Digital Logic Review

- Gates
- Combinational Logic
- Sequential Logic
- Abstraction (sub-circuits)

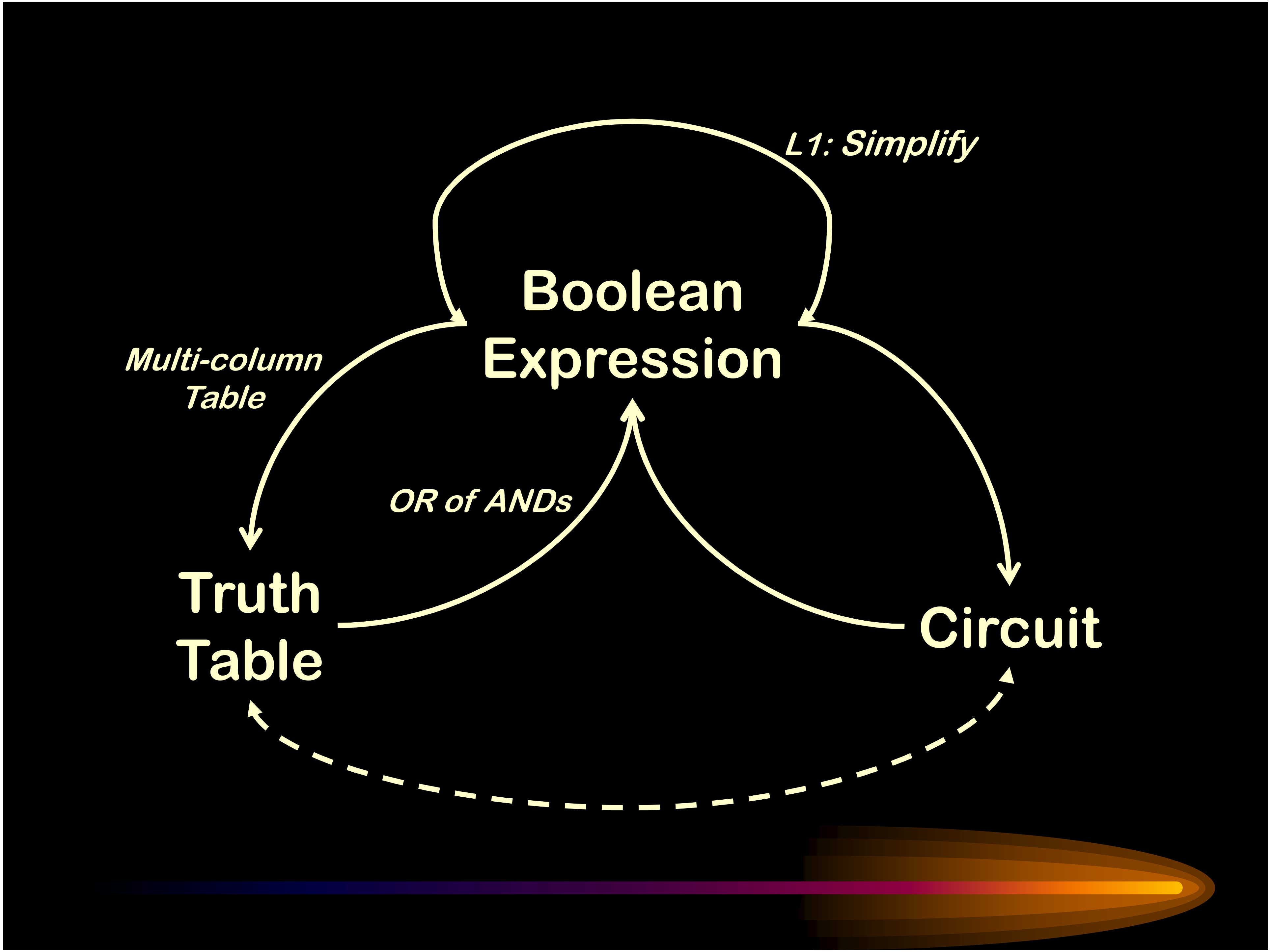
Gates

- Circuit symbol
- Truth Table
- Boolean Expression

- AND
- OR
- NOT
- XOR
- NAND
- NOR

Sequential Logic

- Circuits, Truth-Tables, Boolean Expressions
- Abstraction
  - Simplifying
  - Replicating
  - Example: Adders
    - Half adder vs. full adder
    - "Chaining" adders for n-bit adders

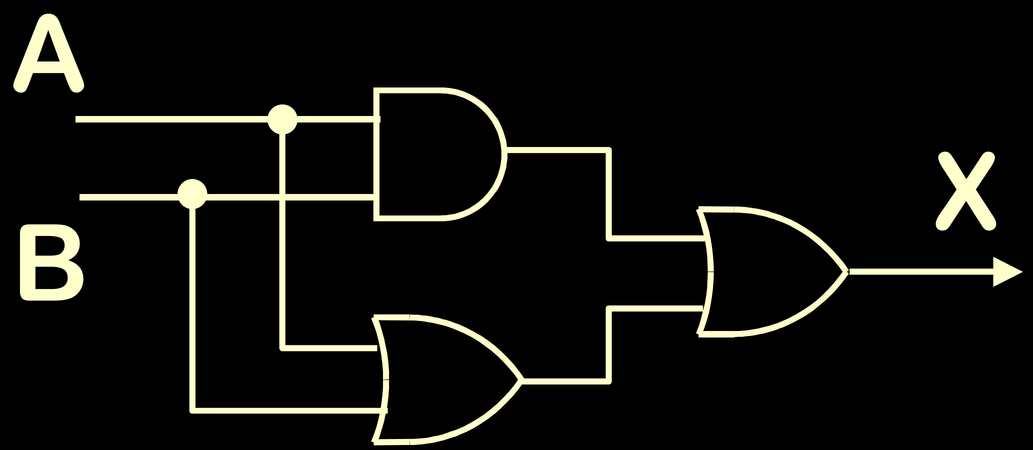

L1: Simplify
Boolean Expression
Multi-column Table
OR of ANDs
Truth Table
Circuit

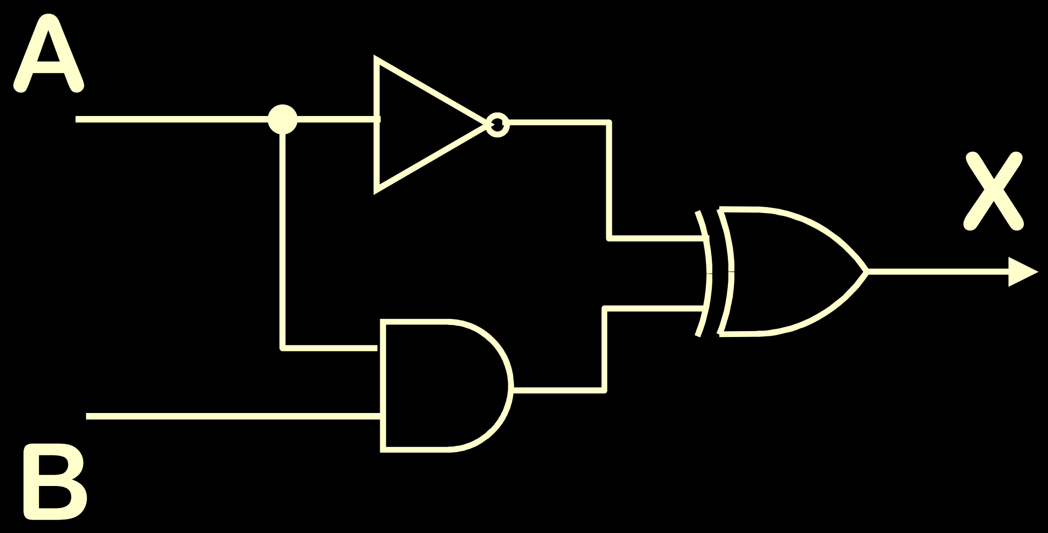
(AB+C)D

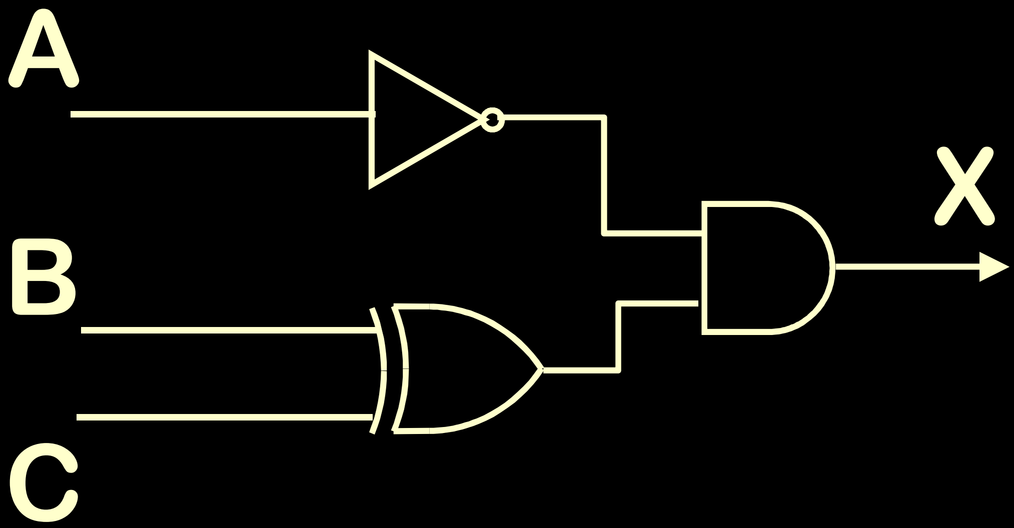
A'B + (B+C)'

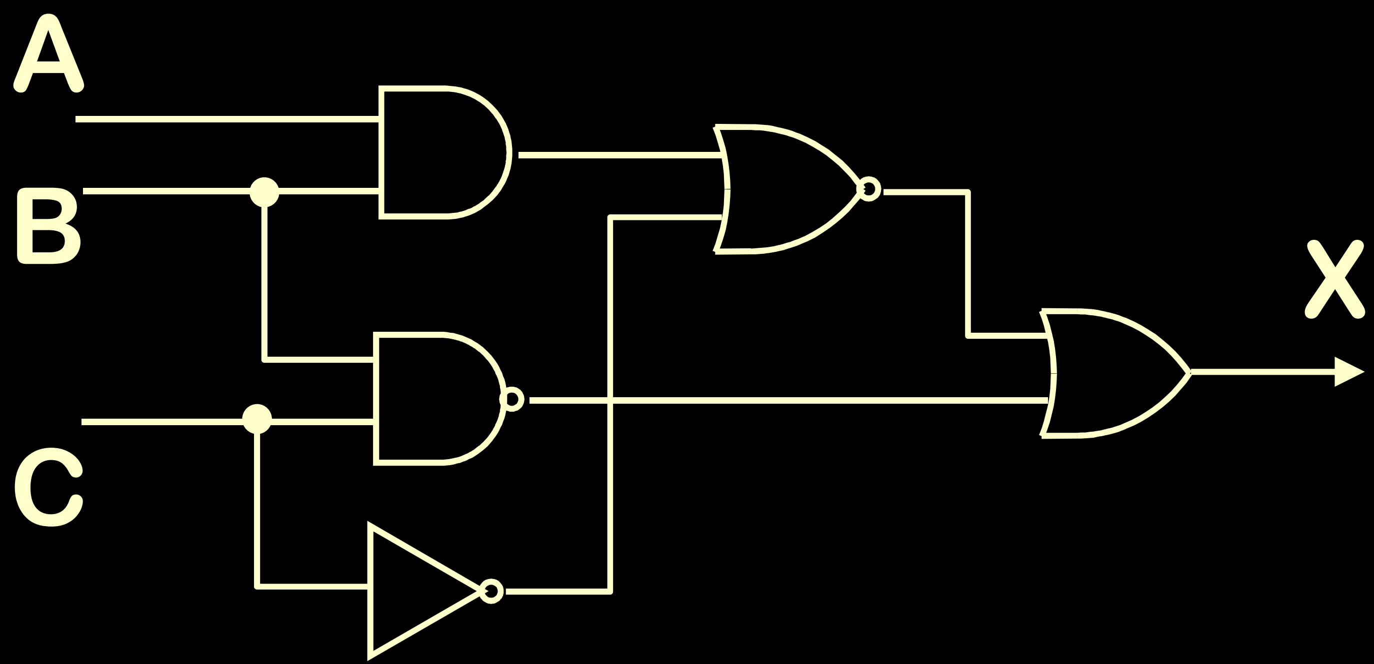
p119: 57

$(AB)' + (CD)'$


A
B
X


A
X
B


A
B
C
X


A
B
C
X

Or of Ands

A	B	⊕
0	0	0
0	1	1
1	0	1
1	1	0

$A \oplus B = A'B + AB'$

A'B

AB'

1. Find all the 1's
2. Write an AND expression for each input, input'
3. OR them together

A'B'C' + A'BC + AB'C + ABC'

A	B	C	X	
0	0	0	1	A'B'C'
0	0	1	0	
0	1	0	0	
0	1	1	1	A'BC
1	0	0	0	
1	0	1	1	AB'C
1	1	0	1	ABC'
1	1	1	0	

Write a Boolean expression for X in terms of A, B, and C.

Hint: Or of Ands…

Properties of Boolean Algebra

AND	OR
$A(1) = A$	$A+0 = A$
$AB = BA$	$A+B = B+A$
$(AB)C = A(BC)$	$(A+B)+C = A+(B+C)$
$A(B+C) = AB + AC$	$A+BC = (A+B)(A+C)$
$AA' = 0$	$A+A' = 1$
$(AB)' = A'+B'$	$(A+B)' = A'B'$

L1

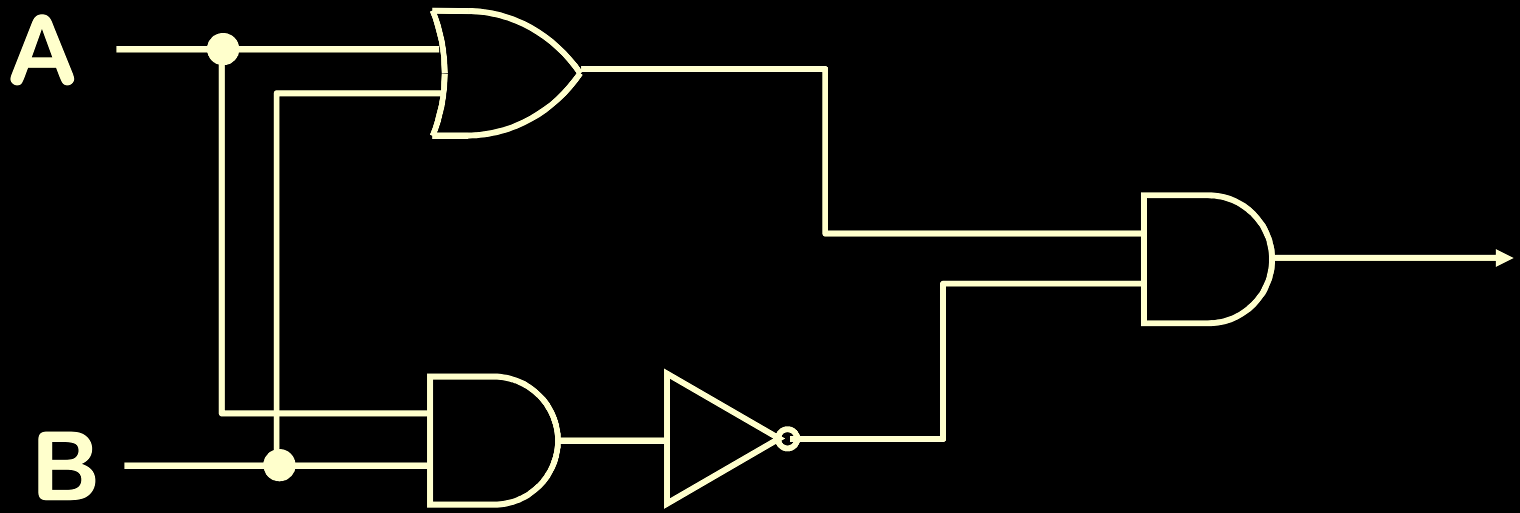
Properties of Boolean Algebra

$$A \oplus B = AB' + A'B$$
$$(A \oplus B)' = AB + A'B'$$

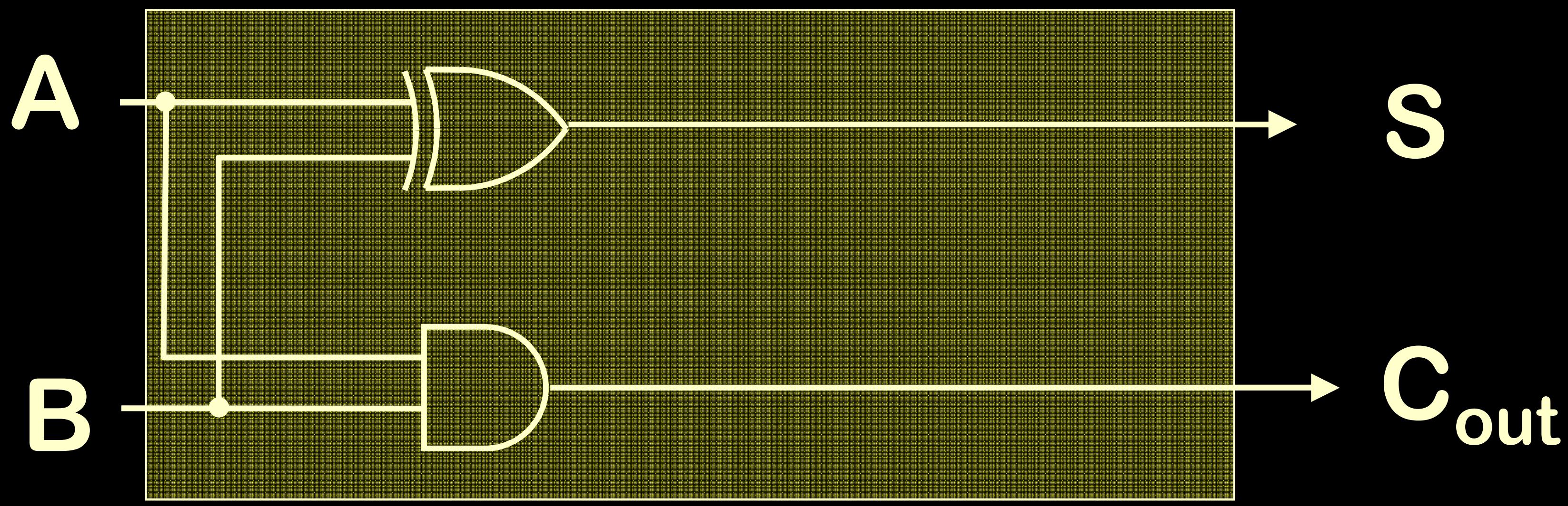
L1

Is this an XOR?

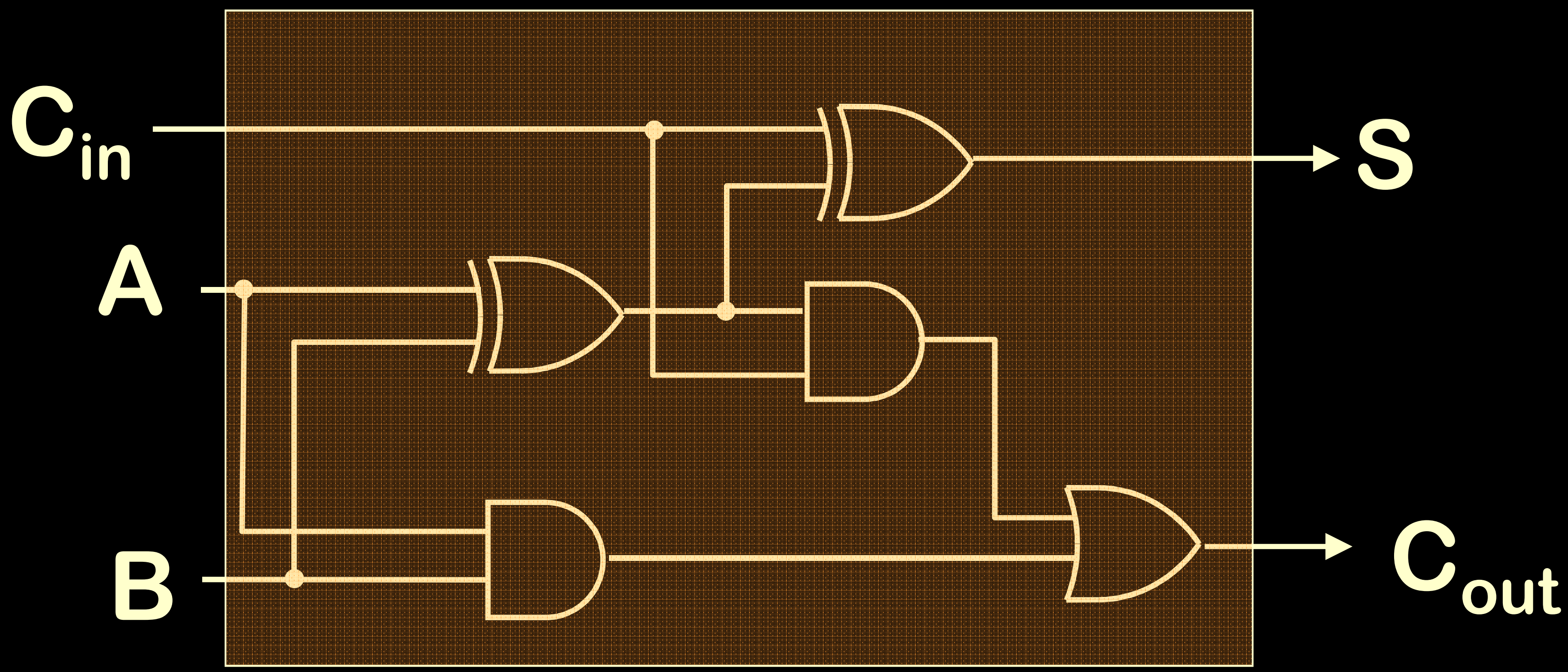
(A+B)(AB)'

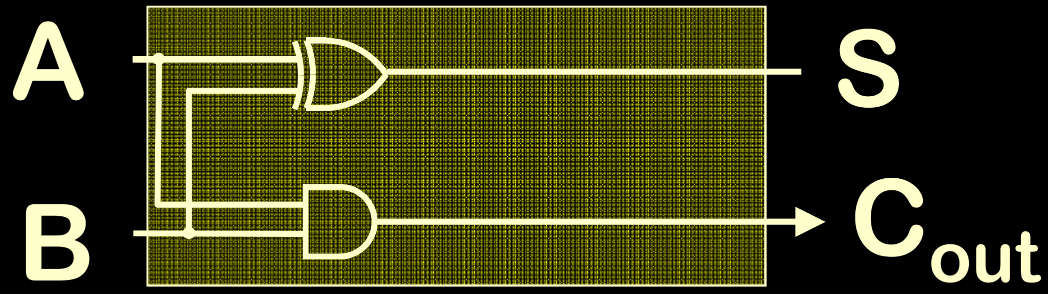


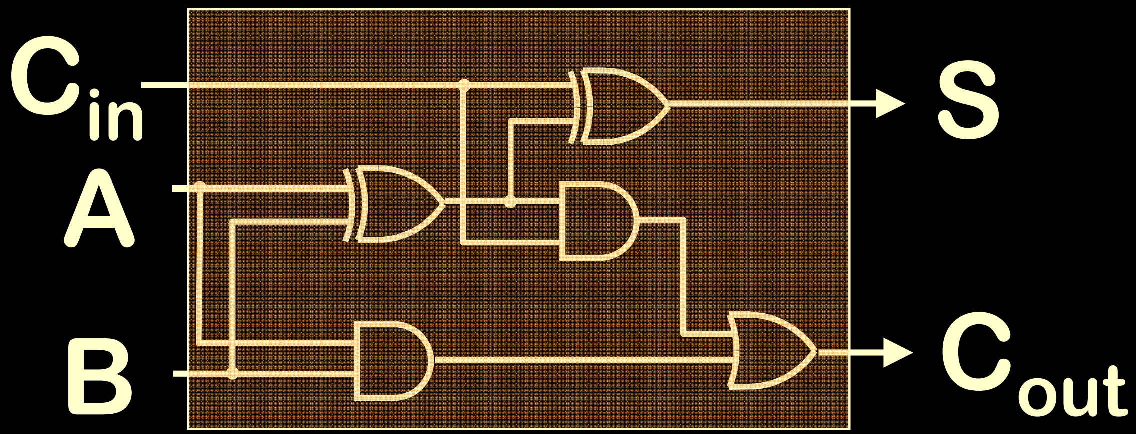
Half Adder

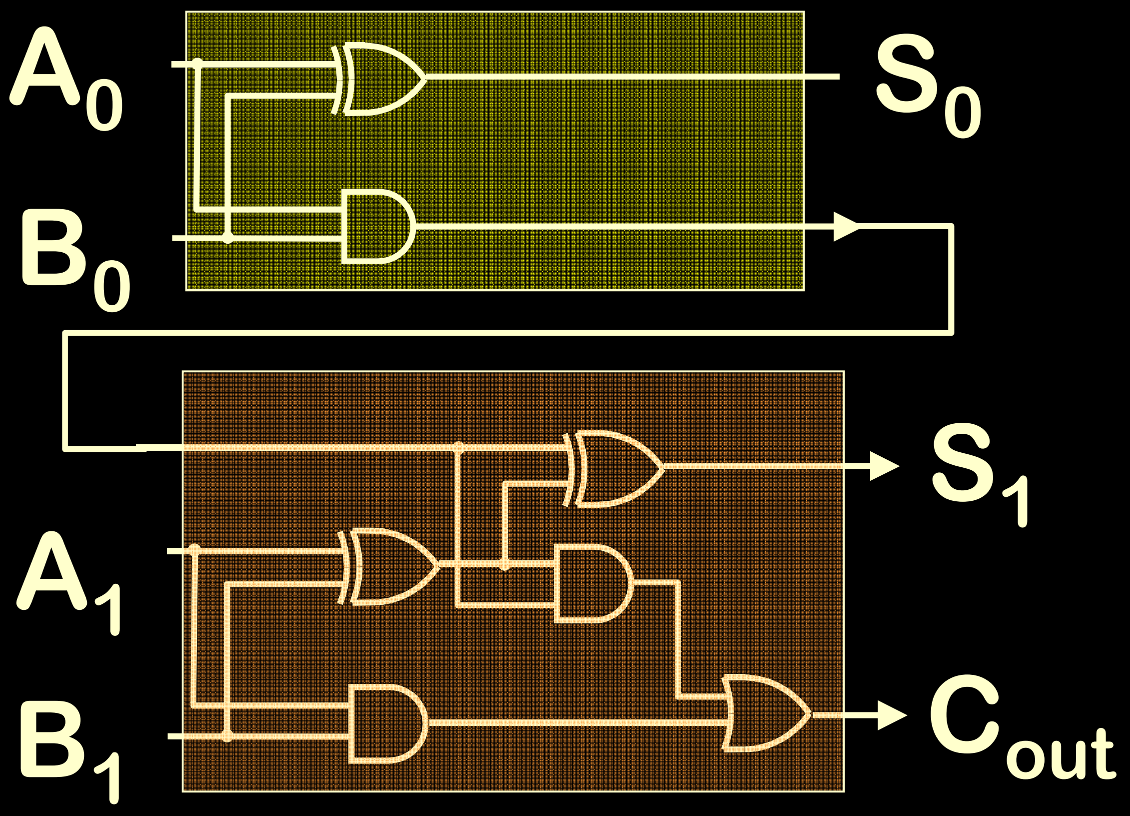


Full Adder

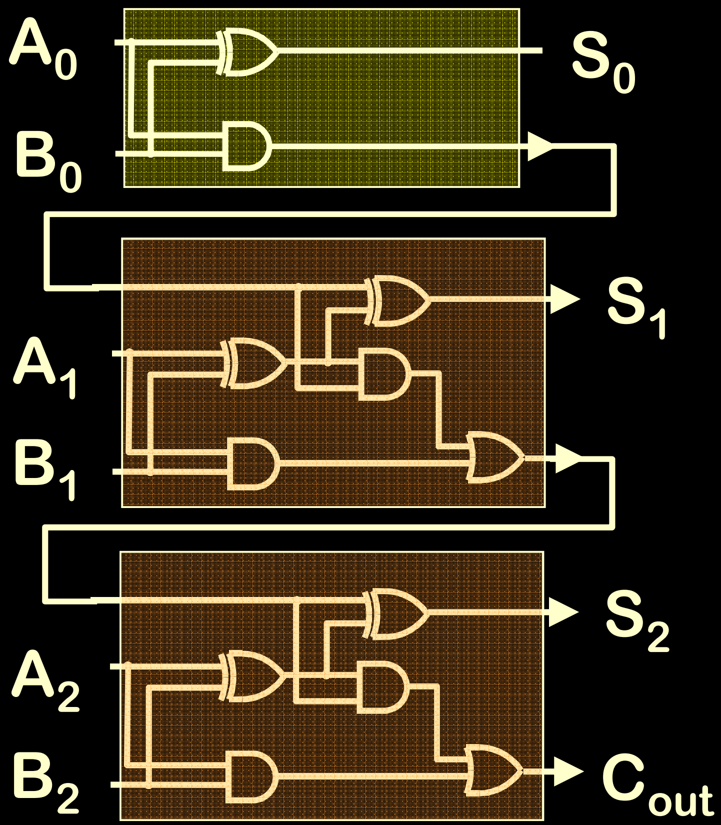


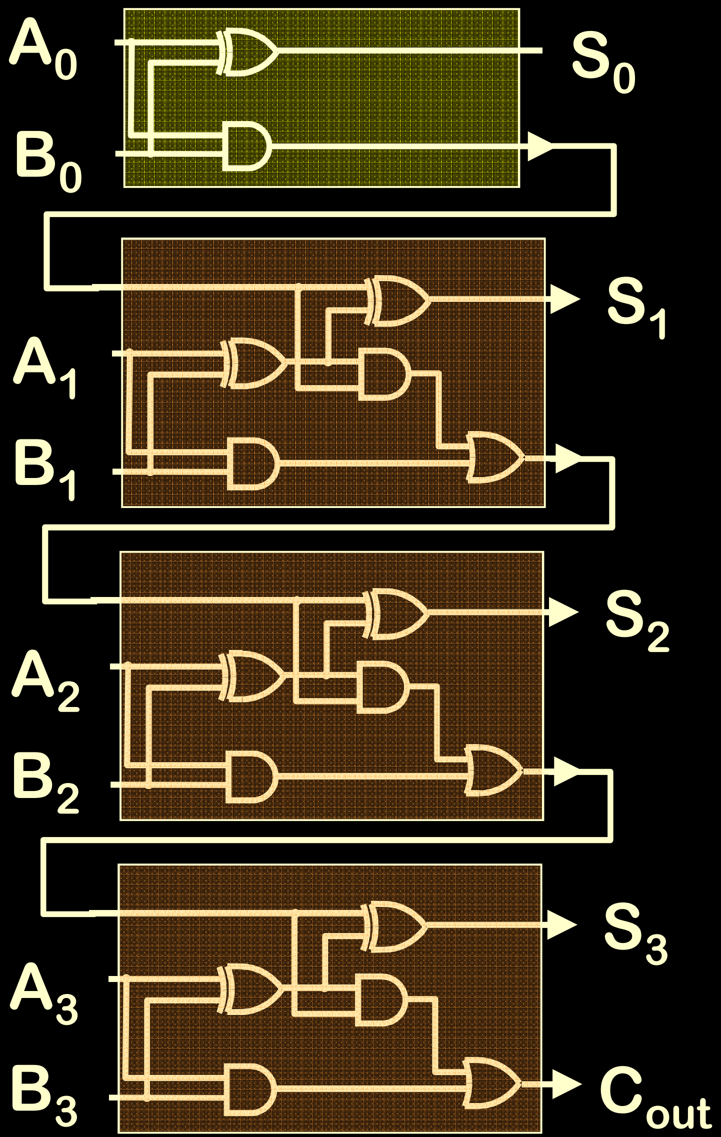

A
B
S
C_{out}


C_{in}
A
B
S
C_{out}



A_0
B_0
S_0
S_1
A_1
B_1
C_{out}


A0
S0
B0
S1
A1
B1
S2
A2
B2
Cout



A0
B0
S0
S1
A1
B1
S2
A2
B2
S3
A3
B3
Cout

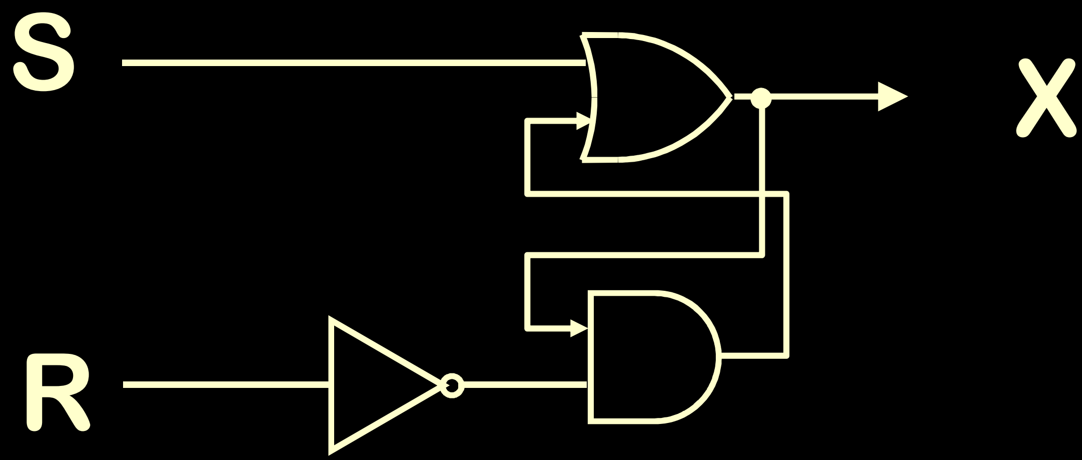
16 Functions of 2 Inputs

AB																
0 0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0 1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1 0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1 1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	0	AND		A		B	XOR	OR	NOR	EQ	B'		A'		NAND	1

Sequential Logic

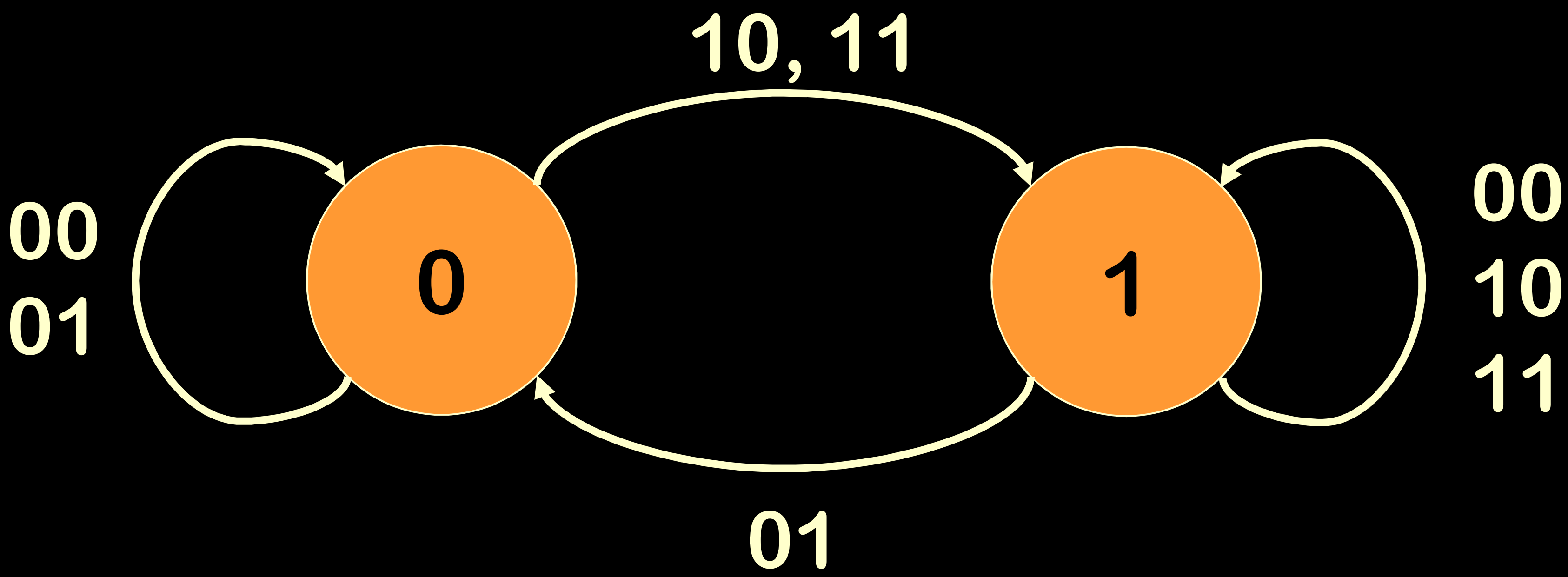
- Truth Table **with state**
- State Diagram
- Summary Table
- Abstraction
  - Change behavior with sequential logic in front and/or behind
- L1: Gate delays, oscillation

Set-Reset Flip-Flop



State	S	R	X (new state)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

State Diagram



00 – unchanged
01 – set to 0
10, 11 – set to 1

State	S	R	X (new state)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

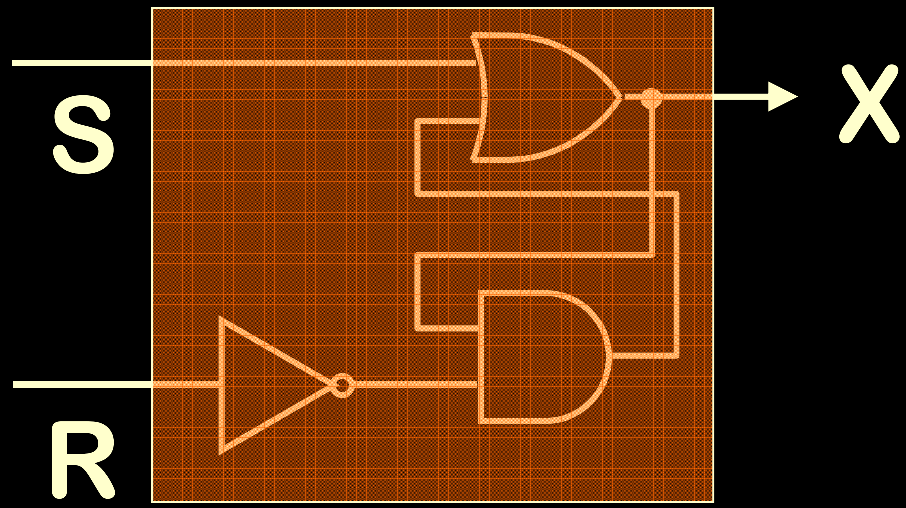
L1

Changing FF Behavior

D

C

D	C	S	R
0	0		
0	1		
1	0		
1	1		



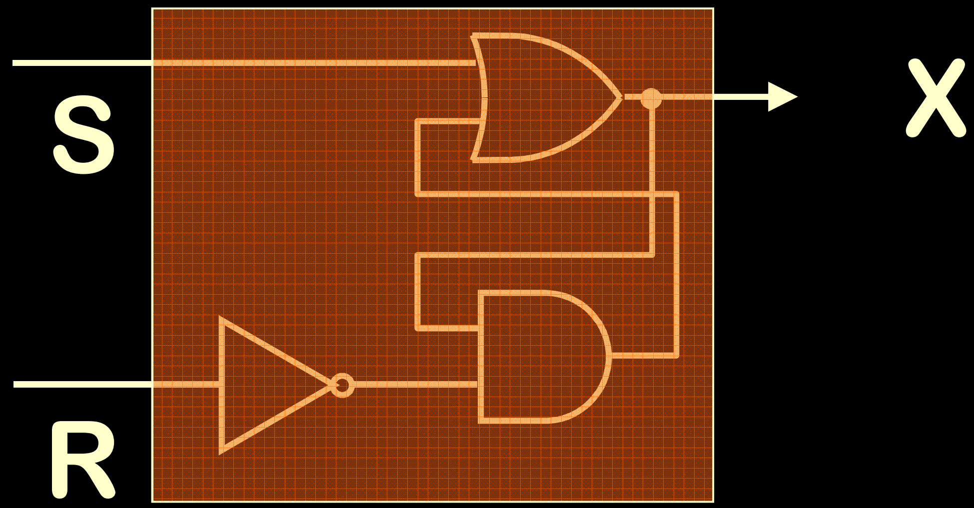
S	R	
0	0	Stays the same
0	1	Resets to 0
1	0	Sets to 1
1	1	Undefined

One Bit Clocked Memory

D

C

D	C	S	R
0	0	0	0
0	1	0	1
1	0	0	0
1	1	1	0

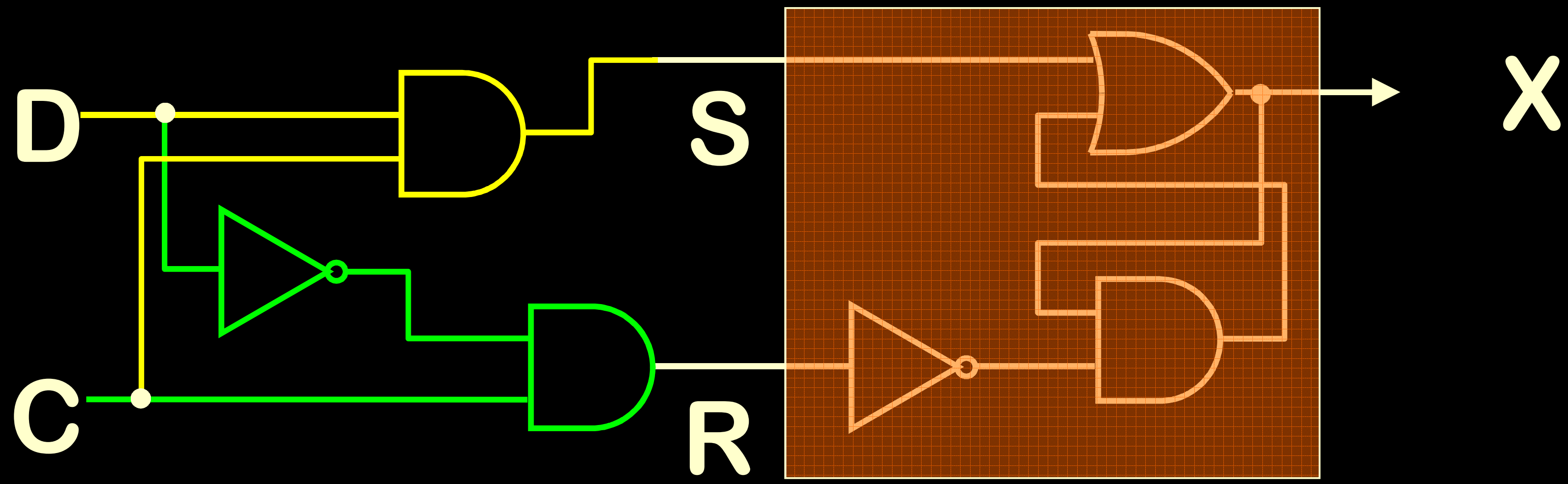


S = DC

R = D'C

S	R	
0	0	Stays the same
0	1	Resets to 0
1	0	Sets to 1
1	1	Undefined

One Bit Clocked Memory

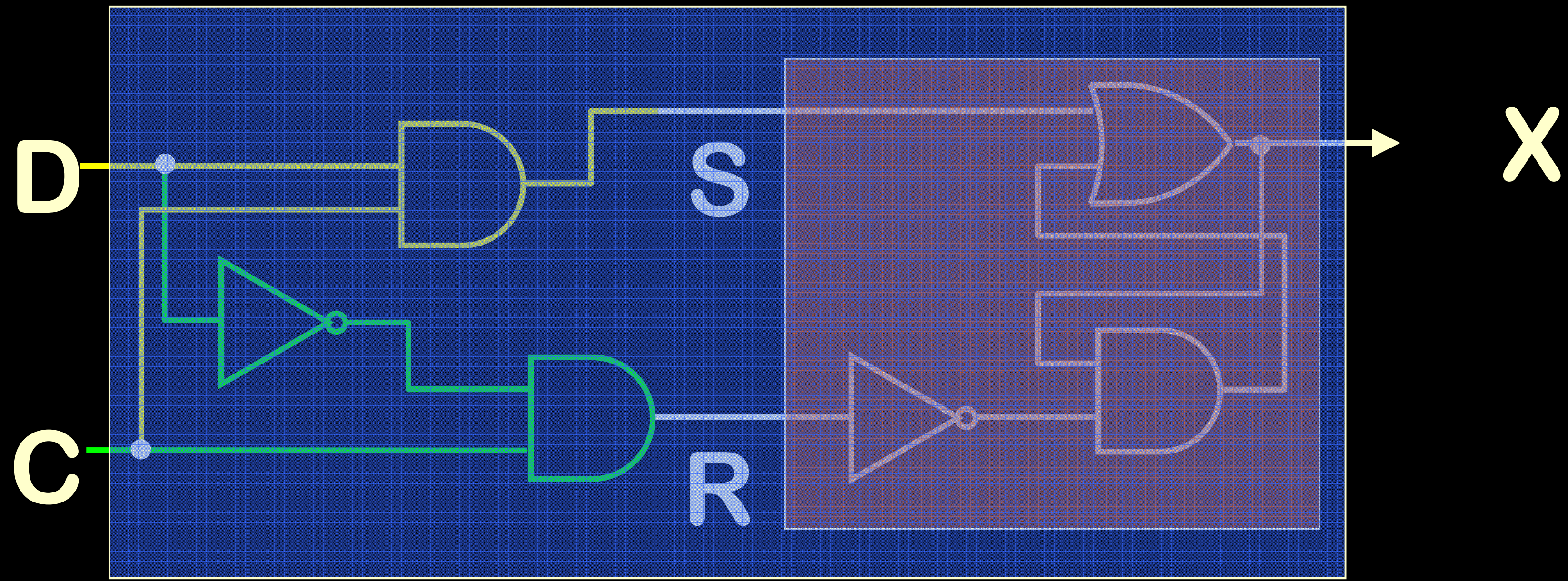


$S = DC$

$R = D'C$

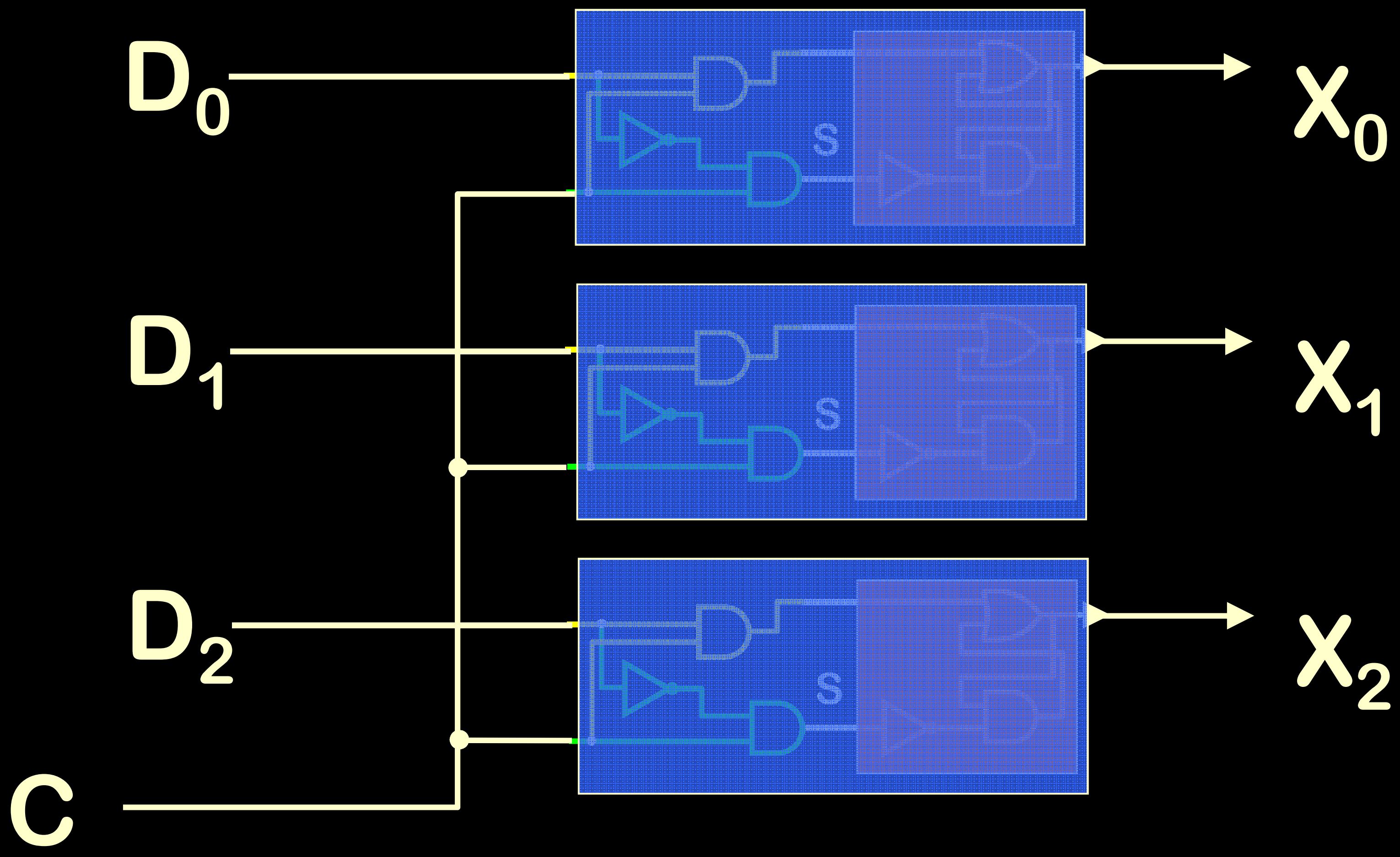
S	R	
0	0	Stays the same
0	1	Resets to 0
1	0	Sets to 1
1	1	Undefined

One Bit Clocked Memory

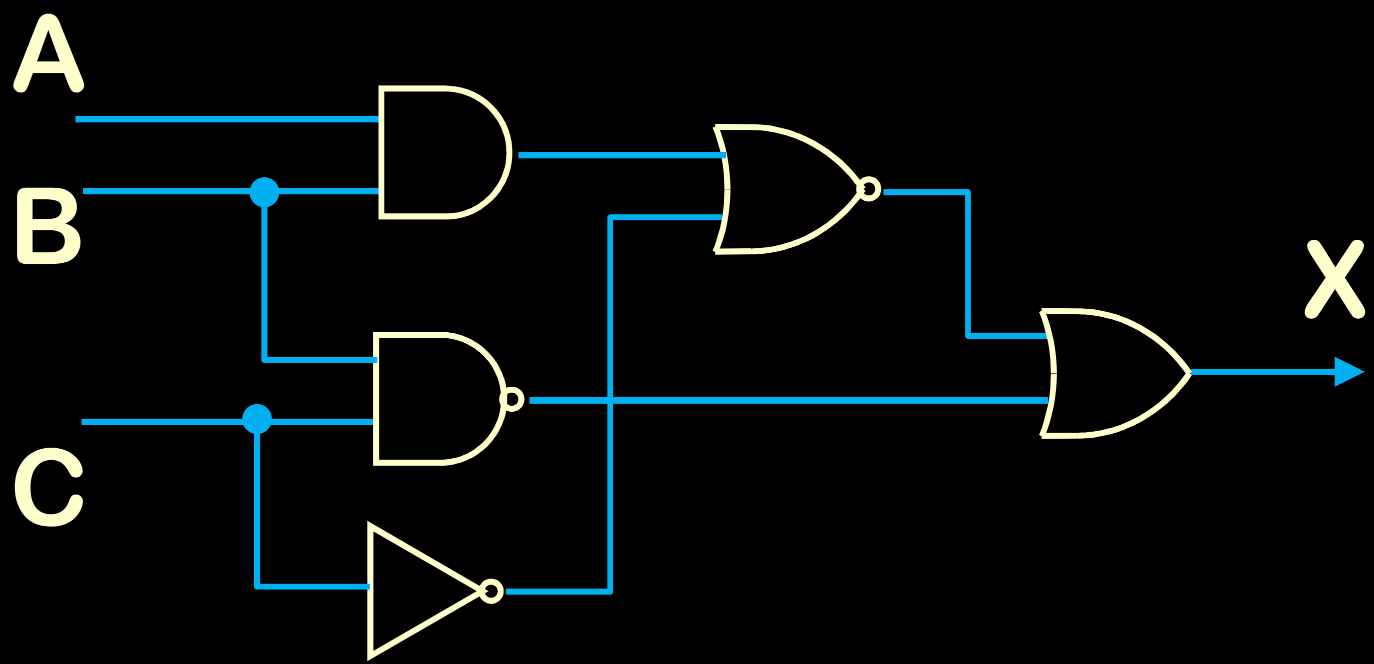


D	C	
0	0	Stays the same
0	1	Sets to 0
1	0	Stays the same
1	1	Sets to 1

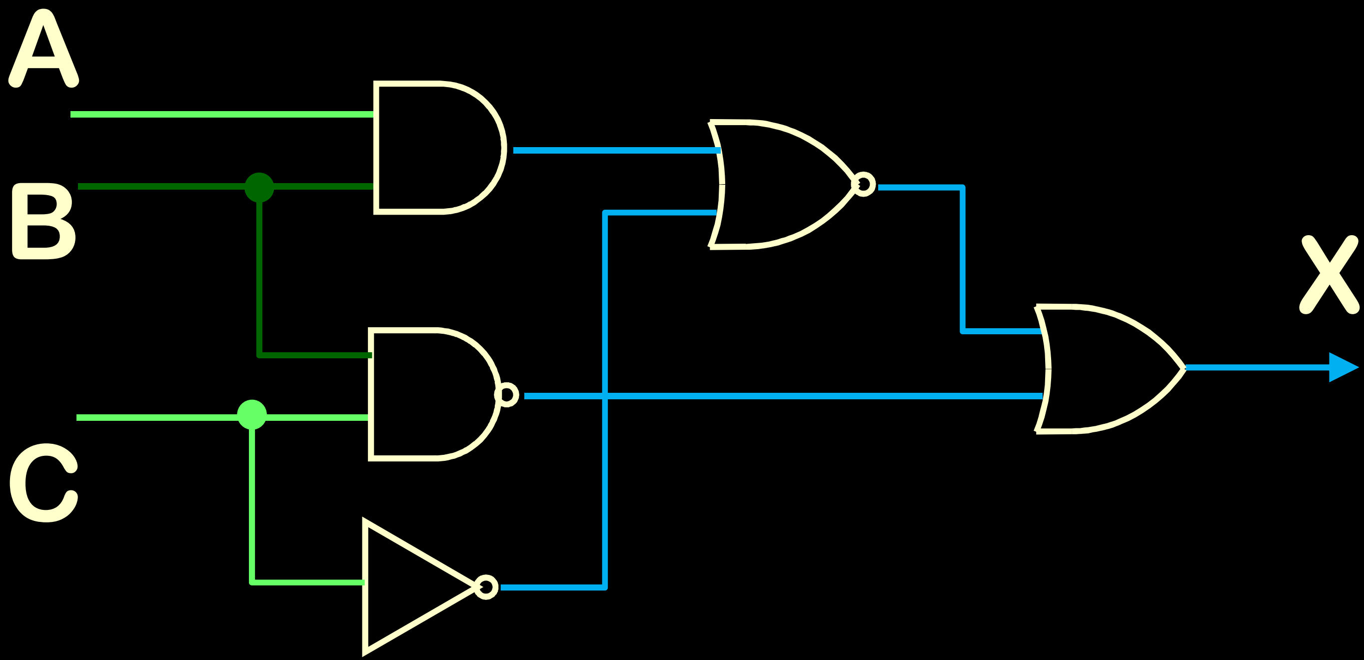
3 Bit Memory



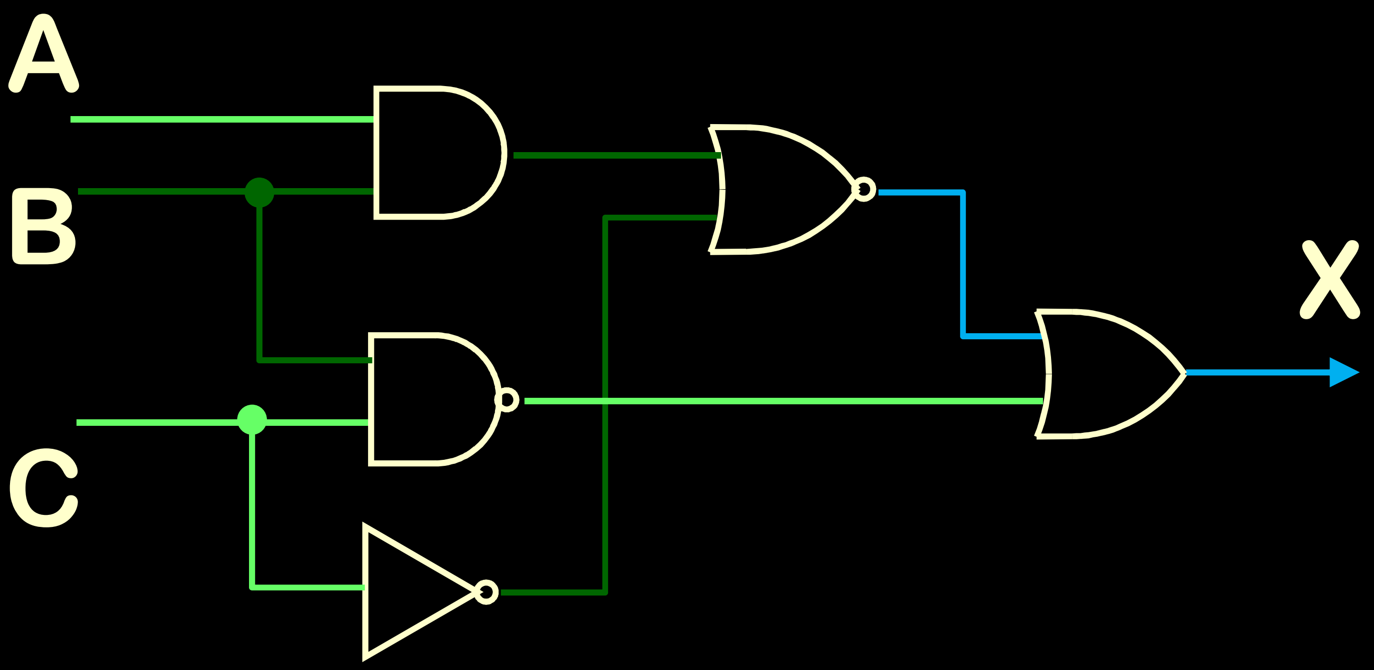
Gate Delays



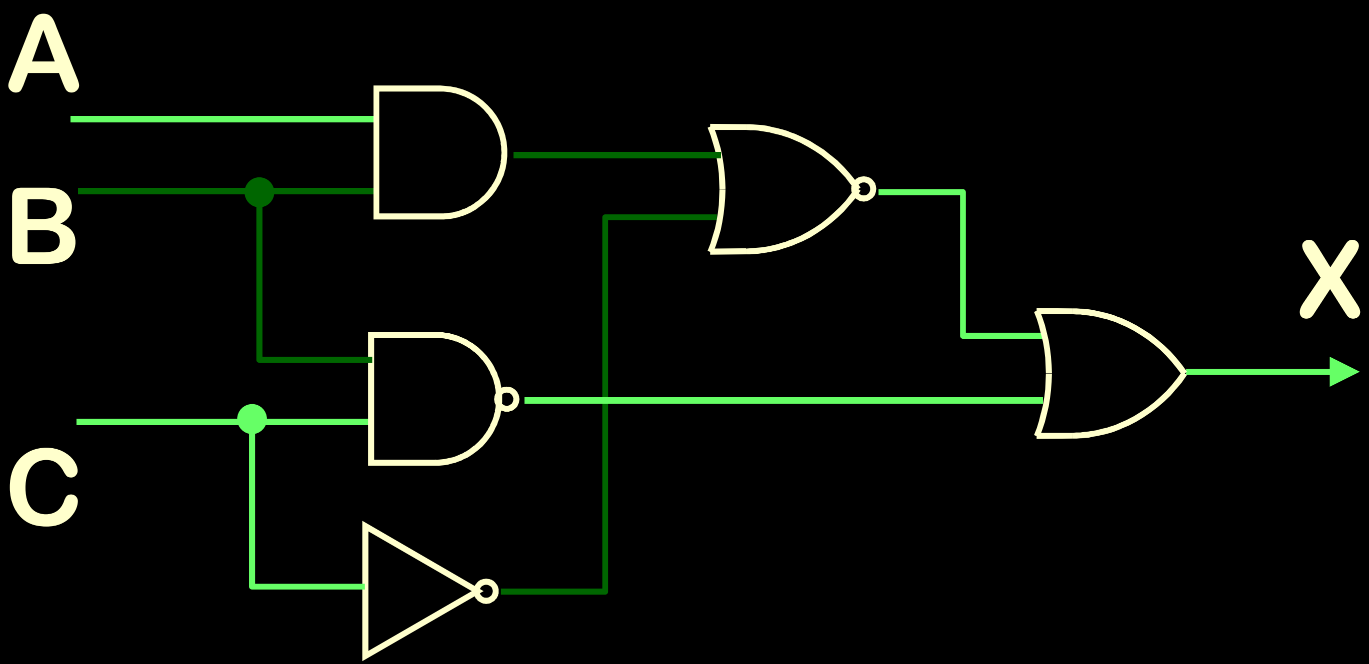
Gate Delays



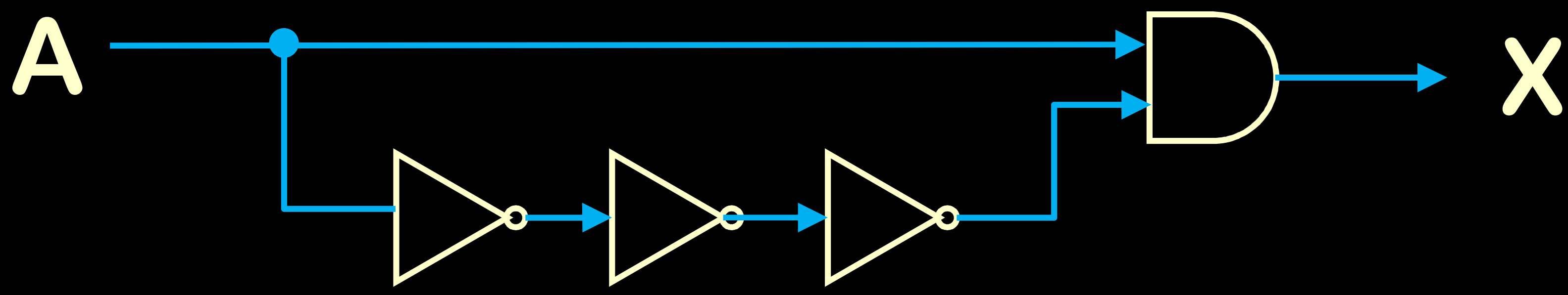
Gate Delays



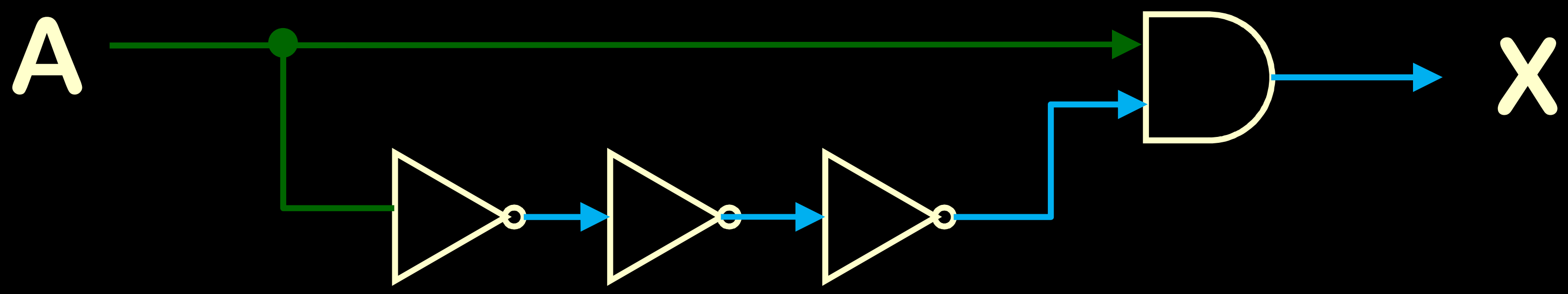
Gate Delays



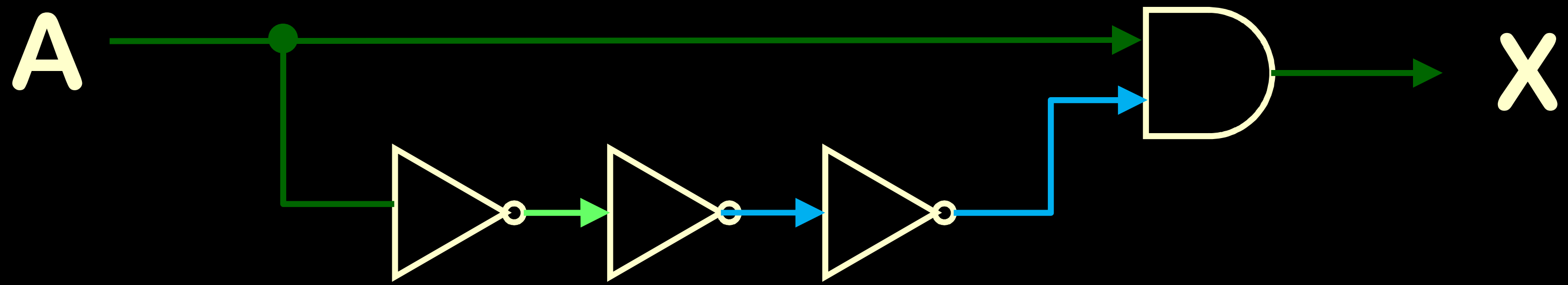
Toggle



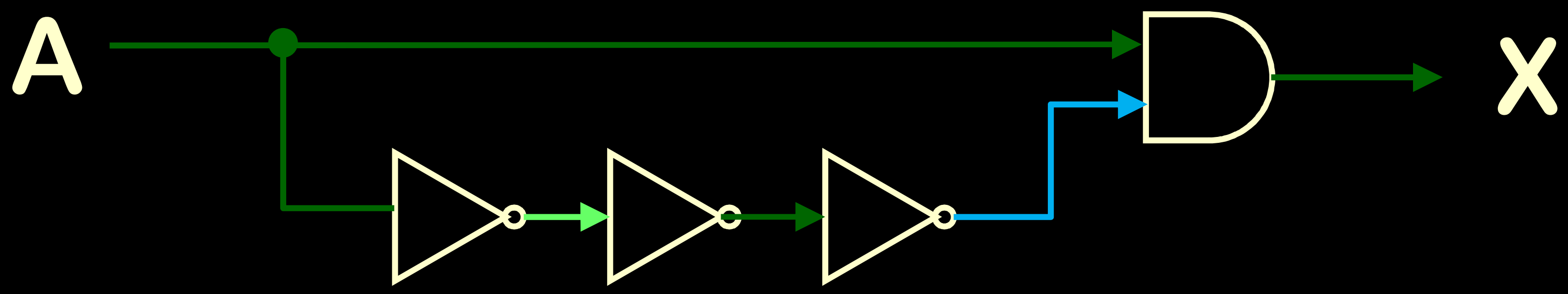
Toggle



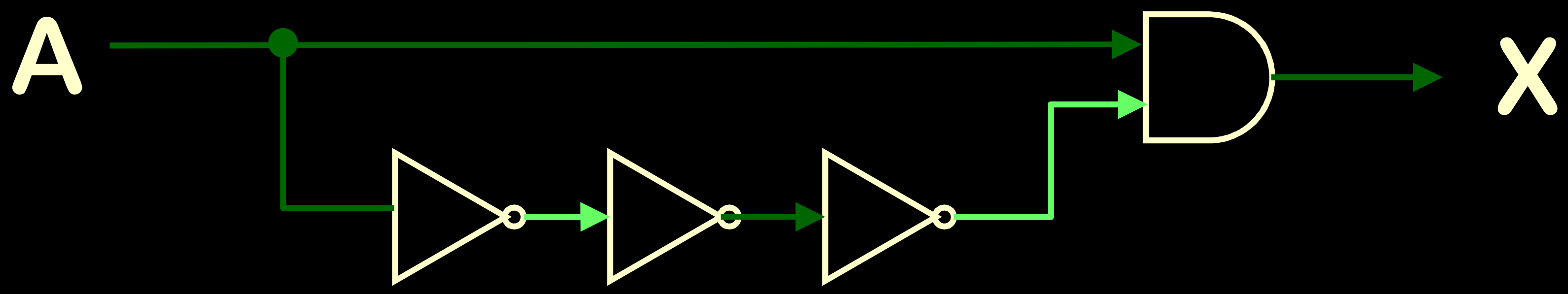
Toggle



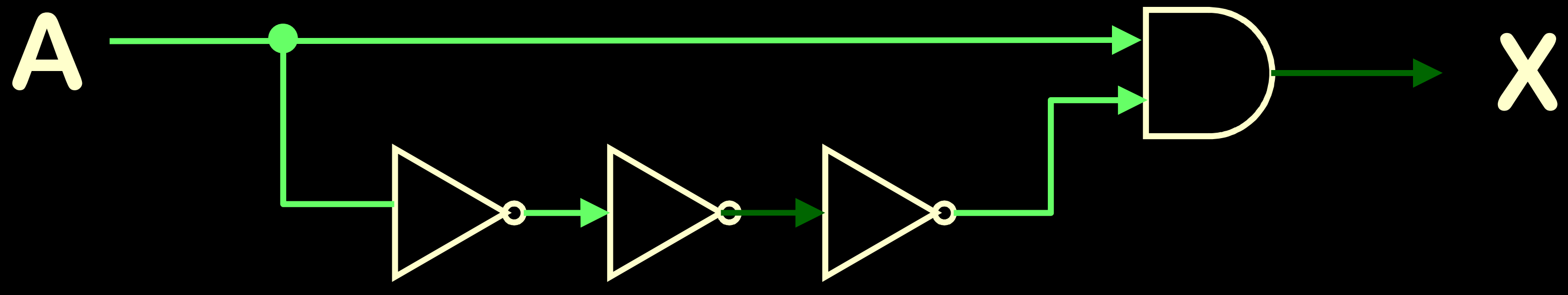
Toggle



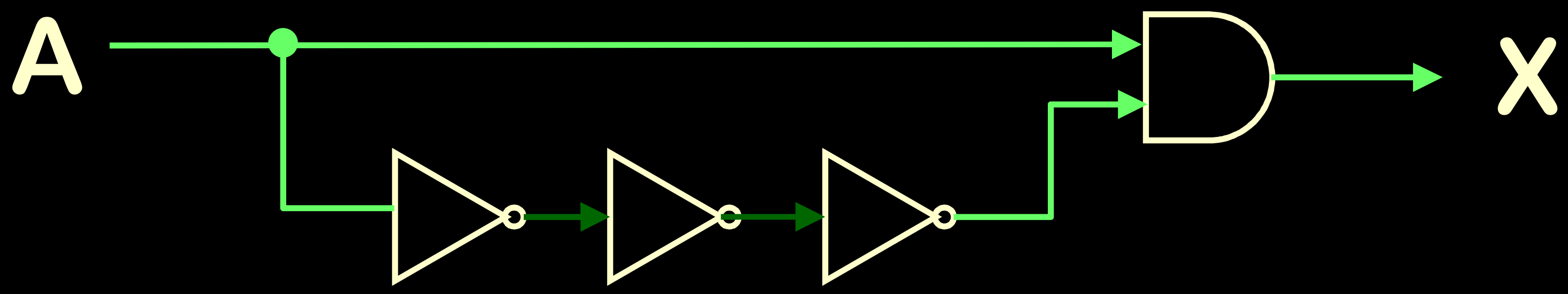
Toggle



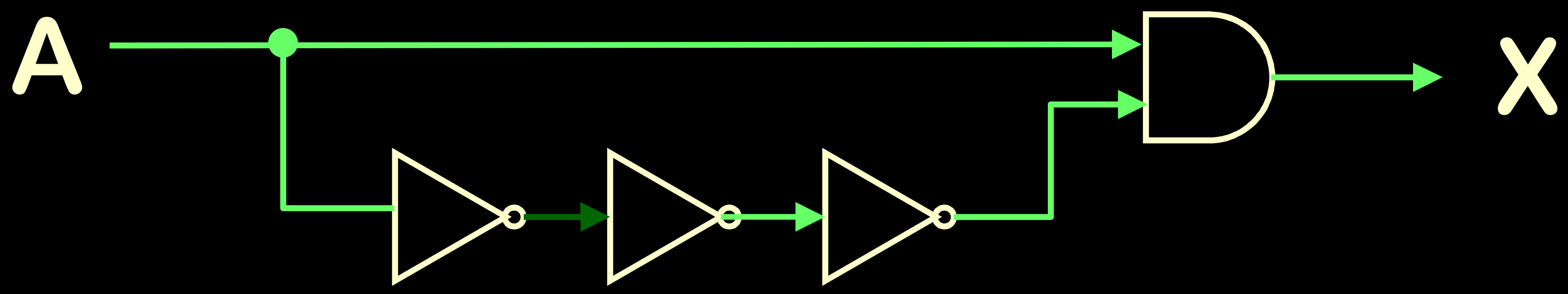
Toggle



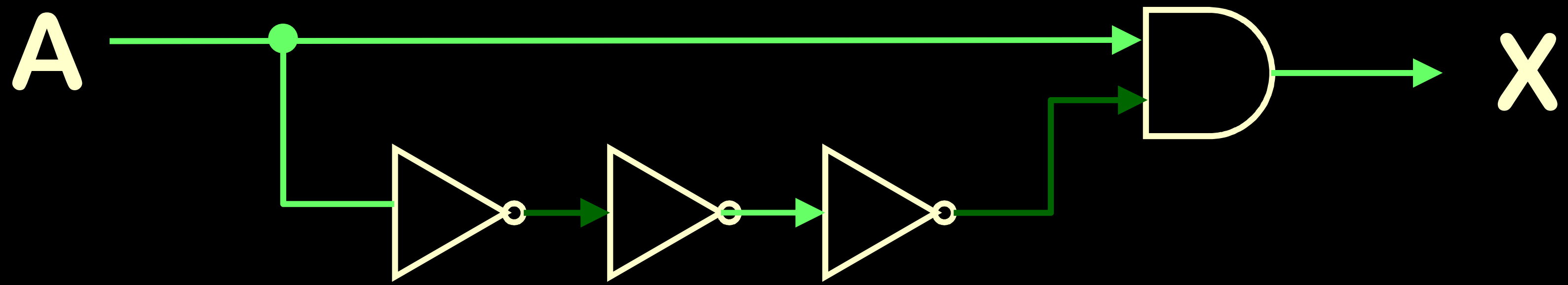
Toggle



Toggle



Toggle



Toggle