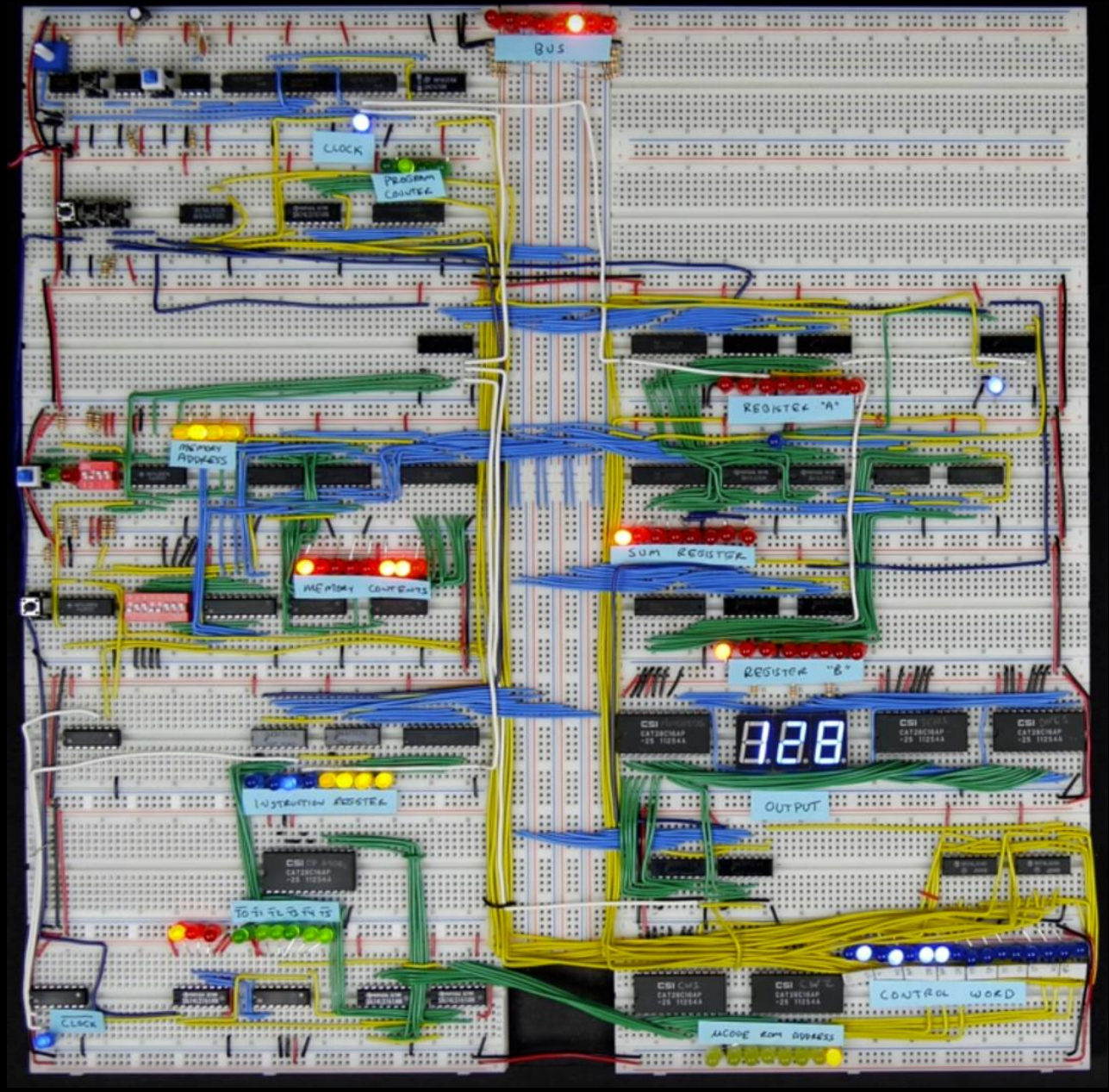
Intro to Computer Logic – Part II

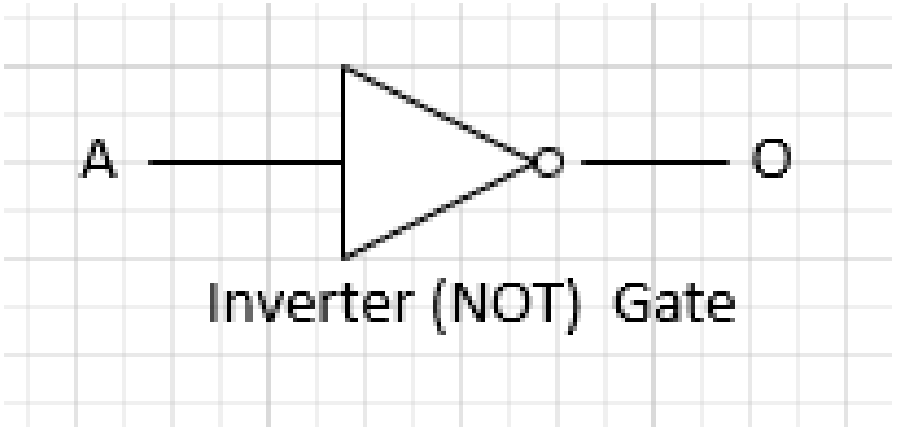


CPSC 240-09
John Overton

NOT

- Inverts (reverses) a boolean value
- Truth table for Boolean NOT operator:

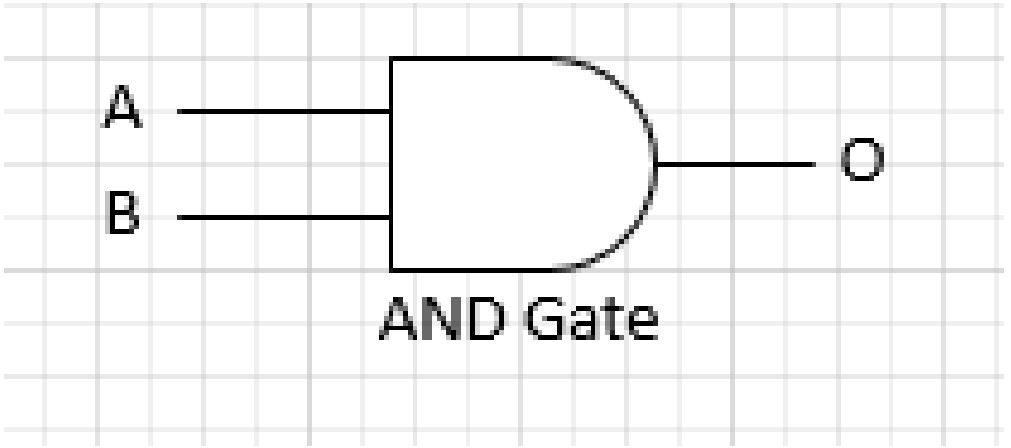
X	¬X
F	T
T	F



AND

- Truth table for Boolean AND operator:

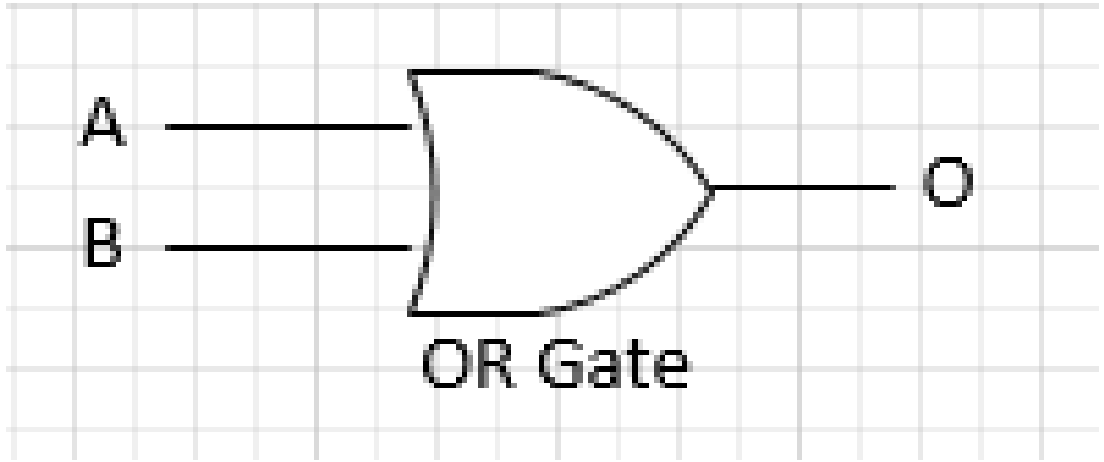
X	Y	X ∧ Y
F	F	F
F	T	F
T	F	F
T	T	T



OR

- Truth table for Boolean OR operator:

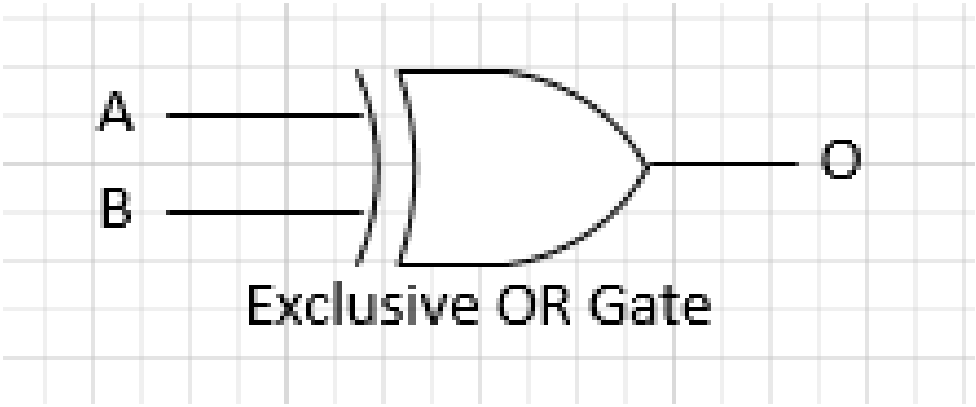
X	Y	X ∨ Y
F	F	F
F	T	T
T	F	T
T	T	T



Exclusive OR

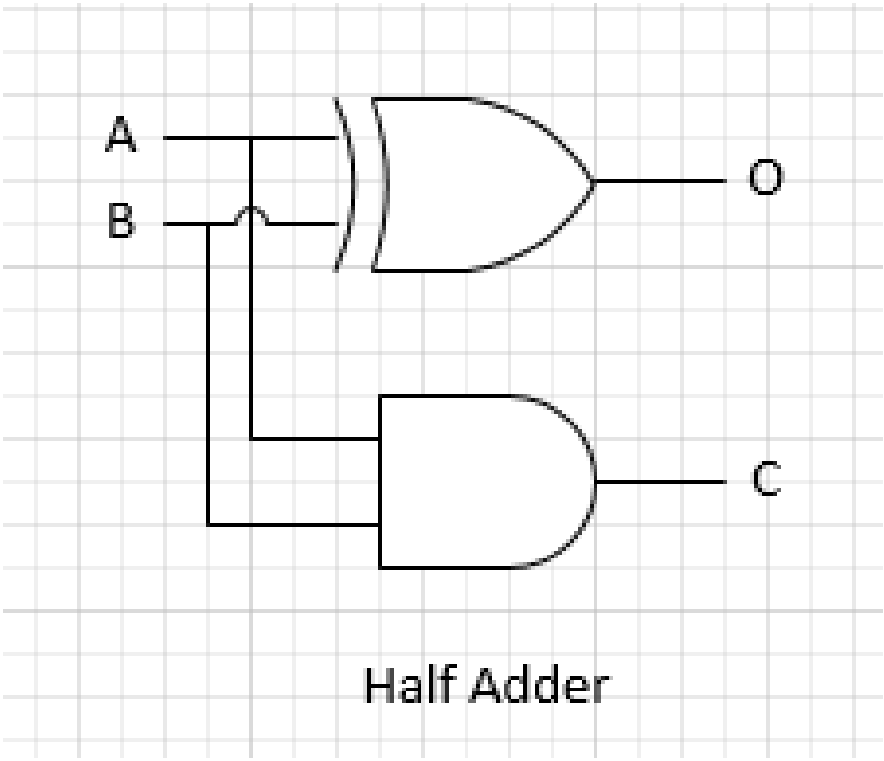
- Truth table for Boolean XOR operator:

Input		Output
A	B	O
0	0	0
0	1	1
1	0	1
1	1	0



Building an Adder

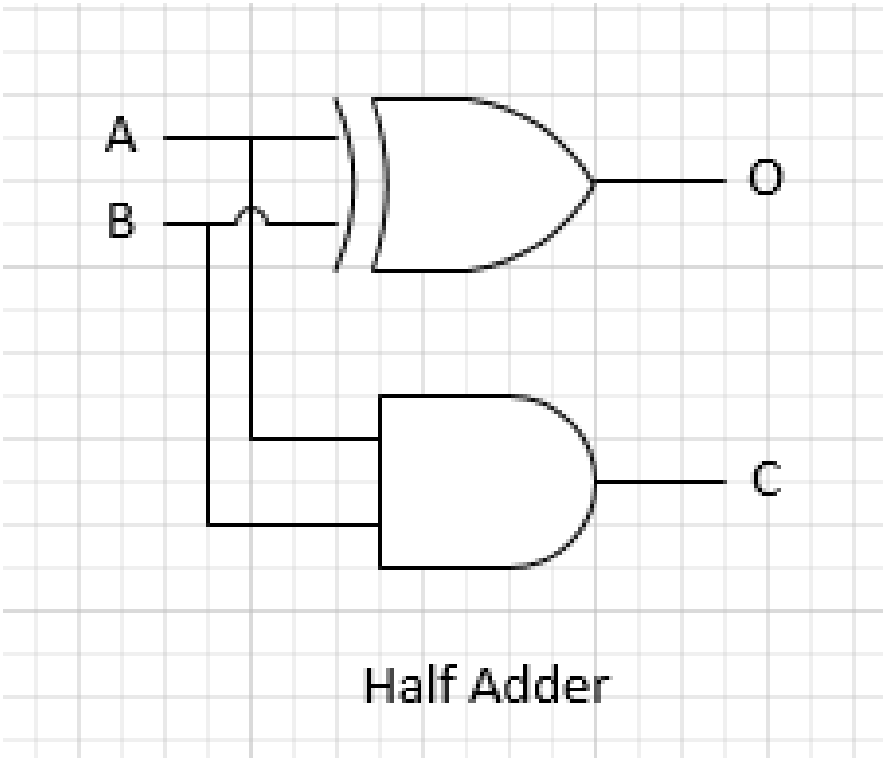
Input		Output	
A	B	O	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Half Adder

Building an Adder

Input		Output	
A	B	O	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

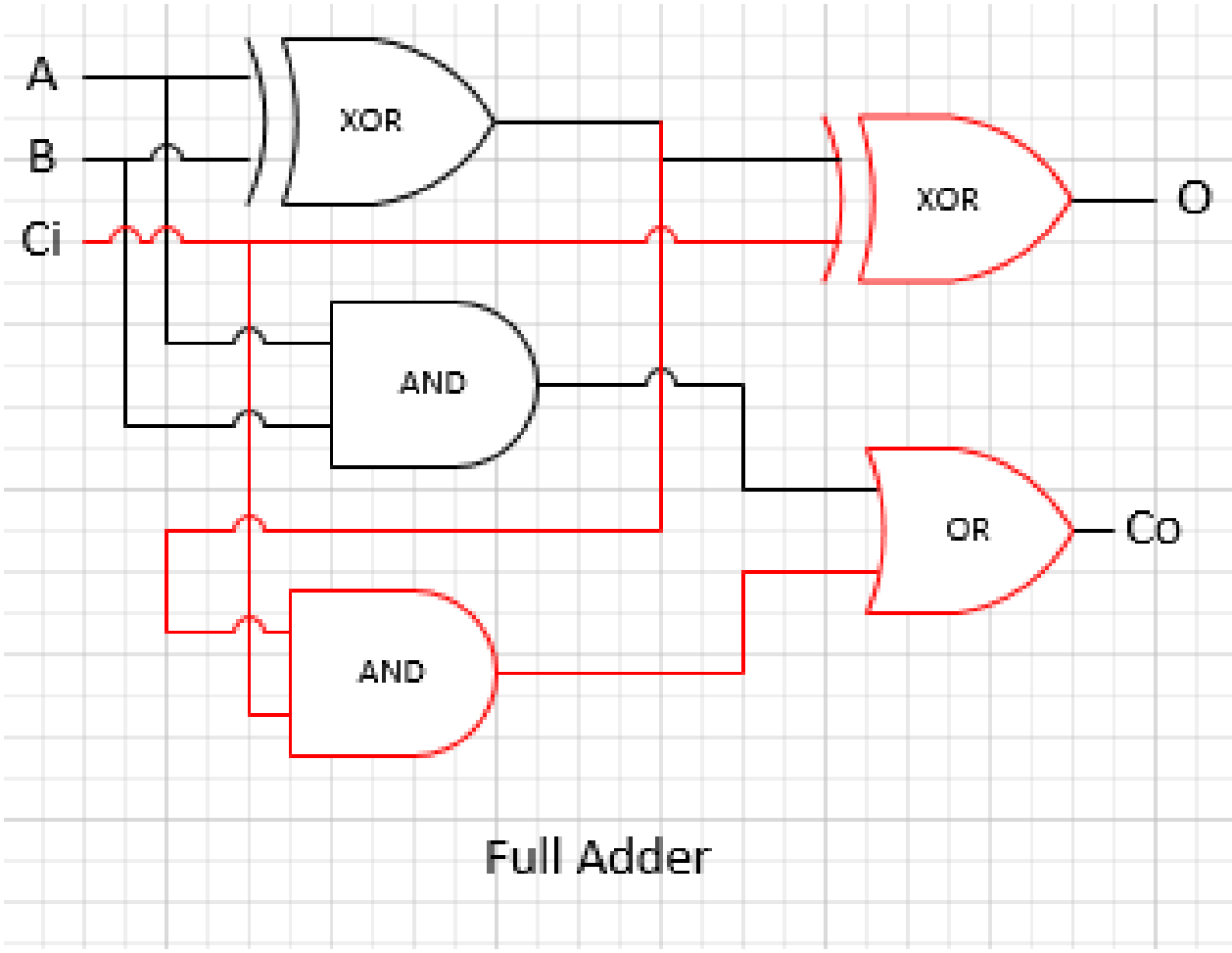


Half Adder

Building an Adder

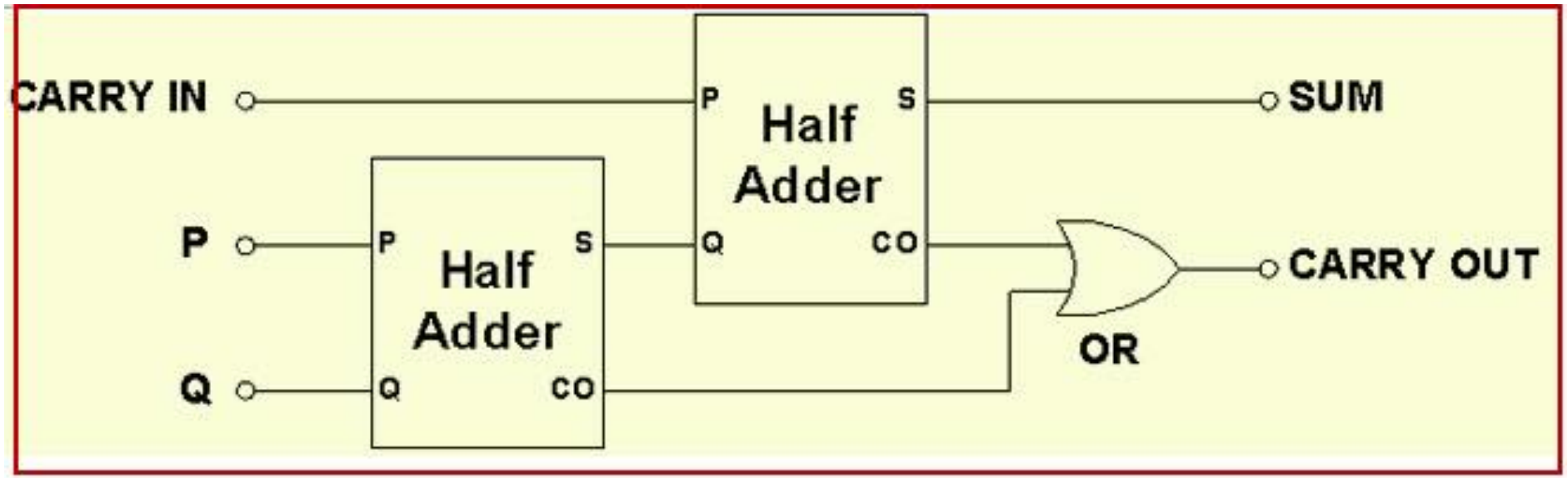
	Input		Output	
Cin	A	B	O	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

The black lines in the schematic on the left are from the original half-adder. The red lines and components are added to make a full adder.



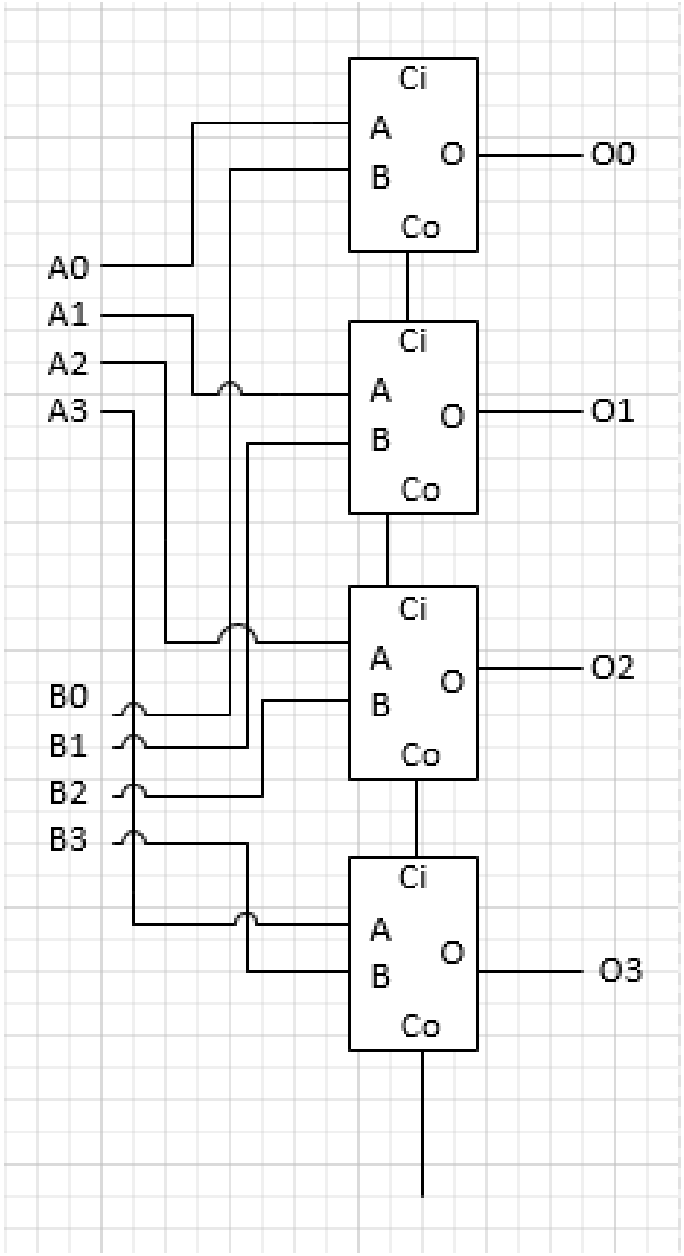
Building an Adder

- Here's another view of the full adder – it is actually two half adders put together plus an additional OR gate for Cout

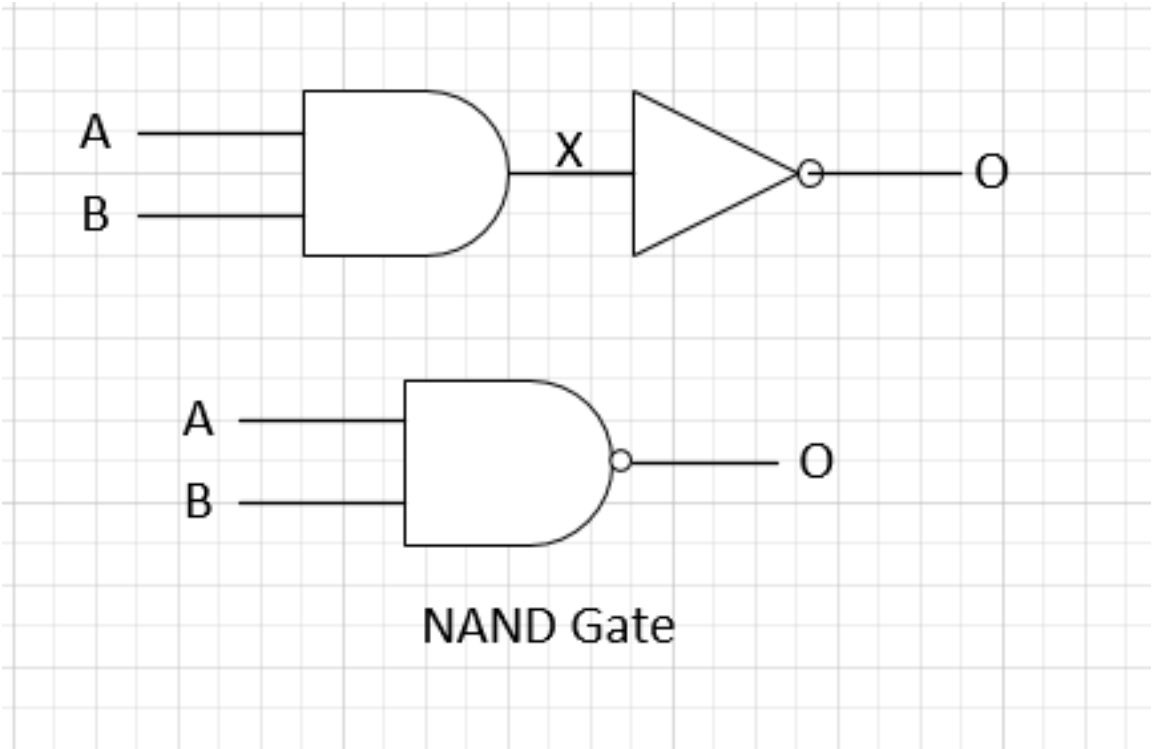


Combining 1-bit adders

- We can combine 1-bit adders to get the number of bits that we need

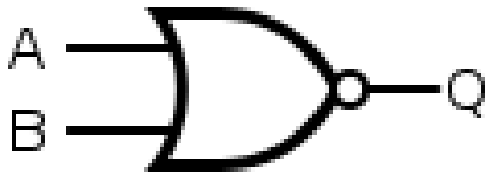


Combining the output of a logic gate with an inverter

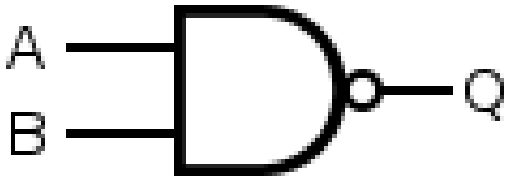


Input		Output	
A	B	X	O
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

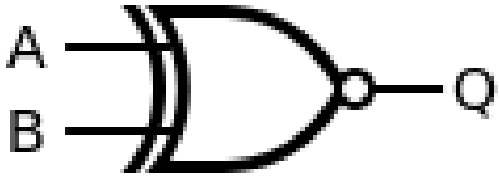
NOR, NAND, NXOR



INPUT		OUTPUT
A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0



INPUT		OUTPUT
A	B	A NAND B
0	0	1
0	1	1
1	0	1
1	1	0



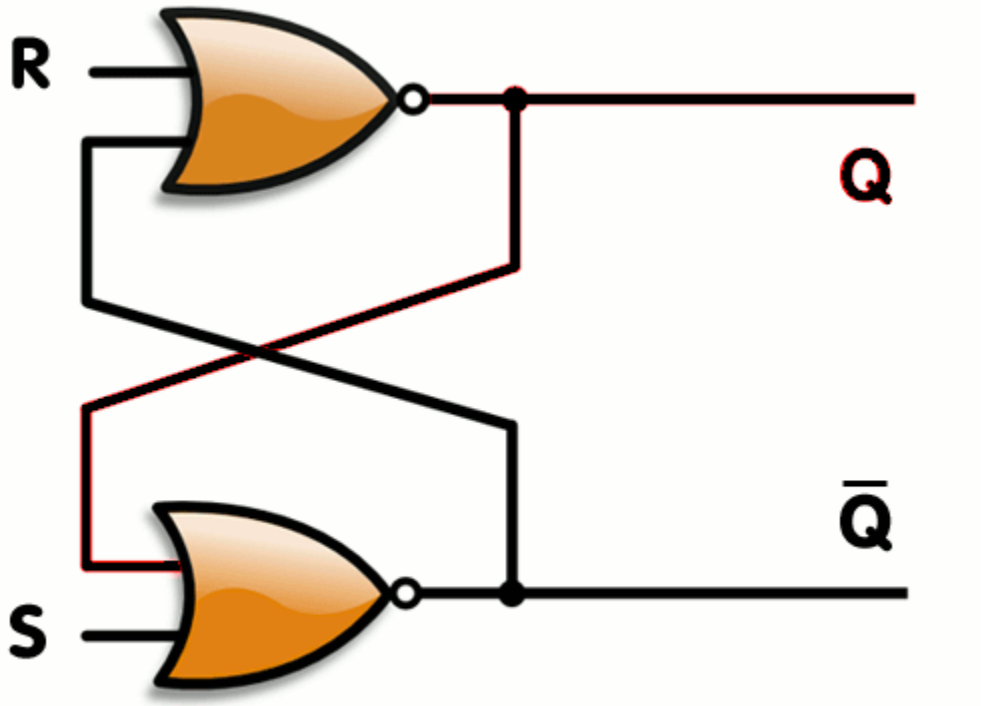
Input		Output
A	B	A XNOR B
0	0	1
0	1	0
1	0	0
1	1	1

Sequential circuits: Flip-Flops.

- Flip-flops are 1 bit circuit that has two stable states
- Generally, there are 4 varieties: SR, D, JK and T
- Flip-flops can be either simple (latched) or clocked
- Clocked flip-flops ignore their inputs until the clock signal is toggled:
  - Some clocked flip-flops change state on the rising edge of the clock signal
  - Other clocked flip-flops change state on the falling edge of the clock signal
- Flip-flops are subject to a problem called metastability – when clock and data change at about the same time and the order is not clear. Not usually a problem because hardware designers follow the flip-flop's timing specifications when using these parts in their designs

The SR Flip-Flop (latch)

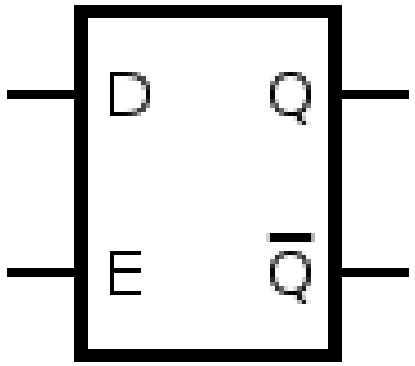
- When S and R are low, the outputs Q and Q’ are in a constant state.
- Q and Q’ are complementary. When Q is high, Q’ is low and vice-versa.
- If S is pulsed high when R is low, then Q goes high and stays high even when S returns to low.
- If R is pulsed high while S is held low, then Q goes low and stays low even when R returns to low.



The D Latch

- When E is high, Q follows D with Q’ the complement of Q
- When E is low, the output remains the same (no state change) and D is ignored

E – Enable
D – Data
Q Q’ - Output



Gated D latch truth table

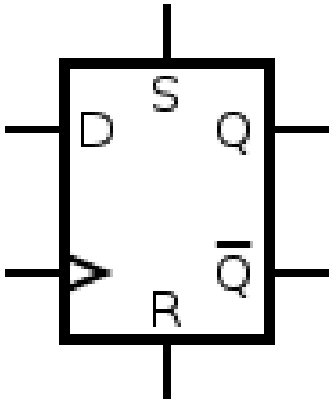
E/C	D		Q	$\overline{Q}$	Comment
0	X		Q_{prev}	$\overline{Q}_{prev}$	No change
1	0		0	1	Reset
1	1		1	0	Set

The D Flip-flop (clocked)

- The D flip-flop captures the value of the D input at a specific portion of the clock cycle – either the rising edge or the falling edge

> – Clock
D – Data
Q Q' – Output
S – Set
R - Reset

What is a clock cycle? It is the clock line going high and then low again. A rising edge is the portion when the clock line goes from low-to-high. The falling edge is the clock line going from high-to-low.



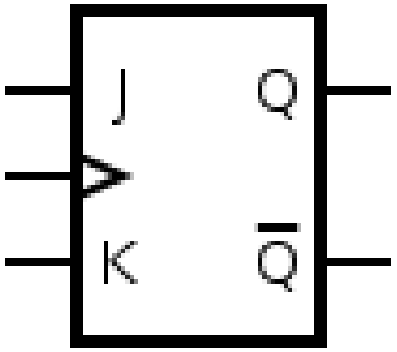
Clock	D	Q_{next}
Rising edge	0	0
Rising edge	1	1
Non-Rising	X	Q

The S (set) and R (reset) inputs can be used to immediately change state

Inputs				Outputs	
S	R	D	>	Q	Q'
0	1	X	X	0	1
1	0	X	X	1	0
1	1	X	X	1	1

JK Flip-Flop

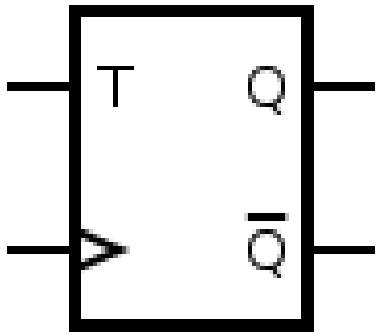
- The JK flip-flop is similar to the SR latch where J is similar to the Set input and K is similar to the Reset input.
- However, all state changes are synced to a clock input: When J is high and K is low, on the next clock rising edge, Q will go high. When K is high and J is low, on the next clock rising edge Q will go low.
- When both J and K are low, no state change happens during clock cycles
- However, an interesting case happens when both J and K are high: In this case, on the rising clock edge, no matter what state Q was in, it will change to the opposite state – it will flip.



J	K	Clock	Comment	Qnext
0	0	Rising	Hold state	Q
0	1	Rising	Reset	0
1	0	Rising	Set	1
1	1	Rising	toggle	Q′

T Flip-flop

- When T is high, every clock cycle will toggle the outputs



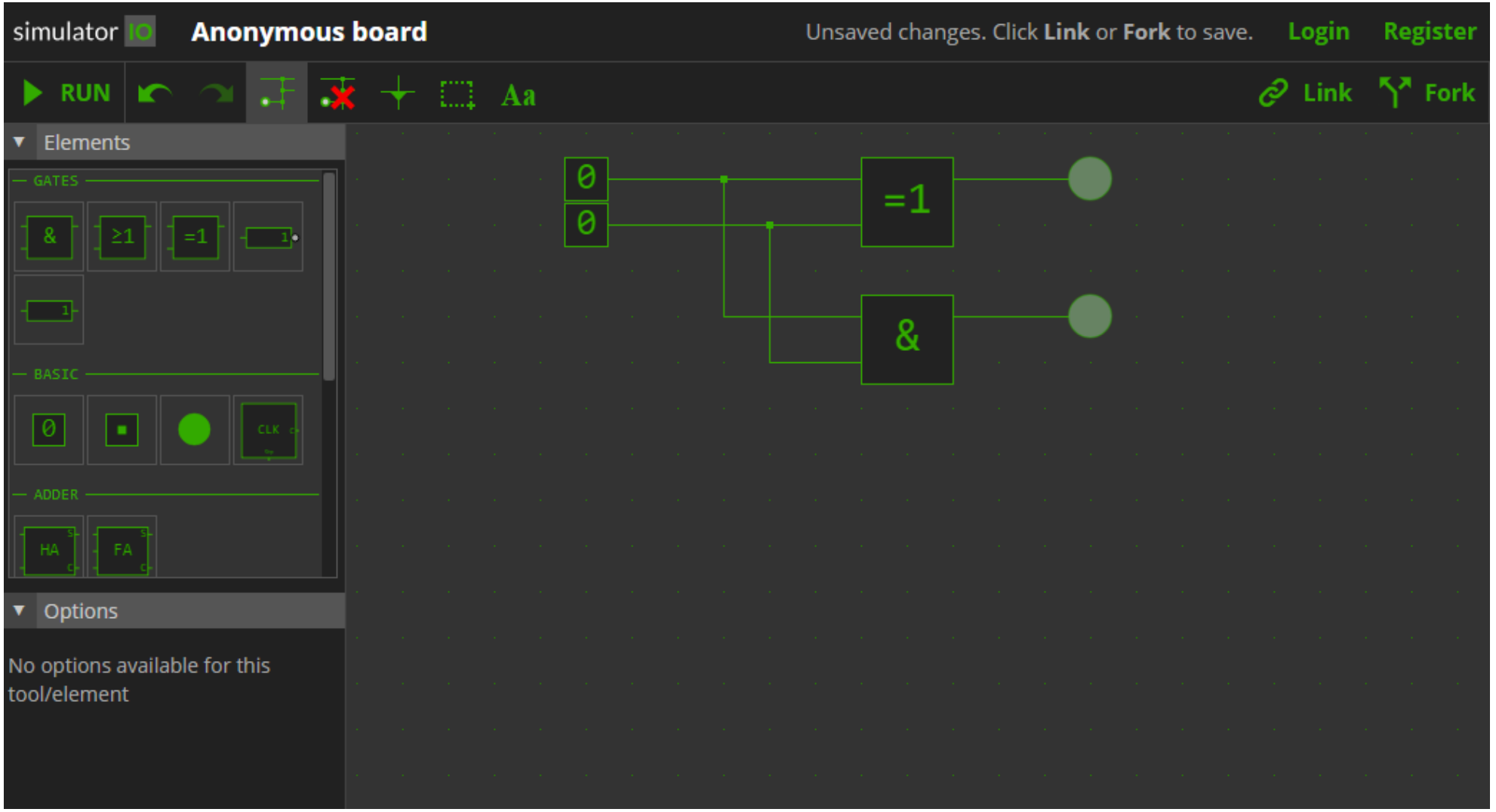
T	Q	Clock	Comment	Qnext
0	0	Rising	Hold state	0
0	1	Rising	Hold state	1
1	0	Rising	Toggle	Q' (1)
1	1	Rising	Toggle	Q' (0)

Assignment

- Use one of the online simulators to see how various flip-flops work:
  1. Design two circuits, one with a D flip-flop and with a D latch.
  2. When you used the clocked D-Type flip-flop in CircuitVerse, did it change state on the rising clock edge or the falling clock edge?
  3. Can you use a JK flip-flop to make a T flip-flop?
  4. Can you build an 8 bit binary counter out of T flip-flops? (For a hint, see the hand-out on the DM74LS393 Dual 4-bit Binary Counter.)

Upload your work as various screen shots. Write your answer for #2 in the text box when uploading your screen shots.

https://simulator.io/



https://circuitverse.org/simulator