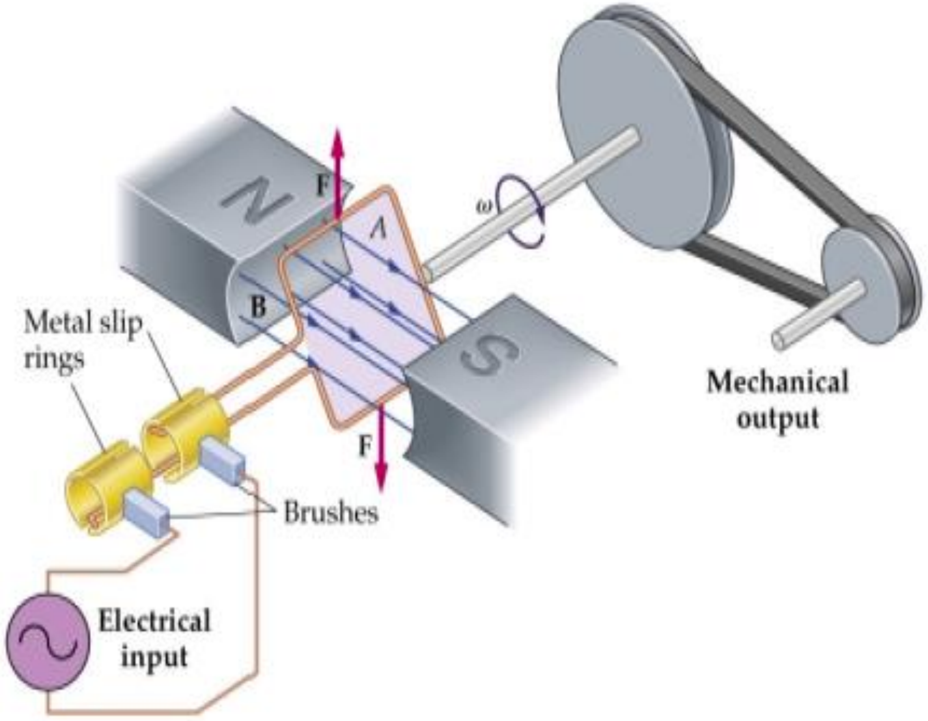
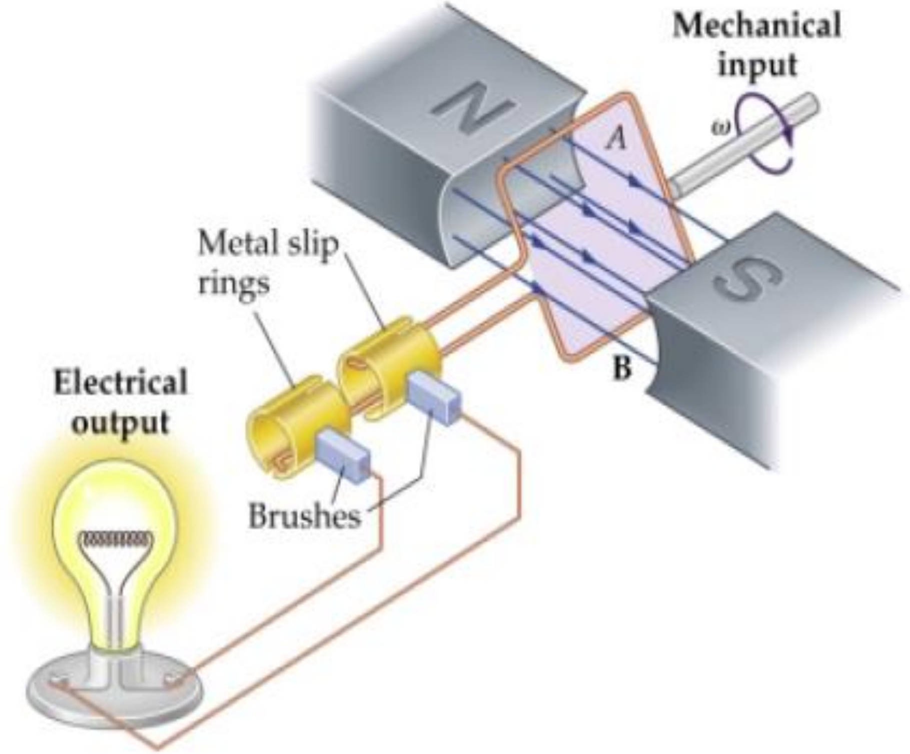
ECOR1044: Boolean Logic

Motors and Drives
Unsigned Binary
Boolean Logic Gates
Algebra

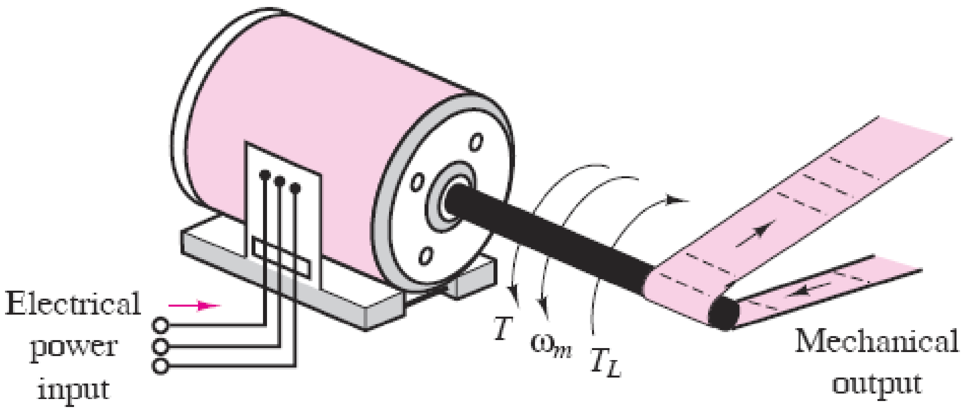
Motor/Generator

A motor/generator consists of a rotor spinning in a magnetic field produced by permanent magnets or by coils.

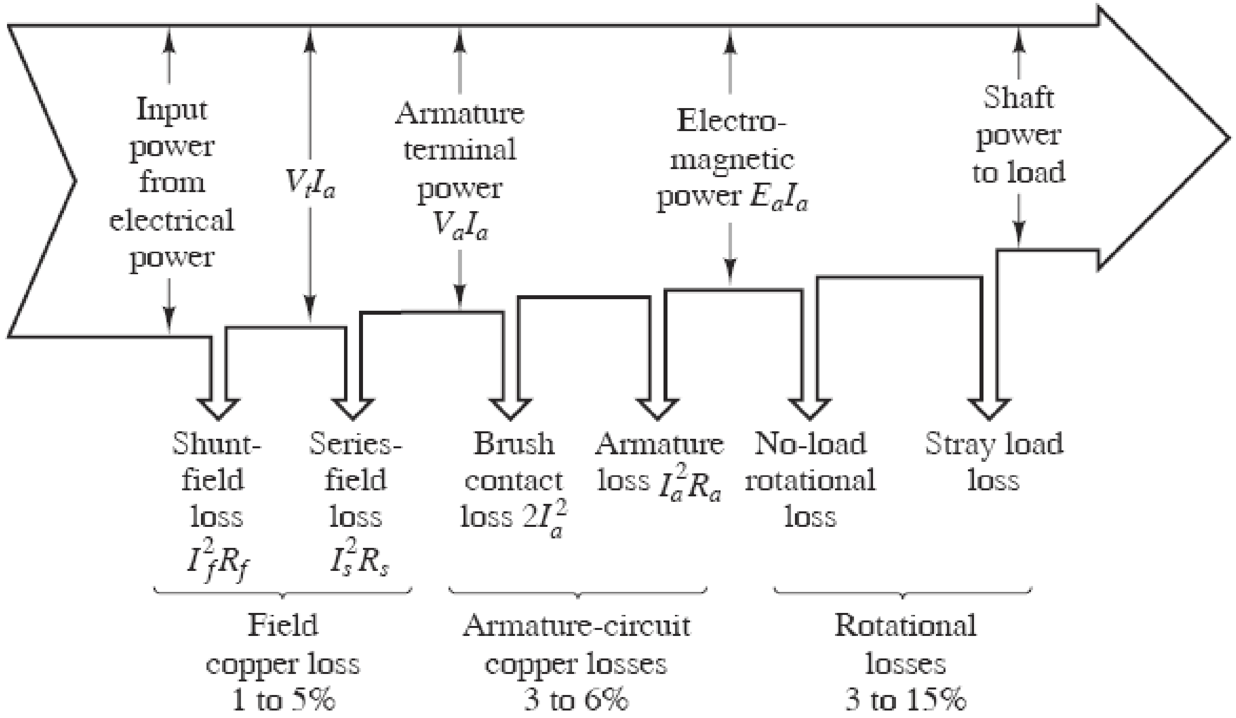


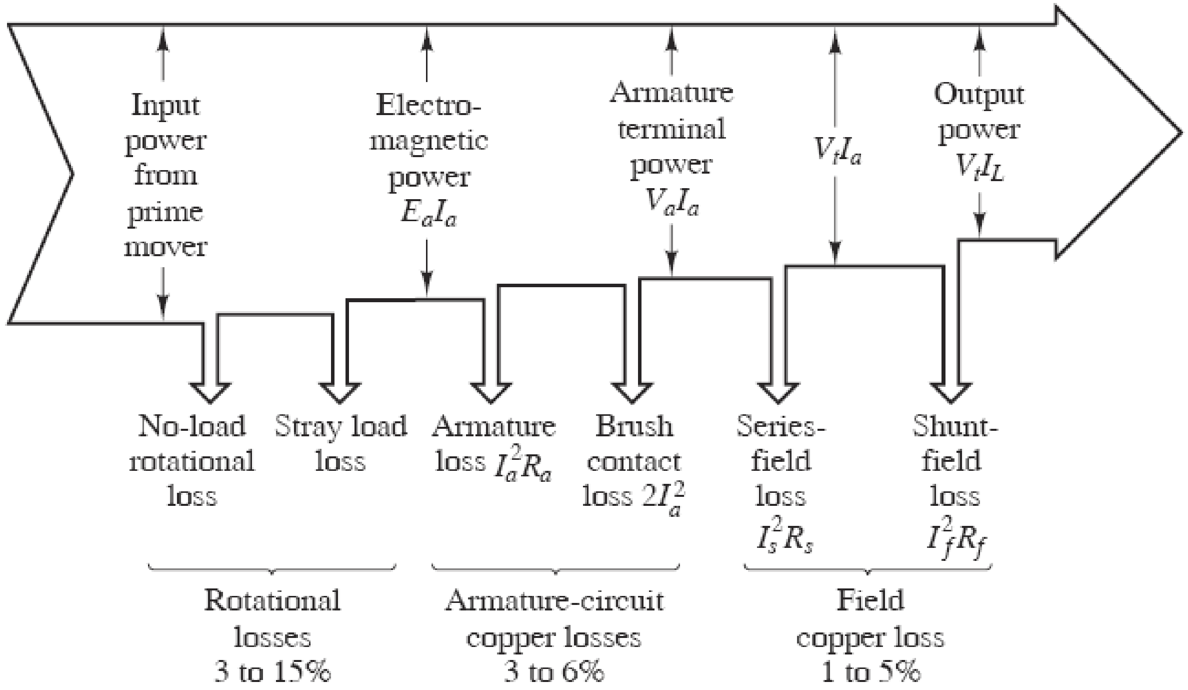


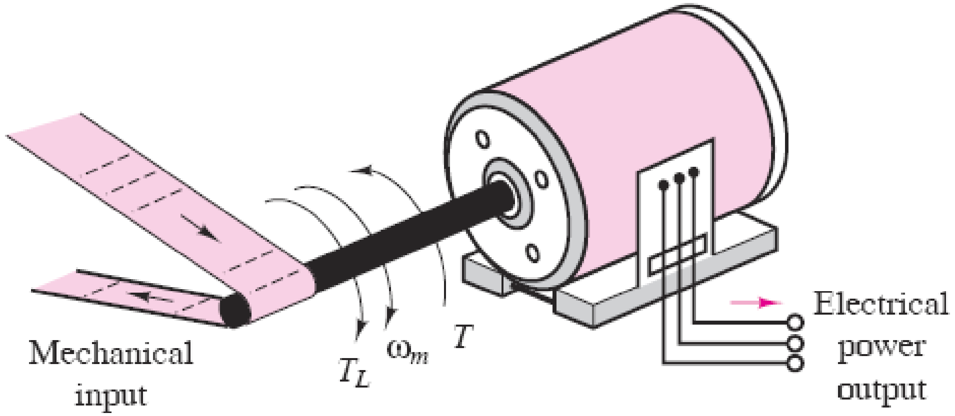
The process of generating a magnetic field by means of an electric current is called ***excitation***.



(a) Motor action

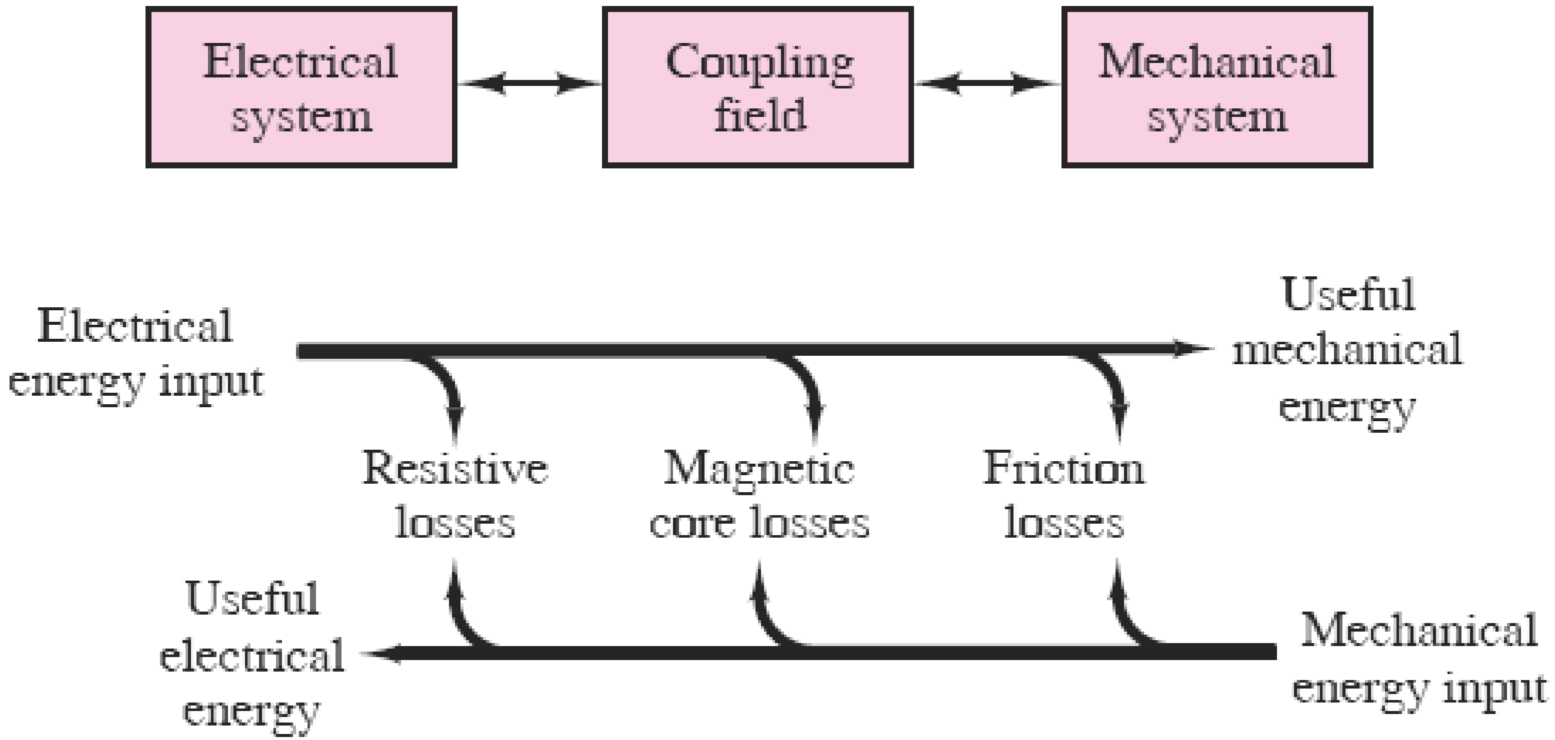






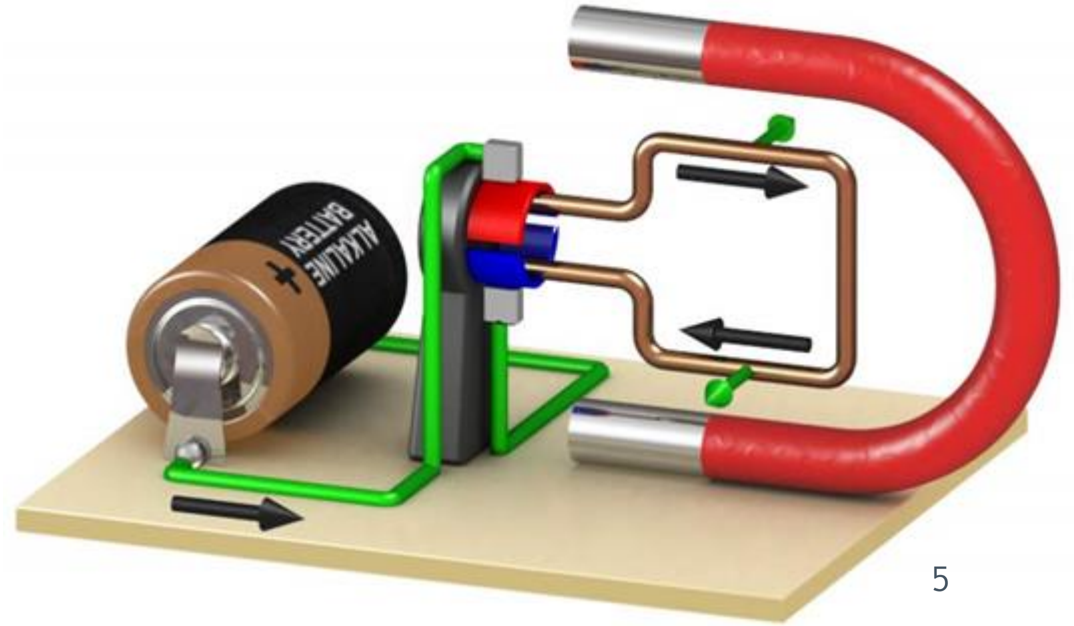
(b) Generator action

Motor/Generator



DC Motor Principle

- When a current flows, a torque is generated producing an angular acceleration.
- When μ and B are aligned, the torque is zero, but the loop has angular velocity and angular momentum. It therefore overshoots the aligned position.
- A split ring causes the current to reverse direction, just as it passes the aligned position.
- This accelerates the loop again through another 180 degree turn.

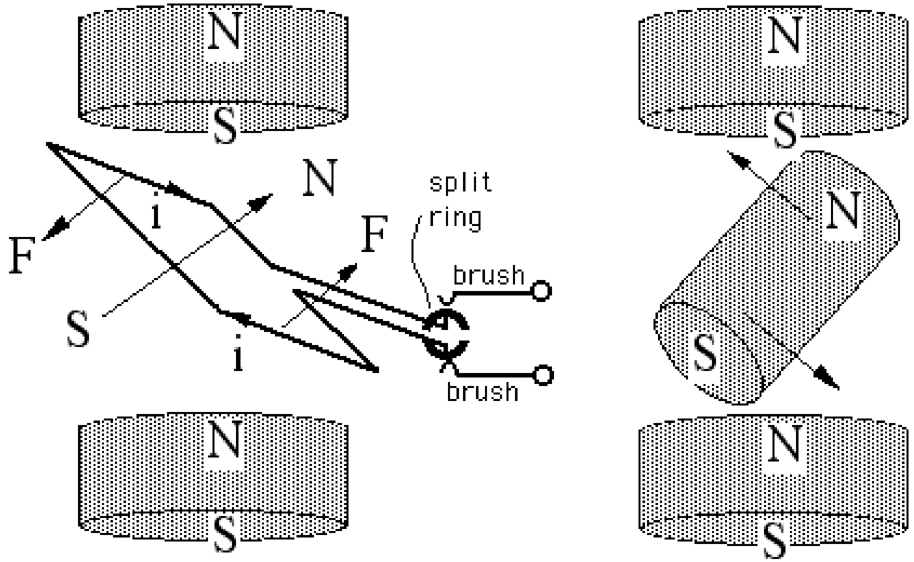


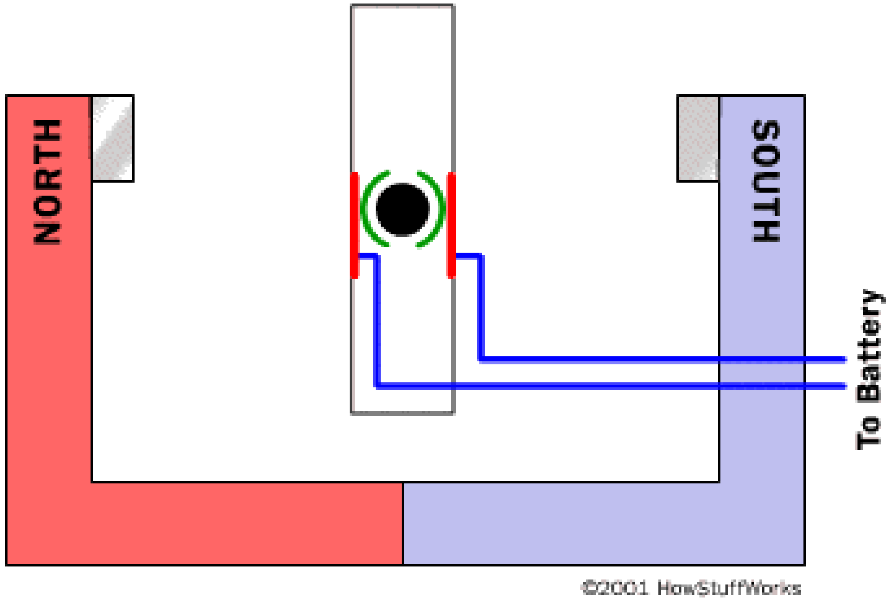
https://www.youtube.com/watch?v=LAtPHANEfQo

DC Motors

› The interaction of the fields produces the movement of the shaft/armature.

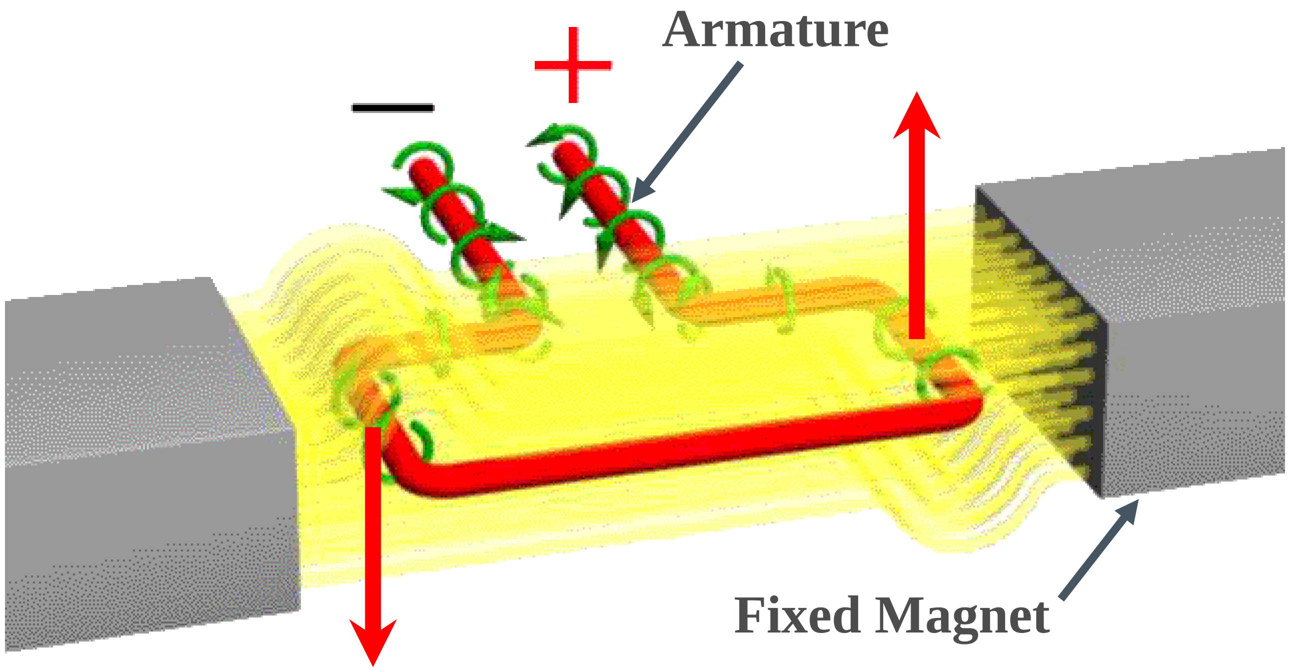
› The "flipping the electric field" part of an electric motor is accomplished by two parts: the **commutator** and the **brushes**.

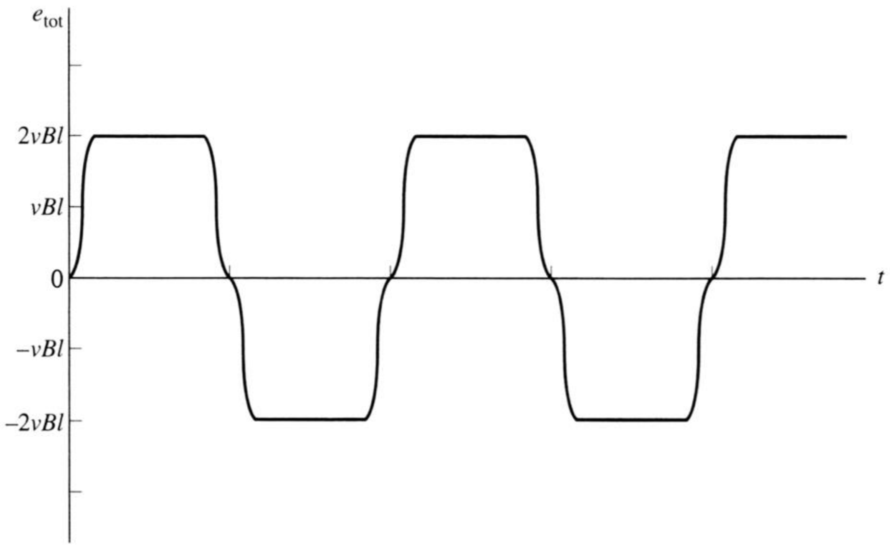




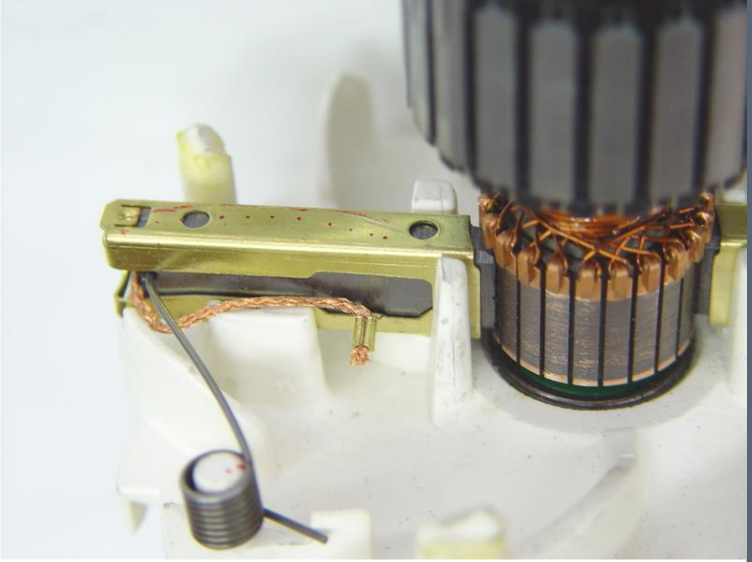
DC Motors

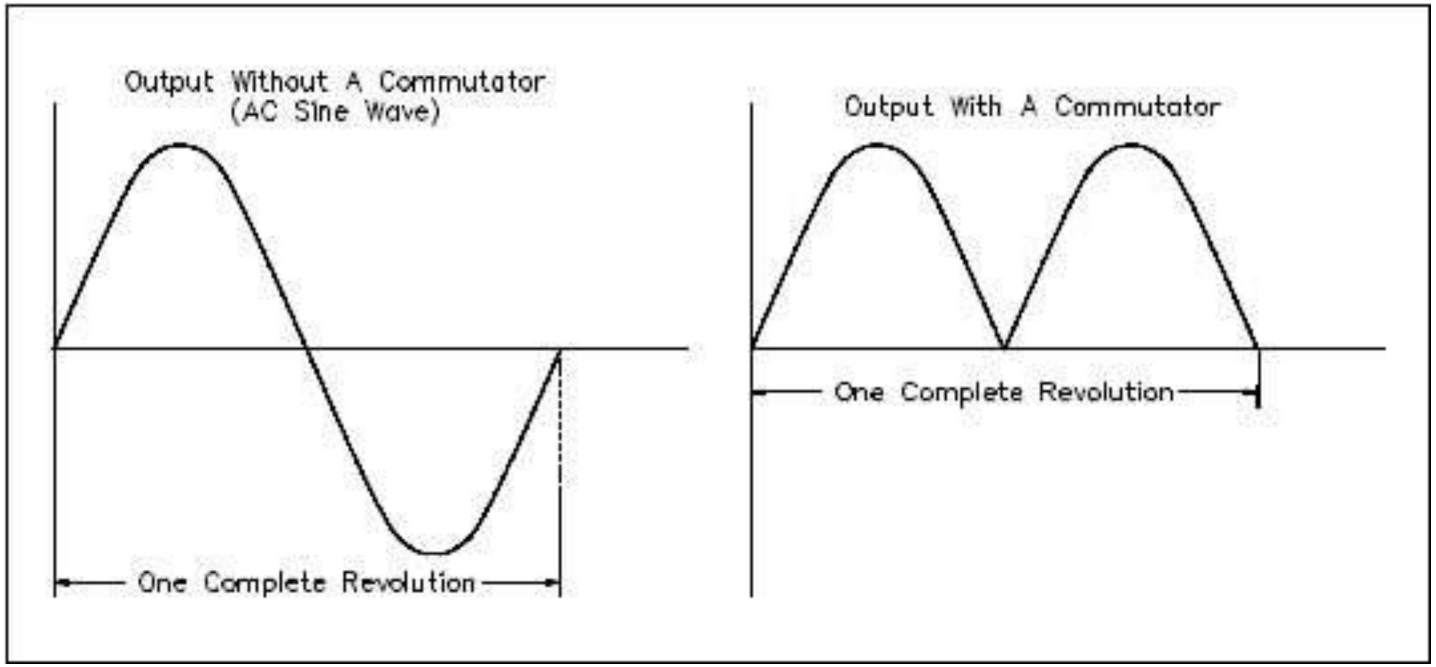
- The interaction of the fields produces the movement of the shaft/armature.
- When the loop rotates through 180°, the direction of the voltage reverses leading to the following total induced voltage:

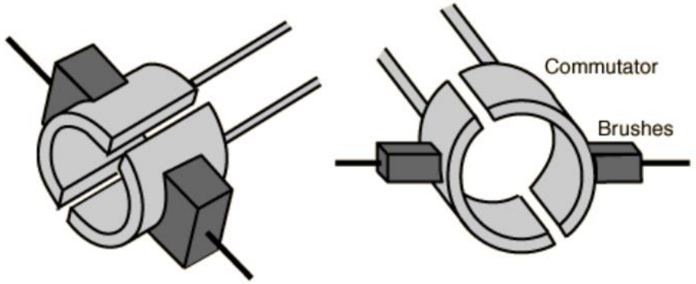




Commutator




Output Without A Commutator
(AC Sine Wave)
One Complete Revolution
Output With A Commutator
One Complete Revolution


Commutator
Brushes

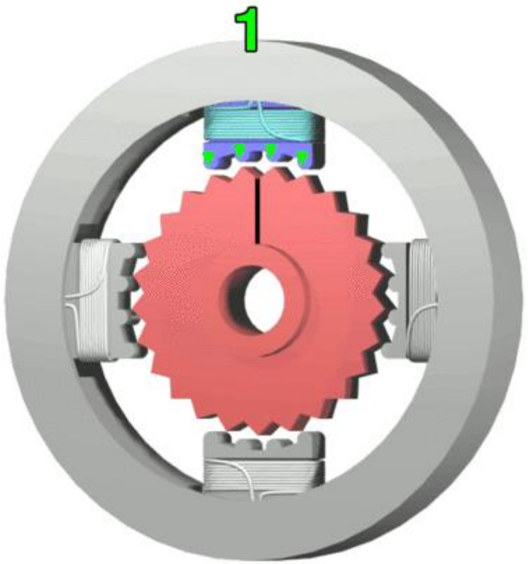
Motor Windings

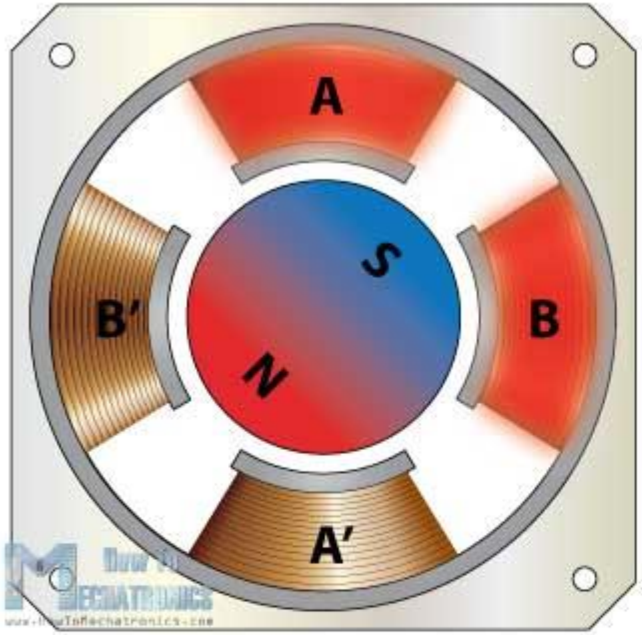
- To avoid irregular motion and for a smooth operation, several winding loops are used.

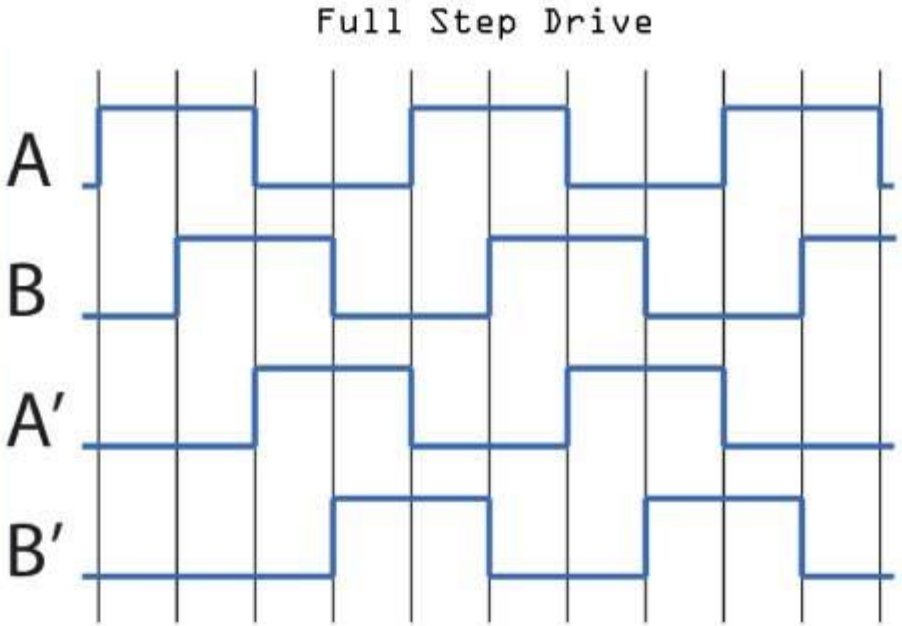
https://www.youtube.com/watch?v=LAtPHANEfQo

Stepper Motor

- No need of a feedback loop.
- Half stepping operation.

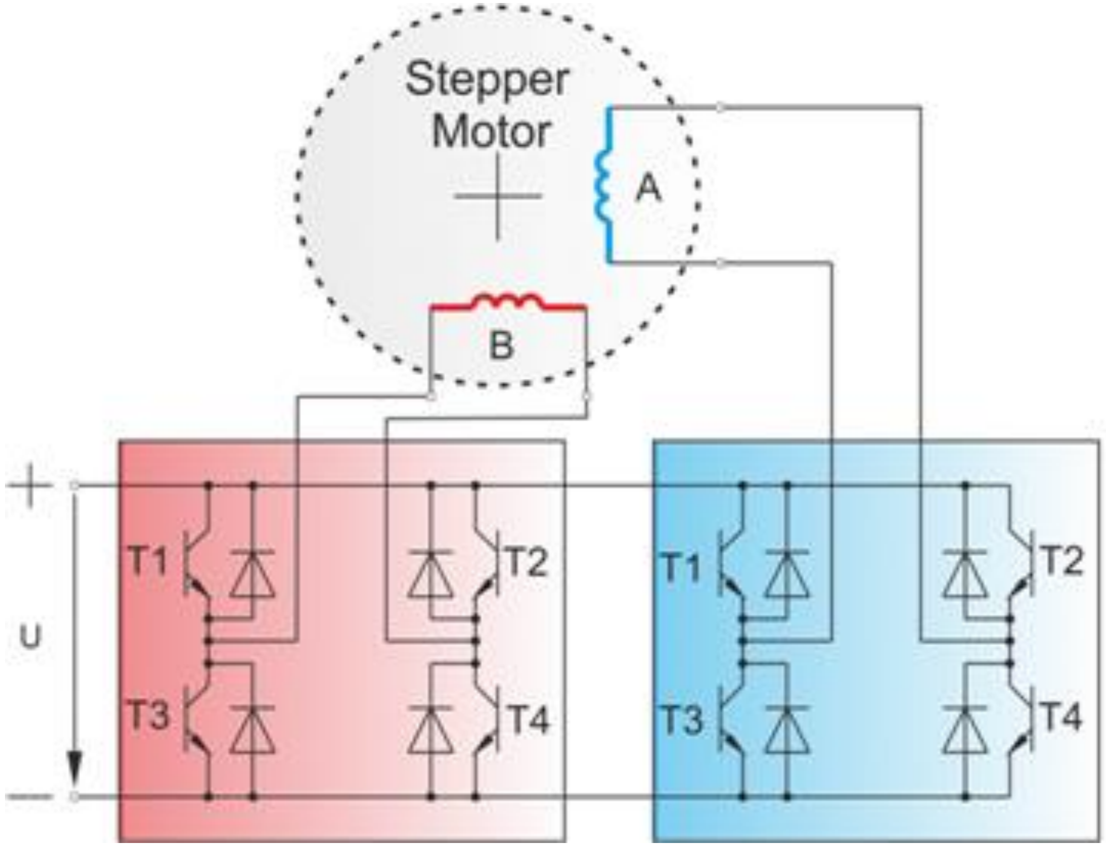


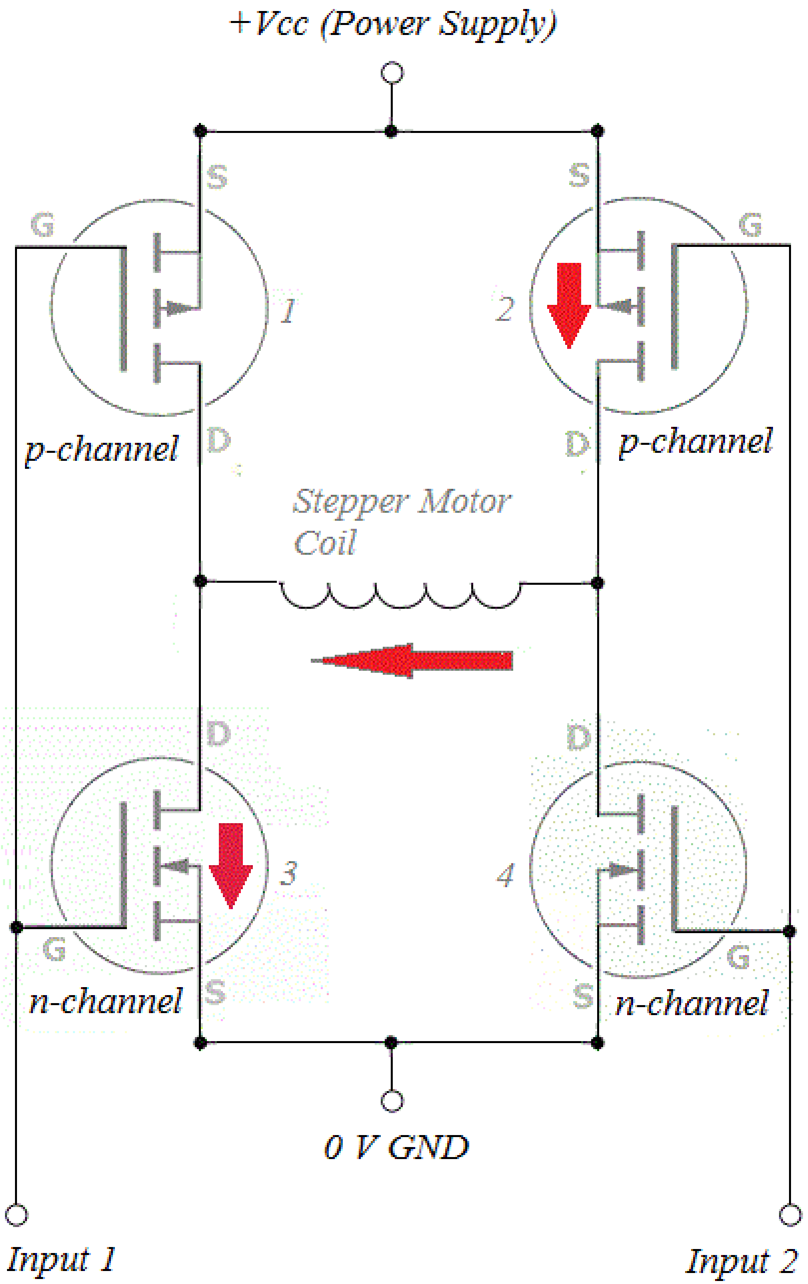




Stepper Motor

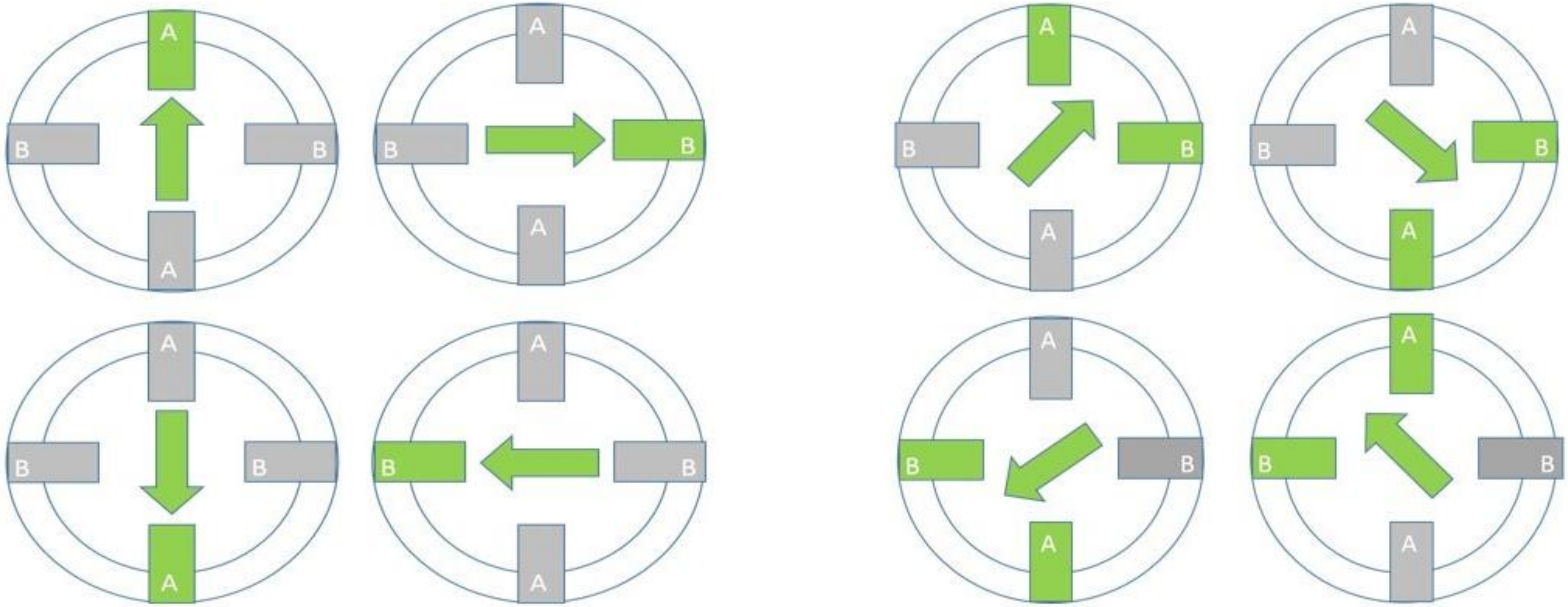
› Control Technique:





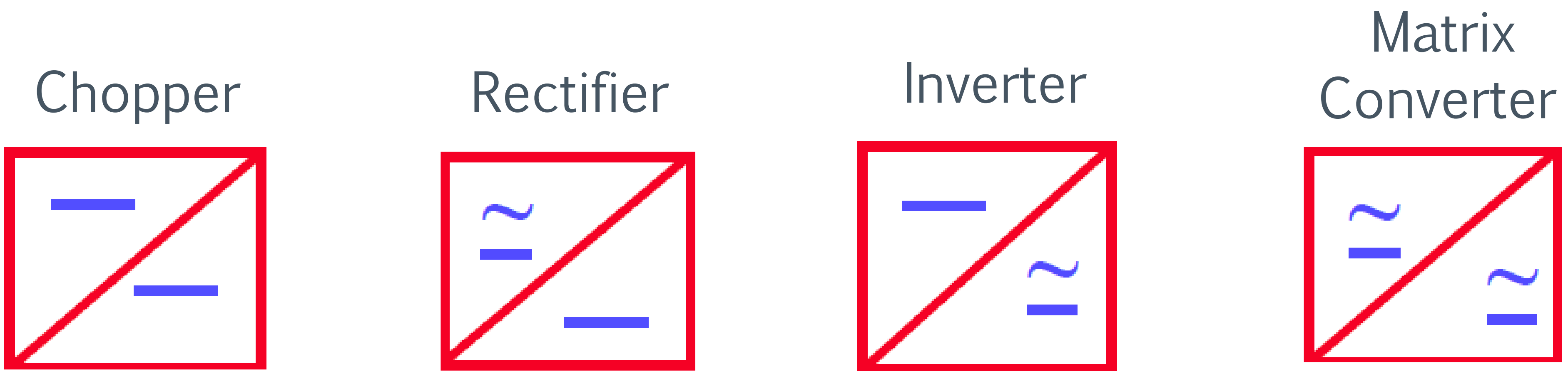
Stepper Motor

› Half stepping operation:



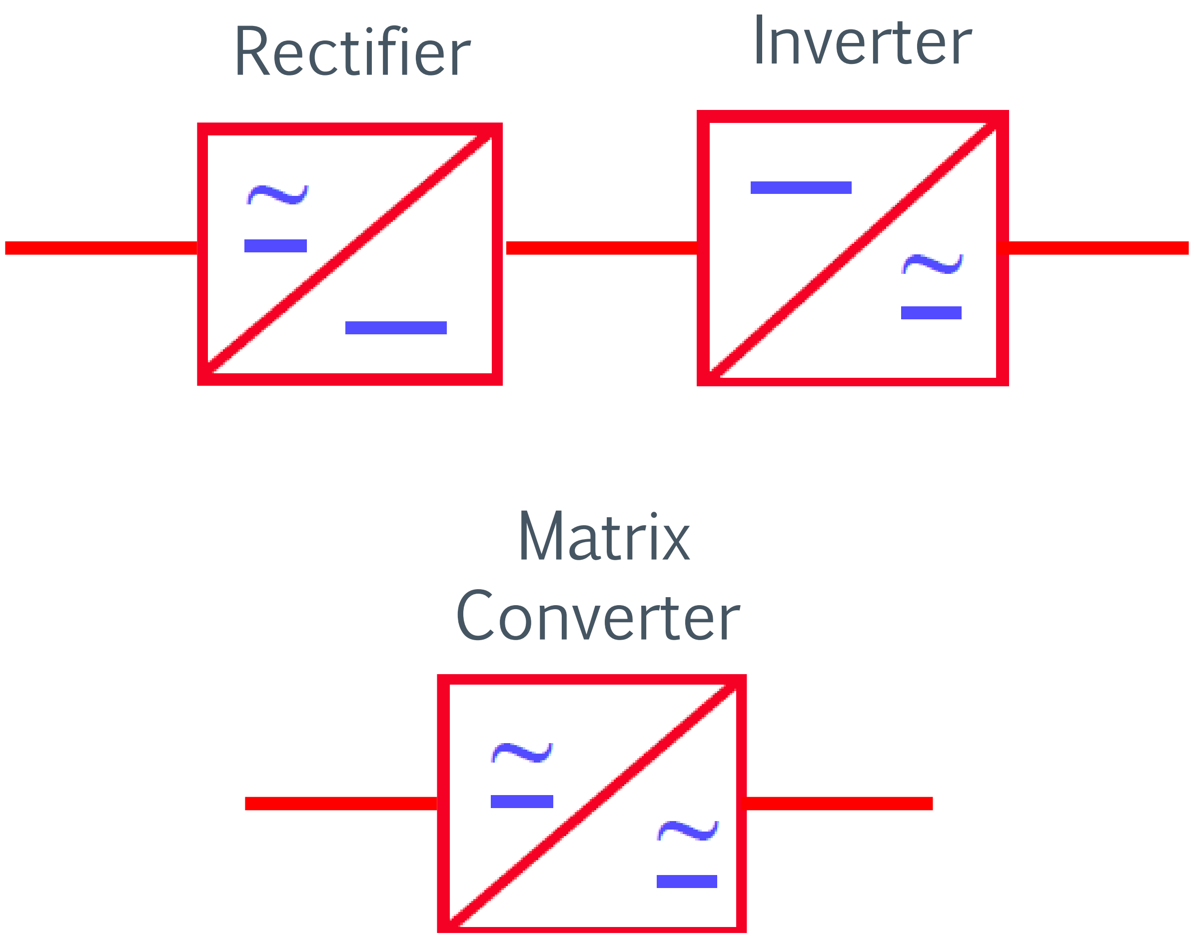
https://www.youtube.com/watch?v=eyqwLiowZiU

Power Converter Topologies

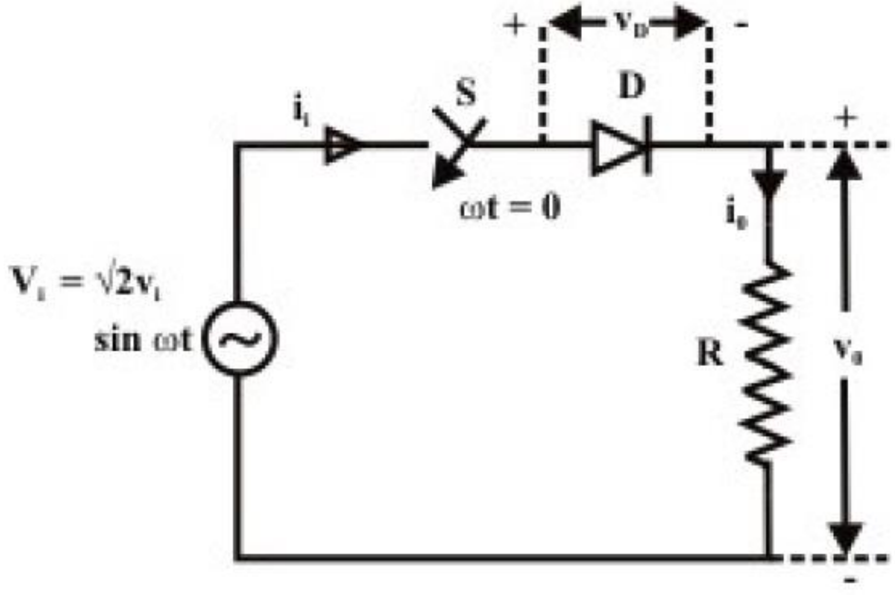


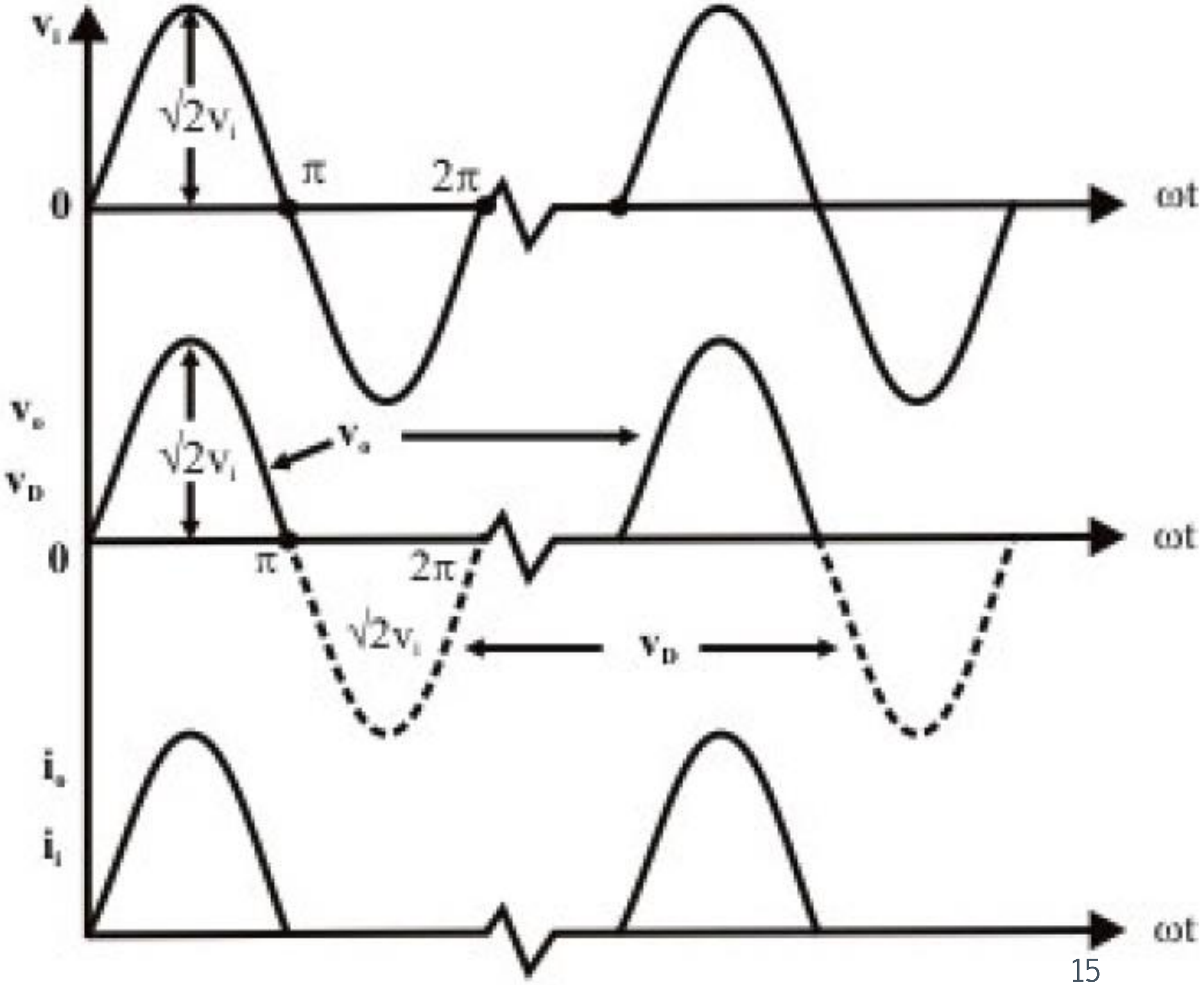
- Chopper: DC-DC converter applications – Buck, Boost, etc.
- Rectifier: AC-DC converter.
- Inverter: DC-AC converter.
- Matrix converter (cyclo-converter): AC-AC converter.

Electric Motor Drive

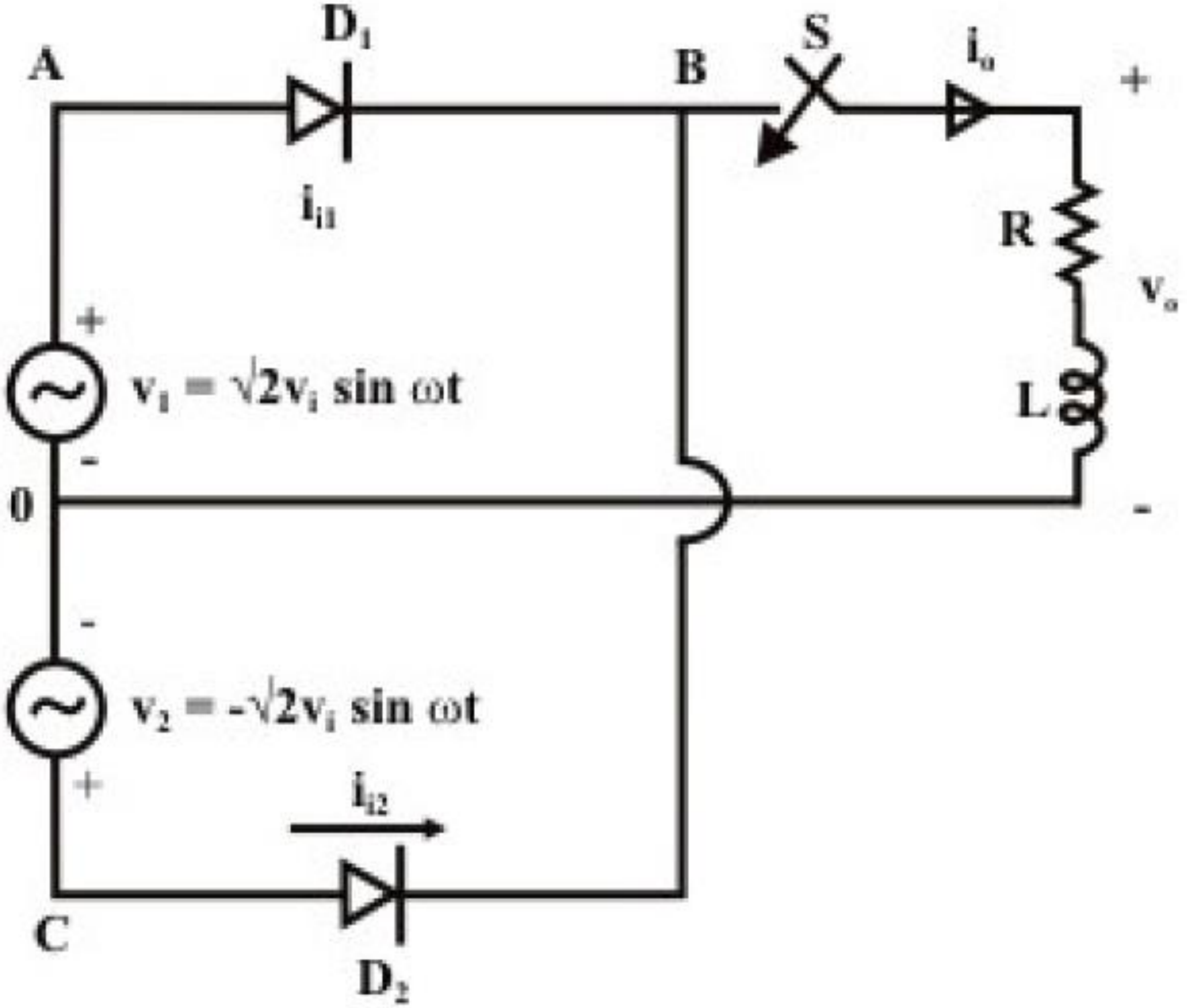


Single Phase Uncontrolled Half Wave Rectifier

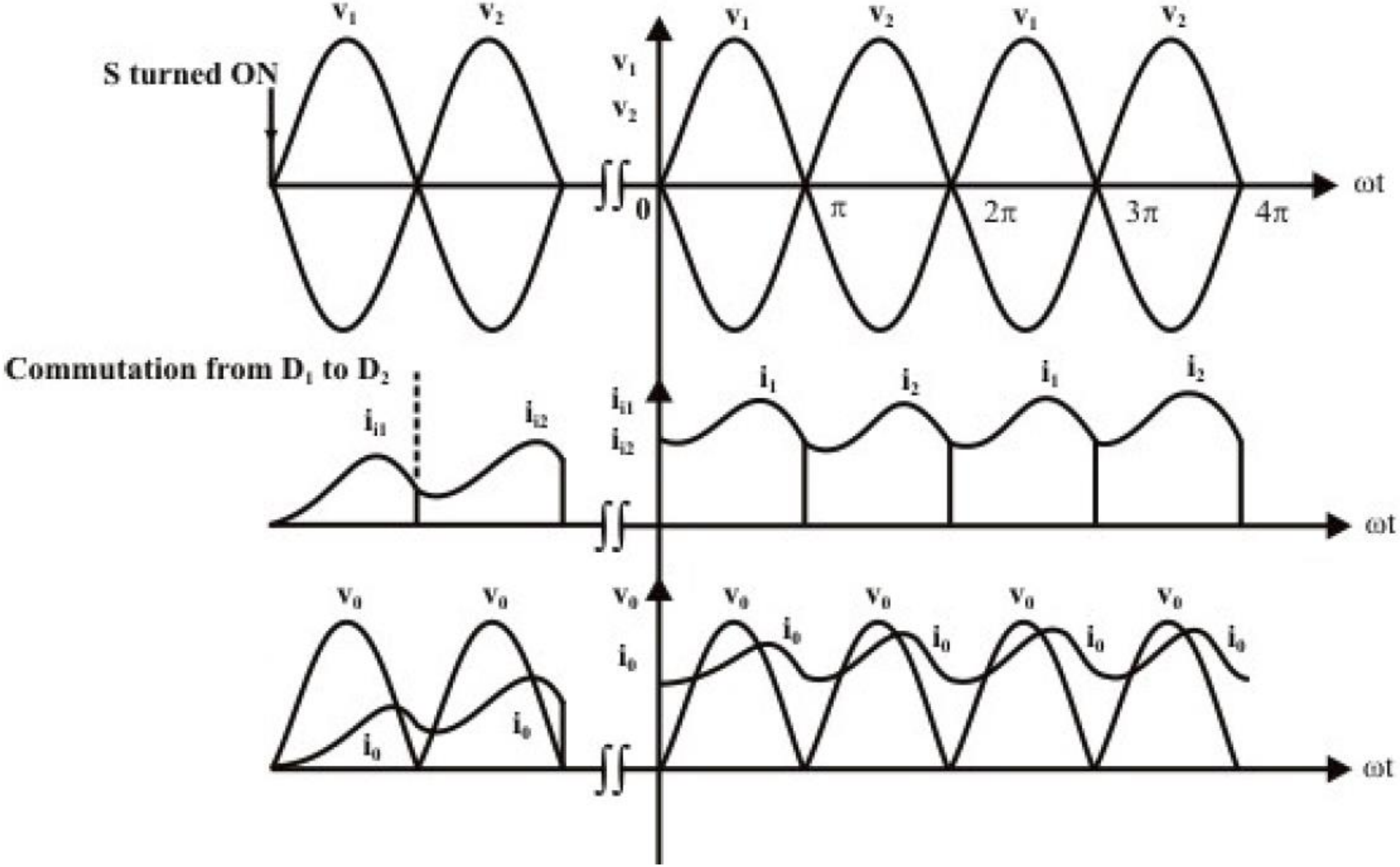




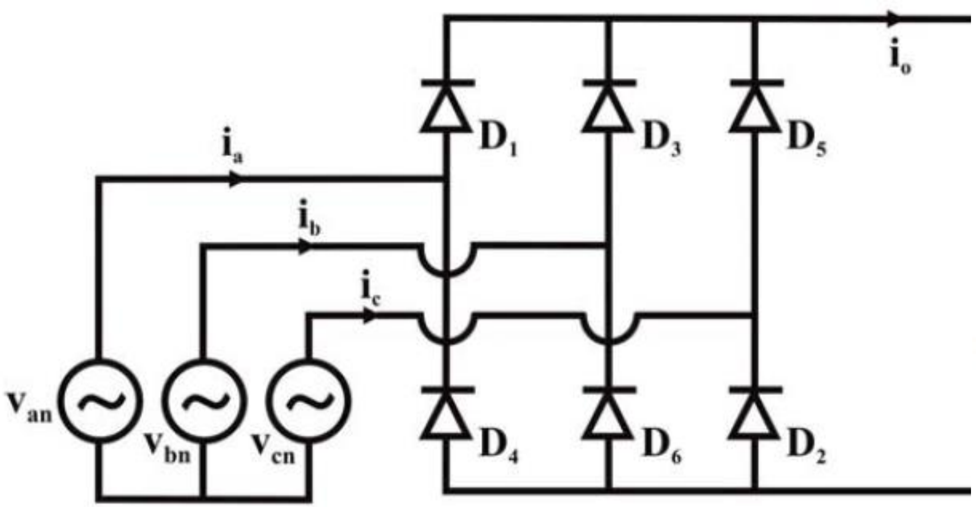
Single Phase Uncontrolled Full Wave Rectifier

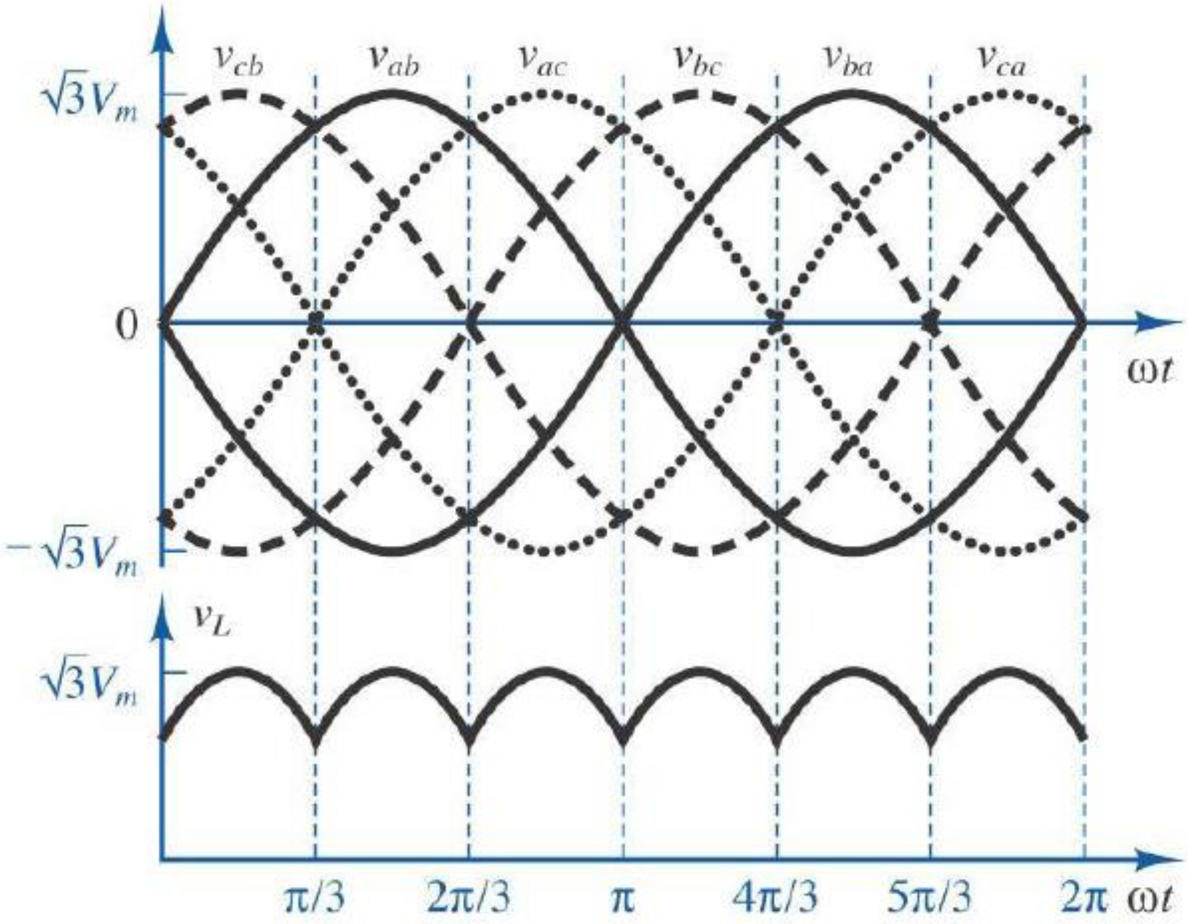


Single Phase Uncontrolled Full Wave Rectifier

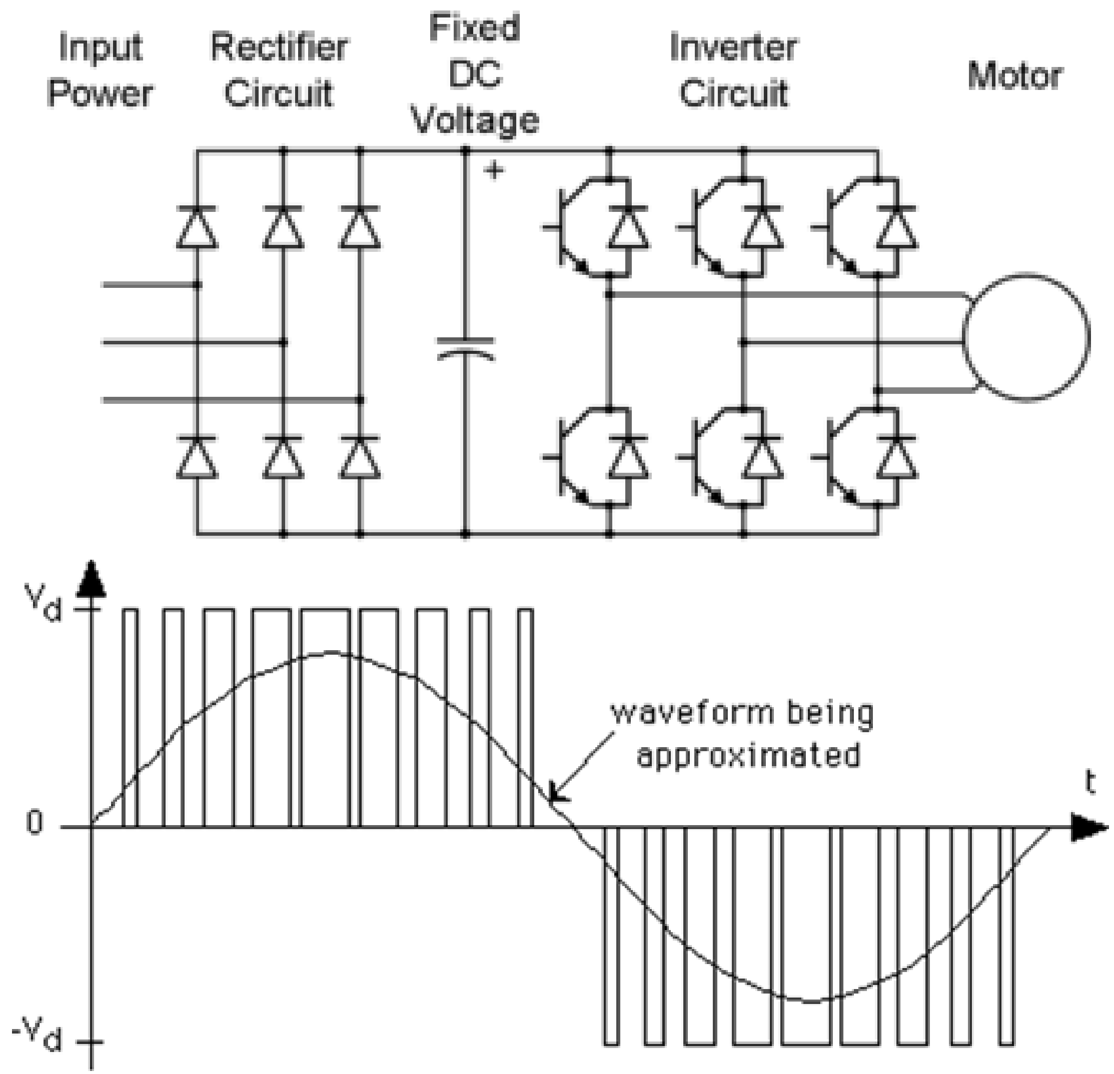


Three Phase Uncontrolled Full Wave Rectifier





Motor Drive

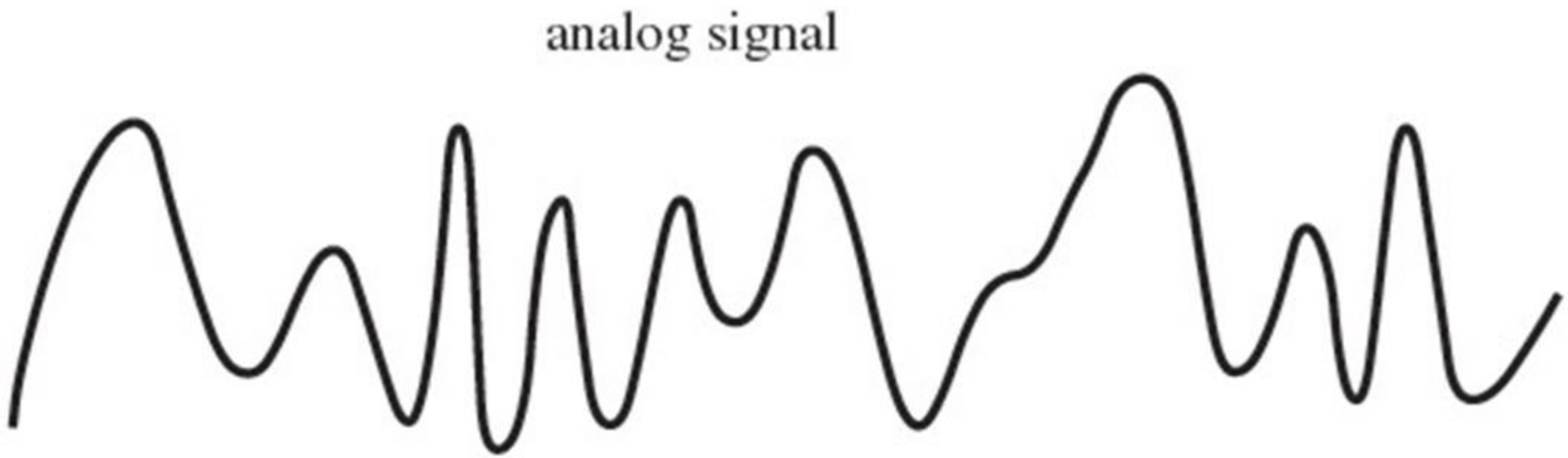


Input Power
Rectifier Circuit
Fixed DC Voltage
+
Inverter Circuit
Motor
V_d
0
$-V_d$
t
waveform being approximated

Self-Commissioning & Auto-Diagnostic Drives

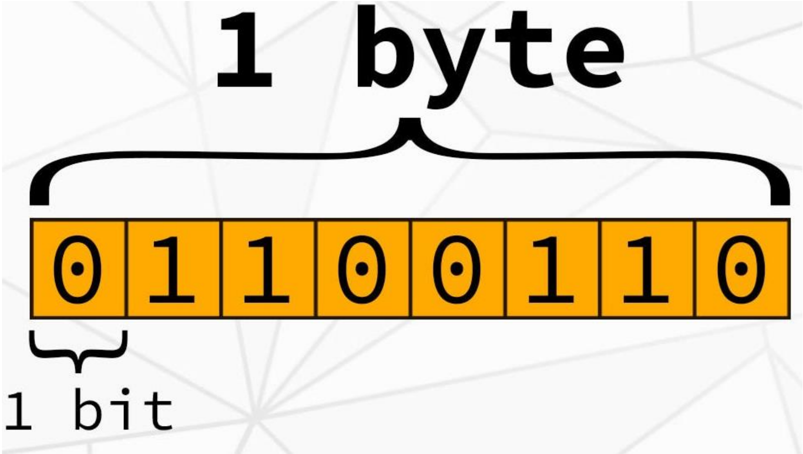
What is Analog and Digital

- **Analog:**
  - Most of the world that we experience is Analog:
    - There are an **infinite** number of ‘increments’
    - There are an **infinite** number of colors to paint an object
    - There are an **infinite** number of sounds levels



What is Analog and Digital

- **Digital:**
  - **Digital** signals and objects deal in the realm of the **discrete** or **finite** things.
    - This means there is a **limited set of values** through which they can be represented.
    - This could mean 2 possible values, 255, or anything ridiculously large, provided it's not infinity.



Challenges of Analog Systems

- **Noise and Interference**

✓ Analog signals are exposed to **noise and signal degradation**, especially over long distances or in electrically noisy environments. This is a major limitation compared to digital systems.

- **Precision**

✓ Analog signals are continuous, the precision is often limited by noise and the quality of the components.

✓ Small variations in signal can lead to **inaccuracies** in measurement or control.

Challenges of Digital Systems

- **Latency**

✓ **Delays** in real-time applications due to signal conversion and processing.

- **Power Consumption**

✓ It needs **more energy** due to increased processing and data handling.

- **Complexity**

✓ In design, **debugging**, and synchronization.

Analog or Digital?

analog signal

digital signal

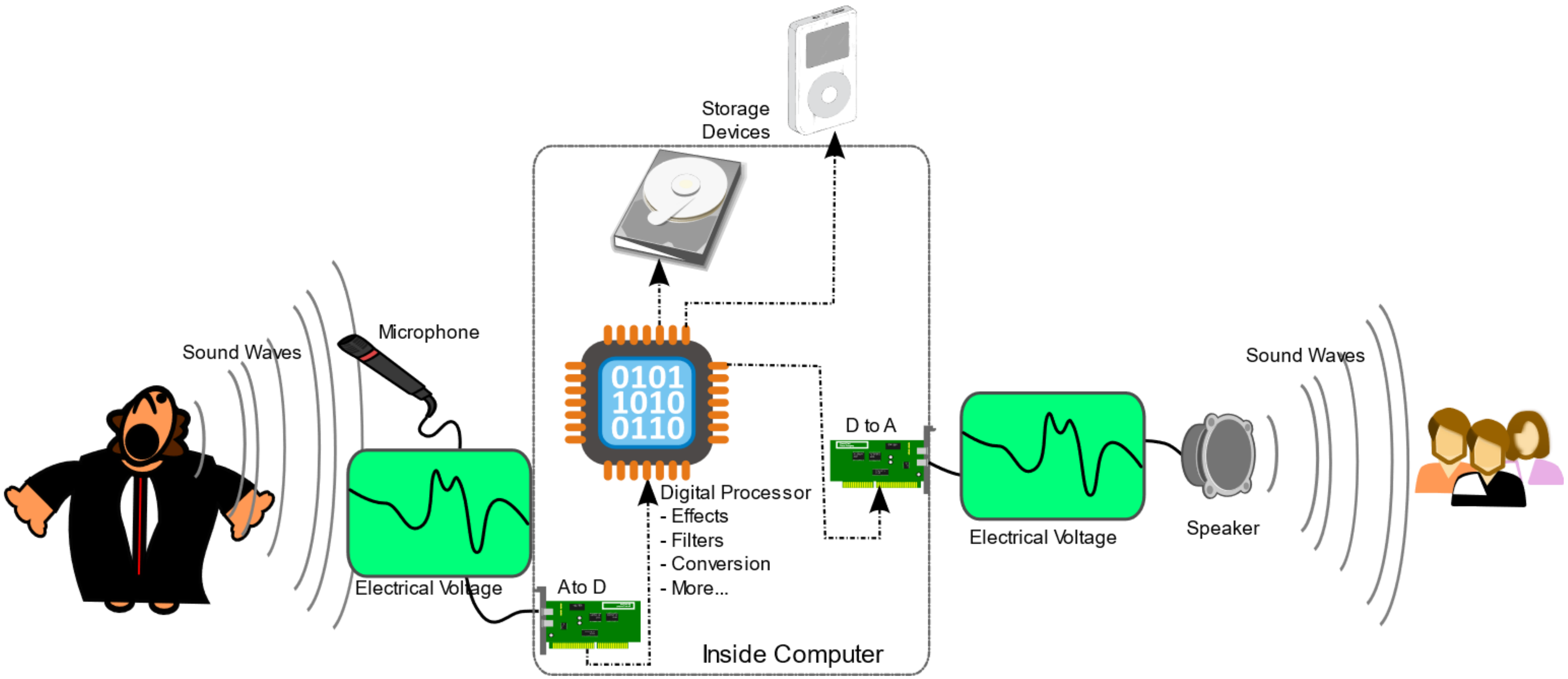
Analog

Digital

Analog

Digital

Analog or Digital?



Analog or Digital?

- **Resolution and Precision**

✓ **Analog** systems offer infinite resolution because the signals can take any value within a range.

✓ **Digital** systems, however, have limited resolution due to the discrete nature of digital signals.

- **Processing and Control**

✓ Digital systems are easier to program and reconfigure.

✓ Analog systems may provide smoother and faster real-time responses but are harder to modify once implemented.

Digital or Analog?

- **Accuracy:**
  - For **analog**, 0.1% is very **good accuracy**.
  - For **digital**, add **more circuitry** to get more accuracy. Subject to accuracy of analog sensor input.

- **Long Term Storage:**
  - Analog circuits store data as a voltage on a capacitor. Max storage time is in minutes.
  - Digital circuits store data in memory. Max storage time is years.

Digital or Analog?

- **Speed**
  - Fastest circuits are analog.
  - The highest frequency circuits in your cell-phone etc. are analog.

- **Design Cost/Time**
  - Analog designers have to worry about:
    - **noise**, power-supply variation, **cross-talk** between-wires, inaccurate values, **temperature variation** of component values, ground bounce, clock feed-through, . . .
    - To digital designers, these are 2nd order effects.

Digital or Analog?

- **Design Cost/Time cont.**
  - **Analog** designers typically need several years experience.
  - There are many more people doing **digital** design than analog.
  - Analog circuits are rapidly being redesigned in digital.

- **Digital is Essentially Analog**
  - Analog problems become especially important in digital for:
    - very fast circuits.
    - very low supply voltages (1.0 V).
    - very large circuits (5 million gates per chip).

When to Choose Analog vs. Digital

Analog:

- ✓ **High** precision.
- ✓ Real-time **response**.
- ✓ When the physical variable is **continuous**.
- ✓ Traditional control systems without a lot of data processing.

Digital:

- ✓ If you need **flexibility** and **programmability** for changing system requirements.
- ✓ For **noise immunity** and **long-distance communication**, as digital signals are less affected by interference.
- ✓ When integrating with **computers**, **microcontrollers**, or any system requiring digital processing.

Summary

Analog	Digital
Continuous signals	Discrete
Infinite possible values	Finite values (0,1)
High Noise Sensitivity	Low Noise Sensitivity
Complex Hardware	Simple Hardware
High resolution and accuracy	Can lose information during sampling

Hybrid System

- **Hybrid** system use both **analog and digital** components. For example, a sensor might produce an analog signal that is then converted to digital using an Analog-to-Digital Converter (ADC), allowing a microcontroller to process the data.

- In many applications, analog systems (sensors, amplifiers) combined with digital control systems (microcontrollers, programmable logic controllers) for optimized performance.

Hybrid System Applications

- **Communication systems**

Hybrid systems are common in radio transceivers, where analog components process signals over the air and digital components manage data processing and storage.

- **Control systems**

Many industrial control systems use analog inputs (from sensors) that are converted into digital signals for more accurate and complex processing.

- **Audio and video equipment**

Analog sound and video inputs (e.g., microphones or cameras) are converted to digital for processing, storage, and editing, then converted back to analog for playback.

Hybrid System

Digital Signals: What is Binary?

https://perexpteamworks.com/en/binary-number-system/

Digital Signals: What is Binary?

- Binary is a number system that consists of only two digits: **0** and **1**.
- It's also known as **Base-2**, unlike the decimal system (**Base-10**) that uses ten digits (0-9).
- Binary is the foundation of digital computing since it maps easily onto the two states of electrical circuits: on (1) and off (0).

Digital Signals: Binary (Base-2)

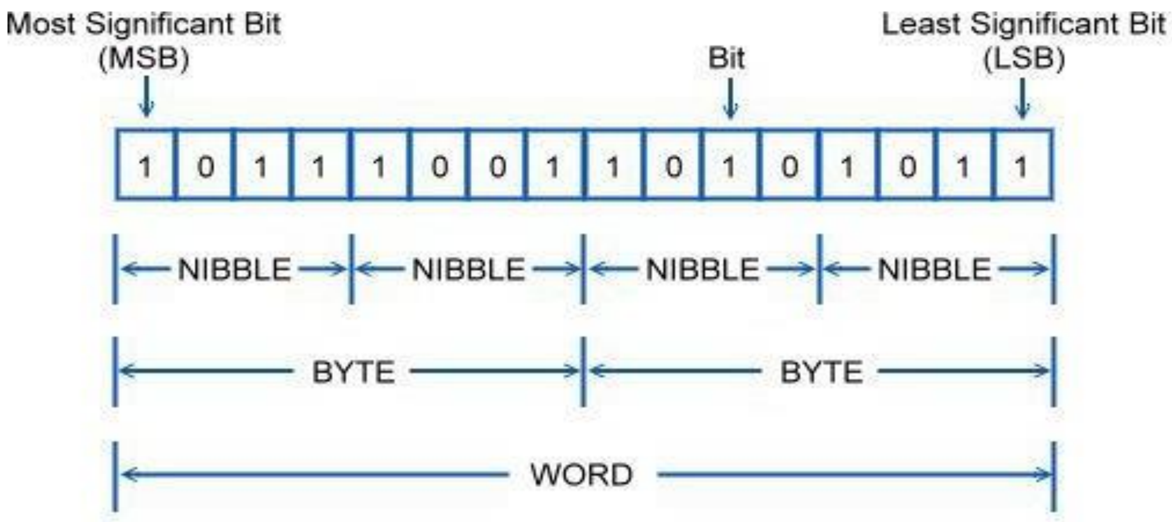
- Before we look at digital signals in depth, first we need to know how to represent them. To do this we use the Binary system.

- The binary number system (base-2) is an **alternative** to the decimal numbering system that we use every day.

- In the decimal system we have 10 symbols to represent numbers
  - The symbols are: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
  - After 9 we run out of symbols, so instead we ‘carry’ the 1 to the left and reset the symbol to the right to 0, resulting in 10 (ten).

Digital Signals: Binary (Base-2)

- After 1, we run out of symbols, so we ‘carry’ the 1 to the left and reset the symbol to the right to 0, resulting in 10 (two).
- Each of these 1s or 0s is called a ‘**bit**’
- A group of 4 bits is called a ‘**nibble’** and 8 bits is called a ‘**byte’**
- ‘**Word**’ is another group of bits whose length is processor dependent. It could be 16, 32, or 64 bits, etc.



Why Binary is Important

- Digital systems, like computers and electronic devices, use binary to process data.
- Binary simplifies the design of circuits in processors, memory, and data storage.
- Logic gates (AND, OR, NOT) in computers operate using binary.
- Binary data can represent anything: numbers, text (ASCII), images, and more.

Application of Binary

- **Computers**

All software and hardware operate in binary (e.g., binary code, machine code).

- **Networking**

IP addresses and network protocols use binary for routing and data transfer.

- **Data Storage**

Files (text, images, video) are stored in binary form.

Decimal versus 4-bit Binary Numbers

- Try converting 1 to 15 into a 4-bit binary equivalent

Decimal	Binary	Decimal	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	10	1010
3	0011	11	1011
4	0100	12	1100
5	0101	13	1101
6	0110	14	1110
7	0111	15	1111

Digital Signals: Binary (Base-2)

- In an n-bit word we can represent up to 2^n states (0 to $2^n - 1$)
- For example:
  - A 4-bit word can represent up to 2^4 (16) states which are numbered from 0 to $2^4 - 1 = 15$ after which we run out of bits. The state 15 would be represented by 1111.

$$\boldsymbol{Num_{MAX} = 2^n - 1}$$

- We can rearrange this equation to determine the minimum number of bits we need to represent a number:

$$\boldsymbol{n_{minimum} = log_2(Num + 1)}$$

- As we can only have an integer number of bits, we must **round this up** to represent the number.

Converting between Binary and Decimal

- Exp. 1: Convert 21 to binary

We can check the minimum number of bits required

$$n_{minimum} = log_2(Num + 1)$$

$$n_{minimum} = 4.46 = 5$$

Dividend	Remainder
21/2 = 10	1
10/2 = 5	0
5/2 = 2	1
2/2 = 1	0
1/2 = 0	1

$$Ans: 10101$$

Converting between Binary and Decimal

- Exp. 2: Convert 11011 to decimal

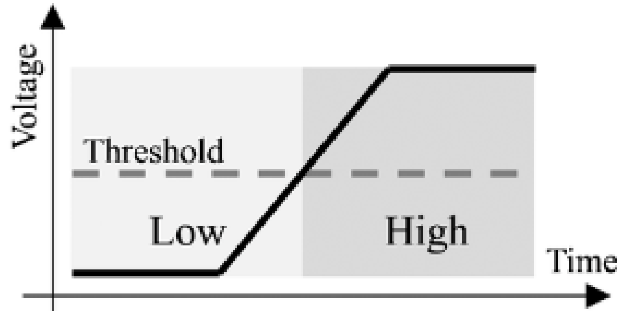
$$\boldsymbol{Num_{MAX} = 2^n - 1}$$

$$\boldsymbol{Num_{MAX} = 31}$$

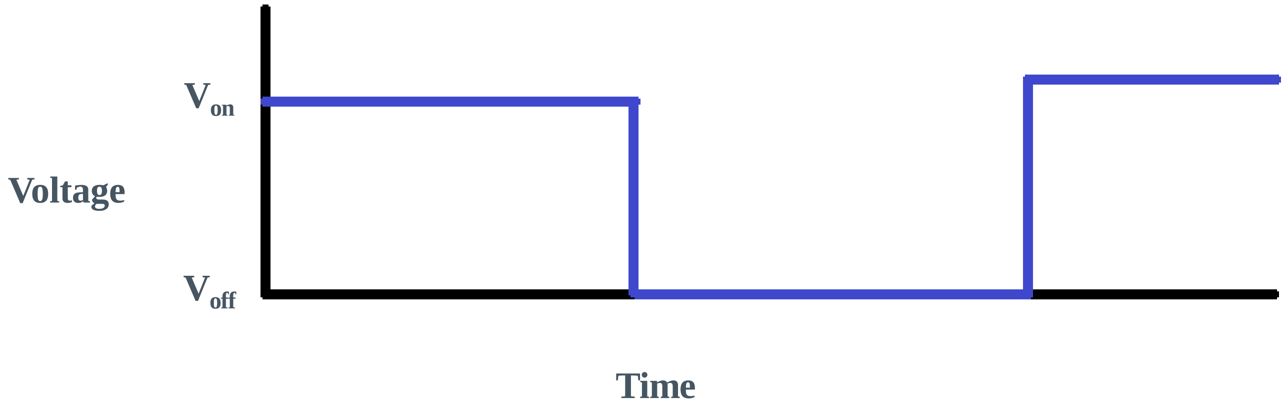
$$(11011)_2 = (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) = (27)_{10}$$

$$\boldsymbol{Ans: 27}$$

Digital Signal Waveform



- Typical digital signal waveform:



- These 1s and 0s are in fact a range of voltages, and are not very sensitive to noise (temperature, supply voltage, etc.)

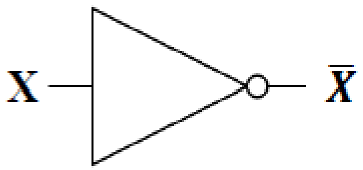
- The ‘On’ and ‘Off’ voltages can be different values, 5V, 3.3V, 1.2V, etc. depending on the device.

Introduction to Boolean Logic/Algebra

- Two value algebra:
  - Values are ‘0’ or ‘1’, ‘True’ or ‘False, and ‘High’ or ‘Low’

- Variables:
  - Variables, for example ‘X’, can have values of 1 or 0

- Operations:
  - Complement, Not, or Inverse:
    - $\overline{X}$ is the opposite of X

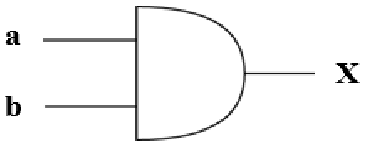


Introduction to Boolean Logic/Algebra

Operations cont.:

– AND (ab or a·b):

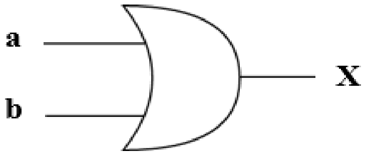
- X is 1 if variables **a** and **b** are both 1



∧	T	F
T	T	F
F	F	F

– OR (a + b):

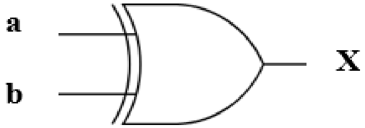
- X is 1 if **a** or **b** or **both** are 1



∨	T	F
T	T	T
F	T	F

– Exclusive OR (XOR) ($a \oplus b$)

- X is 1 if exactly **one** of **a** or **b** is 1

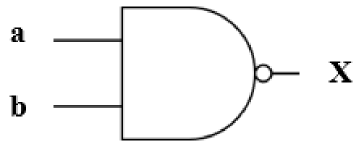


Introduction to Boolean Logic/Algebra

Operations cont.:

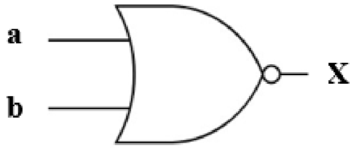
– NAND ($\overline{ab}$ or $\overline{a \cdot b}$):

- X is 0 if variables **a** and **b** are both 1



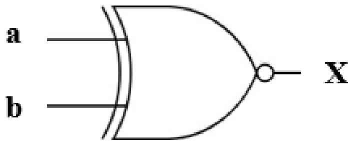
– NOR ($\overline{a + b}$):

- X is 0 if **a** or **b** or **both** are 1



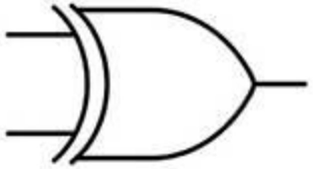
– X-NOR ($\overline{a \oplus b}$)

- X is 0 if exactly **one** of **a** or **b** is 1



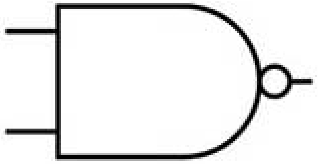
Introduction to Boolean Logic/Algebra

XOR



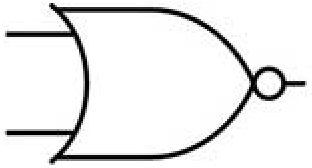
A	B	Output
0	0	0
1	0	1
0	1	1
1	1	0

NAND



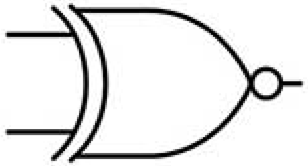
A	B	Output
0	0	1
1	0	1
0	1	1
1	1	0

NOR



A	B	Output
0	0	1
1	0	0
0	1	0
1	1	0

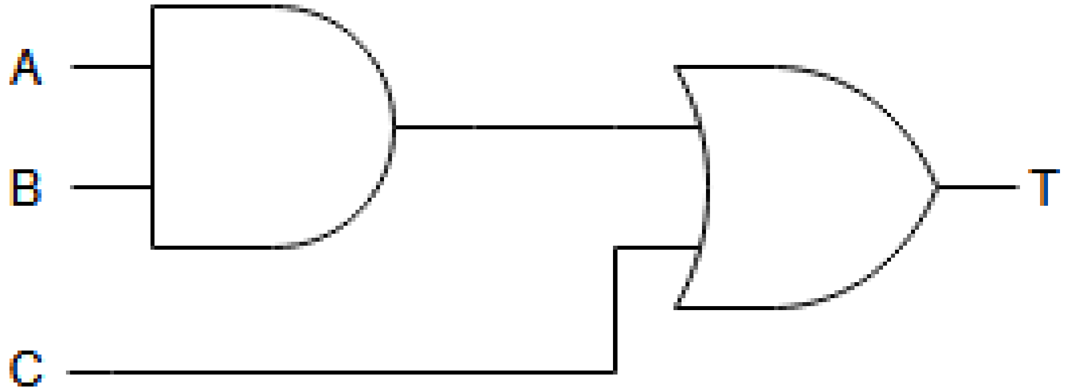
XNOR



A	B	Output
0	0	1
1	0	0
0	1	0
1	1	1

Equations from Digital Circuits

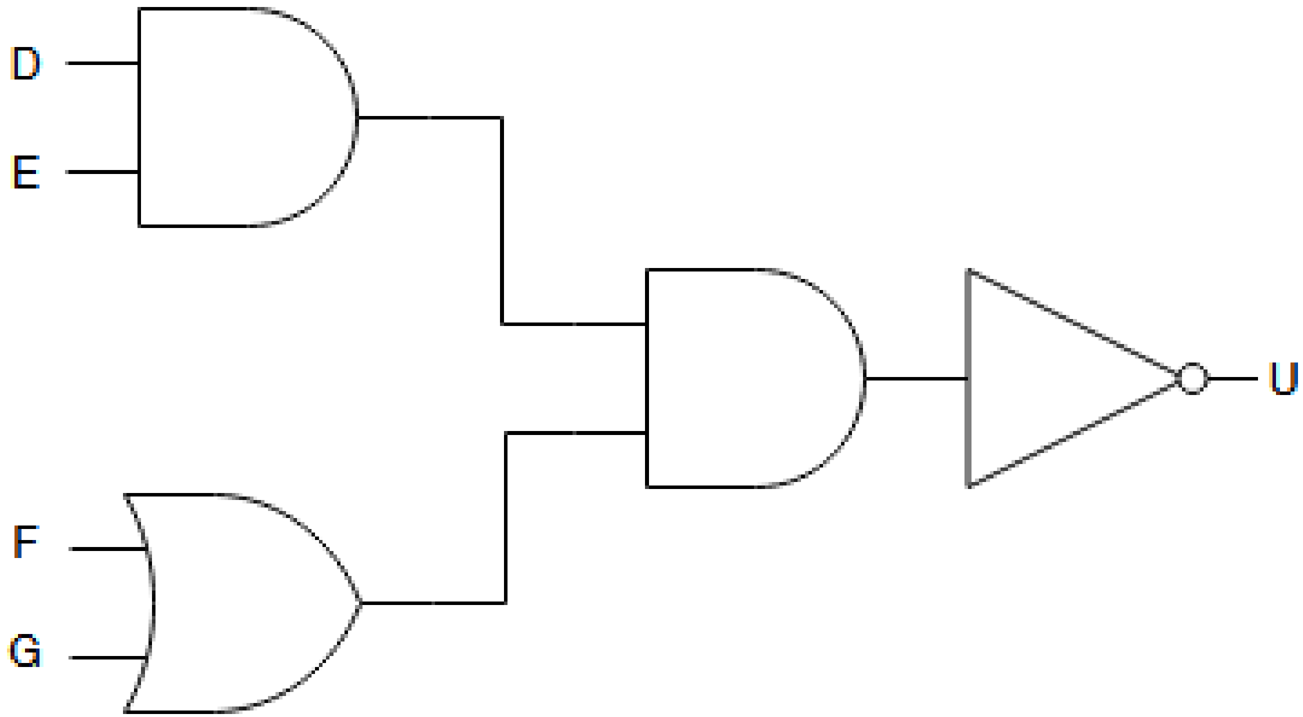
- Ex.1: Determine the equation for ‘T’



$$\boldsymbol{T = AB + C}$$

Equations from Digital Circuits

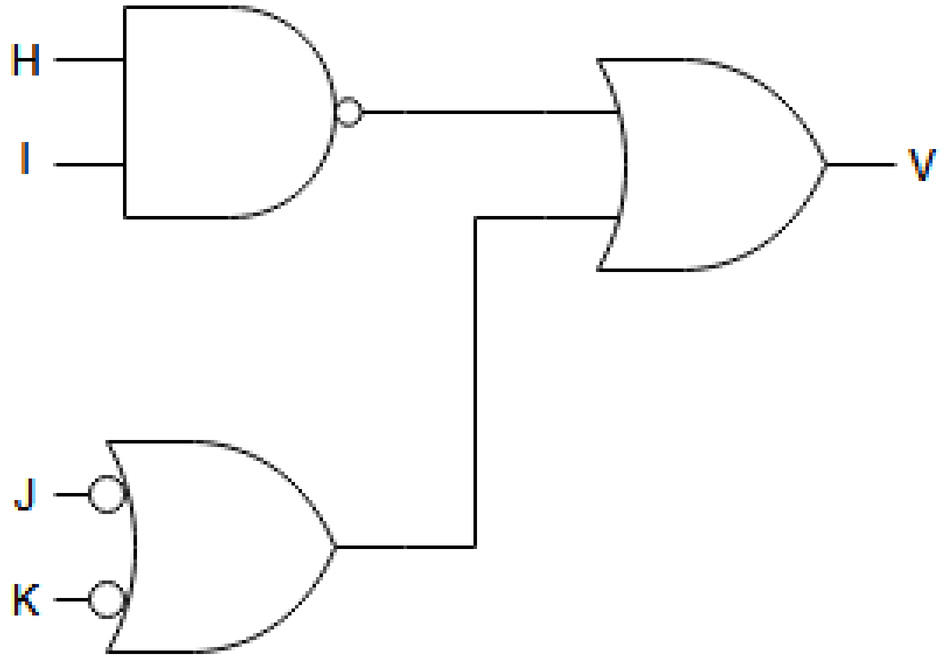
- Ex.2: Determine the equation for ‘U’



$$U = \overline{DE(F+G)}$$

Equations from Digital Circuits

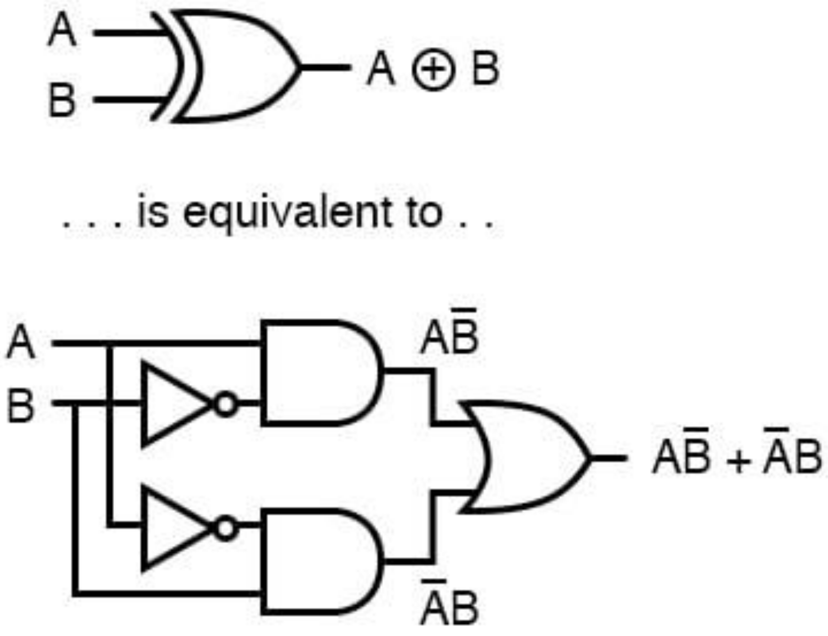
- Ex.3: Determine the equation for ‘V’



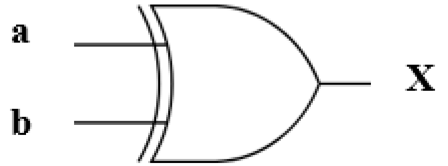
$$V = \overline{HI} + (\bar{J} + \bar{K})$$

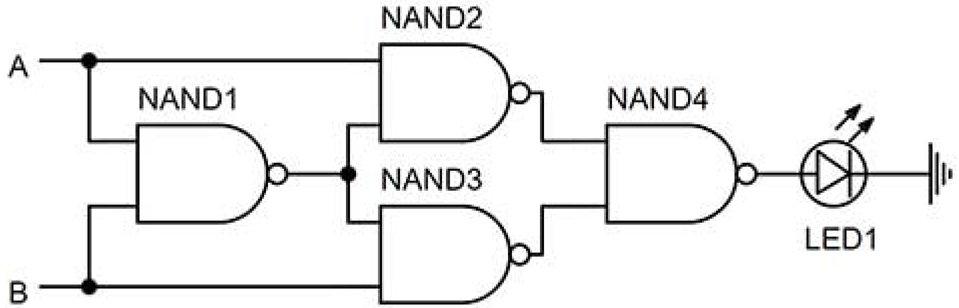
XOR and XNOR

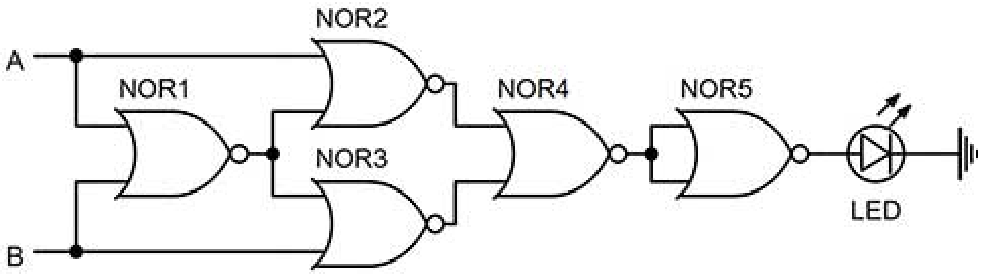
- The formulas for XOR and XNOR are combinations of AND as well as OR gates



$$A \oplus B = A\bar{B} + \bar{A}B$$

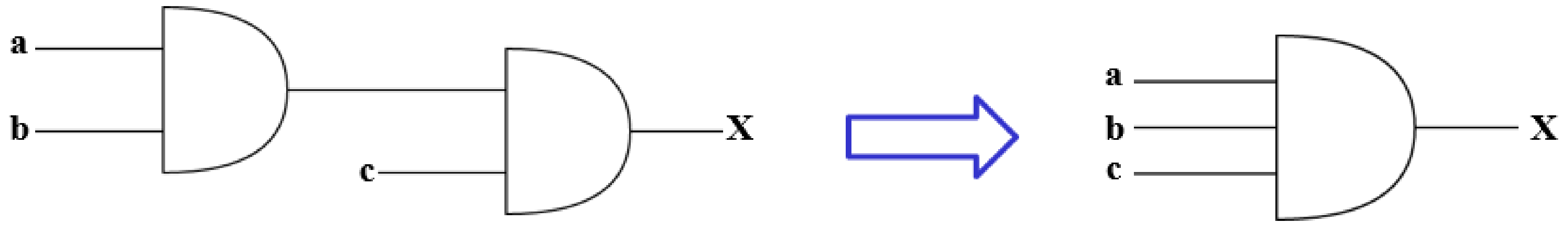






Introduction to Boolean Logic/Algebra

- What if we have more than 2 inputs?
  - $X = a \cdot b \cdot c$



Formulas From Truth Tables

- Exp. 1: Determine the equation for 'J' given the following:

A	B	J
0	0	0
0	1	1
1	0	0
1	1	1

$$J = \bar{A}B + AB$$

Formulas From Truth Tables

- Exp. 2: Determine the equation for 'K' given the following:

A	B	C	K
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

$$K = \bar{A}B\bar{C} + \bar{A}BC + A\bar{B}C$$

Formulas From Truth Tables

- Exp. 3: Determine the equation for ‘L’ for:

$$L = \bar{A}\bar{B}CD + \bar{A}BC\bar{D} + A\bar{B}\bar{C}D + ABC\bar{D}$$

A	B	C	D	L
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

Questions?