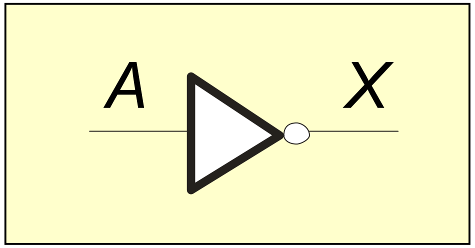
Digital Fundamentals

Thomas L. Floyd

Logic Gates /
Boolean Algebra and Logic Simplification
Chapter 3

The Inverter

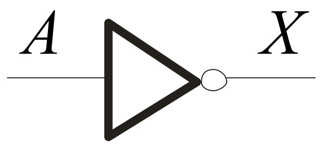
Boolean **NOT** operation



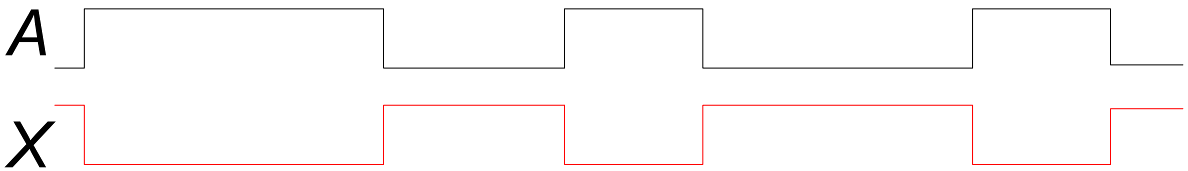
Input	Output
A	*X*
LOW (0)	HIGH (1)
HIGH (1)	LOW (0)

The **NOT** operation (complement) : $\boldsymbol{X = \overline{A}}$.

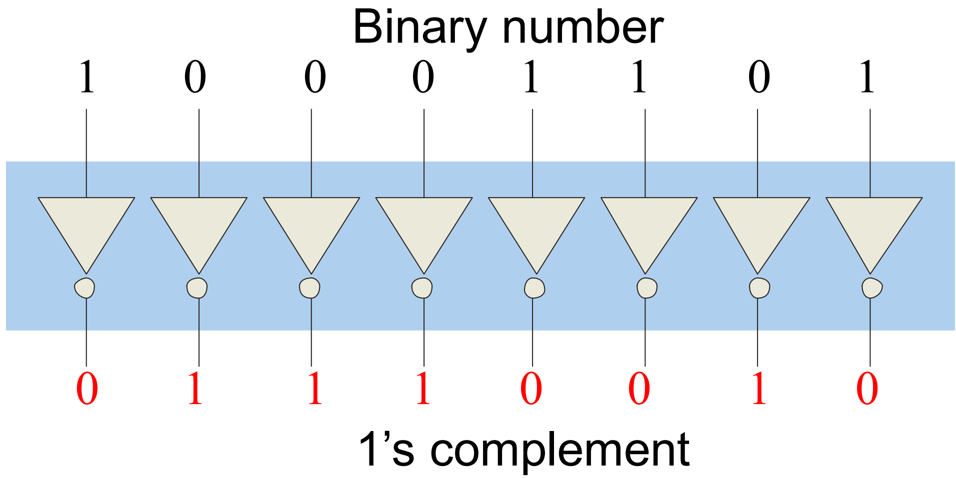
An Inverter Application



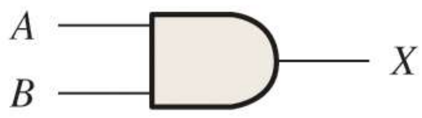
Example waveforms:

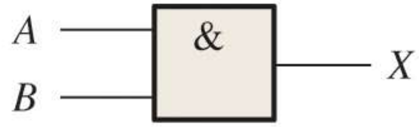


Example: 1's Complement



The AND Gate





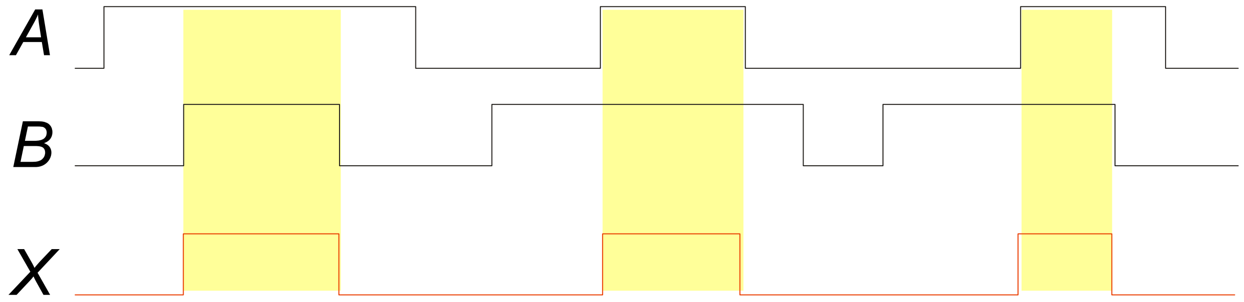
Outputs HIGH when all inputs are HIGH

Inputs		Output
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

The **AND** operation : $X = A \cdot B$ or $X = AB$.

The AND Gate

Example waveforms:



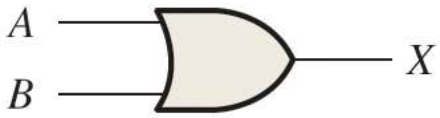
Example: Masking

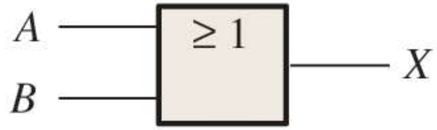
If the binary number 10100011 is ANDed with the mask 00001111, what is the result?

00000011

Ch.3 Summary

The OR Gate



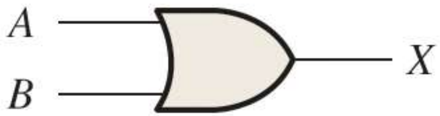


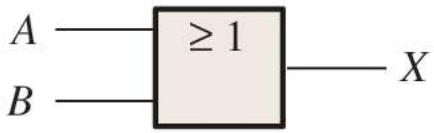
Outputs HIGH if any input is HIGH

Inputs		Output
A	*B*	*X*
0	0	0
0	1	1
1	0	1
1	1	1

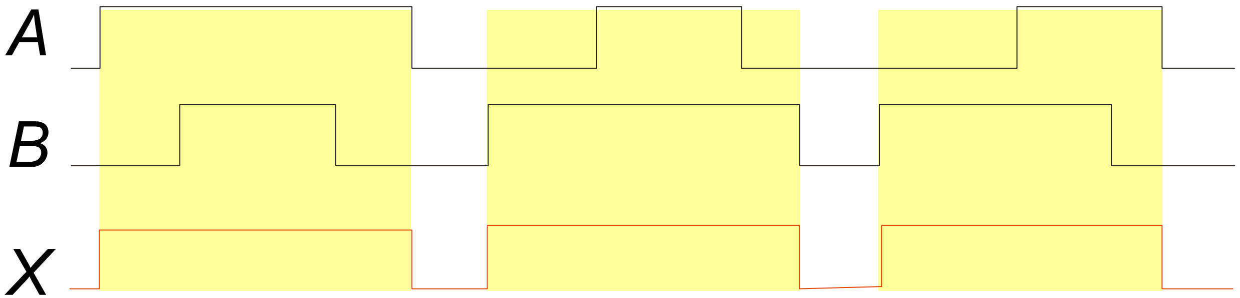
The **OR** operation : $\boldsymbol{X = A + B}$.

The OR Gate





Example waveforms:



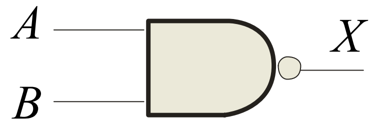
Example: ASCII (Set to specific bit)

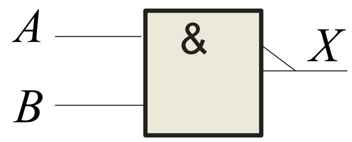
5th bit: 1 if lowercase, 0 if uppercase

(ASCII letter) OR (8-bit mask 00100000)?

The resulting letter will be lower case.

The NAND Gate



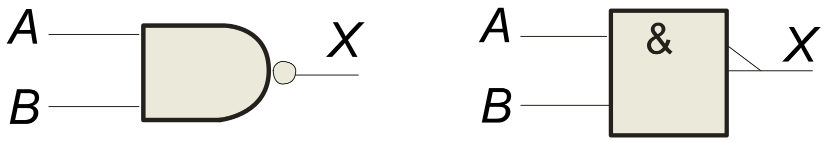


Outputs LOW when all inputs are HIGH

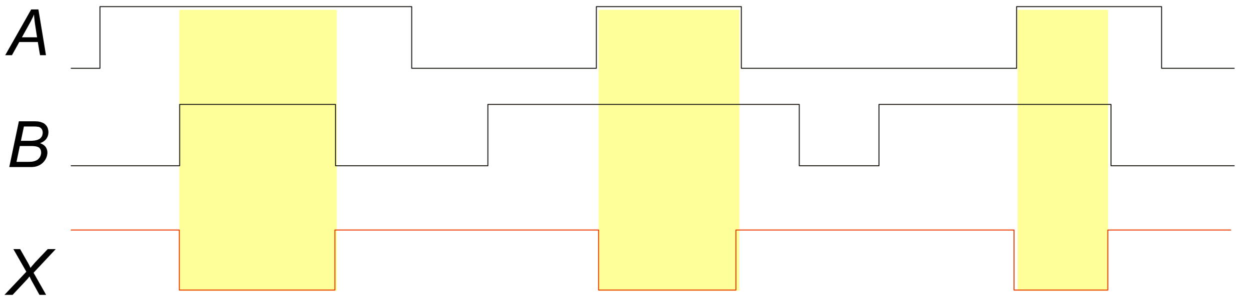
Inputs		Output
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

The **NAND** operation $X = \overline{A \cdot B}$ (Or, $X = \overline{AB}$.)

The NAND Gate

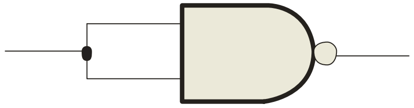


Example waveforms:

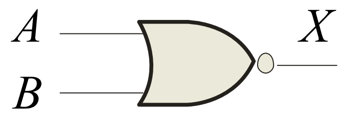


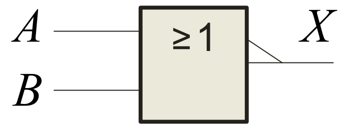
The NAND gate is “universal”.

Inverter using NAND gate:



The NOR Gate



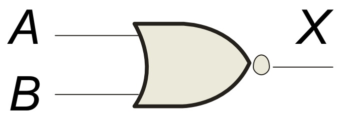


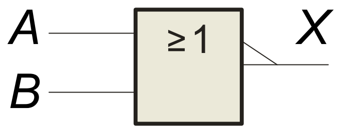
Outputs LOW if any input is HIGH

Inputs		Output
A	*B*	*X*
0	0	1
0	1	0
1	0	0
1	1	0

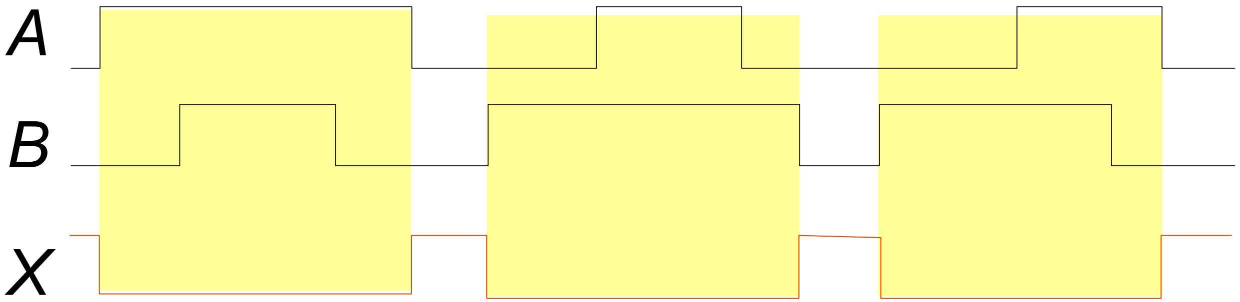
The **NOR** operation : $X = \overline{A + B}$.

The NOR Gate



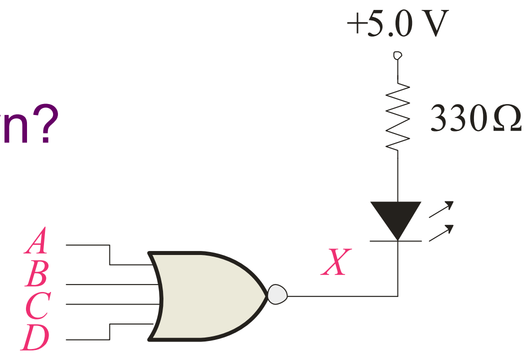


Example waveforms:



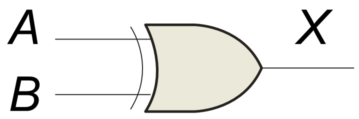
The NOR operation will produce a LOW if any input is HIGH.

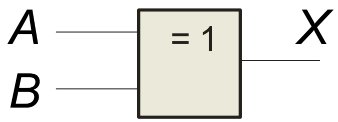
When is the LED is ON for the circuit shown?



The LED will be on when any of the four inputs are HIGH.

The XOR Gate





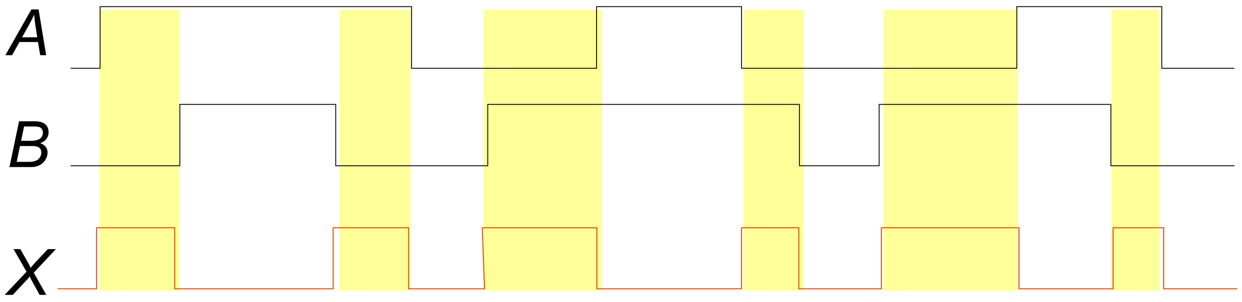
Outputs HIGH only when both inputs are at opposite logic levels.

Inputs		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

The **XOR** operation : $X = A\overline{B} + \overline{A}B$ or $X = A \oplus B$.

The XOR Gate

Example waveforms:

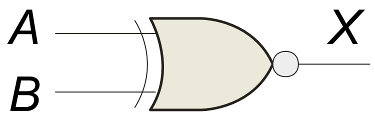


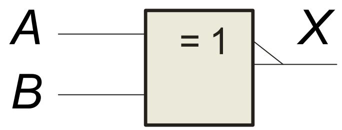
Notice that the XOR gate will produce a HIGH only when exactly one input is HIGH.

If the *A* and *B* waveforms are both inverted for the above waveforms, how is the output affected?

There is no change in the output.

The XNOR Gate



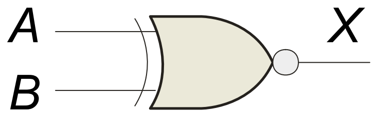


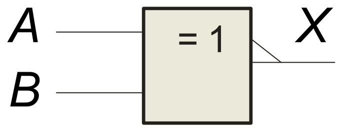
The **XNOR gate** produces a HIGH output only when both inputs are at the same logic level. The truth table is

Inputs		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

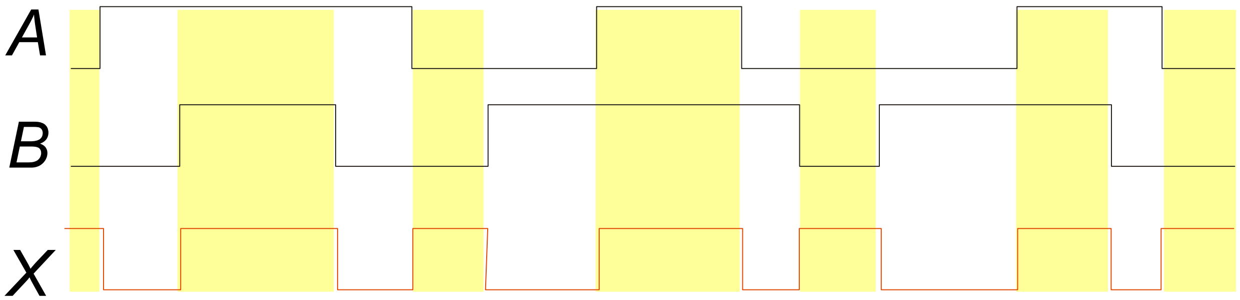
The **XNOR** operation : $X = \overline{A}\overline{B} + AB$ or $X = A \odot B$.

The XNOR Gate





Example waveforms:



Notice that the XNOR gate will produce a HIGH when both inputs are the same. This makes it useful for comparison functions.

If the *A* waveform is inverted but *B* remains the same, how is the output affected?

The output will be inverted.

Fixed Function Logic