio. Thus there exist  $3^n$  various Kronecker Diagrams for a given order of variables. The rules of combining isomorphic nodes in the tree with S, pD and nD expansion nodes are the same as for BDDs and can be find in lecture slides. Here is an example, this transformation applies to any type of nodes.

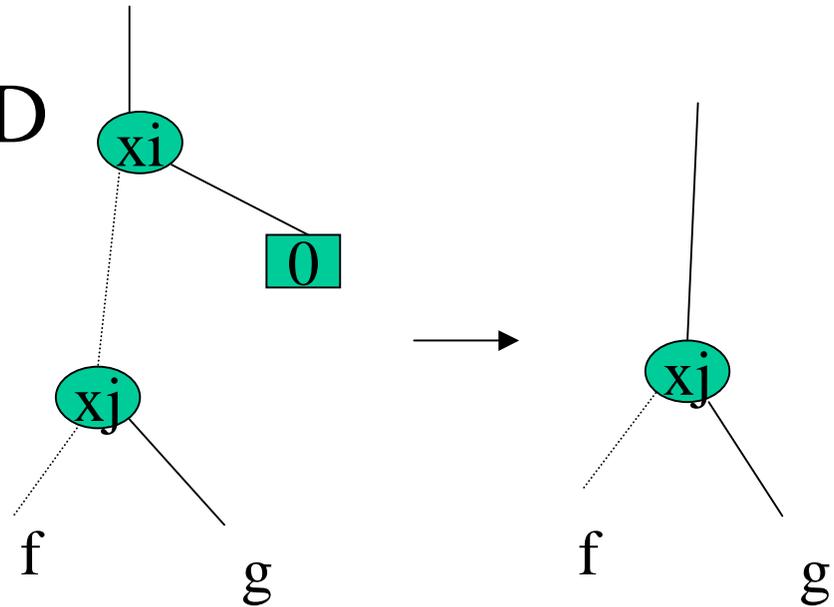


# Another type of KFDD transformation

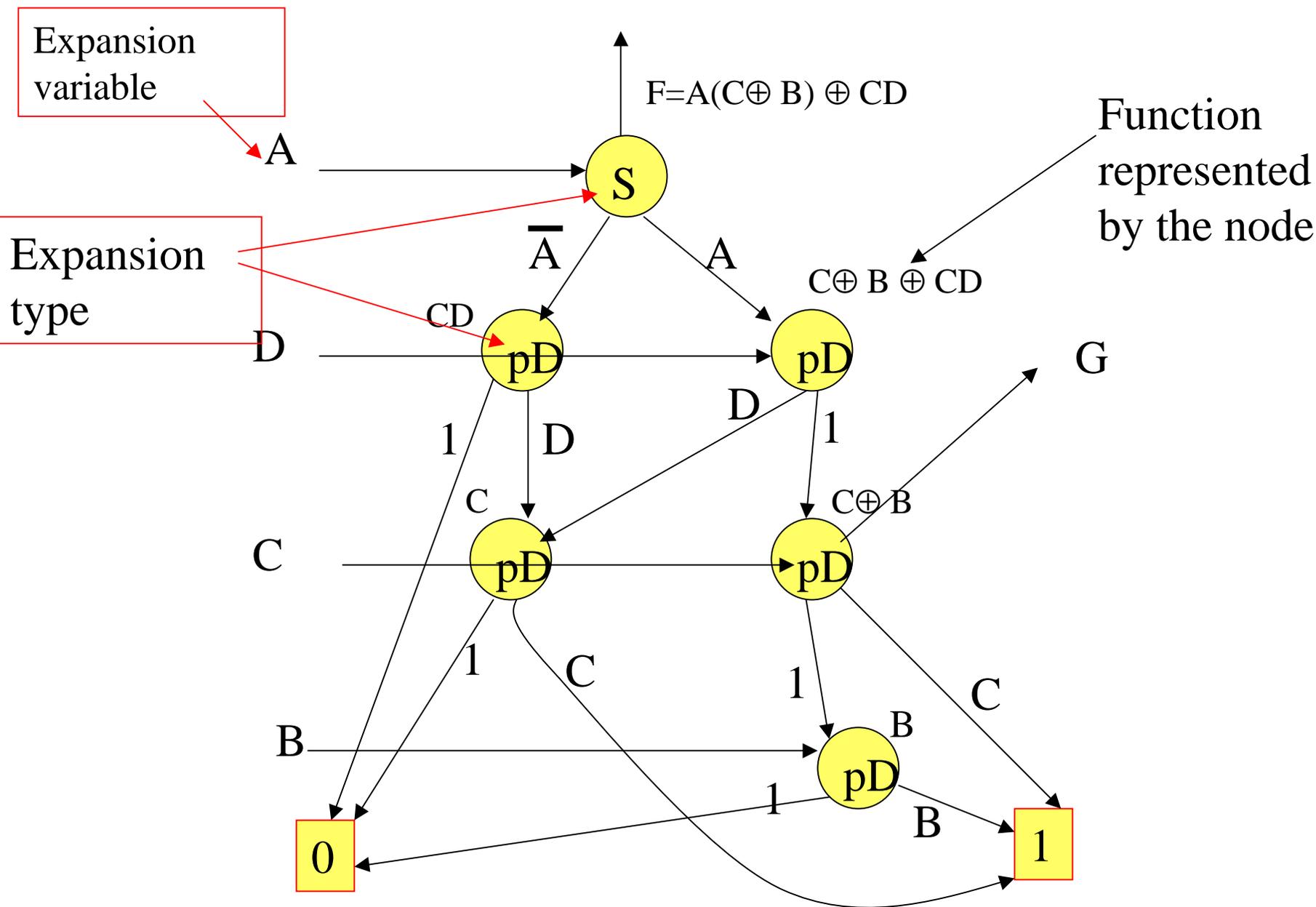
Type S



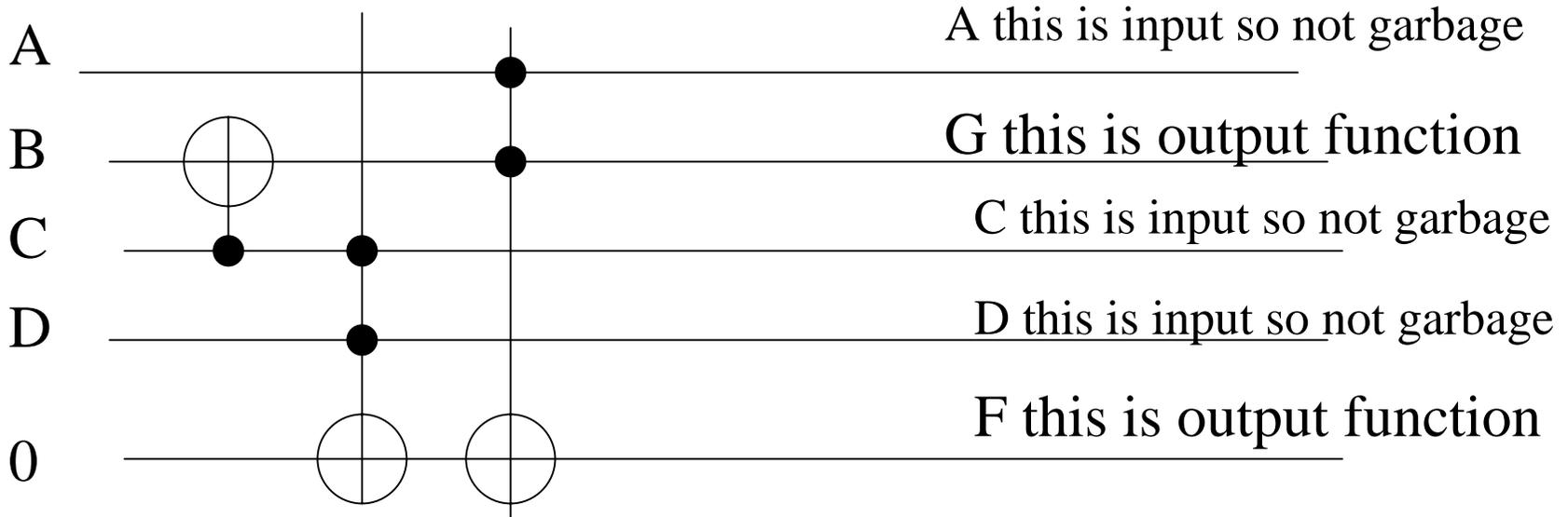
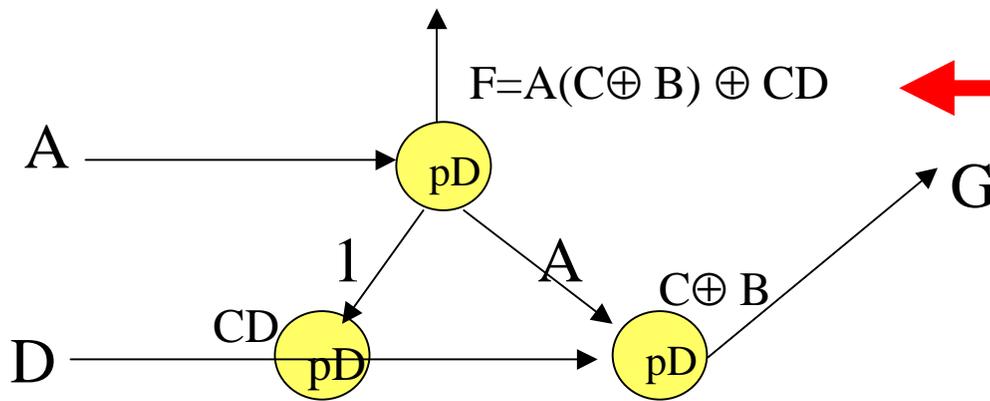
Type D



- B) Derive such a diagram for functions  $F=A(C \oplus B) \oplus CD$ ,  $G=C \oplus B$  sharing as much of the diagram for both functions together.

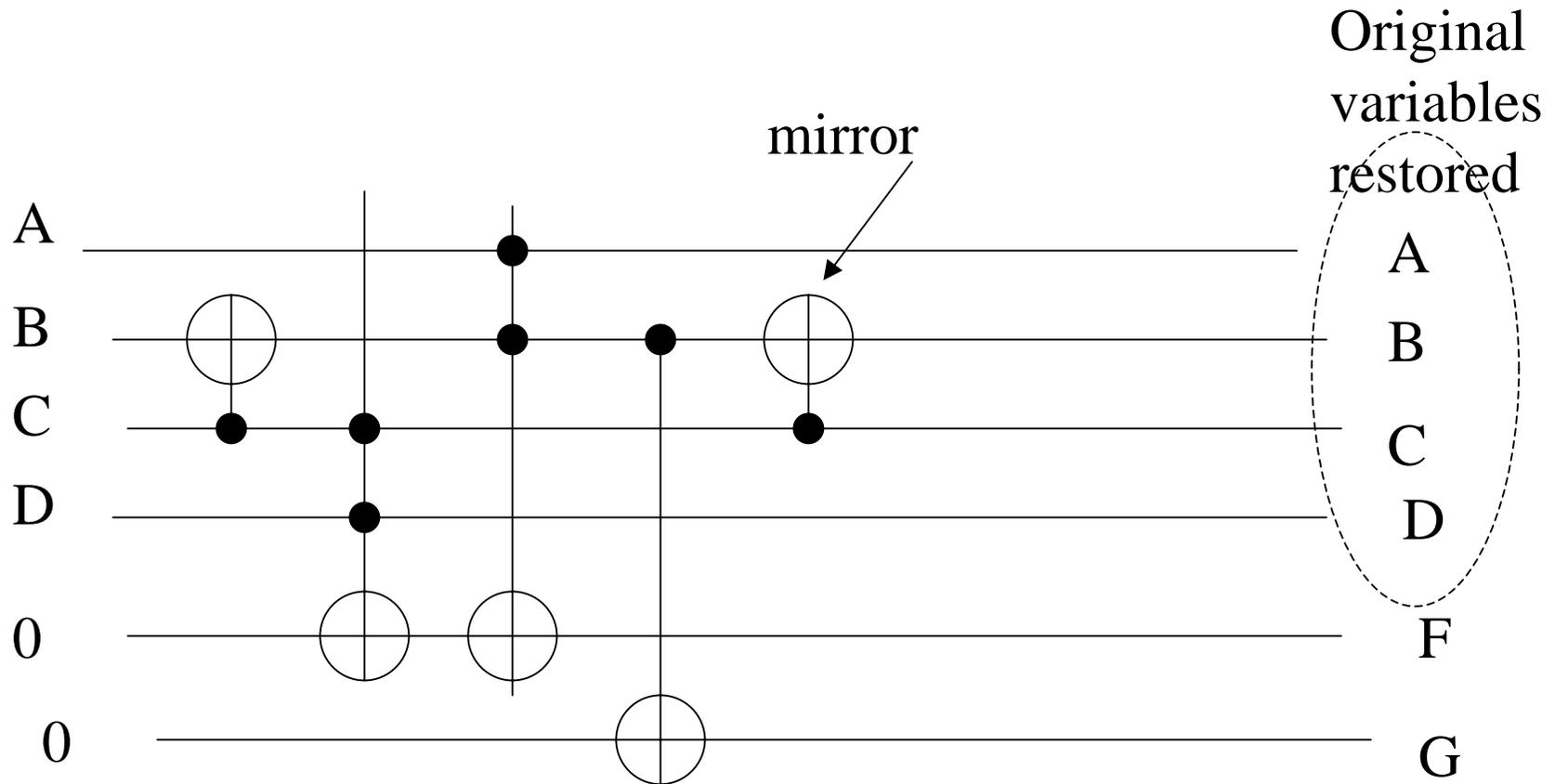


Changing first node to pD (this is also a KFDD - a special case)



- C) Convert the diagram to a circuit with Inverter, Toffoli and Feynman gates. Show all constant (if any) and garbage (if any) signals.
  - Function has four input variables and five output variables. So it has one garbage. On the other hand, because in this solution the potential garbage functions are input variables, some authors do not categorize these outputs as garbages.

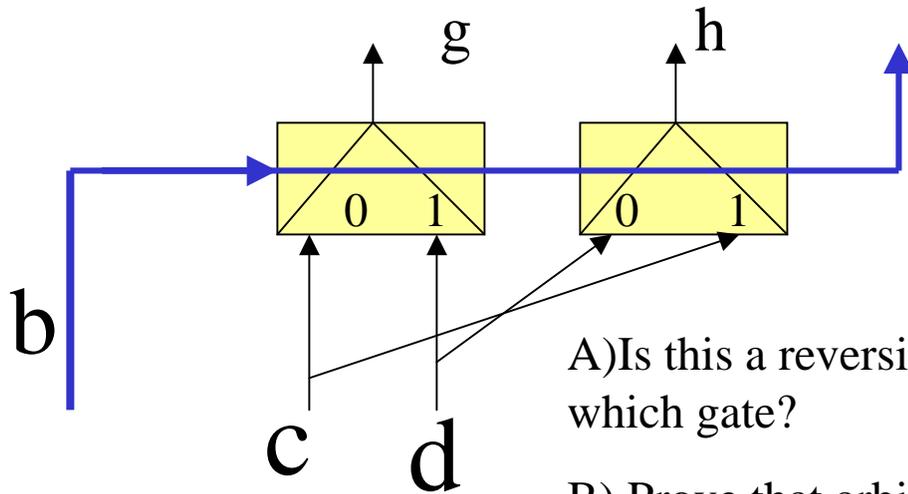
- D) Add the mirror and spy circuits in a standard way. Discuss the garbage in the new circuit.
  - The mirrors are added in order to create original variables at the output. This is useful in quantum computing.



Observe that in this case the mirror is very simple because of smart design.

# Problem 6

Given is cell with 3 inputs and 3 outputs.

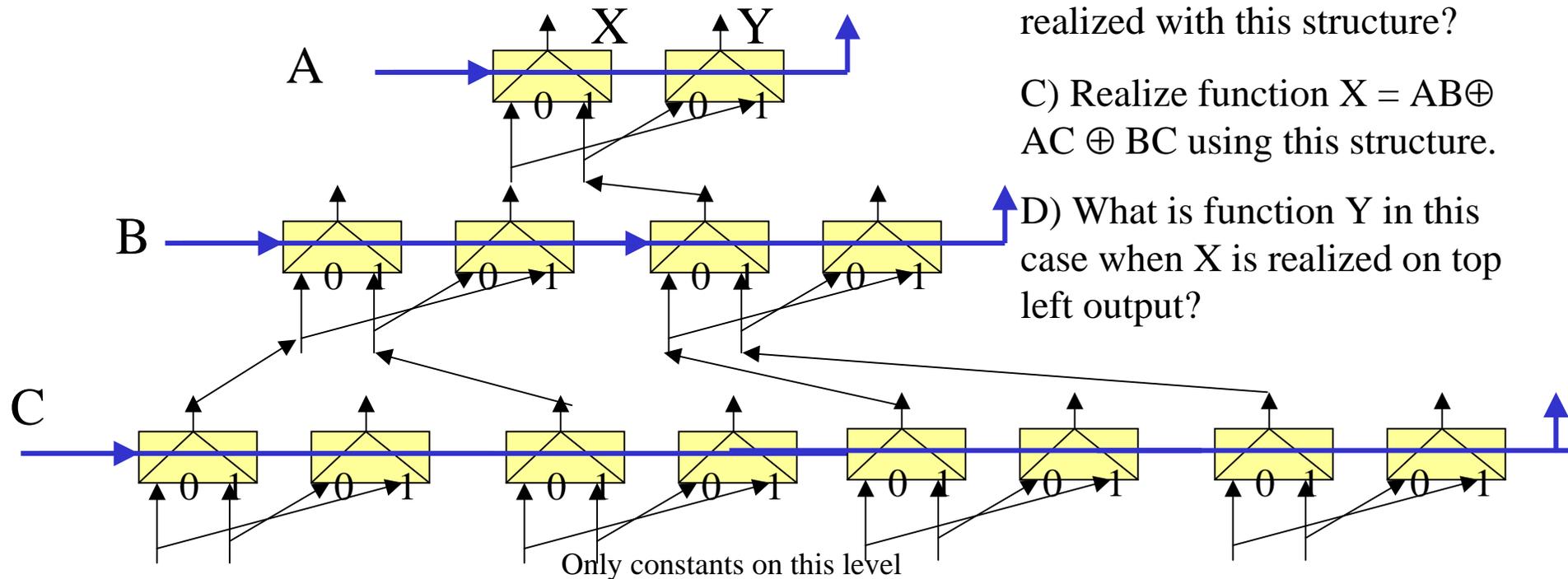


A) Is this a reversible gate? If yes, which gate?

B) Prove that arbitrary symmetric function  $X(A,B,C)$  can be realized with this structure?

C) Realize function  $X = AB \oplus AC \oplus BC$  using this structure.

D) What is function  $Y$  in this case when  $X$  is realized on top left output?



# Solution to Problem 6

A) Is this a reversible gate? If yes, which gate? **Fredkin**

B) Arbitrary symmetric function  $X(A,B,C)$  can be realized with this structure because it includes in itself the binary tree of multiplexers. So not only symmetric but arbitrary function can be realized by assigning constants to the lowest level data inputs

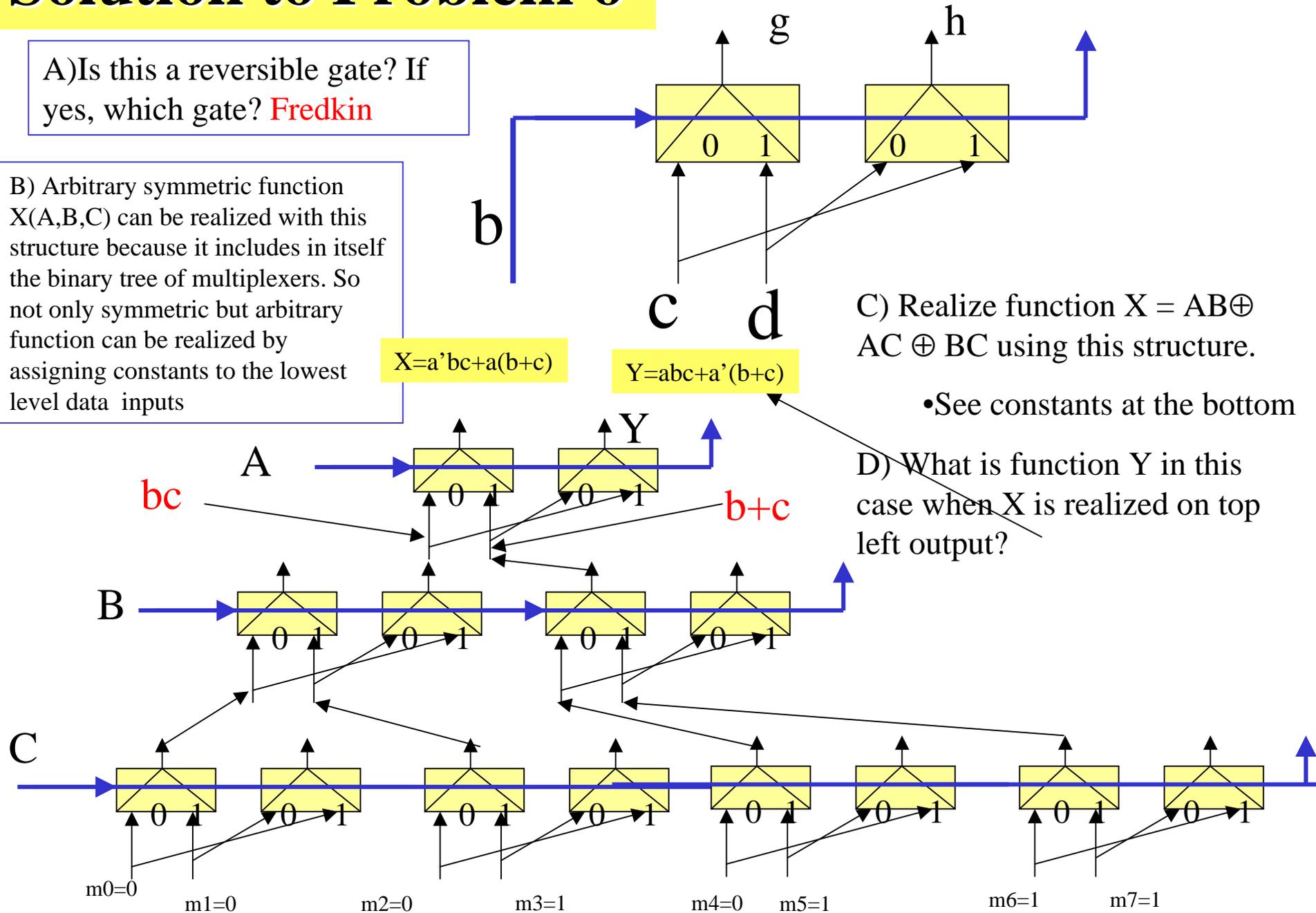
$$X = a'bc + a(b+c)$$

$$Y = abc + a'(b+c)$$

C) Realize function  $X = AB \oplus AC \oplus BC$  using this structure.

• See constants at the bottom

D) What is function  $Y$  in this case when  $X$  is realized on top left output?



## Problem 7

- Realize Converter from Gray code to Binary Natural Code using only reversible gates. Try to minimize Garbage.

## Problem 8

- Discuss the importance of work of Bennett and Landauer related to reversible logic.

# Solution to Problem 7

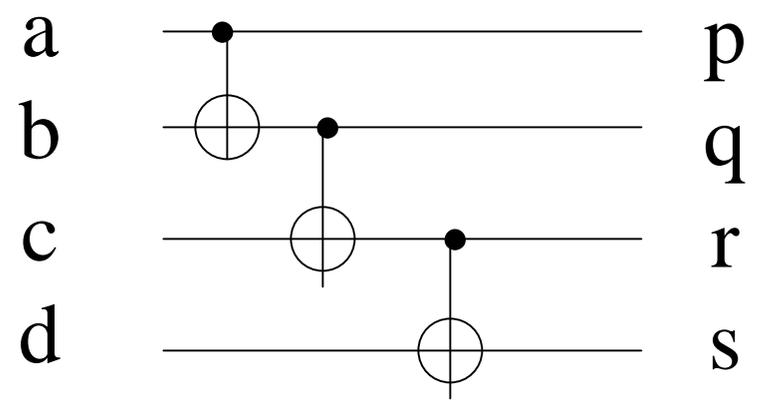
- Realize Converter from Gray code to Binary Natural Code using only reversible gates. Try to minimize Garbage.

inputs	outputs	cd				
		ab	00	01	11	10
0000	0000	00	0	0	0	0
0001	0001	01	0	0	0	0
0010	0011	11	1	1	1	1
0011	0010	10	1	1	1	1

p			
00	0	0	1
01	1	1	0
11	0	0	1
10	1	1	0

q			
00	0	1	0
01	1	0	1
11	0	1	0
10	1	0	1

$p = a$   
 $q = a \oplus b$   
 $r = a \oplus b \oplus c$   
 $s = a \oplus b \oplus c \oplus d$



# Solution to Problem 8

- Discuss the importance of work of Bennett and Landauer related to reversible logic.

## Landauer's Principle

In 1961, Landauer was considering the smallest amount of heat generated per bit processed in computation. He introduced the distinction of **logical and physical irreversibility**. He noted that a physical implementation of a logically irreversible process (defined to be one that cannot be logically reversed, i.e. undone by reversing the flow direction of computation) must be physically irreversible (i.e. cannot be undone or reversed to its prior physical state). A process is **logically reversible** if knowing the binary input to a logic gate, one can deduce the output and **vice versa**. Landauer discovered (rather surprisingly) that the heat coming from computation was **due to the destruction of information** (wiping out bits of information) and *not to the processing of bits*. This is **Landauer's Principle** which states:

- **logic computations that are not reversible, necessarily generate heat:**
  - **i.e.  $kT\log(2)$ , for every bit of information that is lost**, where  $k$  is Boltzmann's constant and  $T$  the temperature.
- For  $T$  equal room temperature, this package of heat is small, i.e.  $2.9 \times 10^{-21}$  joule, but non-negligible. In order to produce zero heat, a computer is only allowed to perform **reversible computations**.

Charles Bennett of IBM proved in 1973 in his famous paper “ Logical reversibility of computation” that there are no unavoidable energy consumption requirements per step in a computer. He discovered a way to make a reversible Turing Machine (by adding a history tape that gets written on and then unwritten (made blank) at the end. Thus, the power dissipation of a reversible computer, under ideal physical circumstances, is zero.

Reversible computing implies no information is wiped out, hence a history of all calculations is kept, then is reversibly restored to its original state. The hardware of a reversible computer cannot be constructed from the conventional gates. On the contrary, it consists exclusively of logically reversible building blocks. If reversible logic gates are computationally universal, then one can build computers based on them which should also be reversible, contradicting Landauer’s original conclusion. Reversible gate is a necessary but not sufficient condition of losing no power. Therefore we have to learn how to design arbitrary logic circuits from reversible gates and we have to build physically the reversible gates. The research is on both.

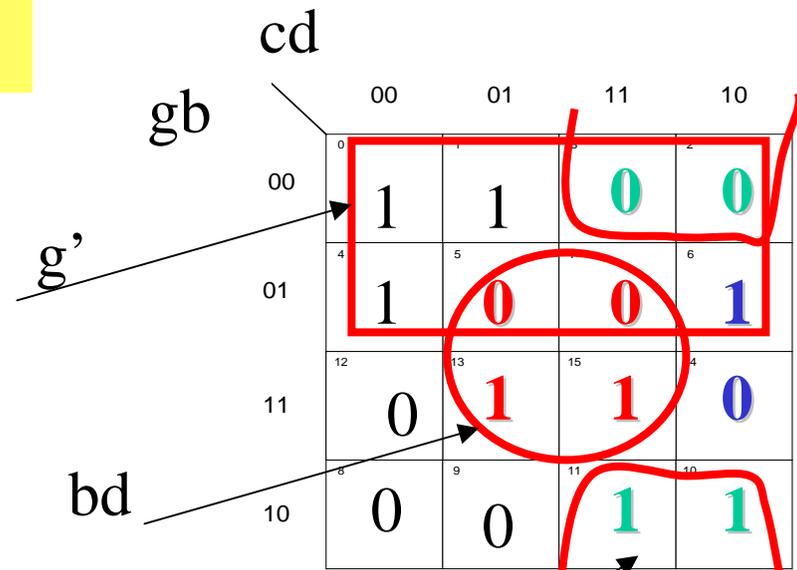
# Problem 9

- A) Realize ESOP for the function shown in Kmap. Minimize the number of gates and inputs to gates.
- B) Write an equivalent Positive Polarity Reed-Muller form for this function.
- C) Draw a reversible cascade in quantum notation in which function  $F(g,b,c,d)$  is one of outputs, and other outputs are arbitrary. Decrease the width of this cascade. Minimize Garbage.

		cd			
		00	01	11	10
gb	00	0 1	1 1	3 0	2 0
	01	4 1	5 0	7 0	6 1
	11	12 0	13 1	15 1	14 0
	10	8 0	9 0	11 1	10 1

# Solution to Problem 9

- A) Realize ESOP for the function shown in Kmap. Minimize the number of gates and inputs to gates.
- B) Write an equivalent Positive Polarity Reed-Muller form for this function.
- C) Draw a reversible cascade in quantum notation in which function  $F(g,b,c,d)$  is one of outputs, and other outputs are arbitrary. Decrease the width of this cascade. Minimize Garbage.



For A)

From groups we have  
 $ESOP = g' \oplus bd \oplus b'c$

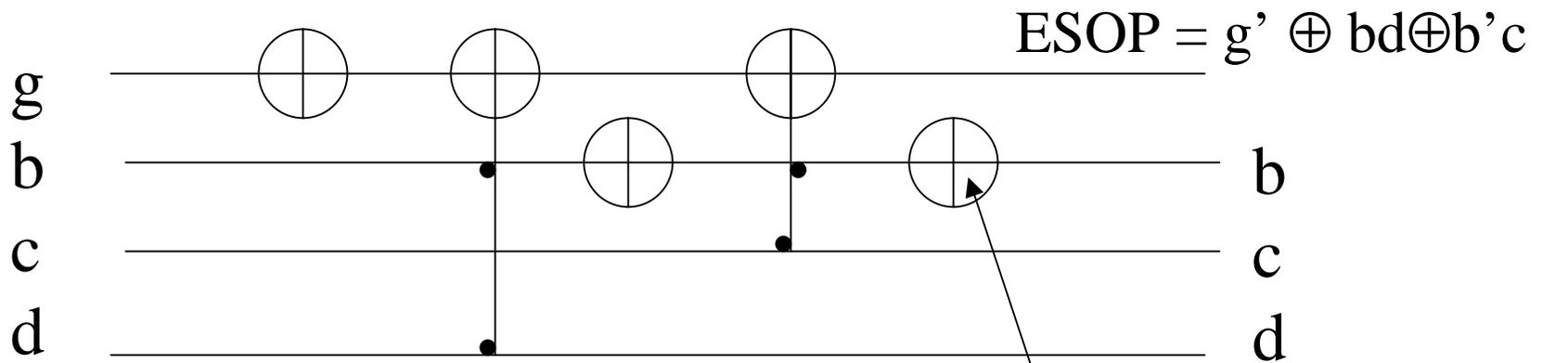
For B)

From ESOP we have PPRM =  $(g \oplus 1) \oplus bd \oplus (b \oplus 1)c = g \oplus 1 \oplus bd \oplus bc \oplus c$

# Solution to Problem 9

- C) Draw a reversible cascade in quantum notation in which function  $F(g,b,c,d)$  is one of outputs, and other outputs are arbitrary. Decrease the width of this cascade. Minimize Garbage.

From  $ESOP = g' \oplus bd \oplus b'c$  we directly can draw this quantum circuit



No garbage, shortest width. In other variant  $g$  is also available but width is 5 and all 4 inputs are restored, try to find it.

This NOT gate is not mandatory when inputs are not required to be restored

# Problem 10

cba	YZ			
	00	01	11	10
000	0	0	-	-
001	0	0	-	-
011	0	-	-	0
010	1	-	-	1
110	-	1	1	-
111	-	1	1	-
101	-	-	1	1
100	-	-	0	0

A) Groups shown in the map correspond to prime terms of an SOP. Draw a BDD of this function.

$$f = a'b + bz + b'ay$$

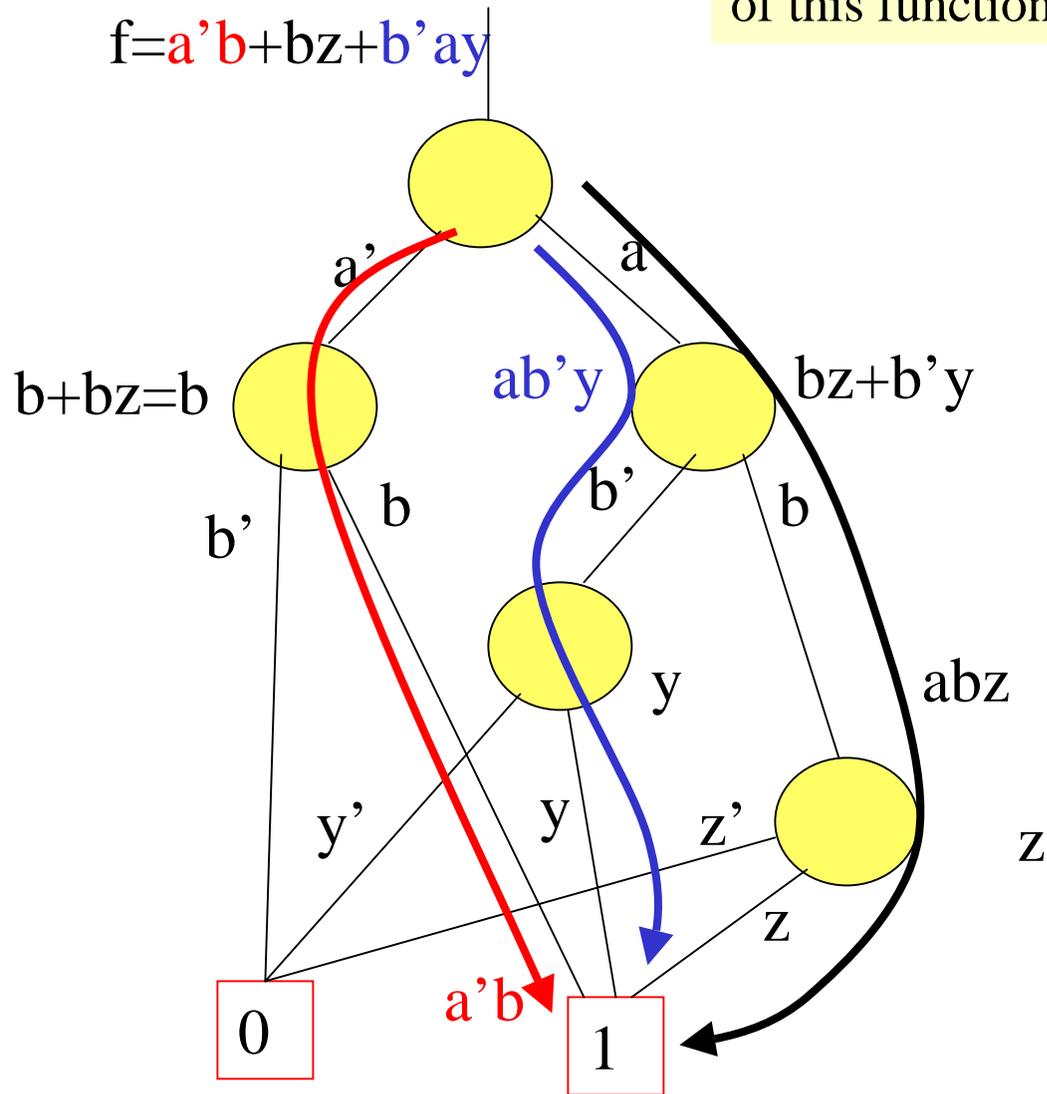
B) Groups shown in the map correspond to product terms in an ESOP. Draw a KFDD of this function that uses only Positive Davio gates.

$$f = a'b \oplus bz \oplus b'ay$$

C) Write the Positive Polarity Reed Muller form and ESOP expression.

# Solution to Problem 10

A) Groups shown in the map correspond to prime terms of an SOP. Draw a BDD of this function.



Remember to verify always a BDD as shown here by finding all paths and comparing with initial description. Observe that  $a'bz$  is included in first path  $a'b$



# Problem 11

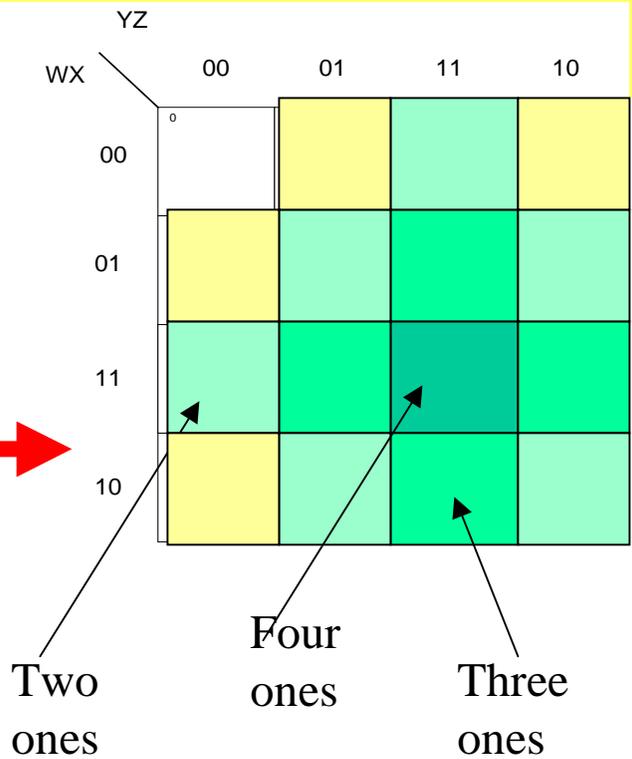
- A) How to recognize a symmetric function in a Karnaugh Map?
- B) How to recognize a linear function in a Karnaugh Map?
- C) Give an example of Kmap of 4 variable function that is both linear and symmetric?
- D) Can you give an example of function that is linear but not symmetric? If not, why?
- E) Realize function  $S_{0,1,3,5}$   
(A,B,C,D,E) in the most efficient way.

	YZ			
WX	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

# Solution to Problem 11

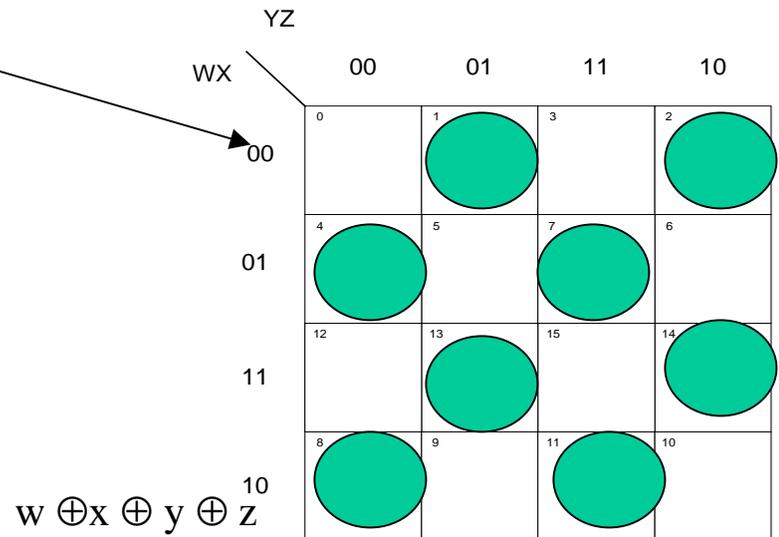
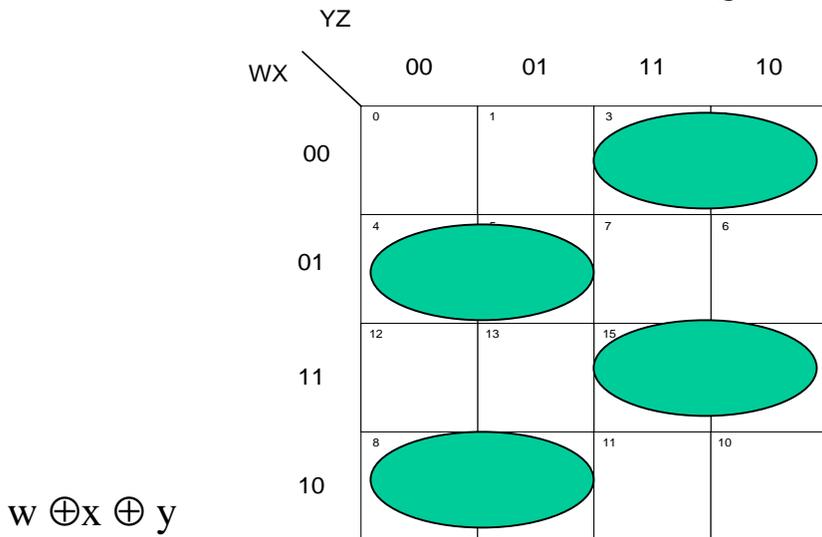
A) How to recognize a symmetric function in a Karnaugh Map?

For every cell described by  $k$  ones the values are the same or don't cares,  $k=0$  to 4. See the sets of cells with the same index (number of ones in argument) marked by the same color.



B) How to recognize a linear function in a Karnaugh Map?

It is an exor of a set of literals, looks like a chess pattern. Some are shown here.



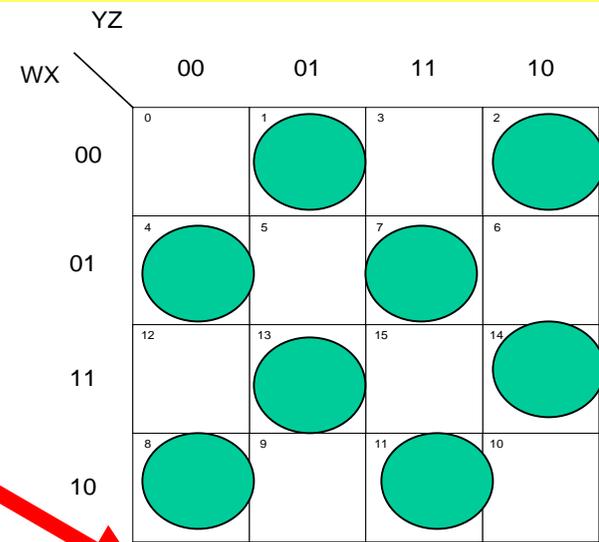
# Solution to Problem 11

C) Give an example of Kmap of 4 variable function that is both linear and symmetric?

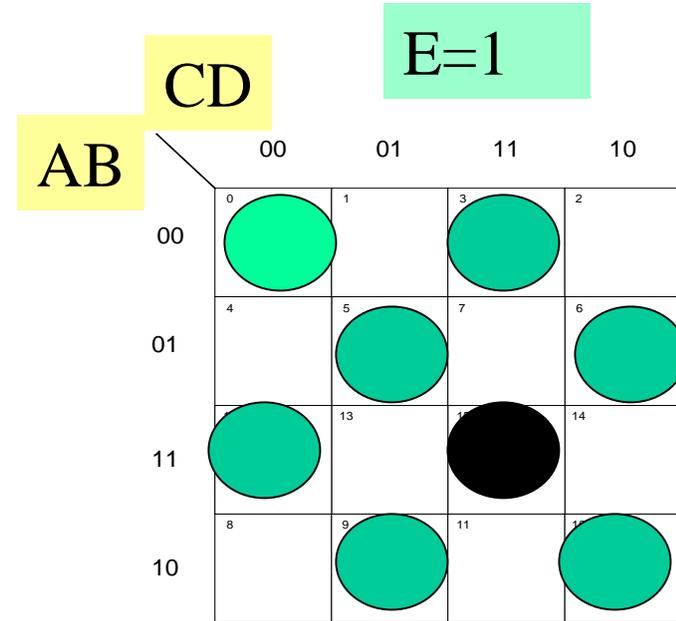
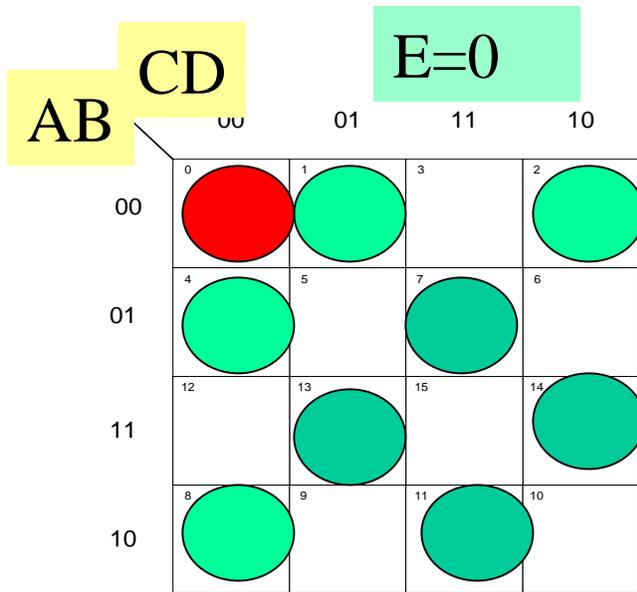
D) Can you give an example of function that is linear but not symmetric? If not, why?

**The function  $w \oplus x \oplus y$  is linear but is not a symmetric function of 4 variables. It is however a symmetric function of 3 variables from definition.**

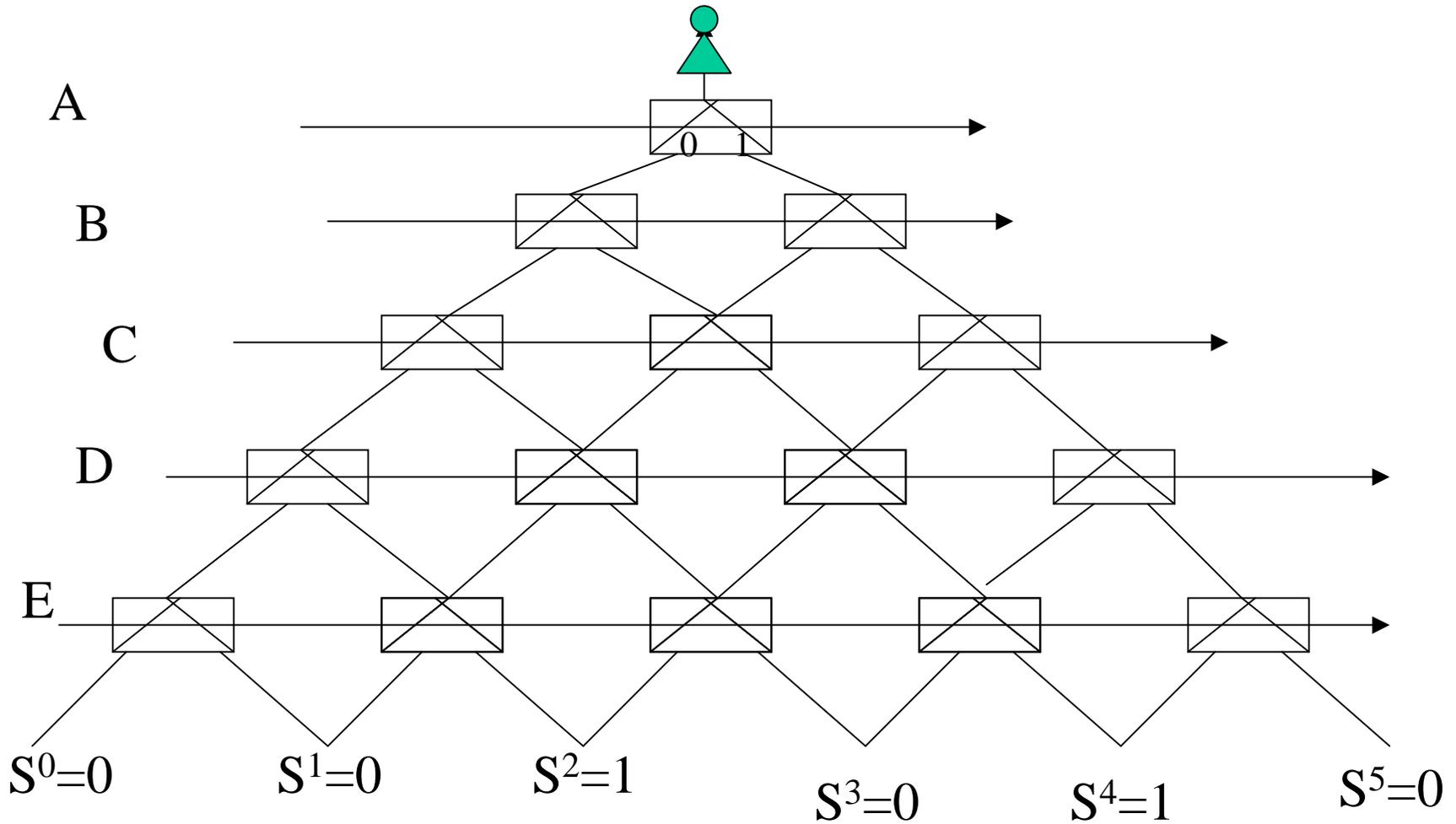
E) Realize function  $S^{0,1,3,5}(A,B,C,D,E)$  in the most efficient way.



$S^{1,3}(A,B,C,D)$



$S^{0,1,3,5}(A,B,C,D,E) = S^{2,4}(A,B,C,D,E)$  each of them can be realized



Some muxes can be removed and variables E and E' added. Check it! This is only one solution. You can also use Davio Lattice, Reed-Muller Lattice or ESOP.

## Problem 12

- Given is a circuit described by equations:

- $A = ab + cd' + aef$

- $B = ab + cef + (d \oplus f)$

- $C = acd' + (a' fg + (bd') * (c + fd))$

A) Draw a reversible realization of this circuit with arbitrary gates.

B) Prove that your circuit is reversible

C) Find an inverse circuit

## Problem 13

Realize Margolus gate with a minimum number of Toffoli and Feynman gates (and inverters).

# Solution to Problem 12

- Given is a circuit described by equations:
  - $A = ab + cd' + aef$
  - $B = ab + cef + (d \oplus f)$
  - $C = acd' + (a'fg + (bd')) * (c + fd)$

A) Draw a reversible realization of this circuit with arbitrary gates.

B) Prove that your circuit is reversible

C) Find an inverse circuit

This is a rather big function, it has 6 variables. So the only method that you can practically apply is approximate, especially that intentionally you were not asked to minimize the circuit. We cannot find known parts of reversible gates either. So we just convert the circuit using Toffoli, Feynman gates and many constants and garbage.

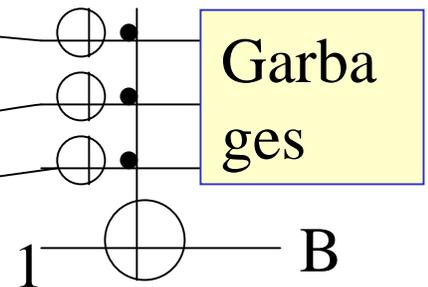
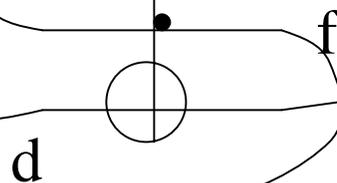
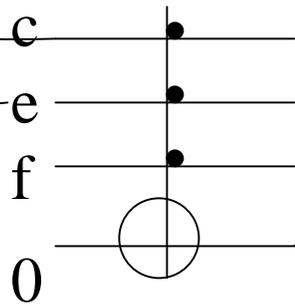
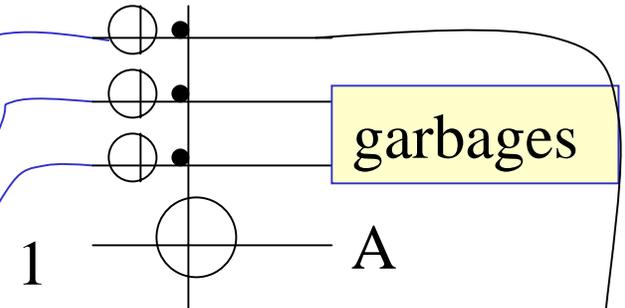
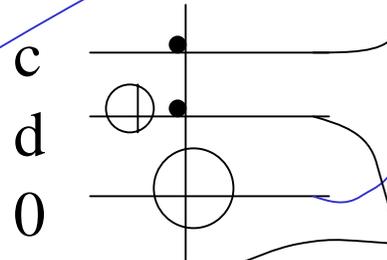
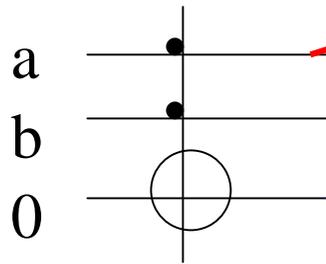
# Solution to Problem 12

- We get equations:

- $A = ab + cd' + aef$

- $B = ab + cef + (d \oplus f)$

- $C = acd' + (a'fg + (bd')) * (c + fd)$



Now you understand the method and you can realize also function C similarly.

# Solution to Problem 12

B) Prove that your circuit is reversible

The circuit is composed from reversible gates and satisfies axioms (no fan-out, no loops) of reversible circuit. So it is reversible. Of course, with a lot of constants and garbages.

C) Find an inverse circuit

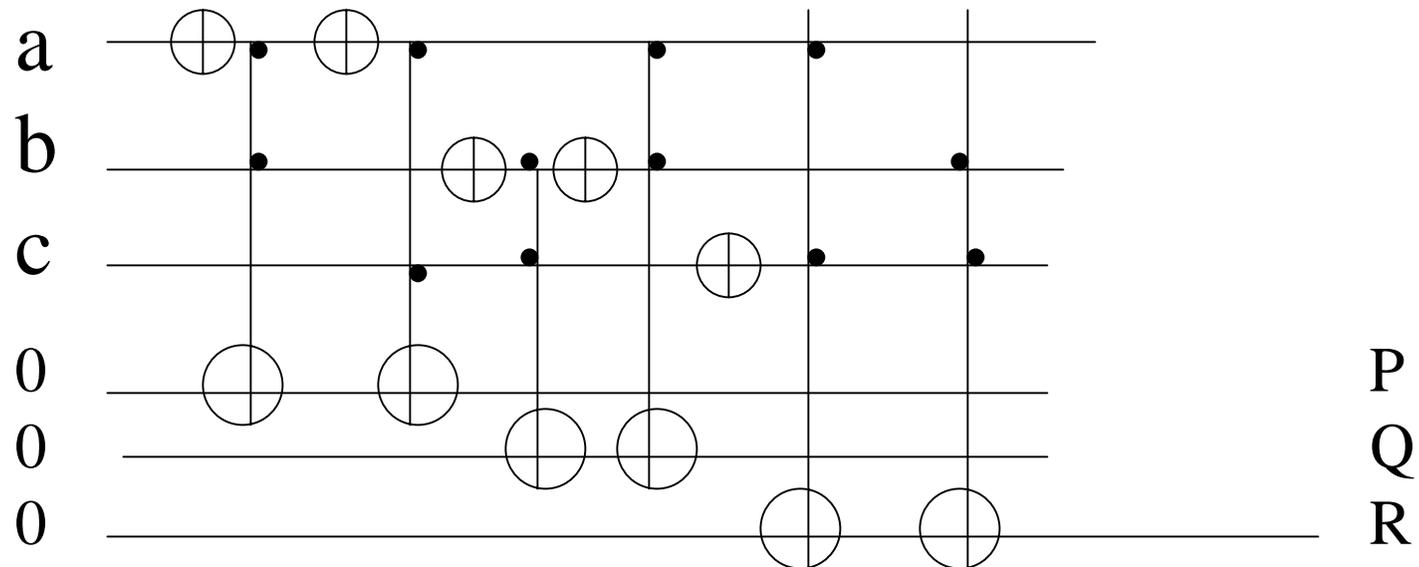
This can be achieved remembering that Toffoli is its own inverse, so you have just to draw a mirror circuit by mirroring your circuit to restore original inputs and constants. Use Feynman as spy circuit - as shown in class.

# Solution to Problem 13

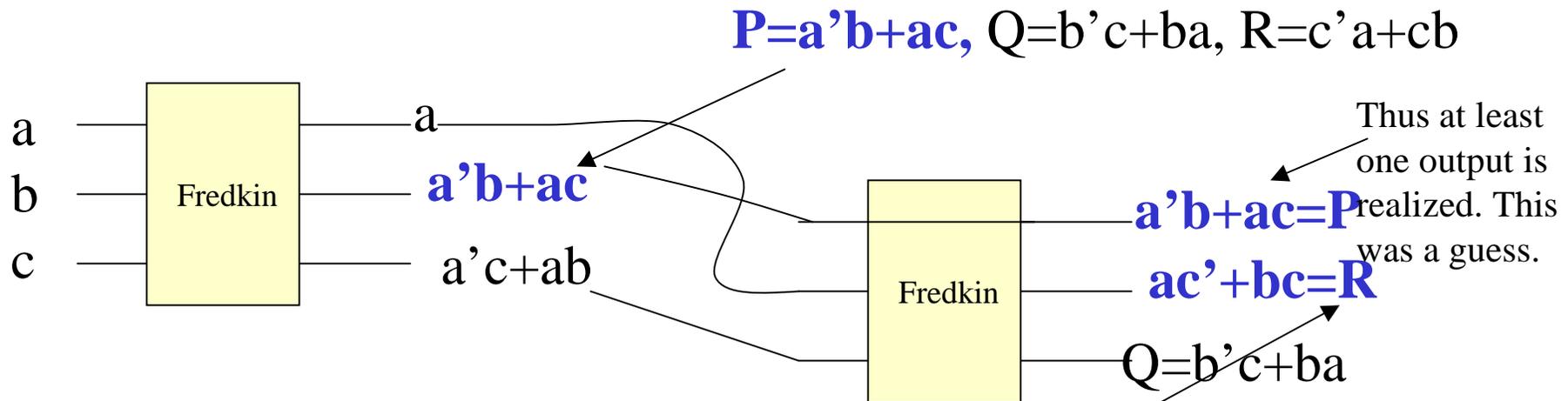
Realize Margolus gate with a minimum number of Toffoli and Feynman gates (and inverters).

Margolus:  $P=a'b+ac$ ,  $Q=b'c+ba$ ,  $R=c'a+cb$  - observe the shift property of this circuit. We do not call it symmetry, because this function is not symmetric, but we can expect certain kind of “regularity” or “symmetry” (not in Boolean sense) since of the cyclic place of signals a, b and c in the equations P, Q and R.

It would be easy to assume one garbage in each function and use standard ESOP method, like this:



We have however enough knowledge to design this gate in a much smarter way, decreasing the width and the length.



$$(a'b+ac)'a + (a'b+ac)(a'c+ab) =$$

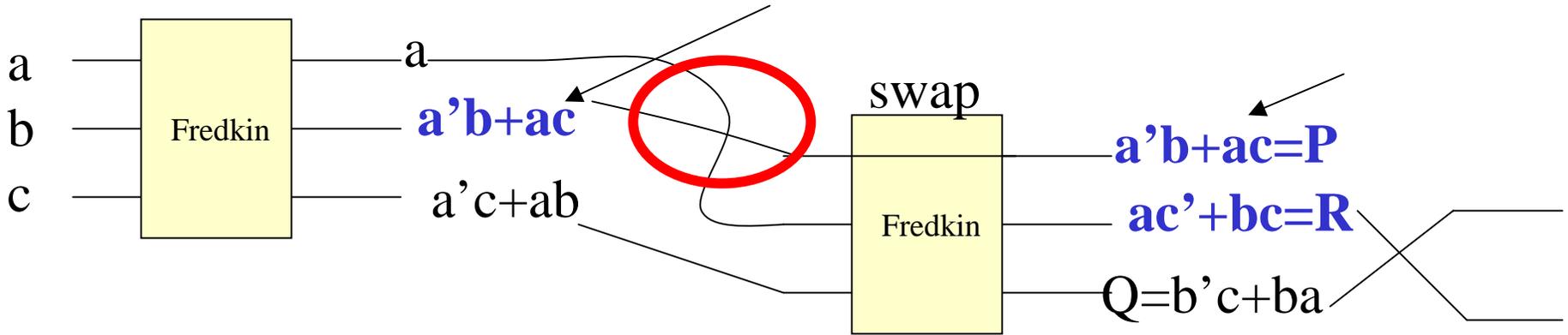
after standard De Morgan and  
Boolean algebra =  $ac'+bc$

We see that our guess was good with accuracy to output signal permutation.  
Let us now check the last signal.

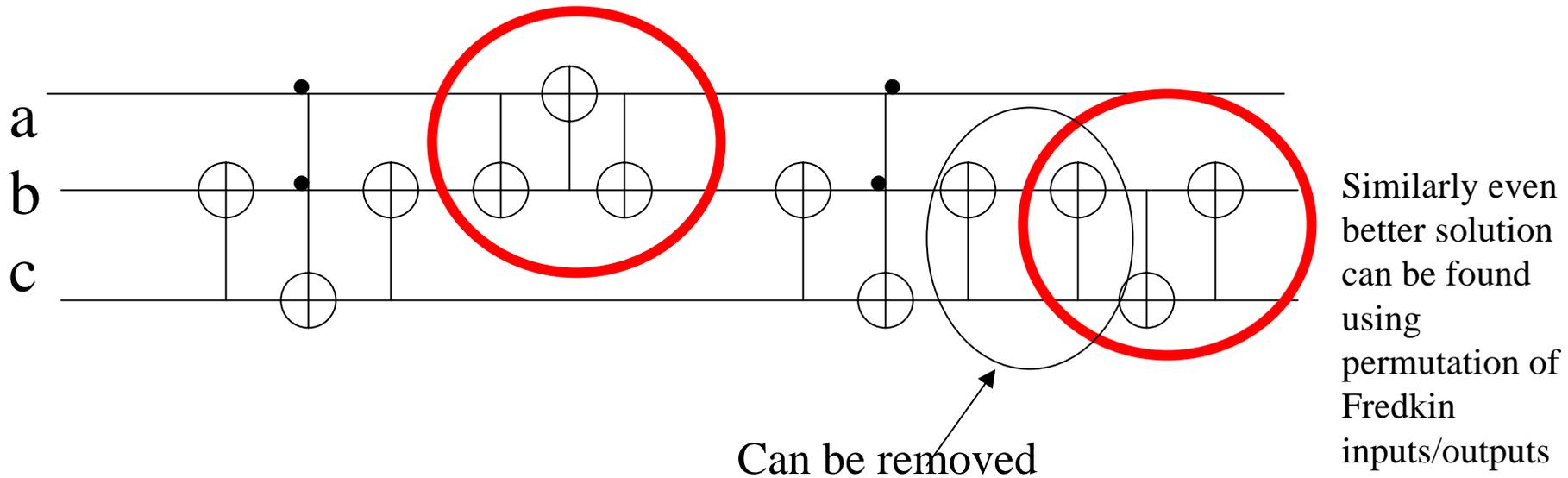
$$(a'b+ac)a + (a'b+ac)'(a'c+ab) =$$

after standard De Morgan and Boolean algebra =  $ab+b'c=Q$  so our guess was good. We just need swap gates.

$$P=a'b+ac, Q=b'c+ba, R=c'a+cb$$

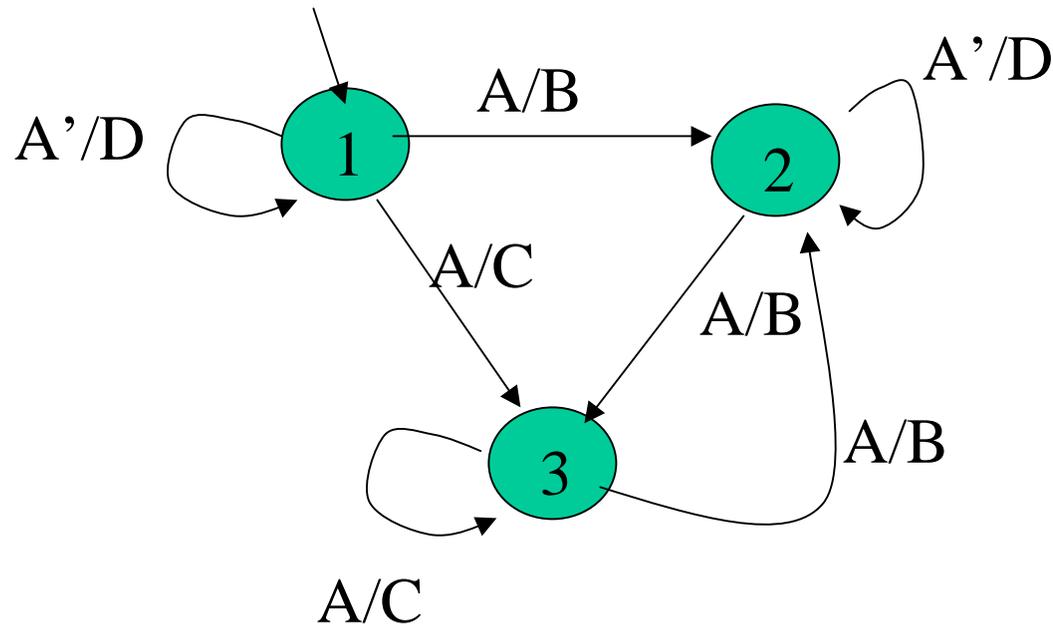


Now we can rewrite to quantum notation



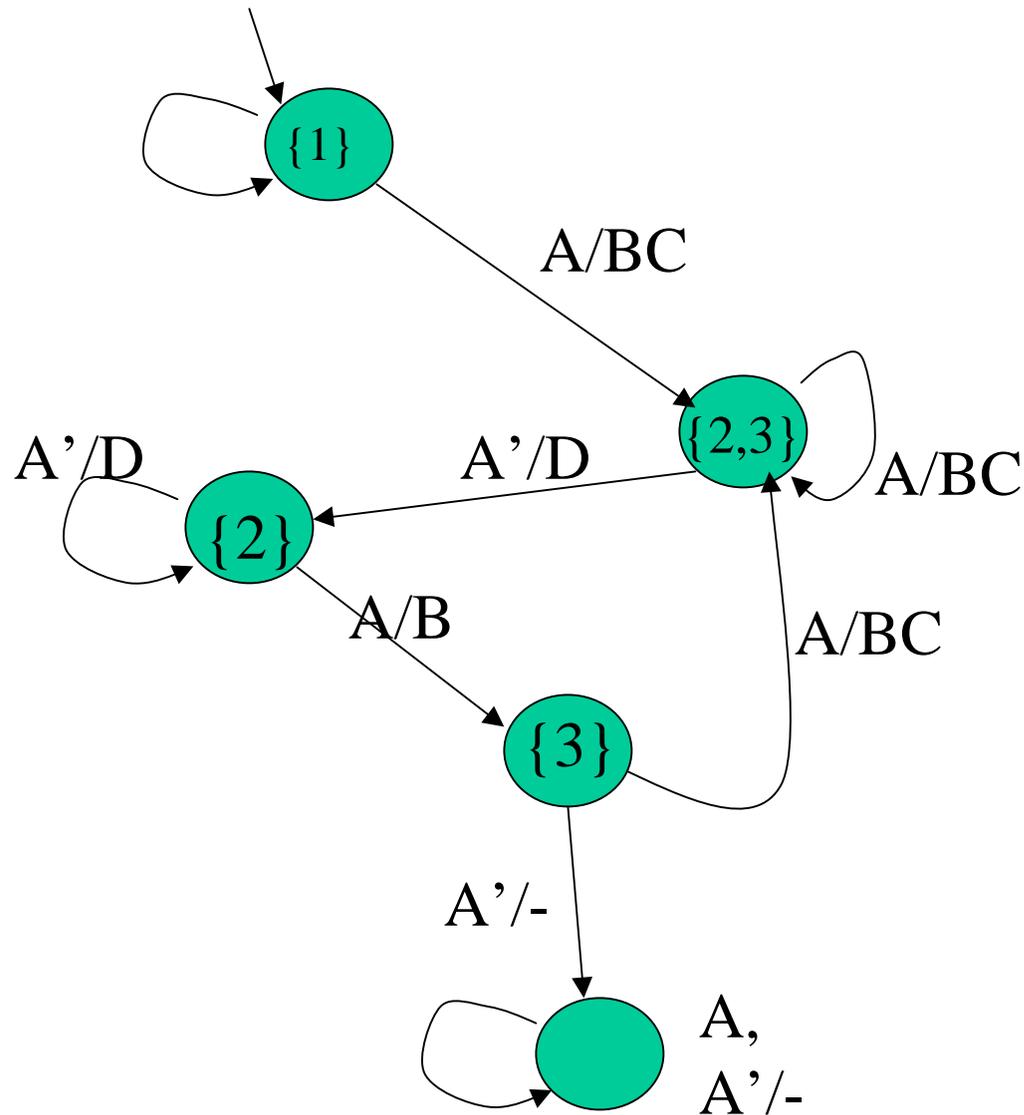
# Problem 14

- A) Convert non-deterministic machine to a deterministic one.
- B) Realize the non-deterministic (Mealy) machine using standard one-hot code realization shown in class.



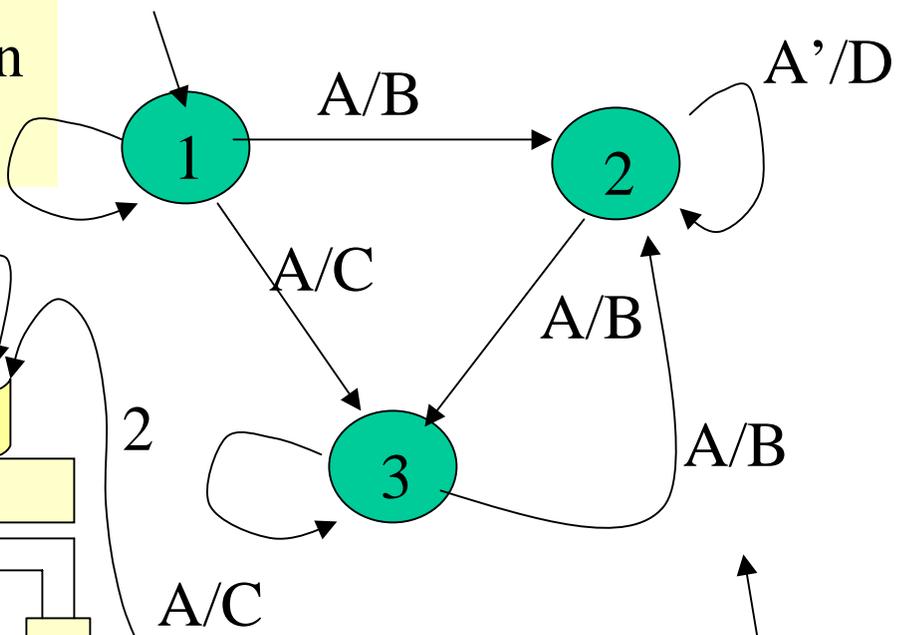
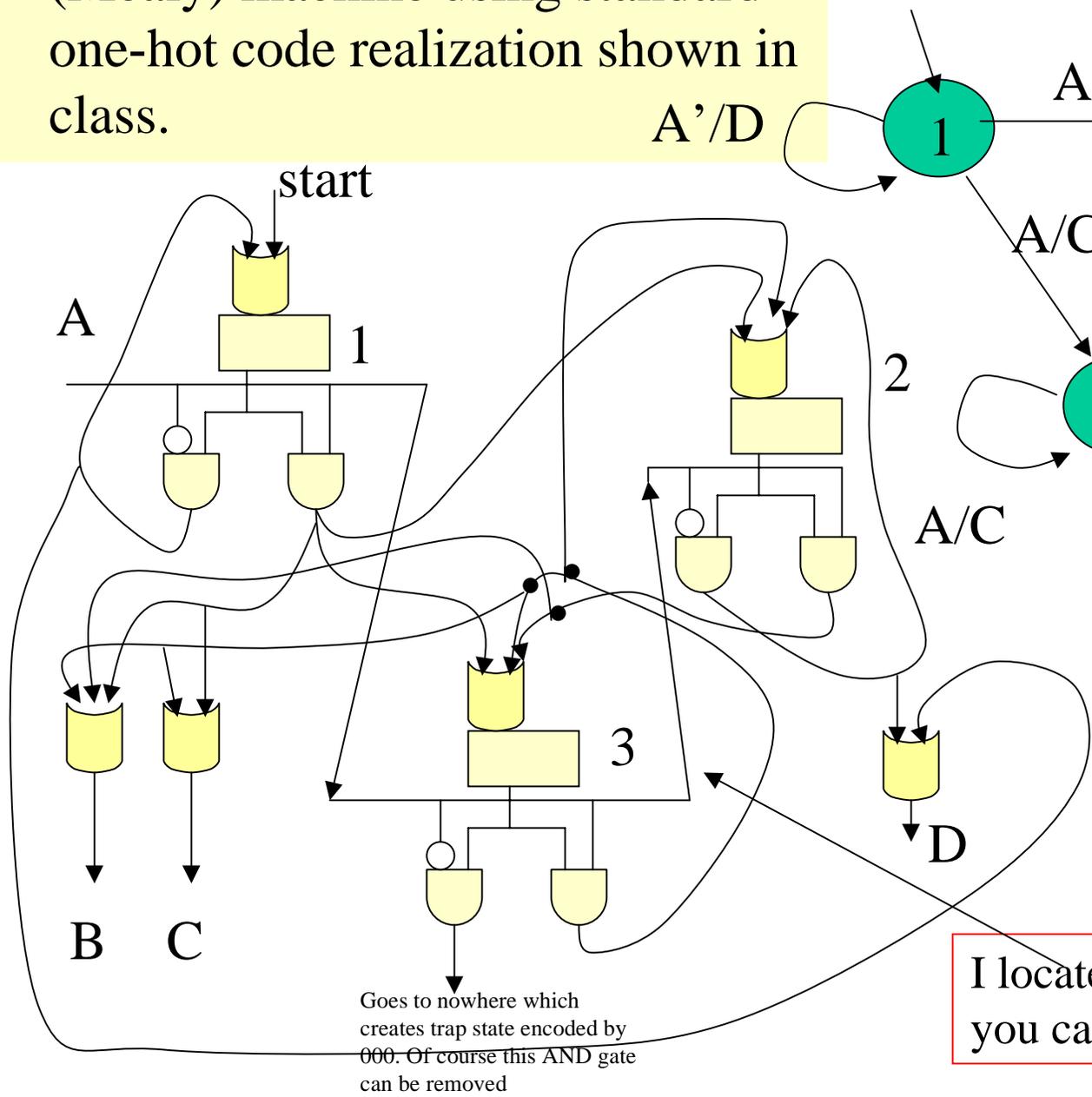
# Solution to Problem 14

- A) Convert non-deterministic machine to a deterministic one.



# Solution to Problem 14

- B) Realize the non-deterministic (Mealy) machine using standard one-hot code realization shown in class.



I located FFs in such positions that you can compare with the graph.

# Problem 15

A ternary function with inputs Z and X and outputs U and V that is specified by the map

Z	X <sub>0</sub>	1	2
0	0,0	0,1	0,2
1	1,1	1,2	1,0
2	2,2	2,0	2,1

U,V

A) Is this function symmetric?

B) Is this function reversible?

C) Draw this function using the minimum number of ternary reversible gates.

D) Using this function, draw a realization of a 3-qubit ternary linear function with the minimal number of gates.

E) Extend the concept of binary controlled gate to the concept of ternary controlled gate and show two examples of such gates.

# Solution to Problem 15

Z	X <sub>0</sub>	1	2
0	0,0	0,1	0,2
1	1,1	1,2	1,0
2	2,2	2,0	2,1

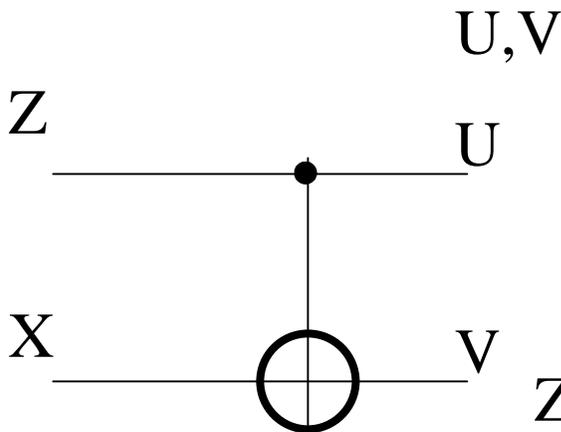
A) Is this function symmetric?

- Function U is not symmetric, Function V is symmetric.

B) Is this function reversible?

- Yes, by inspection of its Kmap or truth table.

C) Draw this function using the minimum number of ternary reversible gates.



Modulo 3  
addition of Z  
and X

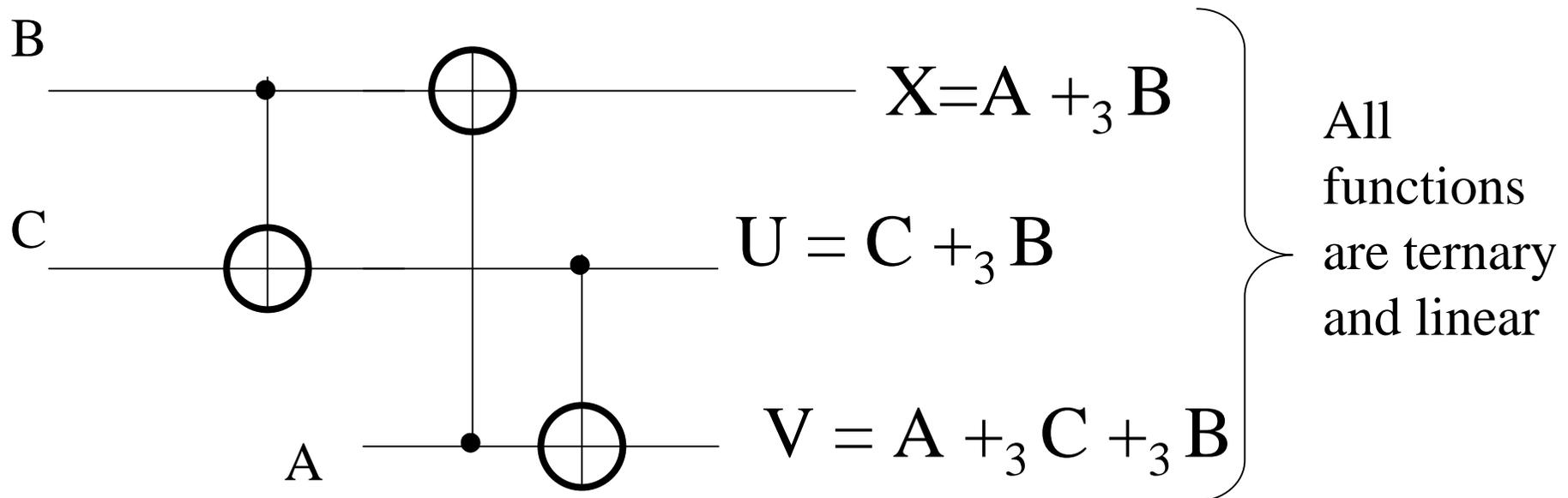
Z	X <sub>0</sub>	1	2
0	0	1	2
1	1	2	0
2	2	0	1

• Thus we created a ternary counterpart of Feynman Gate.

• It uses modulo-3 and not modulo-2 addition

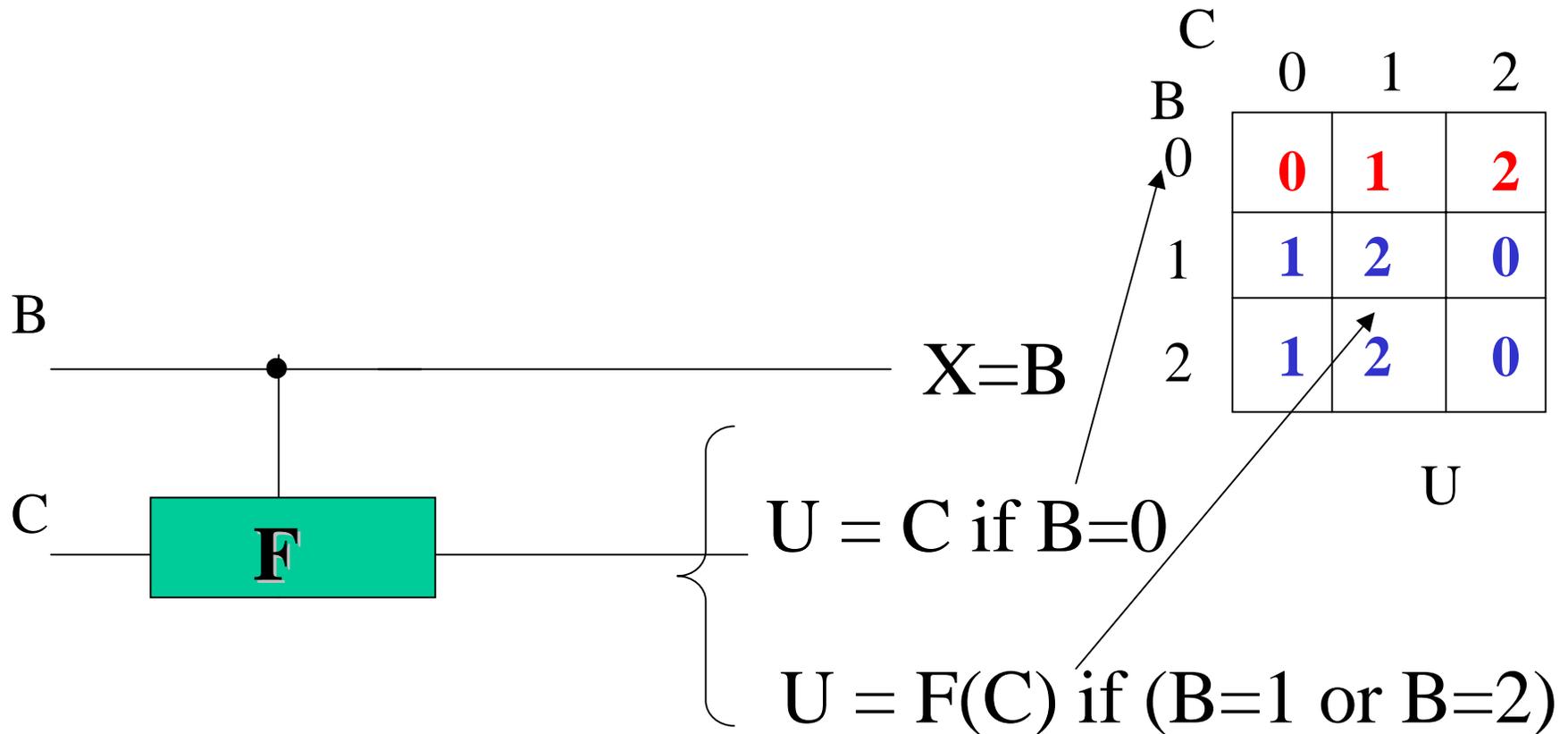
# Solution to Problem 15

D) Using this function, draw a realization of a 3-qubit ternary linear function with the minimal number of gates.



# Solution to Problem 15

E) Extend the concept of binary controlled gate to the concept of ternary controlled gate and show two examples of such gates.



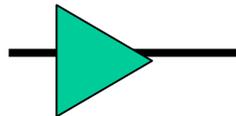
In this case,  $F = C +_3 1$

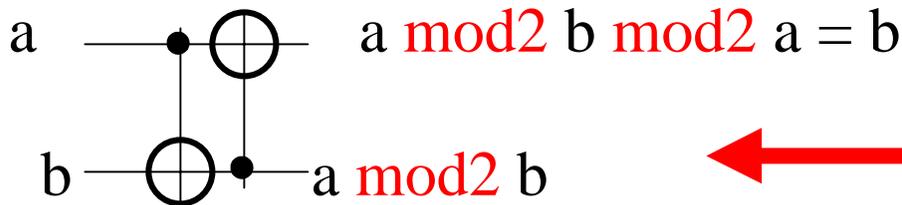
# Problem 16

- Realize a ternary swap gate with arbitrary gates that are generalizations of binary gates. Prove that it really works as a swap of arbitrary ternary signals.

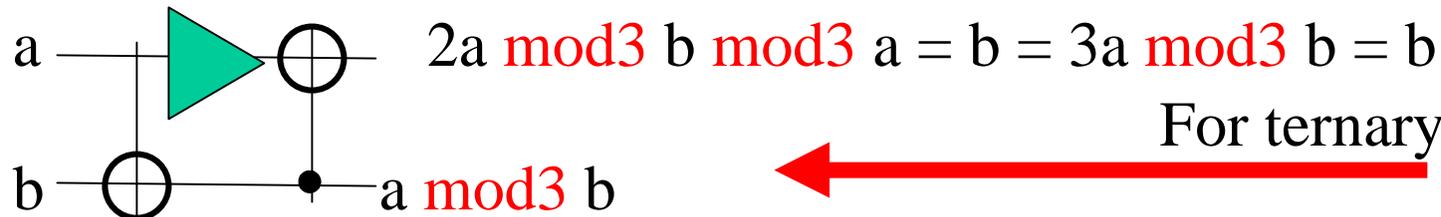
# Solution to Problem 16

- Realize a ternary swap gate with arbitrary gates that are generalizations of binary gates. Prove that it really works as a swap of arbitrary ternary signals.
- We solve this problem by analogy with binary swap that is done using Feynman gates:

 This gate multiplies by two



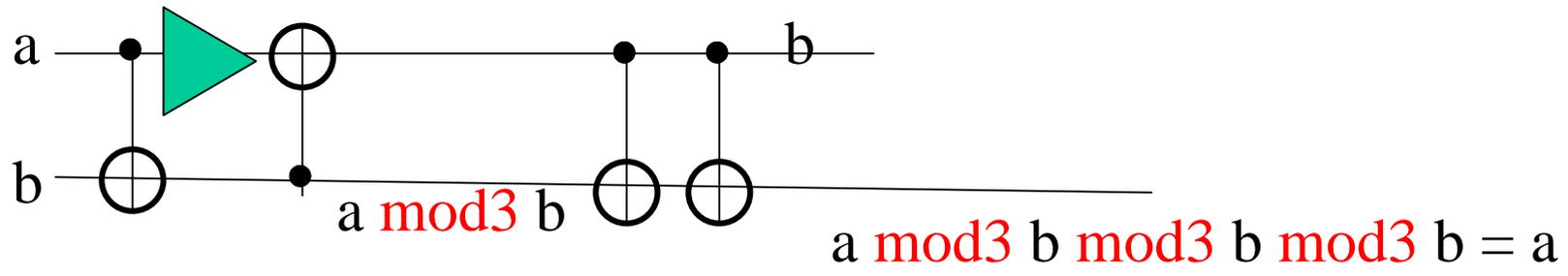
For binary



For ternary



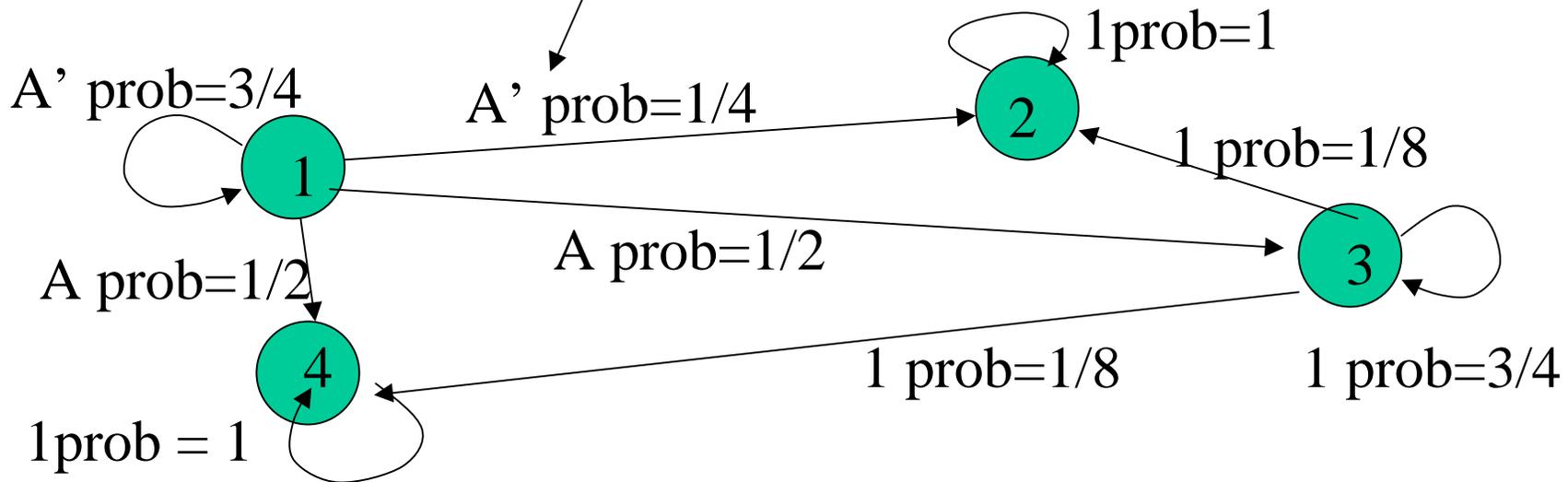
- In binary we are adding mod2 b to the bottom line.
- Thus to complete the ternary swap circuit we have to add 2b to the bottom line



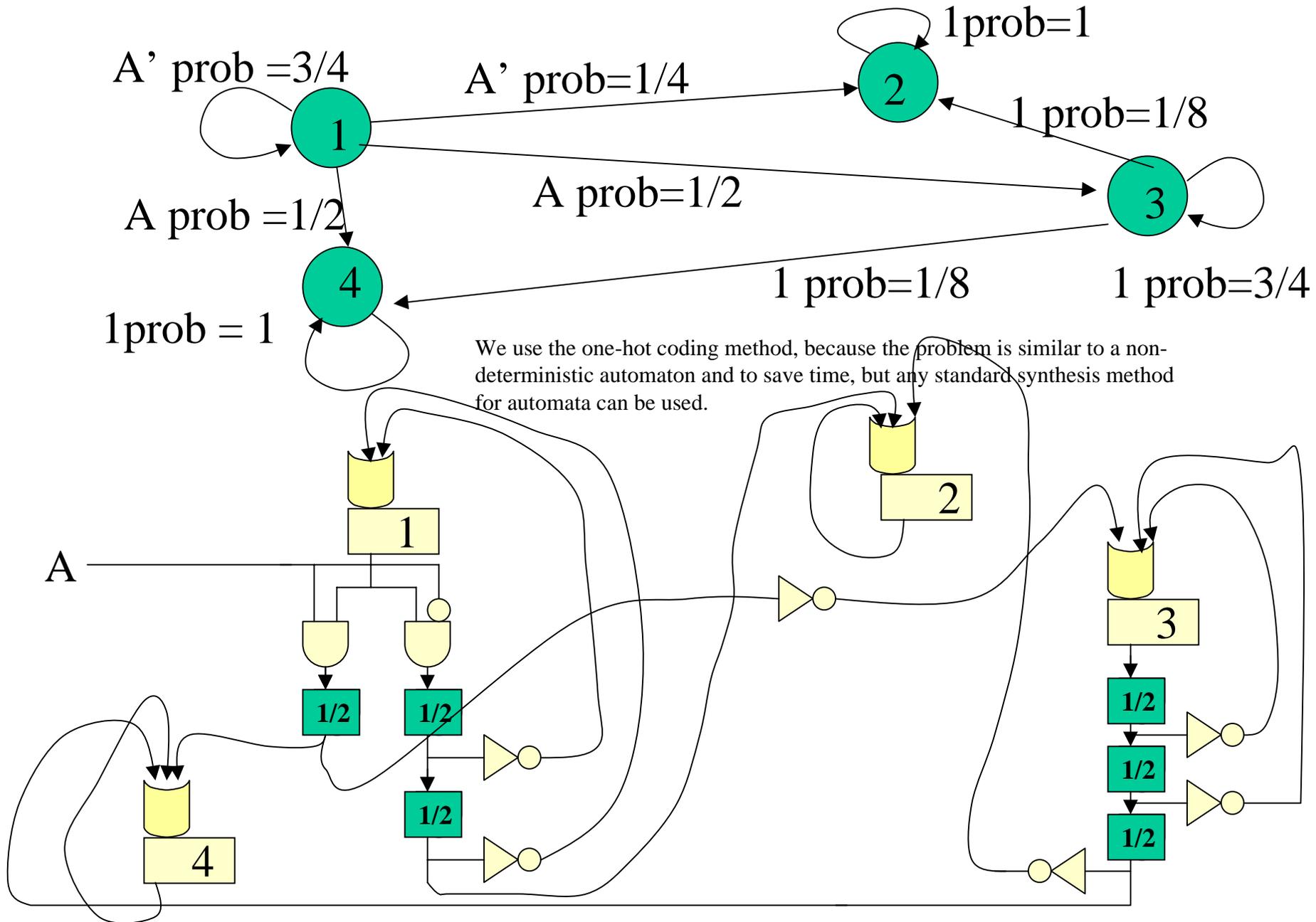
# Problem 17

- Assuming that you have a generator of probability  $1/2$  and arbitrary logic gates and flip-flops, realize the following probabilistic state machine.

Means transition  
with  $A=0$  input that  
has probability  $1/4$

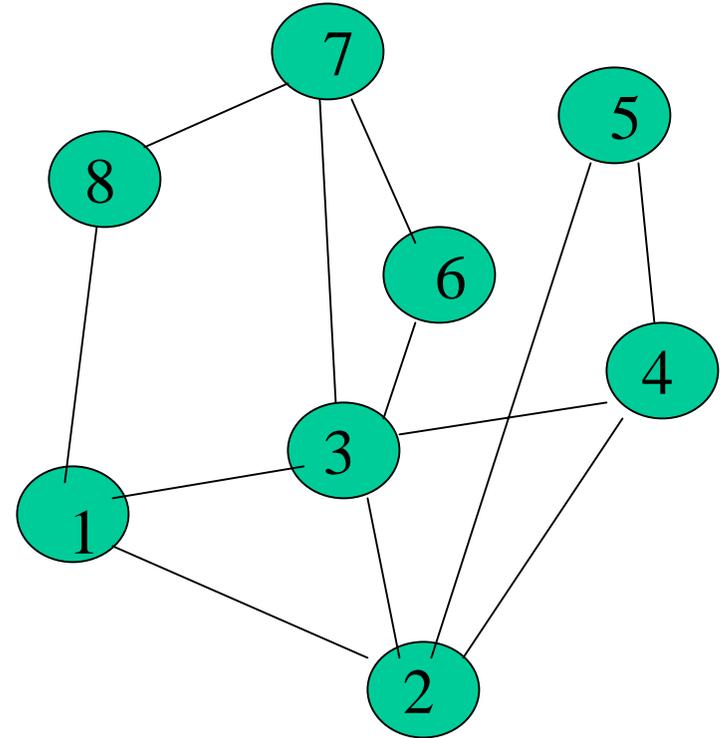


# Solution to Problem 17

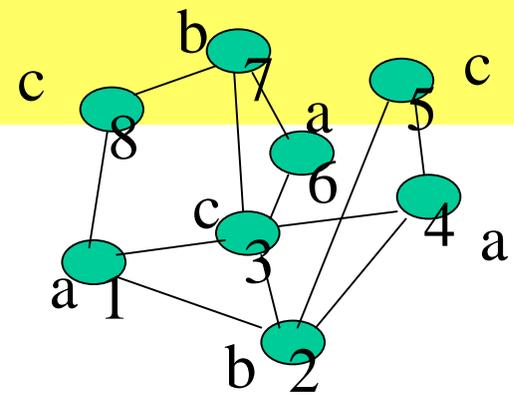


# Problem 18

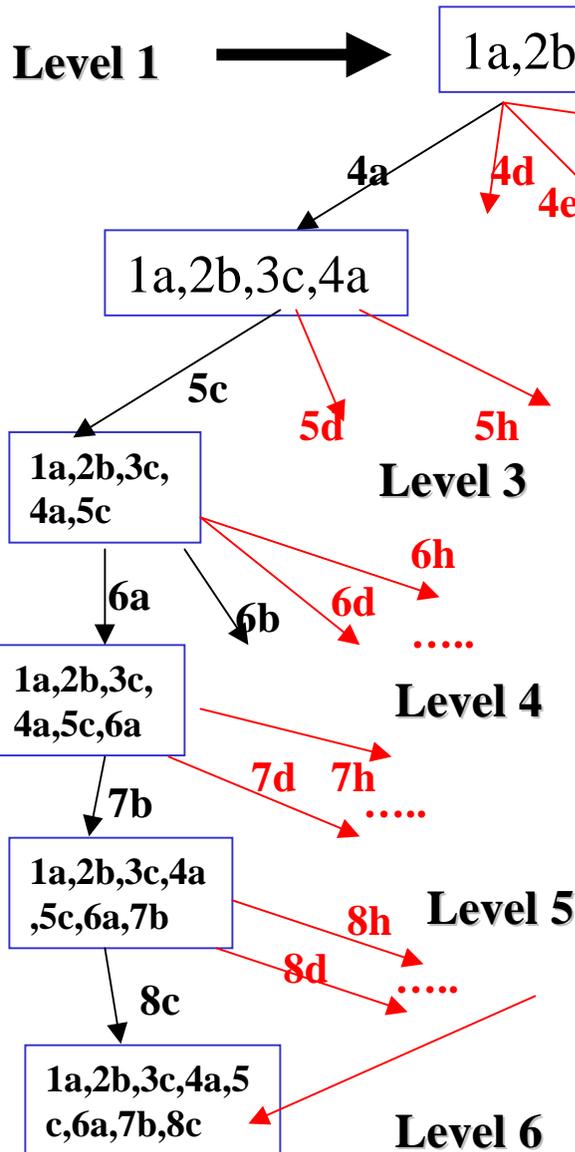
- Given is a graph.
- Show and explain a backtracking algorithm that finds the exact minimal coloring to this graph.
  - Or, if you do not know the backtracking tree-searching algorithm show any other algorithm to find the minimum coloring for arbitrary graphs.
  - You may use trees to explain operation of your algorithm.



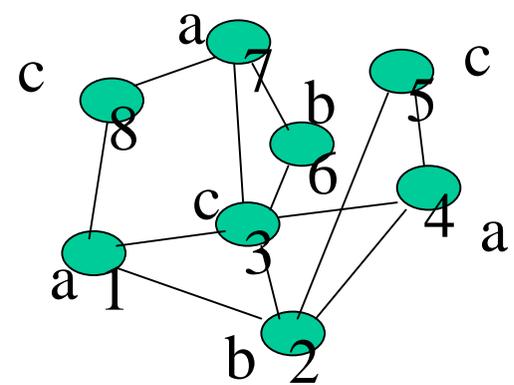
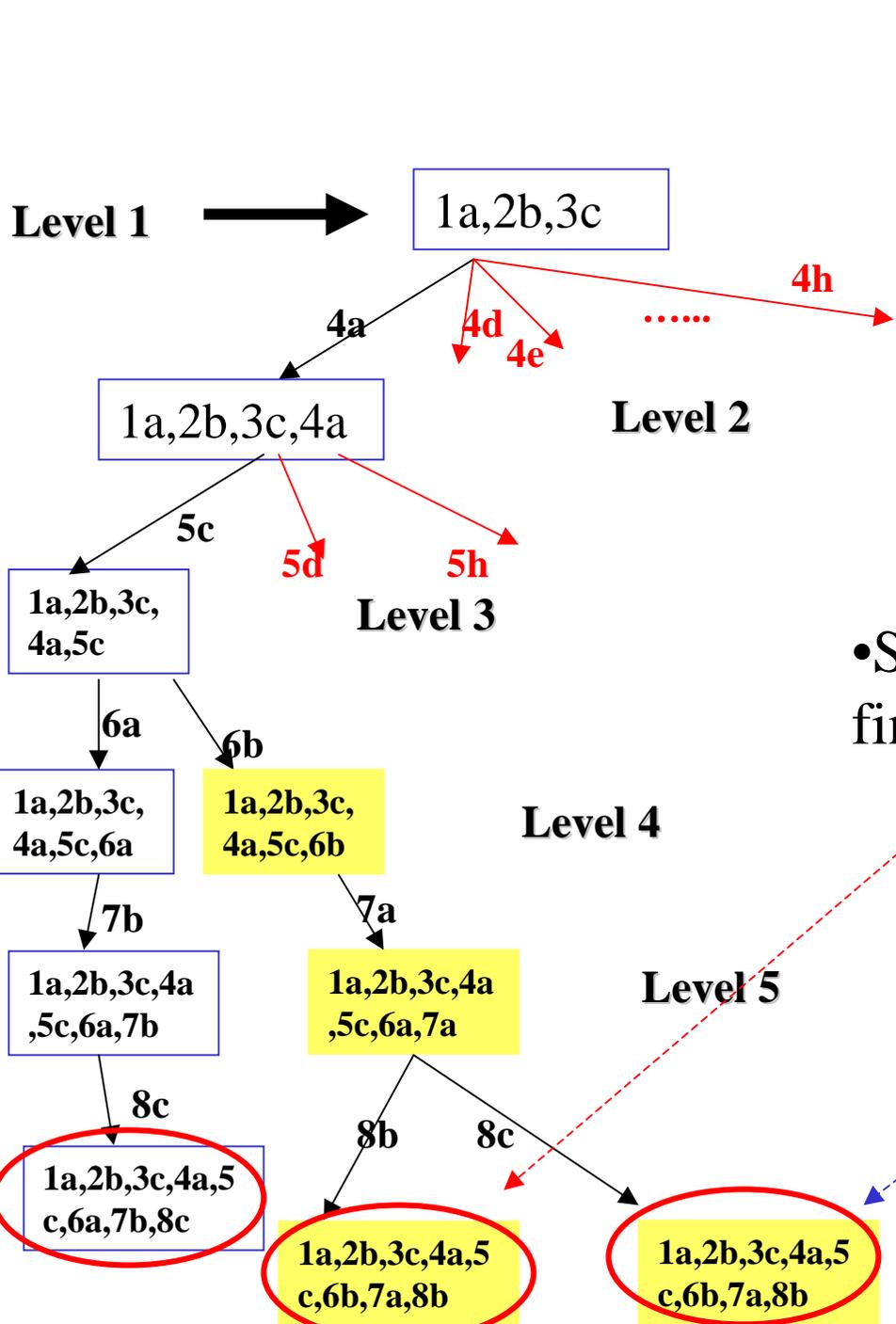
# Solution to Problem 18



Colors={a,b,c,d,e,f,g,h}



- With this 3-coloring we can stop since the number of colors cannot be smaller than 3, having clique with 3 nodes.
- However, this will not work in general. What we can always do is to remove from the graph all hanging nodes that have colors other than a,b,c. These are denoted by color red in the tree.
- When node at depth 6 is reached all nodes with arrows labeled with colors d,e,f,g,h are removed.
- This backtrack to node on depth 5 finds that there are no more possibilities to investigate since all arrows going out are red. Backtrack to node at level 4. Again all arrows going out are red so we backtrack
- Level 3. There is arrow 6b that is not red. We expand the tree in depth, see next slide.



Colors={a,b,c,d,e,f,g,h}

• Searching in depth of the tree we find **solution** and next **solution**

All other arrows are red so backtracking to levels 3, 2 and 1 will not cause creating next processes of searching the tree to depth, and the returned solutions (in circles) are optimal