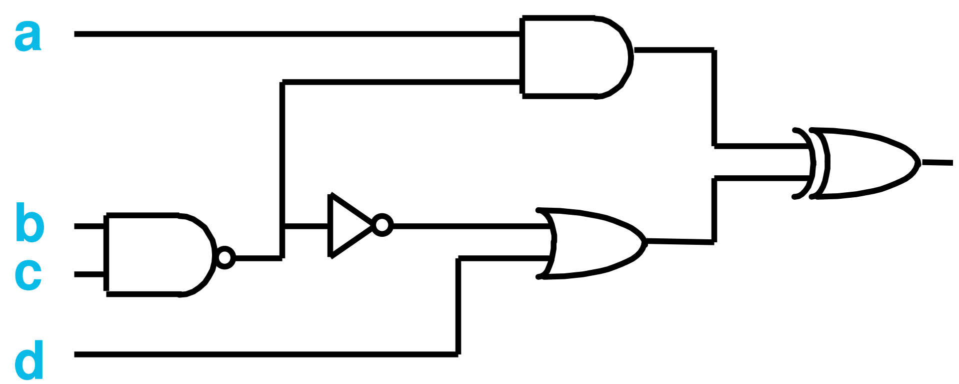
Timing Analysis

What is the delay of a combinational logic circuit?



Depends on:

- delay model
- topology of circuit
- Boolean behavior ←

Static Timing Analysis

Longest or shortest path, topologically speaking, is assumed to be responsible for circuit delay

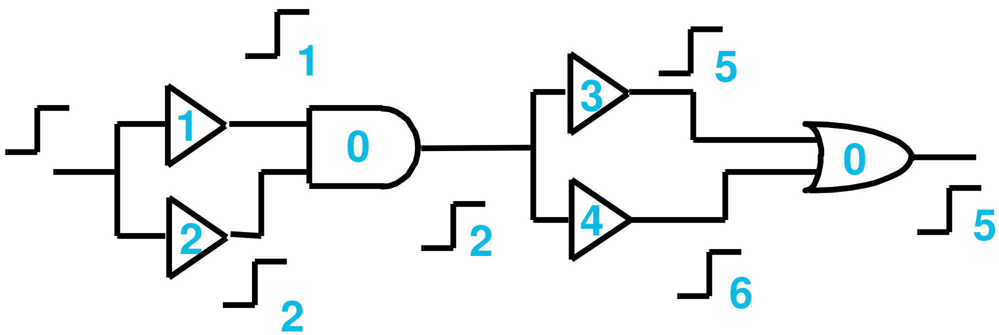
- $\Rightarrow$ **Can be needlessly pessimistic since paths can be false**

Parameter variations call for bounded gate delays [dmin, dmax] which current static timing verifiers do not handle

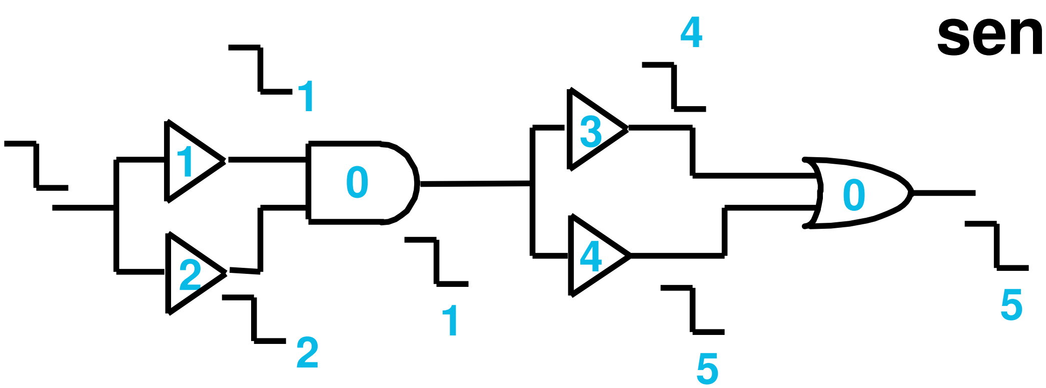
False Paths

A path is false if it cannot be responsible for the delay of a circuit

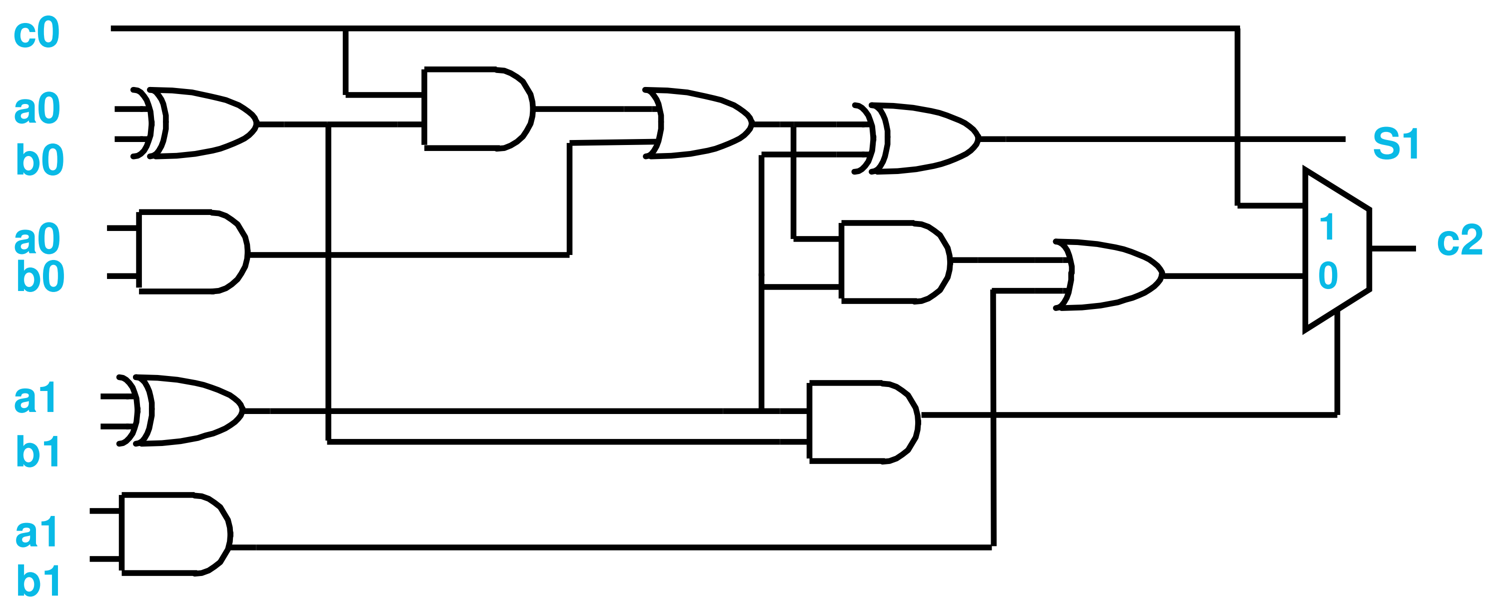
No sequence of vectors results in the propagation of an event along the path



Length of sensitized path = 5

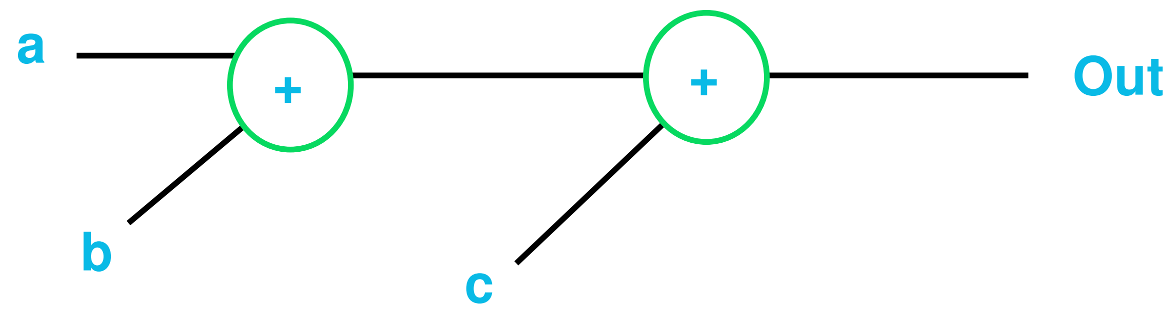


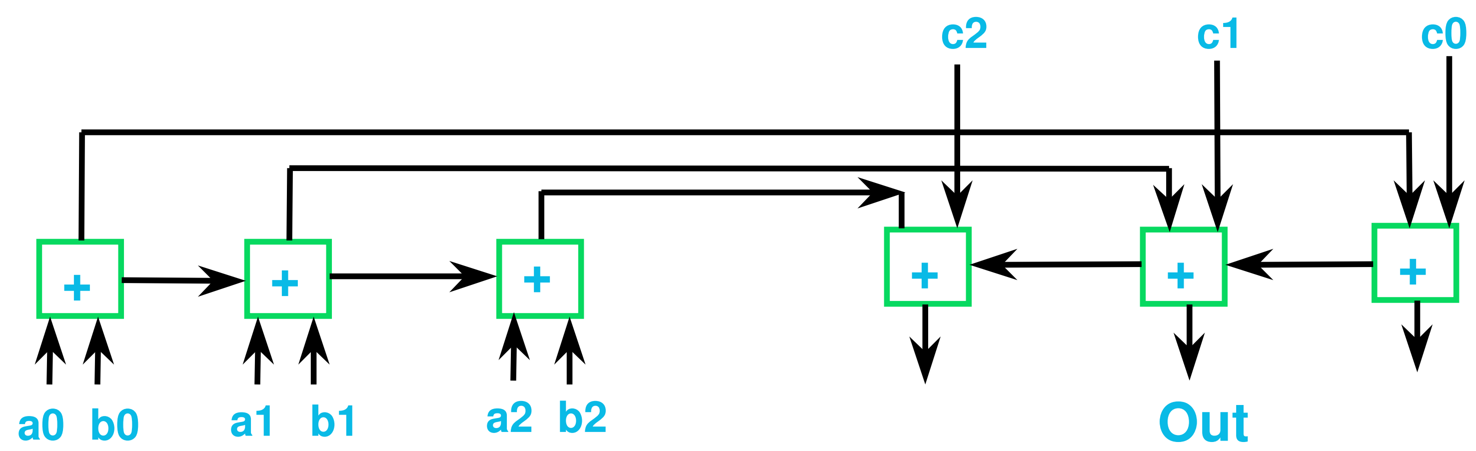
Where Do False Paths Arise?



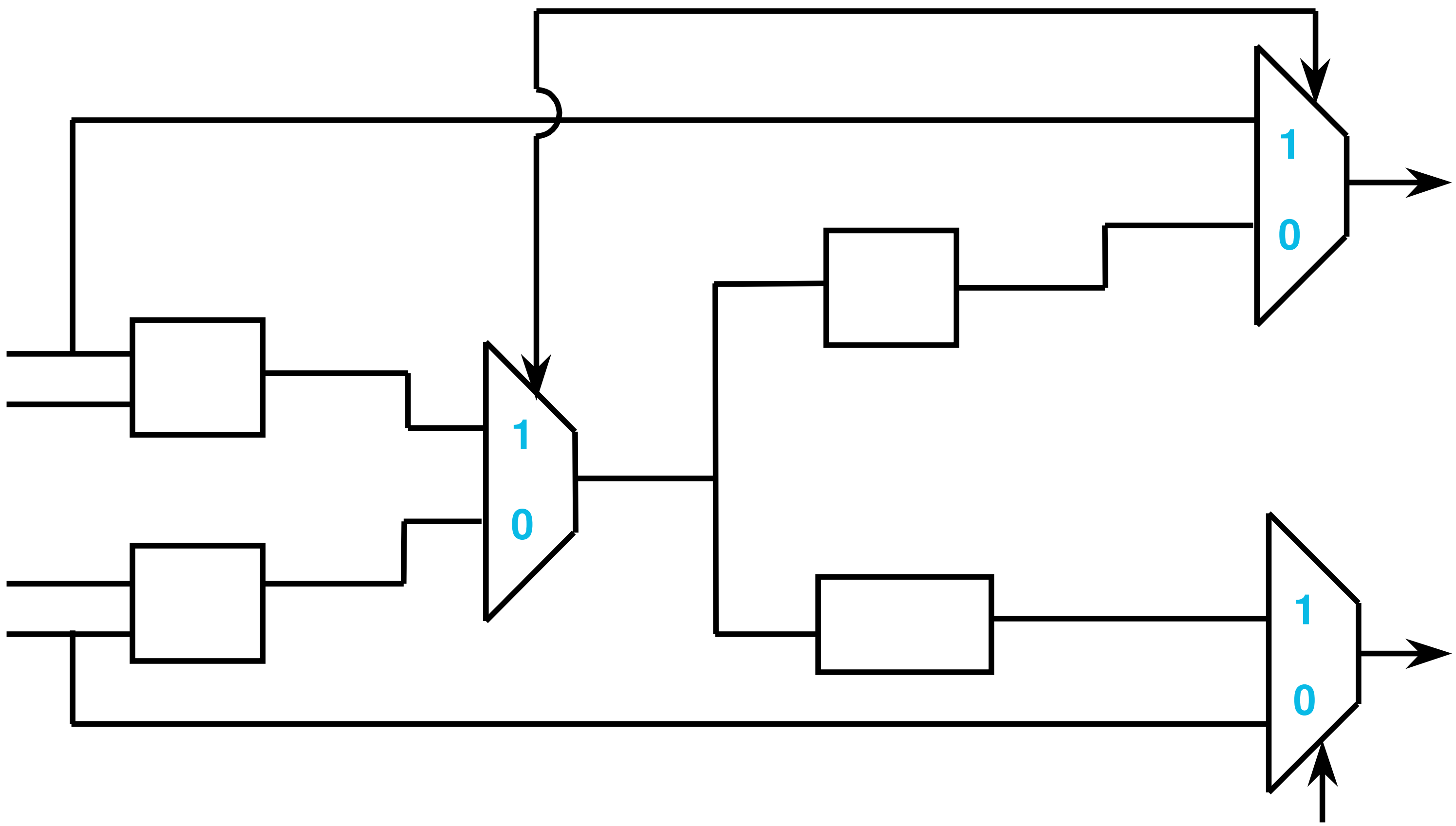
Carry bypass adder

False Path Structures

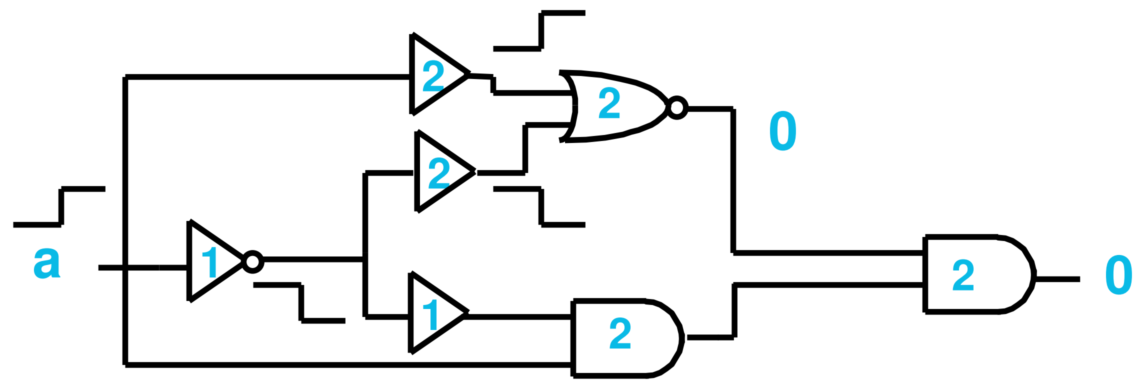




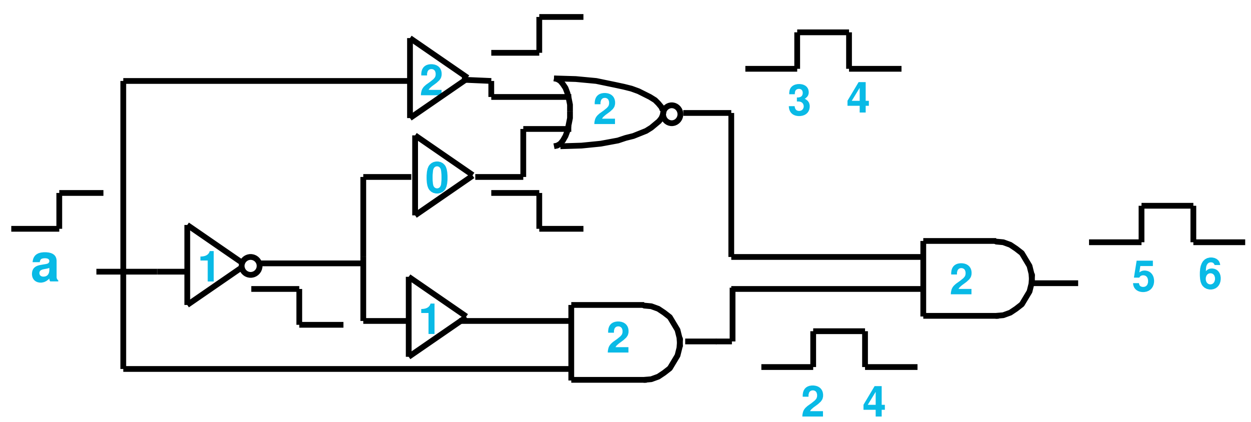
False Path Structures - 2



Problem With Fixed Gate Delays



Transition delay is 0, for both input transitions.
Consider the "faster" circuit



Bounded Gate Delays

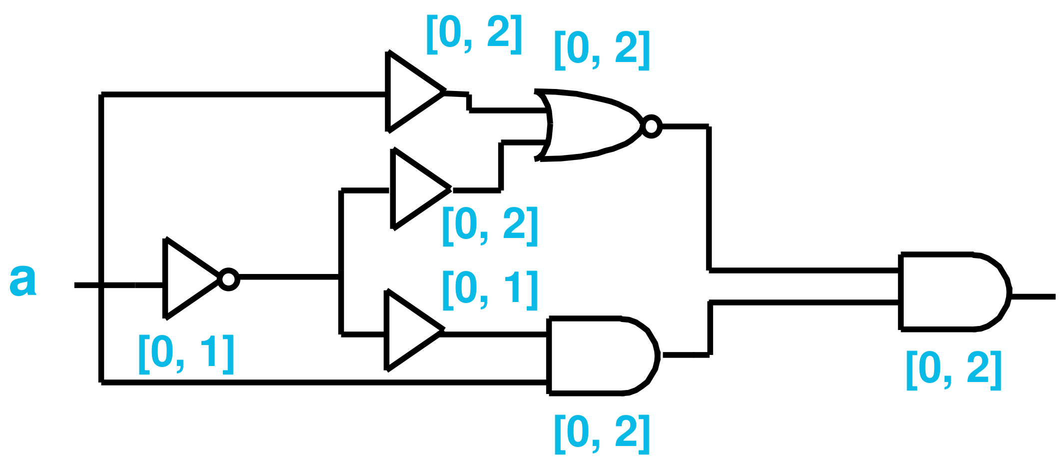
Fixed gate delays are unrealistic

Dealing with an abstraction of a fabricated circuit

Need best-case, worst-case bounds on the wire delays

Analyze a circuit family not a single circuit

Bounded Delays



When all lower bounds are 0 called monotone speed-up.

Want algorithms that report the critical path delay of the slowest circuit in the circuit family.

Report 6 for above circuit family.

Search Space

If we are dealing with transition delay, over two vectors, the search space is of size 2^{2N}, where N is the number of inputs

Bounds on the gate delays make analysis/simulation much more difficult.

Floating mode delay:

- Single-vector condition
- pessimistic
- easier to compute

Floating Mode Delay

Single vector condition

Assume $\langle V_1, V_2 \rangle$ has been applied, but only look at V_2

- Previous logic values due to the application of V_1 are pessimistically assumed to provide values to sensitize paths upon application of V_2

Assume the 1 arrives before the 0 (even if in reality it arrives later and the gate output stays at 0 throughout, and no path is sensitized).

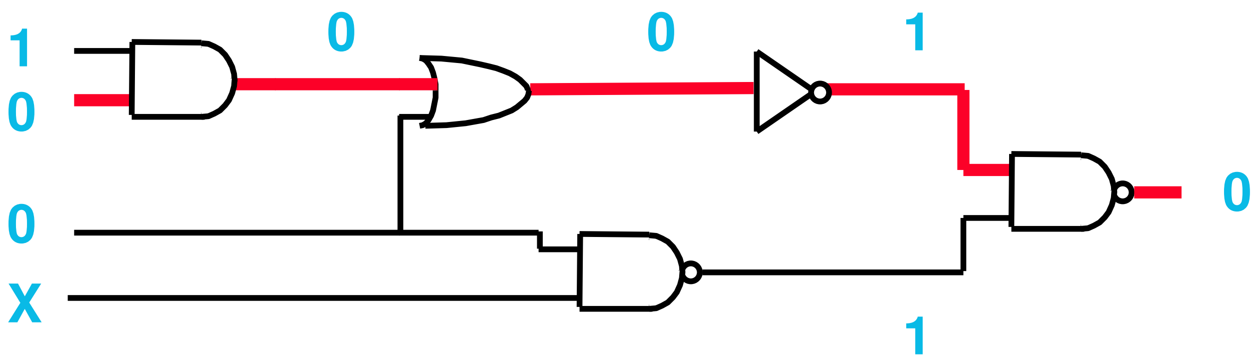
Single-Vector Sensitization

A controlling (non-controlling) value for an AND gate is a 0 (1). Vice versa for OR gate.

Definition: A path is statically sensitized by a vector V, if along each gate on the path, if the gate output is a controlling value, the input corresponding to the path is the <u>only</u> input with a controlling value

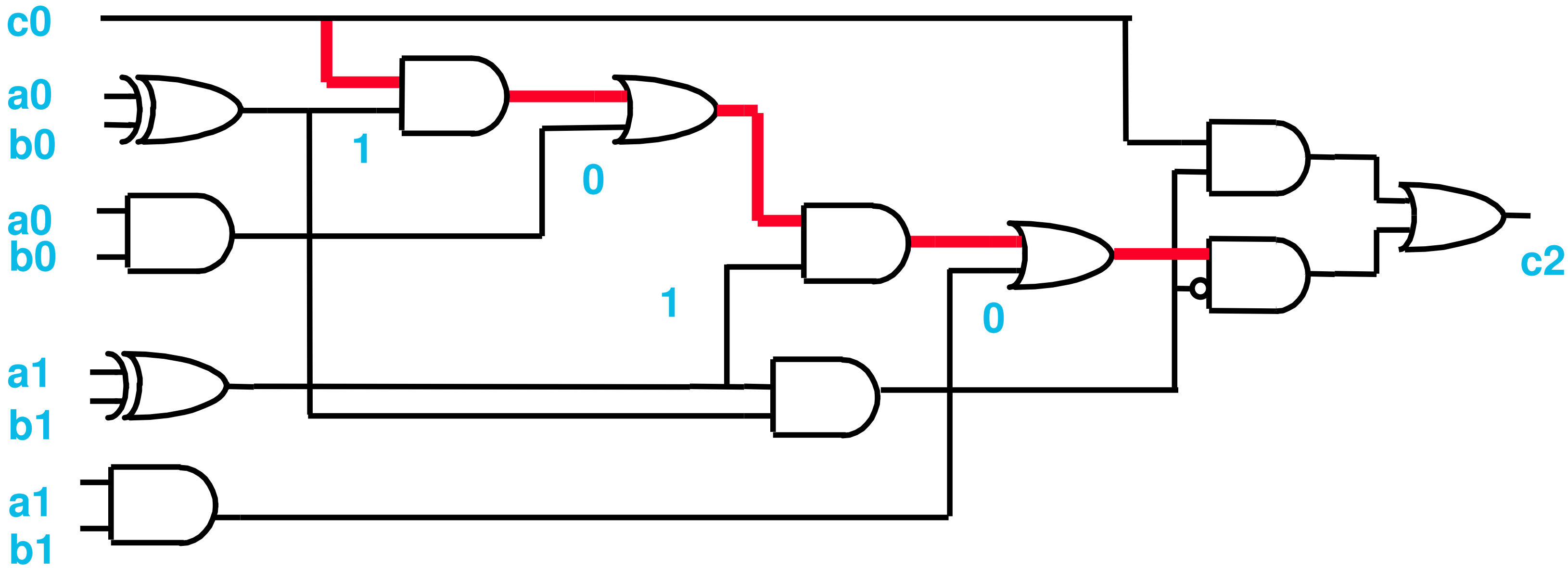
NOTE: independent of delays of gates

Static Sensitization



Bold path is statically sensitized by 100X

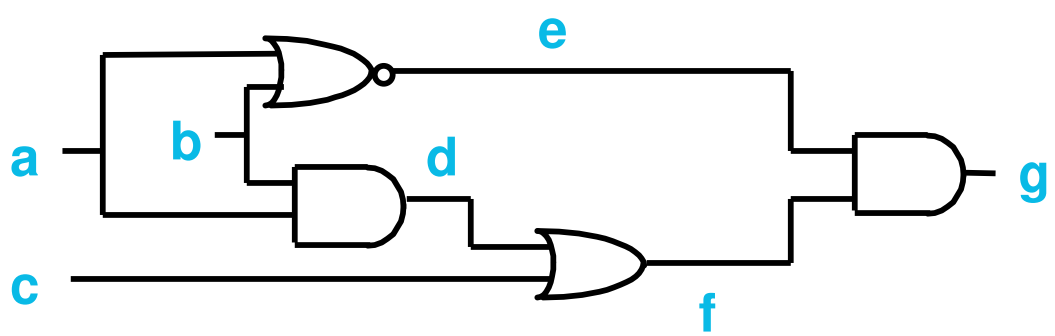
Static Sensitization - 2



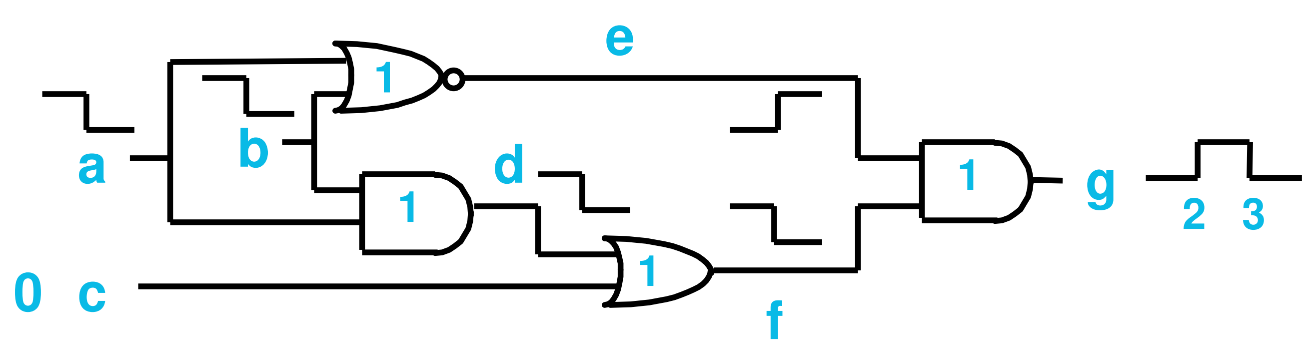
Bold path is not statically sensitizable

Static Sensitization Fails

A path may not be statically sensitizable but can be responsible for the delay of a circuit

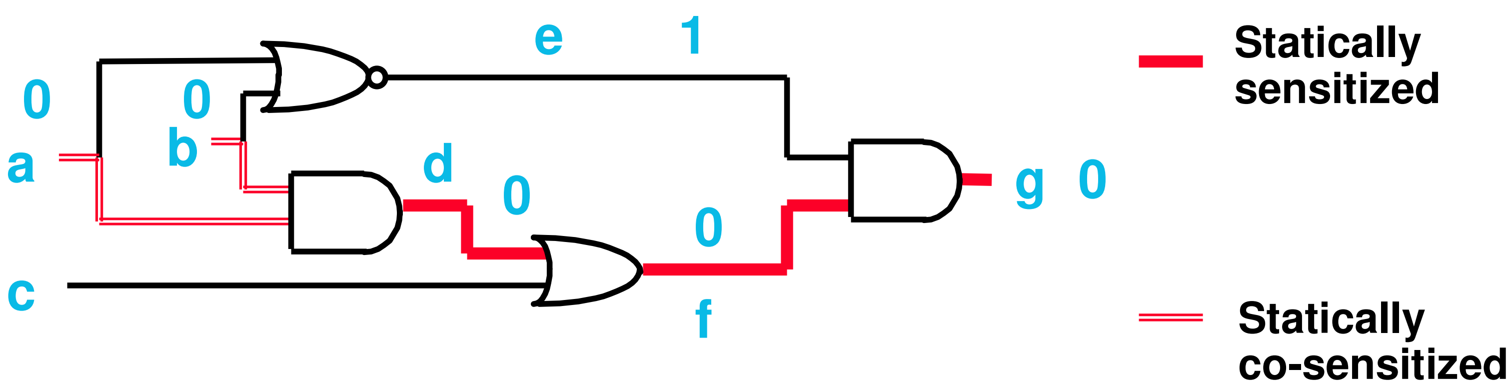


a,d,f,g and b,d,f,g not statically sensitizable



Static Co-Sensitization

Definition: A path is statically co-sensitized by a vector V, if the input corresponding to the path presents a controlling value at each gate along the path, whose output is a controlling value.

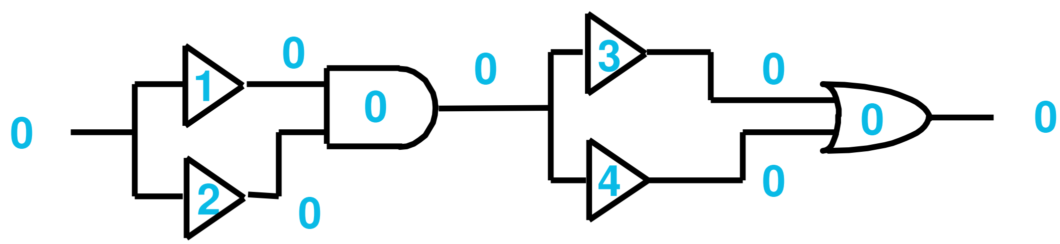


Paths a,d,f,g b,d,f,g are statically co-sensitized.

Static Co-Sensitization and Delay

Static co-sensitization is necessary for a path to be responsible for the delay of a circuit

However …



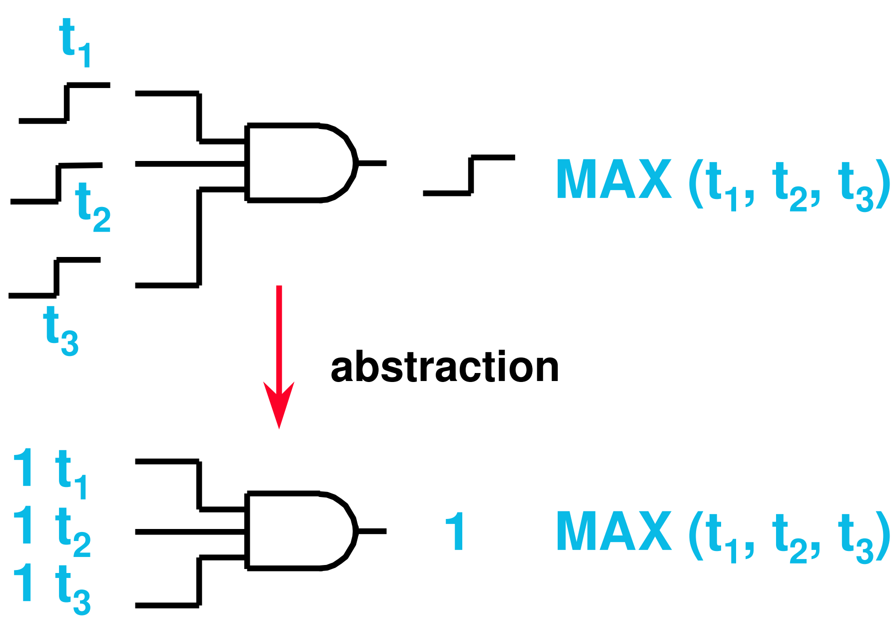
Path of length 6 is statically co-sensitized

Delay of circuit is 5

Augmenting Co-Sensitization

Static co-sensitization is independent of delay values in the circuit

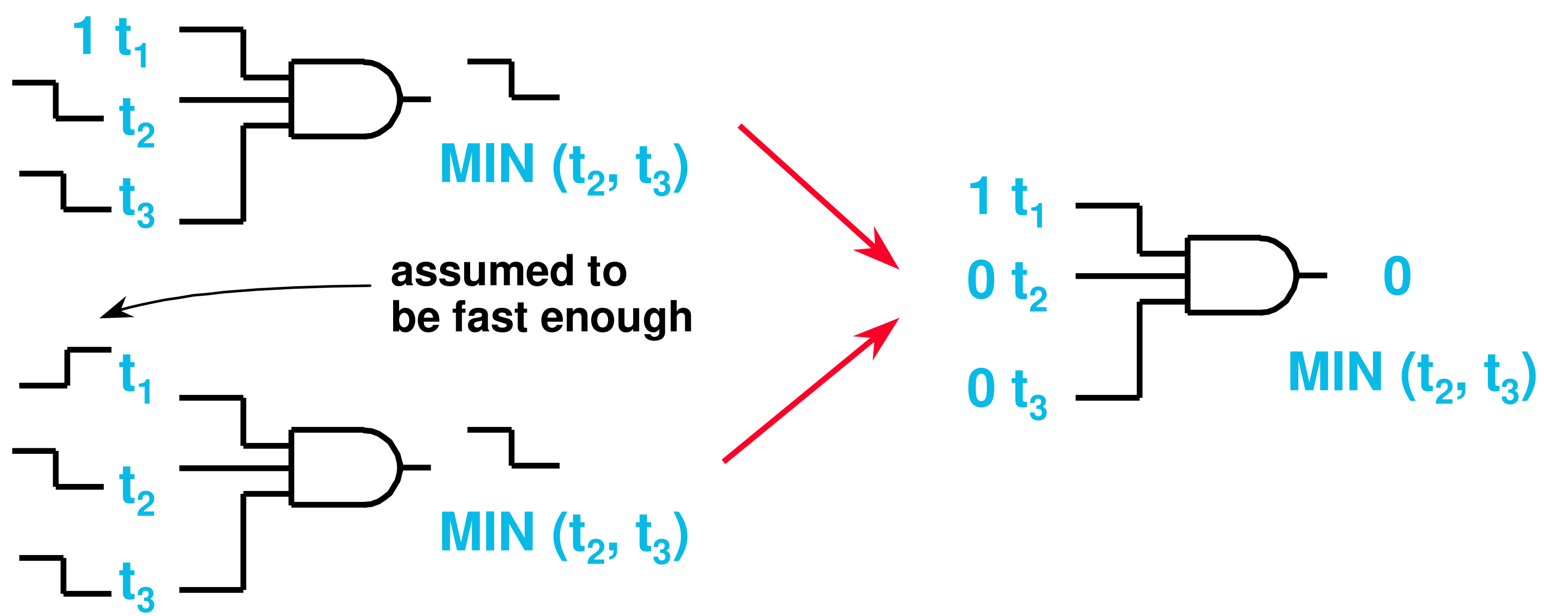
However, if we have a situation where the gate output has a non-controlling value, the slowest of all non-controlling values at the gate inputs is propagated through



Augmenting Co-Sensitization - 2

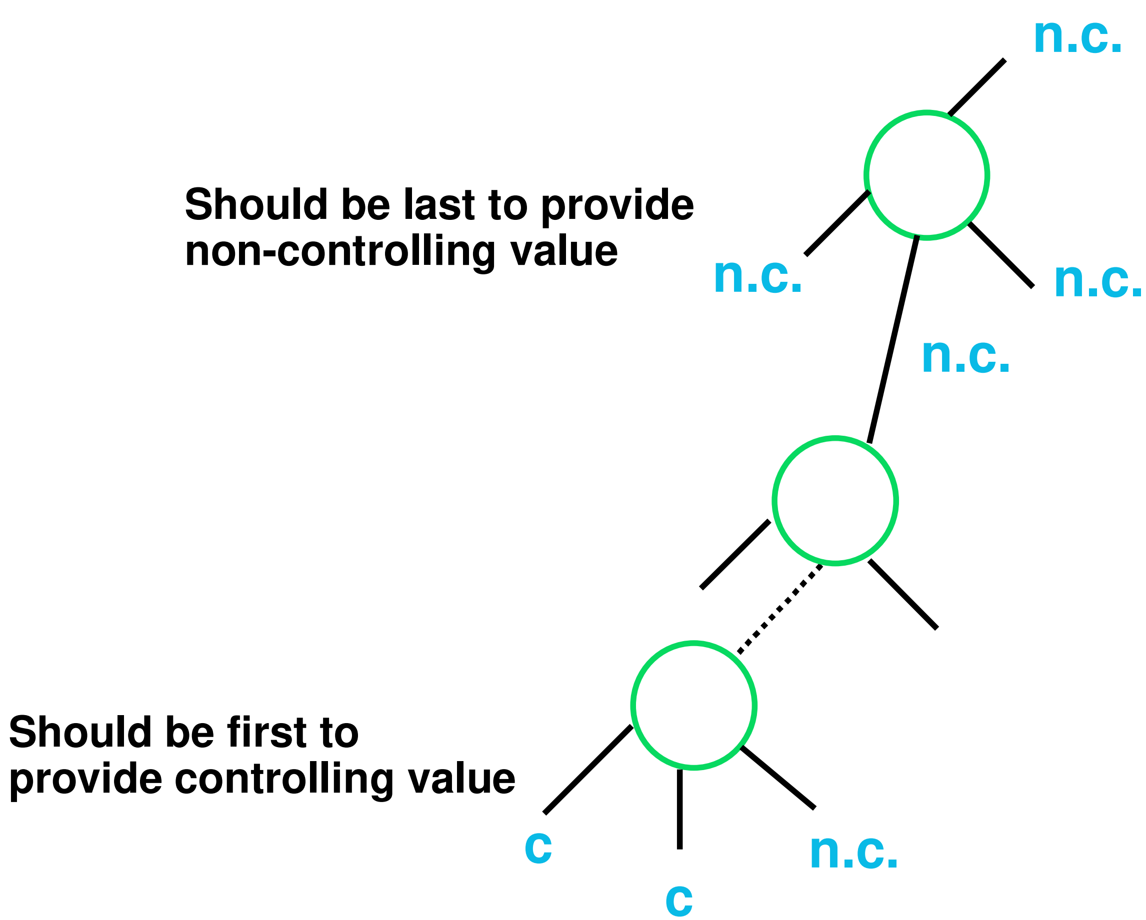
Similarly, if the gate output has a controlling value, the fastest of the controlling values at the gate inputs is propagated through

Non-controlling values are assumed to not matter (floating mode assumption)



Topological Conditions

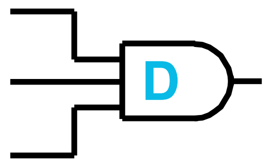
Necessary and sufficient conditions for a path to be true under floating mode of operation



Delay Calculus

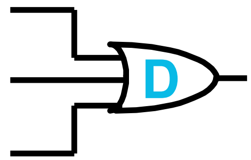
Have delays associated with 0's and 1's

$0(d_1)$, $0(d_2)$, $0(d_3)$



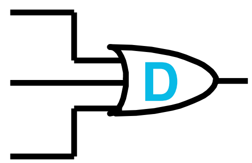
$0\ (\mathrm{MIN}\ (d_1, d_2, d_3) + D)$

$0(d_1)$, $0(d_2)$, $0(d_3)$



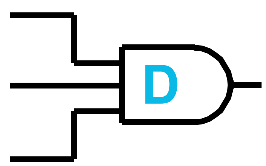
$0\ (\mathrm{MAX}\ (d_1, d_2, d_3) + D)$

$0(d_3)$, $1(d_1)$, $1(d_2)$



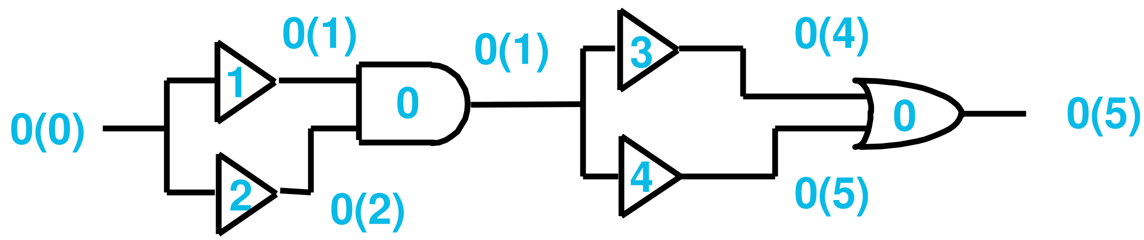
$1\ (\mathrm{MIN}\ (d_1, d_2) + D)$

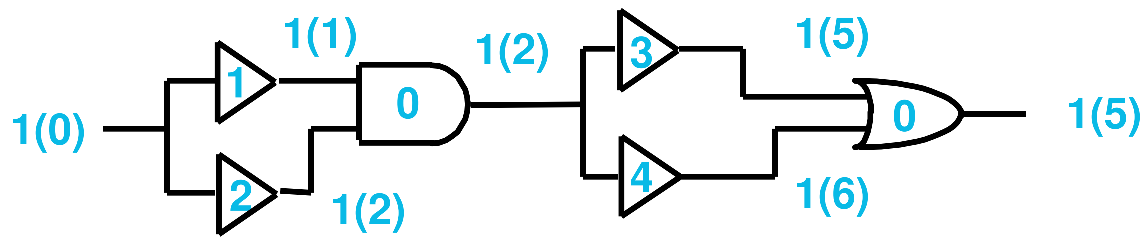
$1(d_1)$, $1(d_2)$, $0(d_3)$



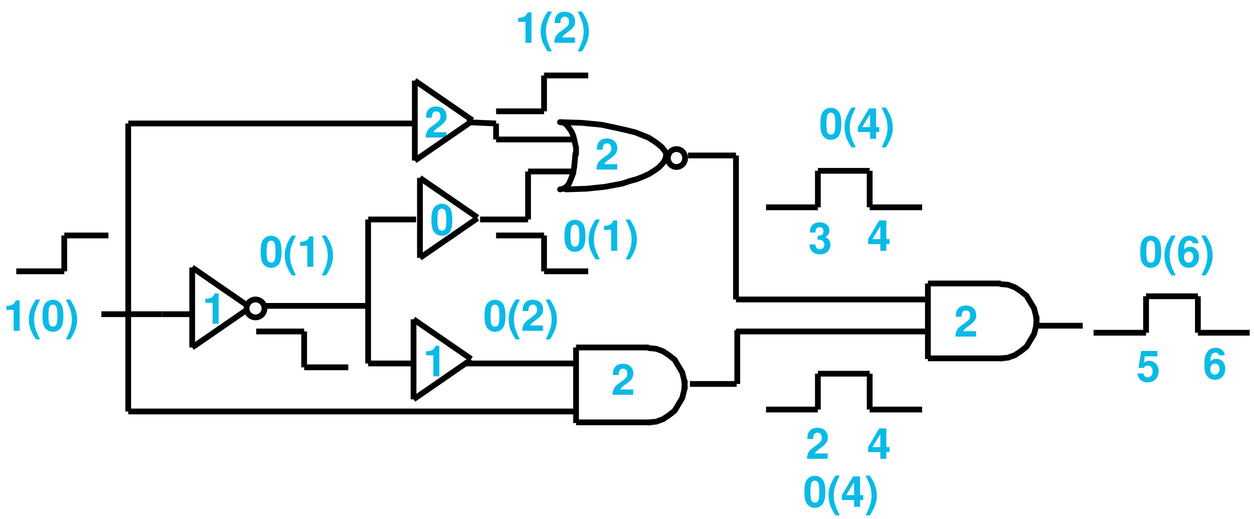
$0\ (d_3 + D)$

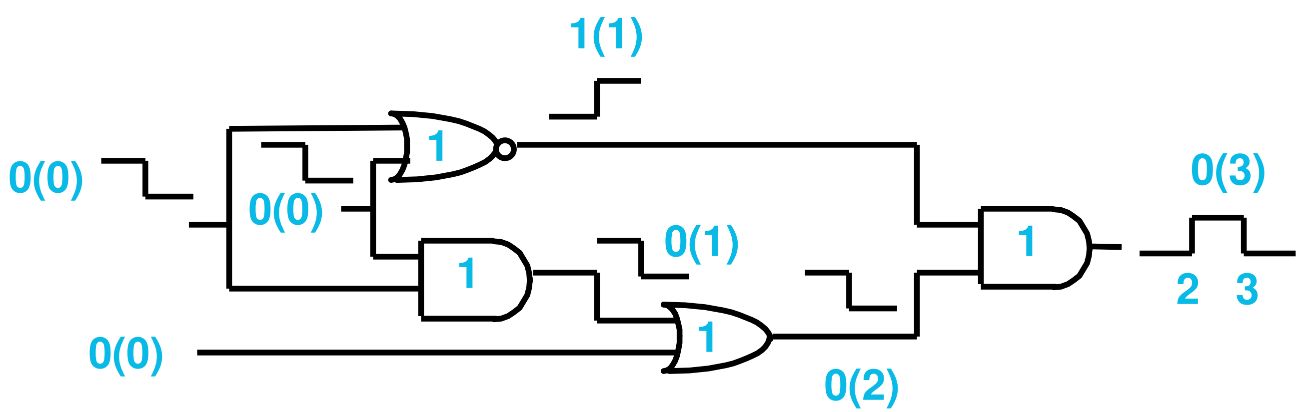
Examples



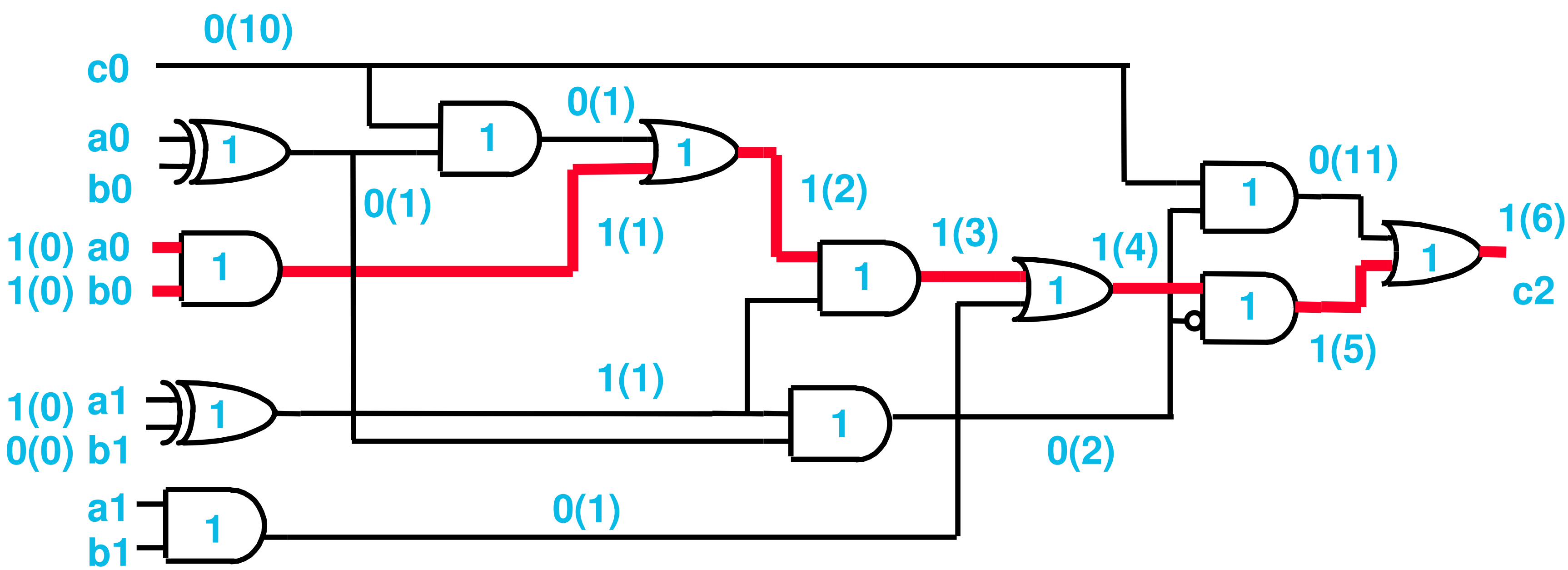


Examples - 2





Examples - 3


0(10)
c0
a0
b0
1(0) a0
1(0) b0
1(0) a1
0(0) b1
a1
b1
0(1)
0(1)
1(1)
1(2)
1(3)
1(4)
0(11)
1(6)
c2
1(5)
1(1)
0(2)
0(1)
1
1
1
1
1
1
1
1
1
1
1
1