Contents

1 Gates, Expressions, Circuits, and Analysis 1/13 and 1/15

Topics:

- Digital Logic Gates
- Boolean Algebra
- Combination Logic Circuits
- Sum of Products
- Karnaugh Maps

1.1 Logic Gates

A gate has (for example NOT gate):

1. Name
2. Schematic Diagram
   - Input, A, for example, with boolean (0 or 1)
   - Output, Y, for example, boolean (0 or 1)
3. Boolean Expressions, i.e. $Y = \overline{A}$
4. Truth Table

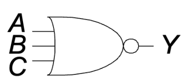
Example

We can also have two or more input gates:

- AND $\rightarrow Y = AB$, A and B must be true
- OR $\rightarrow Y = A + B$, A or B must be true
- XOR $\rightarrow Y = A \oplus B$
- NAND $\rightarrow Y = \overline{AB}$
- NOR $\rightarrow Y = \overline{A + B}$
- XNOR $\rightarrow Y = \overline{A \oplus B}$

A nice to know is that if the NOT's are the actual gate, then it would turn, for example, $X = \overline{A}\,\overline{B} \neq X = \overline{AB}$.

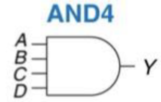
NOR3



$Y = \overline{A+B+C}$

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

AND4



$Y = ABCD$

A	C	B	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Figure 1: It can also have more than 2 inputs as seen here with their truth tables

1.2 Boolean Algebra

Symbols and Boolean operators:

$$x \cdot y,\ \ xy,\ \ x \wedge y,\ \ \text{AND}(x,y),\ \ x\ \text{AND}\ y$$

$$x + y,\ \ x \vee y,\ \ \text{OR}(x,y),\ \ x\ \text{OR}\ y$$

$$\overline{x},\ \ x',\ \ \neg x,\ \ \text{NOT}(x),\ \ \text{INV}(x)$$

$$\overline{x \cdot y},\ \ \overline{x \wedge y},\ \ \overline{xy},\ \ \text{NAND}(x,y),\ \ x\ \text{NAND}\ y$$

$$\overline{x + y},\ \ \overline{x \vee y},\ \text{NOR}(x,y),\ \ x\ \text{NOR}\ y$$

$$x \oplus y,\ \ \text{XOR}(x,y),\ \ x\ \text{XOR}\ y$$

$$x \overline{\oplus} y,\ \ \overline{x \oplus y},\ \ \text{XNOR}(x,y),\ \ x\ \text{XNOR}\ y$$

Figure 2: Notation before we get started

Moreover, here are some basic identities of boolean algebra

Basic Identities of Boolean Algebra

1.	$X + 0 = X$	2.	$X \cdot 1 = X$	
3.	$X + 1 = 1$	4.	$X \cdot 0 = 0$	
5.	$X + X = X$	6.	$X \cdot X = X$	
7.	$X + \overline{X} = 1$	8.	$X \cdot \overline{X} = 0$	
9.	$\overline{\overline{X}} = X$			
10.	$X + Y = Y + X$	11.	$XY = YX$	Commutative
12.	$X + (Y + Z) = (X + Y) + Z$	13.	$X(YZ) = (XY)Z$	Associative
14.	$X(Y + Z) = XY + XZ$	15.	$X + YZ = (X + Y)(X + Z)$	Distributive
16.	$\overline{X + Y} = \overline{X} \cdot \overline{Y}$	17.	$\overline{X \cdot Y} = \overline{X} + \overline{Y}$	DeMorgan's

Figure 3: Some basic identities

Definition

Variable Substitution, is a way of substitution that makes it more tangible and math more easy

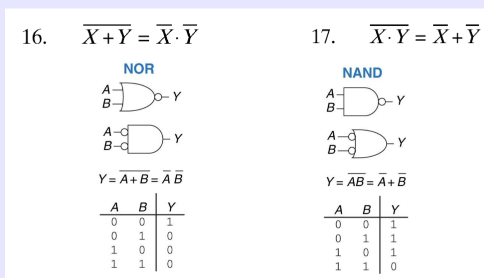
$$ABC + YZ = (ABC + Y)(ABC + Z)$$

Substitute X for ABC

$$X + YZ = (X+Y)(X+Z)$$

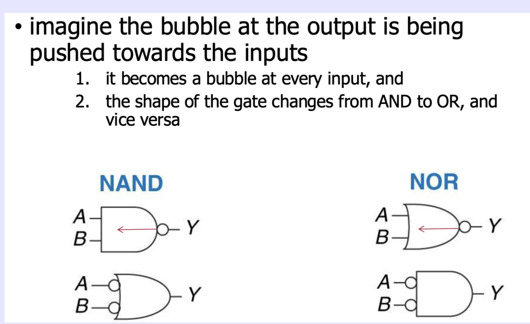
DeMorgan's Identity is used a lot and is very useful. As shown in these examples:

Example



Where the two equivalencies share a truth table due to this Identity

We can also see that it is like "pushing the bubble" as seen in this example:



1.3 Combinational Logic Circuits

Definition

Stateless Digital Logic Circuits:

- Combinational logic combination of logic gates
- Change input values
- Immediate change in output values
- No Memory
- No feedback

Remark 1. There are specifics types of wire connections

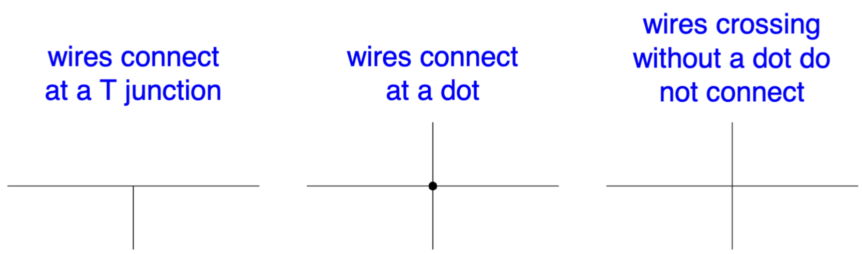
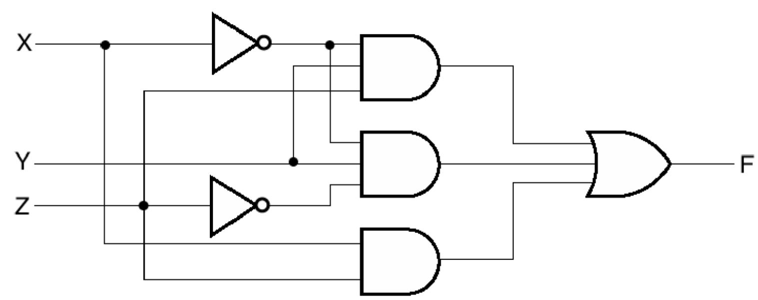


Figure 4: Here are the various ways wires can connect/not connect

Example

Here is an example of a circuit and the resulting algebra to "solve" it and how to simplify it

$$F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$$



$$F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$$

Apply $14. \quad X(Y+Z) = XY + XZ$

$$F = \overline{X}Y(Z + \overline{Z}) + XZ$$

Apply $7. \quad X + \overline{X} = 1$

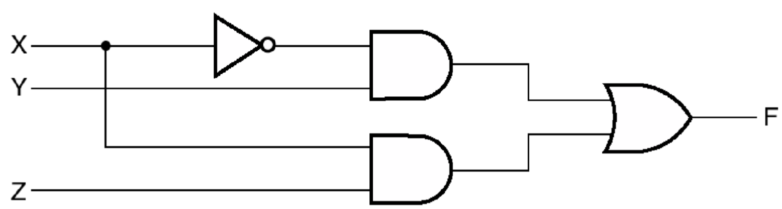
$$F = \overline{X}Y \cdot 1 + XZ$$

Apply $2. \quad X \cdot 1 = X$

$$F = \overline{X}Y + XZ$$

Fewer Gates

$$F = \overline{X}Y + XZ$$



(b) $F = \overline{X}Y + XZ$

Where output variables are either equivalent to 0 or 1 and input is the same. Moreover, simplifying this circuit and circuits in general allow for greater efficiency.

1.4 Standard Design Approach Sum of Products (SOP)

The three step apporach:

1. Define truth table
2. Write down a Boolean expression for every row with the '1' in the output, for example, $Y = \overline{C}\overline{B}A + \overline{C}BA + CB\overline{A} + CBA$
3. Wire up all of the gates

Truth Table

C	B	A	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Figure 5: Here is the truth table given the example

1.5 Karnaugh Map

Definition

Karnaugh maps, aka k-maps, are graphical representations of truth tables that use a grid with one cell for each row of the truth table

C \ BA	00	01	11	10
0	0	1	1	0
1	0	0	1	1

C	B	A	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Figure 6: An example k-map and its respective truth table!

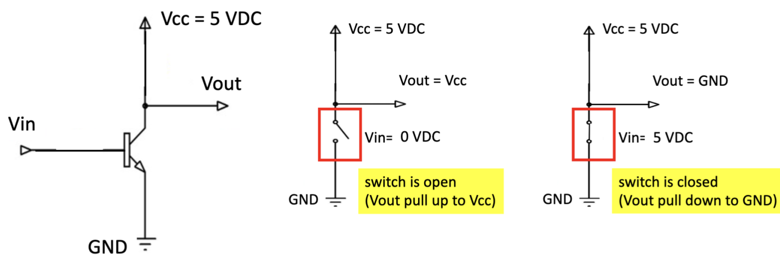
You pretty much put the 1's and 0's onto the cell given the values

Here are some rules given to the k-map

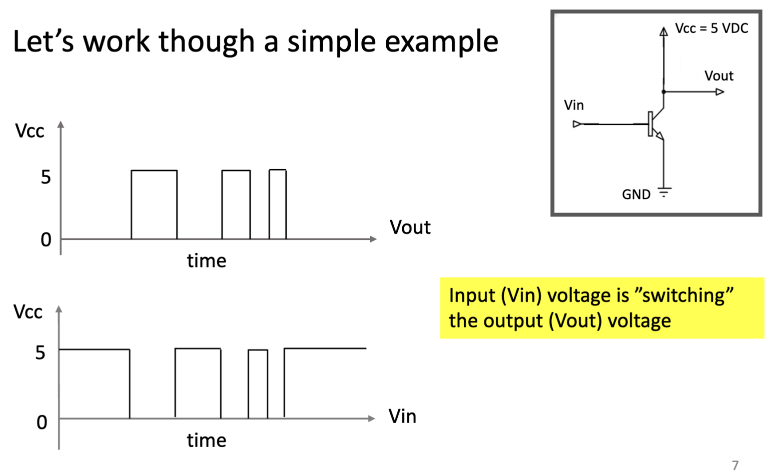
1. The groupoing must be in the shape of a rectangle. There are no diagonal adjacencies allowed
2. All cells in the rectangle must contain ones. No zeros are allowed
3. The number of cells in groupings must be in powers of 2
4. Outside edges of K-maps are considered adjacent, so it may wrap around
5. Cells may be contained in more than one rectangle, but every rectangle must have at least ONE unique cell to
6. Every rectangle must be as large as possible
7. Everyone 1 must be covered by at least one rectangle

2 CMOS Gate Design and Analysis 1/27

The basic design of a transistor is as follows:

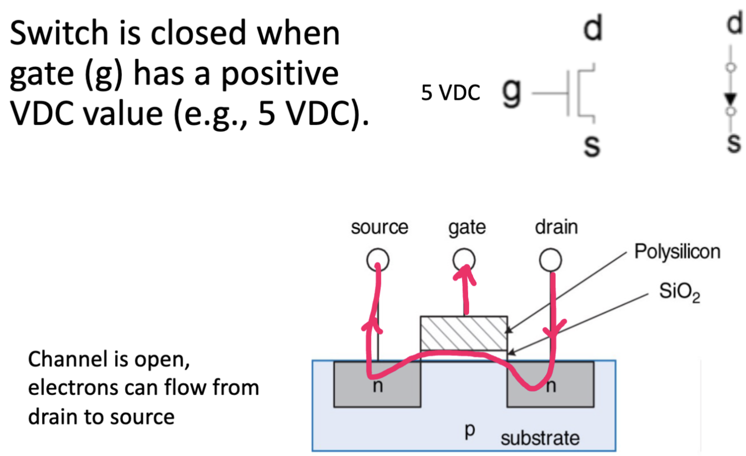


Basic operations:

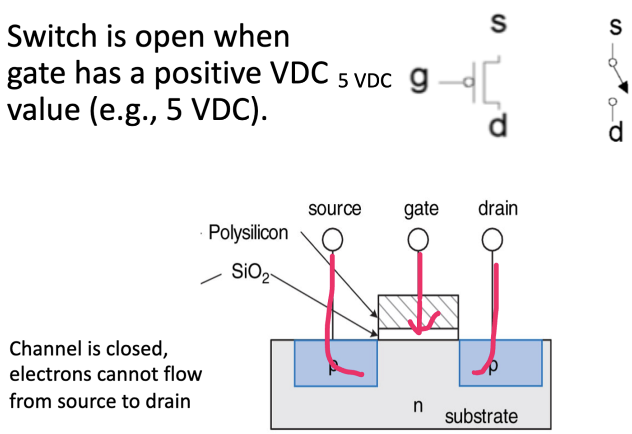


2.1 Metal Oxide Semiconductor (MOS) Transistor

nMOS The n-channel Metal Oxide Semiconductor (nMOS) transistor

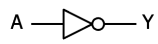


pMOS The p-channel Metal Oxide Semiconductor (pMOS) Transistor
It is similar to a dam, where the analogy states, there is a lot of "water" on one side and then directly flows down depending on amount of "water"



NOT Gate MOS Gate Design A strong 5V and strong no 5V

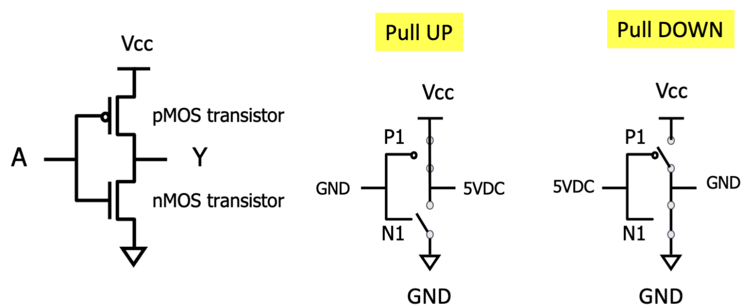
Schematic Diagram



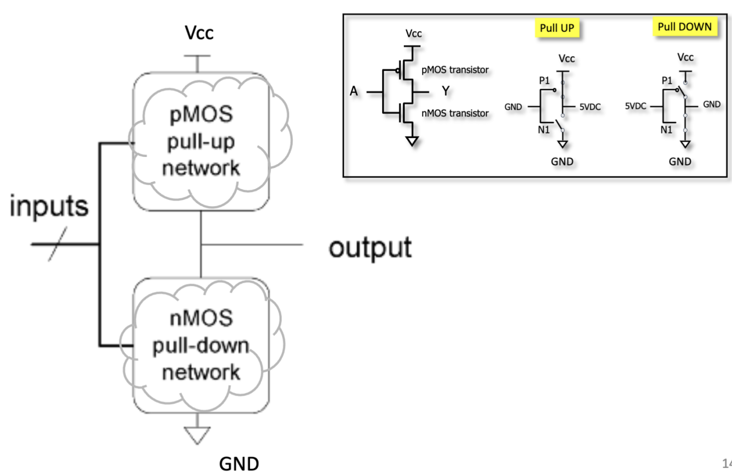
Truth Table

Vcc = 5 VDC

A	P1	N1	Y
GND	5 VDC	GND	5 VDC
5 VDC	GND	5 VDC	GND



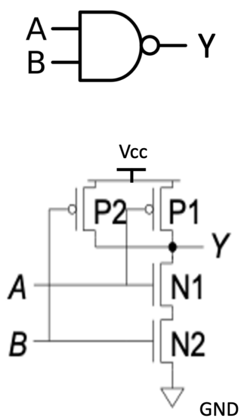
Complementary MOS Designs here



3 CMOS oscillator (clock) properties and design 1/29

This lecture, we are finishing up the remaining lecture from last time...

NAND Gate: MOS Design Only needs one or NONE of them on



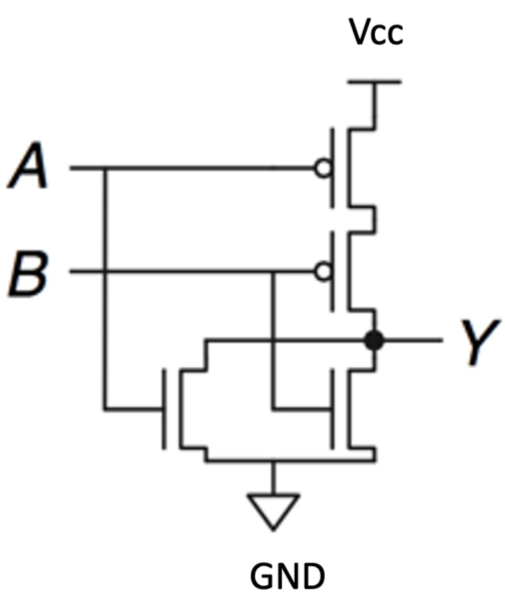
A	B	P1	P2	N1	N2	Y
GND	GND	5VDC	5VDC	GND	GND	
GND	5VDC	5VDC	GND	GND	5VDC	
5VDC	GND	GND	5VDC	5VDC	GND	
5VDC	5VDC	GND	GND	5VDC	5VDC	

Vcc = 5 VDC

15

For the first three, the output should be 5V, while the last output should be 0

NOR Gate: MOS Design Opposite of OR, where only 5V output when all off.



3.1 Relationship between Voltage and Logic

Digital Abstraction

Definition

Voltage a continuous value

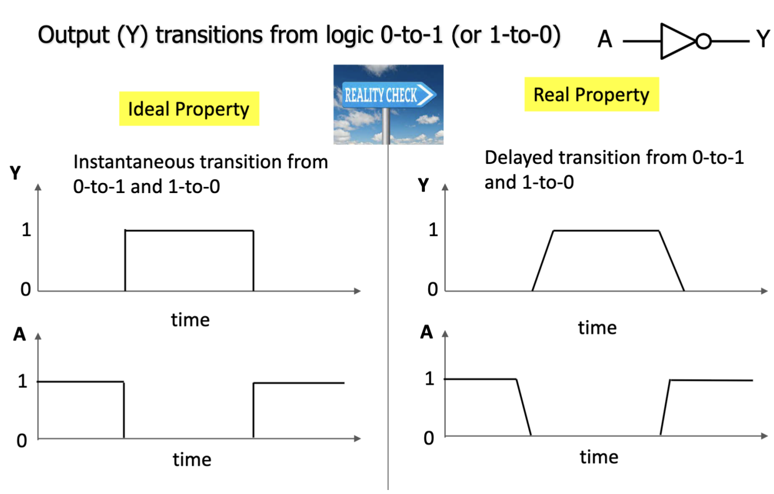
- Has a defined range of values, e.g. from 0 to 5VDC
- And any VDC value between, e.g., 0.1, 0.11, etc…
- Hardware understands voltage values

Boolean Logic (Logic) is a discreate value of 0 or 1

- Abstractions that humans understands
- Apply the rules of Boolean algebra
- Simplifies circuits

Continuous to Discrete conversion can be defined as having:

- Logic 1 - Has voltage range from 5 to 2 VDC
- Logic 0 - Has voltage range from 0 to 0.8 VDC
- Invalid - Less than 2 VDC, greater than 0.8 VDC, unreliable measurements

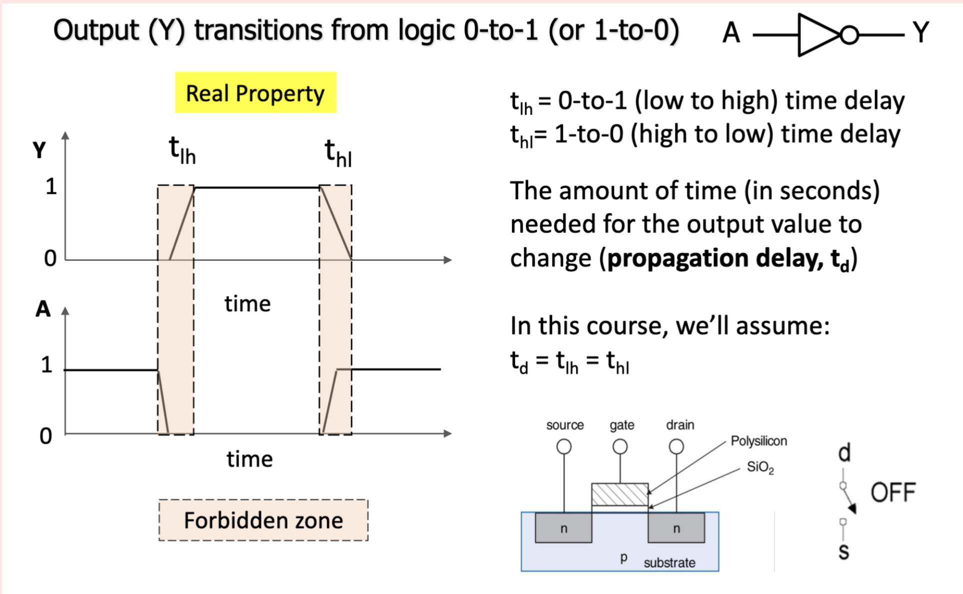


NOT Gate: Closer Inspection It will take time to transition from 0 to 1, and vice versa

This moves onto to our definition of *Gate Delay*

Definition

Gate Delay is defined as the transition from logic 0-to-1 and vice versa



Moving onto 1/29's actual lecture:

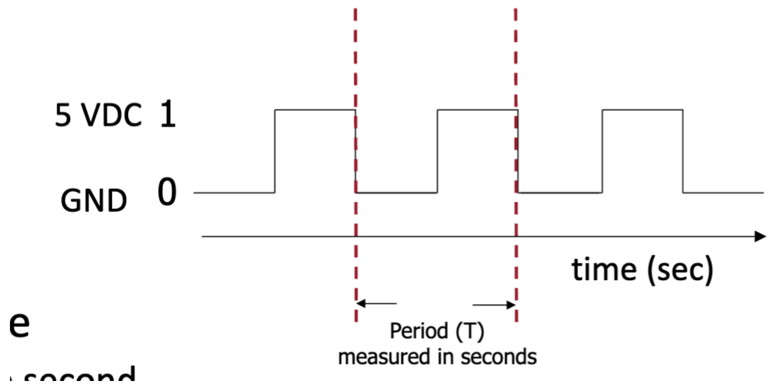
3.2 Clock and Clock design

Properties: Period and Frequency

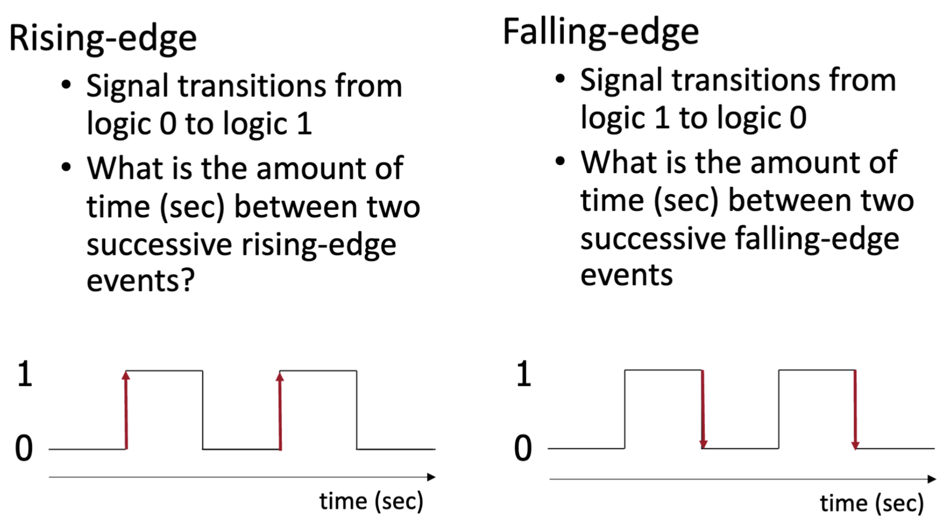
- Clock period (T)
  - One **COMPLETE** cycle
  - Typical period of 1ns
  - Measured in **seconds**
- Clock Frequency (F) or rate
  - How many cycles in 1 seconds
  - Frequency $= F = \frac{1}{T}$

– Measured in **Hertz**

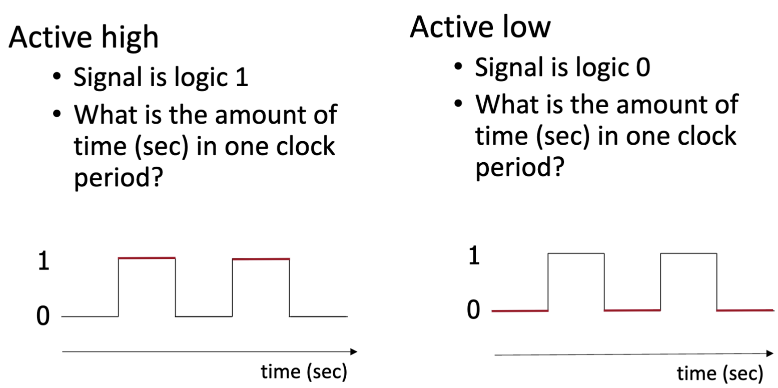
Shown below is what these ways would look like:



Properties: Events What are the different edges?



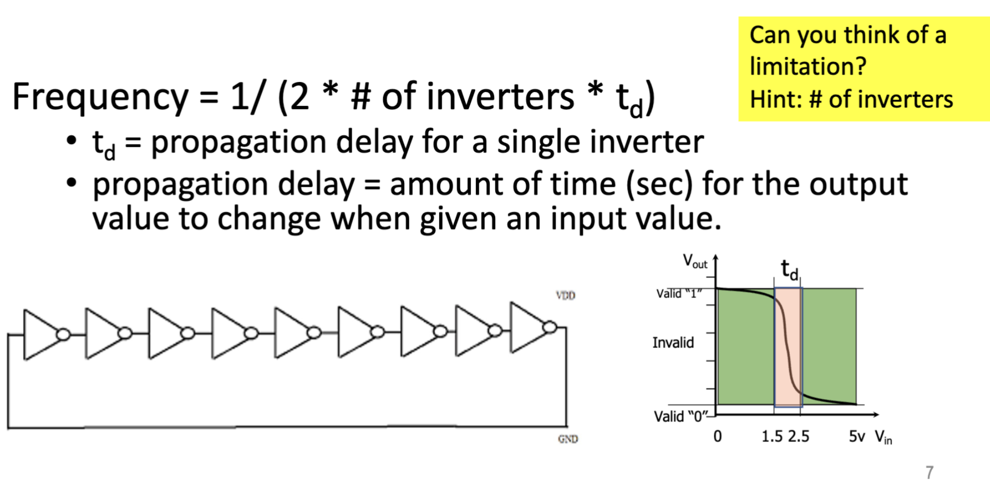
Properties: Active High and Low Now lets look at the highs and lows



3.3 Clock design

Ring Oscillator clock that oscillates using inverter logic gates

> Definition
>
> **Clock Period** is the propagation delay for a sequence of inverters



Quartz Crystal Used in modern processor

Depending on the crystal's physical thickness and size, it can control:

- Frequency of oscillations
- Inversely proportional to its physical thickness between 2 metallic surfaces