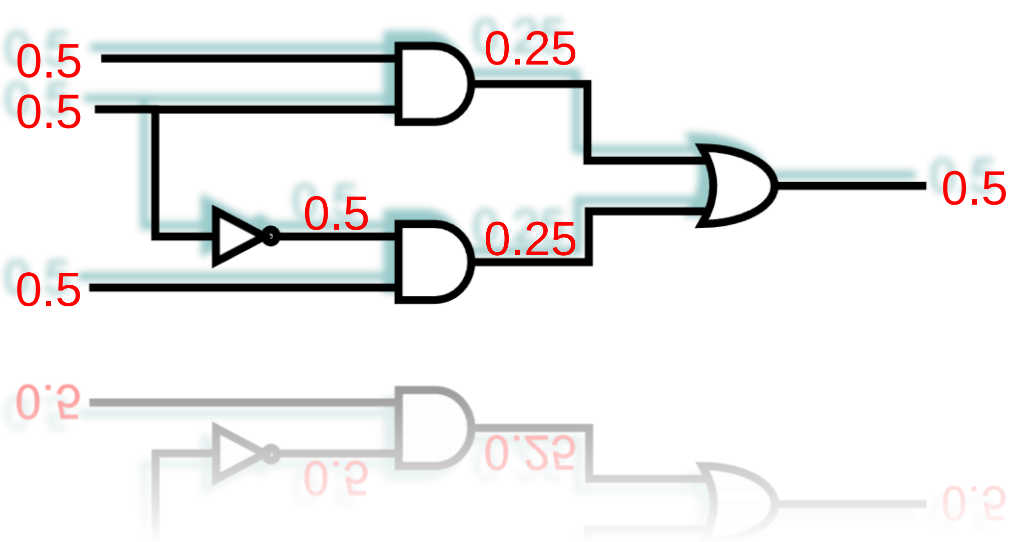
Testability Measure

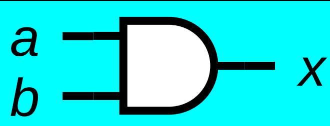
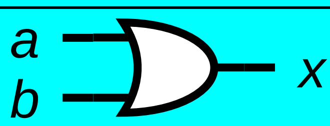
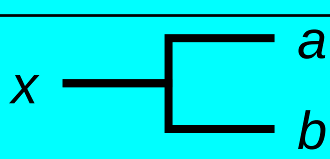
- Introduction
- SCOAP (1979)
- **COP (1984)**
- High-level testability measures
- Conclusion



COP

- *Signal probability* of *x* = probability of *x* being logic 1
  - Actual signal probability requires exhaustive simulation
  - Hard to obtain in practice
- COP = Controllability/Observability Program [Brglez 84]
  - Fast algorithm to estimate signal probability
  - C_x = estimated prob(x = *1)*
  - $1-C_x$ = estimated prob(x = 0)
  - O_x = estimated probability of *fault effect* in *x* being observed at PO
- C_x and O_x are numbers between 0 and 1
  - *Larger number* means *easier* to control or observe
- Assumptions
  - 1. Ignore fanout reconvergence for fast run time
  - 2. PI are independent random numbers: ½ zero and ½ one

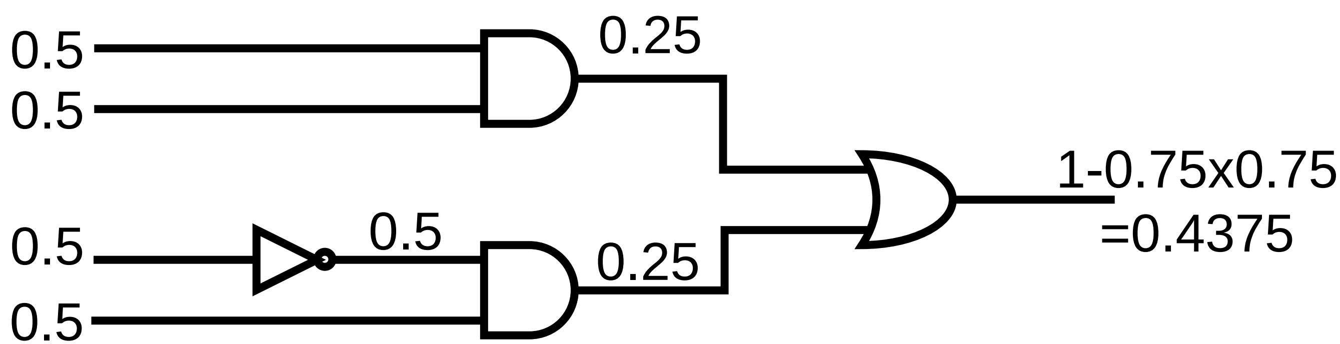
COP

<table>
<tr><th></th><th>C_x</th><th>O_a</th></tr>
<tr><td>x = PI</td><td>0.5</td><td></td></tr>
<tr><td>x = PO</td><td></td><td>0.5</td></tr>
<tr><td>
</td><td>$C_x = C_a \times C_b$</td><td>$O_a = O_x \times C_b$</td></tr>
<tr><td>
</td><td>$C_x = 1 - (1- C_a) \times (1- C_b)$</td><td>$O_a = O_x \times (1- C_b)$</td></tr>
<tr><td>
</td><td>$C_x = C_a = C_b$</td><td>$O_x = 1 - (1- O_a) \times (1- O_b)$</td></tr>
</table>

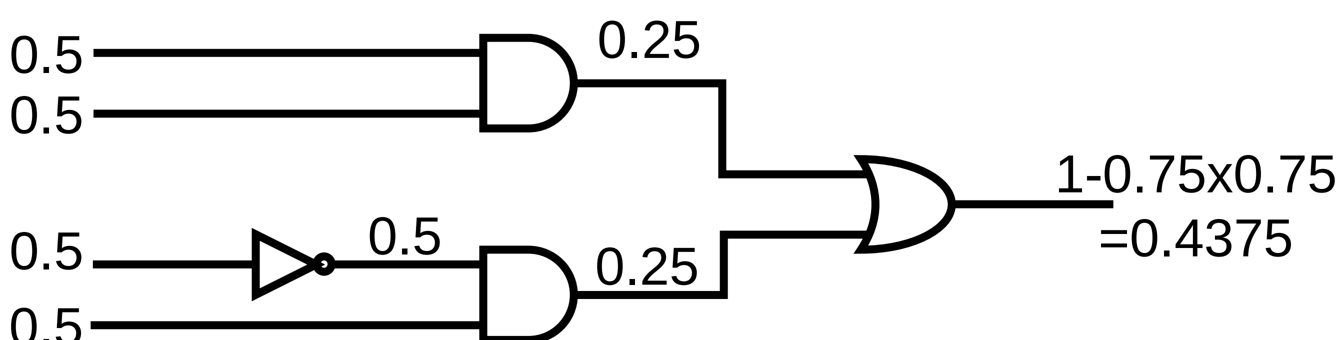
Example – Controllability

- **Calculate from PI to PO**
- **Fanout-free circuit, COP = actual signal probability**

COP C_x



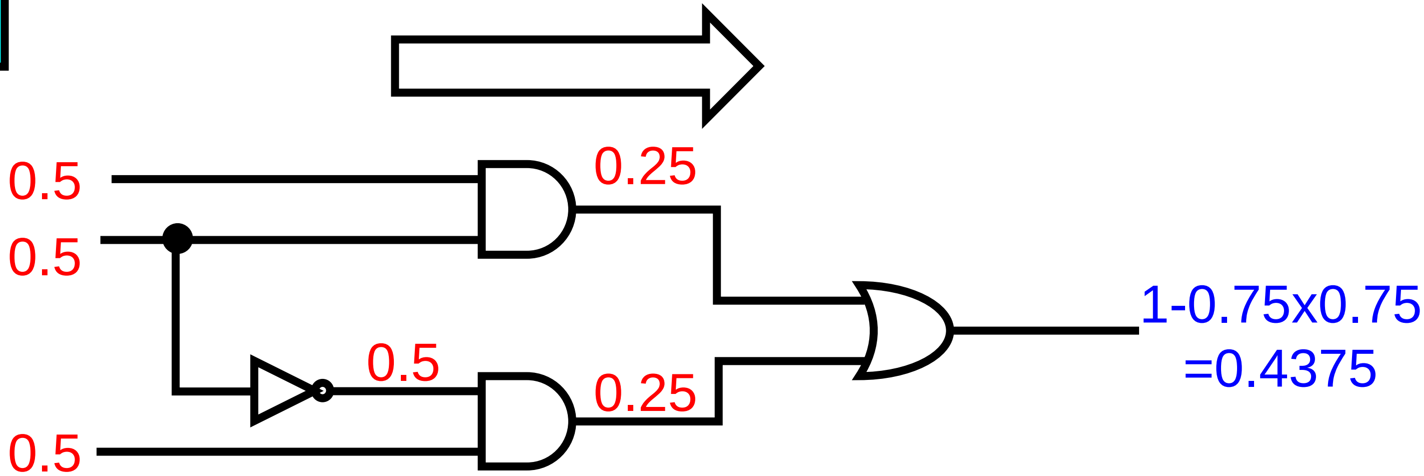
Actual signal probability



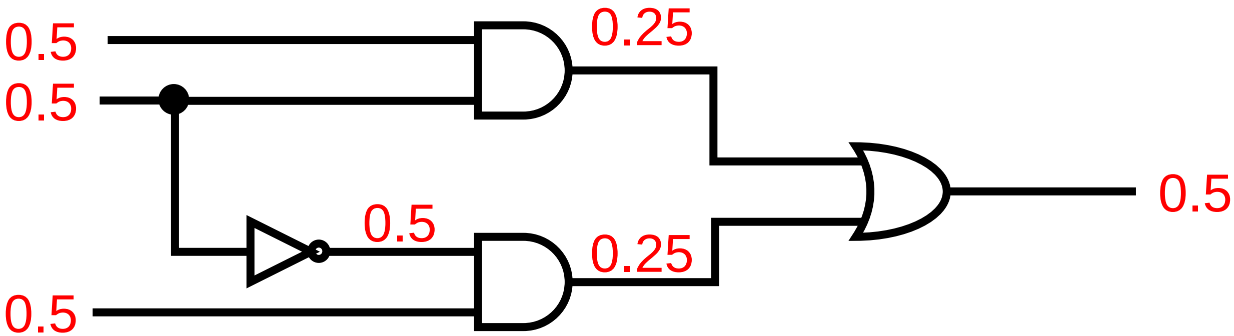
Example (2) – Controllability

- When fanouts reconverge, COP ≠ actual signal probability

COP C_x

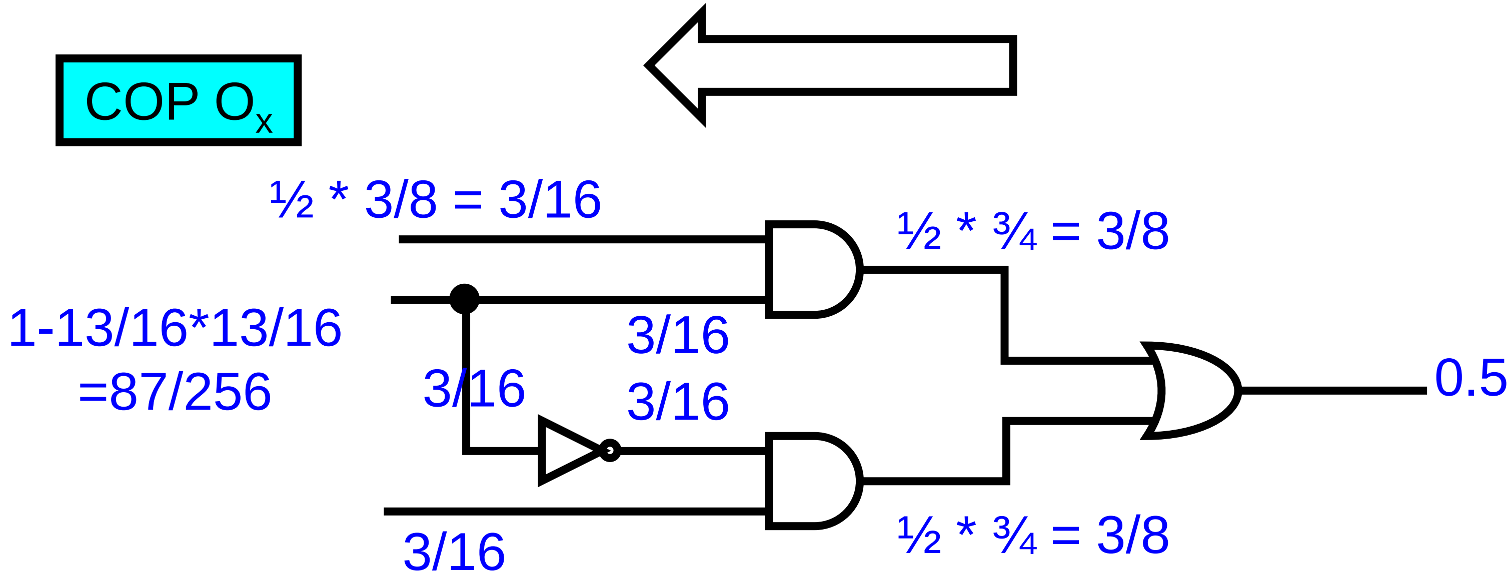


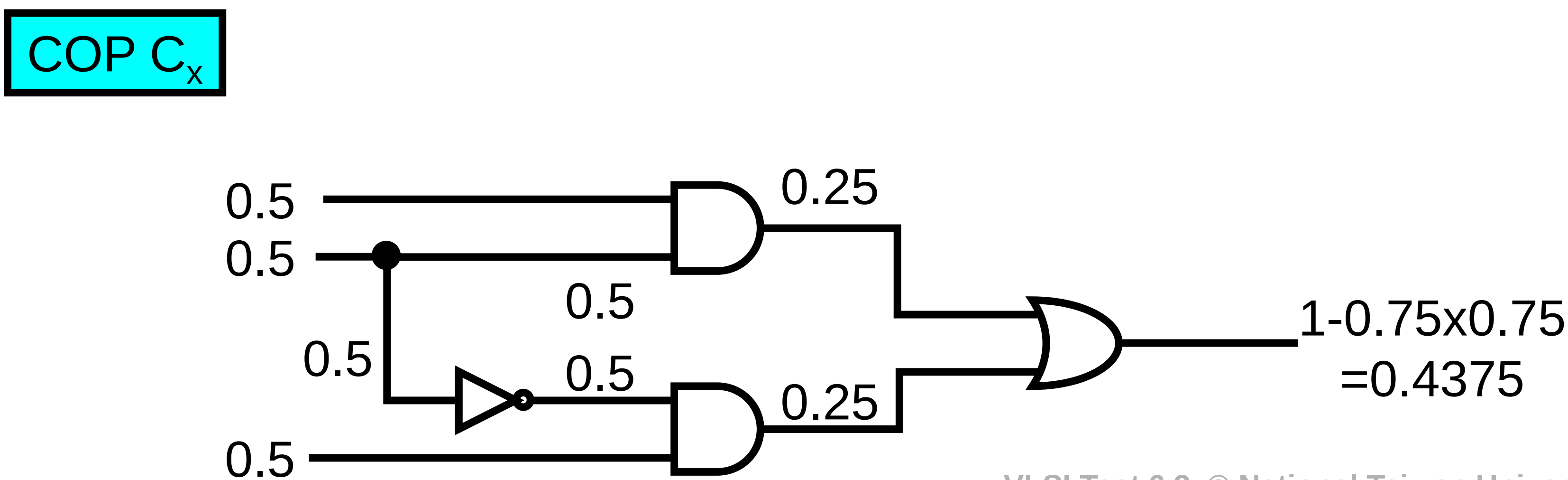
Actual signal probability



Example (2) – Observability

- **Calculate from PO to PI**



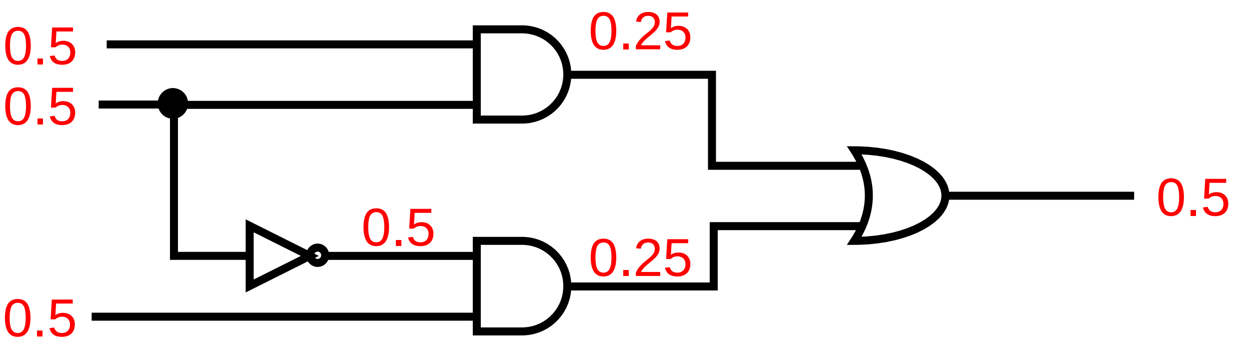


Quiz

Q: verify actual signal probability by exhaustive test patterns

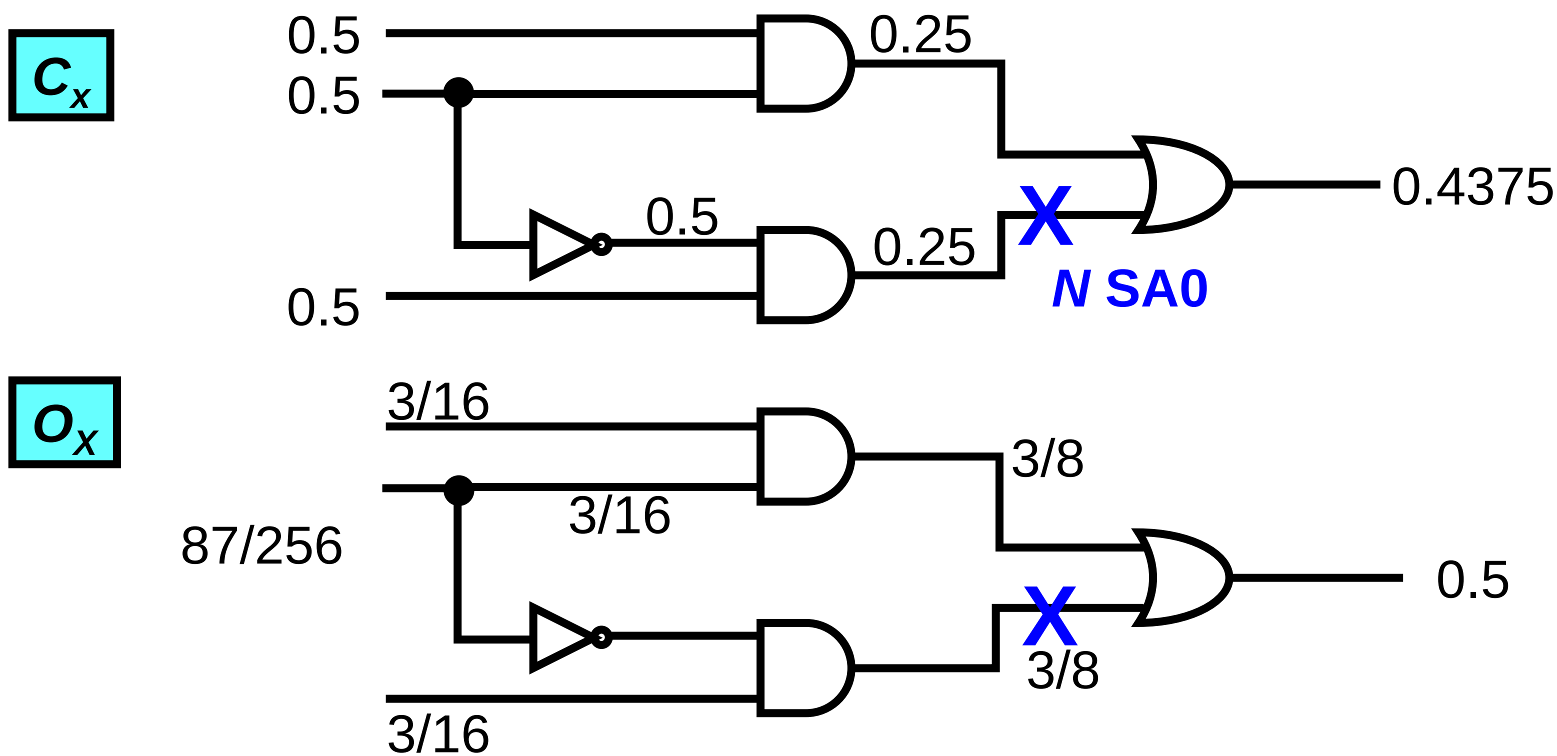
input	output
000	
001	
010	
011	
100	
101	
110	
111	

Actual signal probability



Detection Probability, DP

- DP_f = Probability of detecting a fault f
  - $DP_{N\ SA0} = C_N \times O_N$
  - $DP_{N\ SA1} = (1-C_N) \times O_N$
- Larger DP_f means *easier* to detect fault f
- Example: $DP_{N\ SA0}$ = 1/4 x 3/8 =3/32

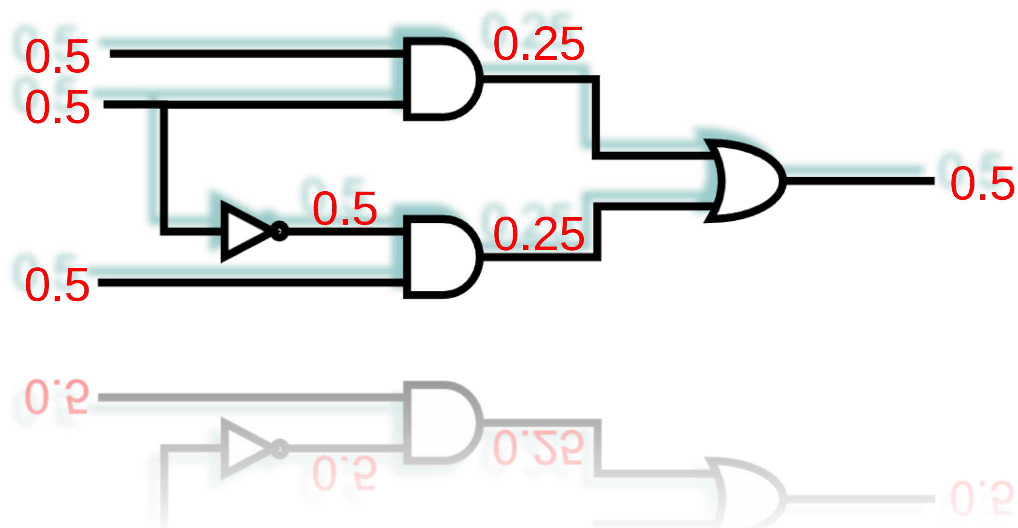


Random Pattern Resistant Faults

- RPRF = Faults that are difficult to be tested by random patterns
  - *Low detectability*
  - aka. *Hard-to-detect faults, difficult faults*
- Example:
  - stuck-at-0 fault at an *n*-input AND gate output
  - Need test pattern (1,1,1,…1)
  - Assume equal signal probability of 0.5 at each input
    - $C_x = 0.5^n$
- Test generation for RPRF is difficult
  - Solutions:
    - 1. Insert test points (See DFT lecture)
    - 2. Weighted random patterns (see BIST lecture)

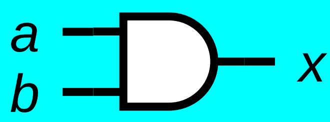
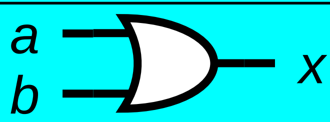
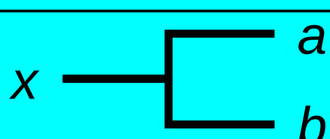
Summary

- **COP**
  - C_x = estimated prob($x = 1$)
  - $1\text{-}C_x$ = estimate prob($x = 0$)
  - O_x = estimated probability of *fault effect* in x being observed
  - COP ≠ actual signal probability because fanout reconvergence



FFT

- **Q: Why observability at PO is 0.5, not 1?**

	C_x	O_a
***x* = PI**	***0.5***	
***x* = PO**		***0.5***
	$C_x = C_a \times C_b$	$O_a = O_x \times C_b$
	$C_x = 1 - (1- C_a) \times (1- C_b)$	$O_a = O_x \times (1- C_b)$
	$C_x = C_a = C_b$	$O_x = 1 - (1- O_a) \times (1- O_b)$