University of Nottingham
UK | CHINA | MALAYSIA

LECTURE 9

DC Motors &
Boolean Algebra

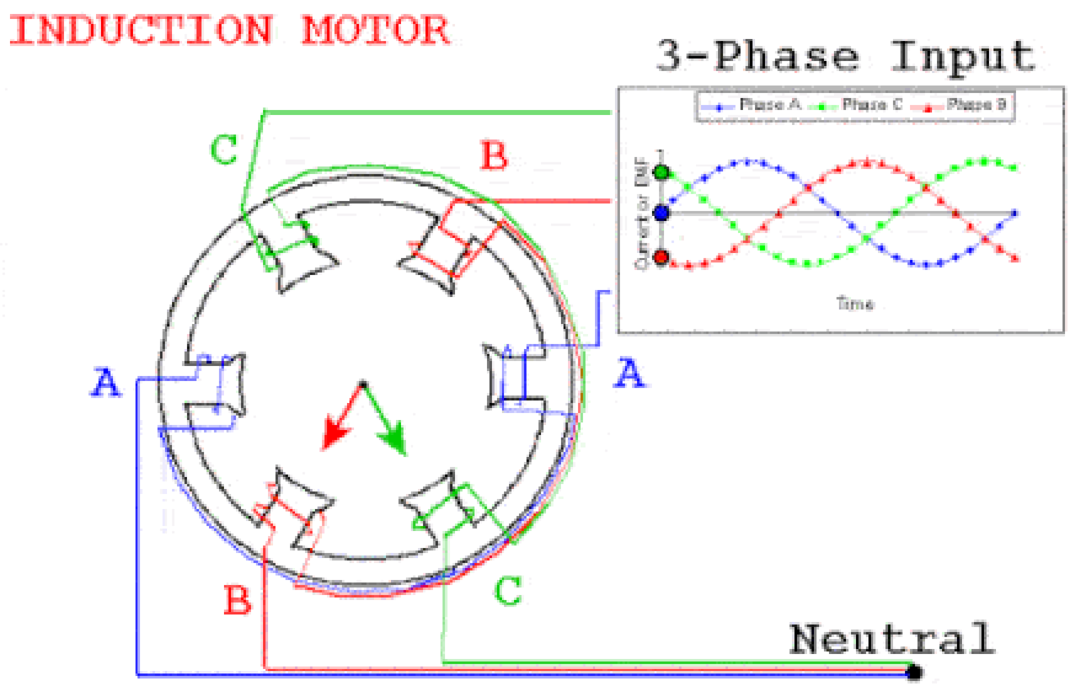
Electromechanical Devices
MMME2051

Module Convenor – Surojit Sen

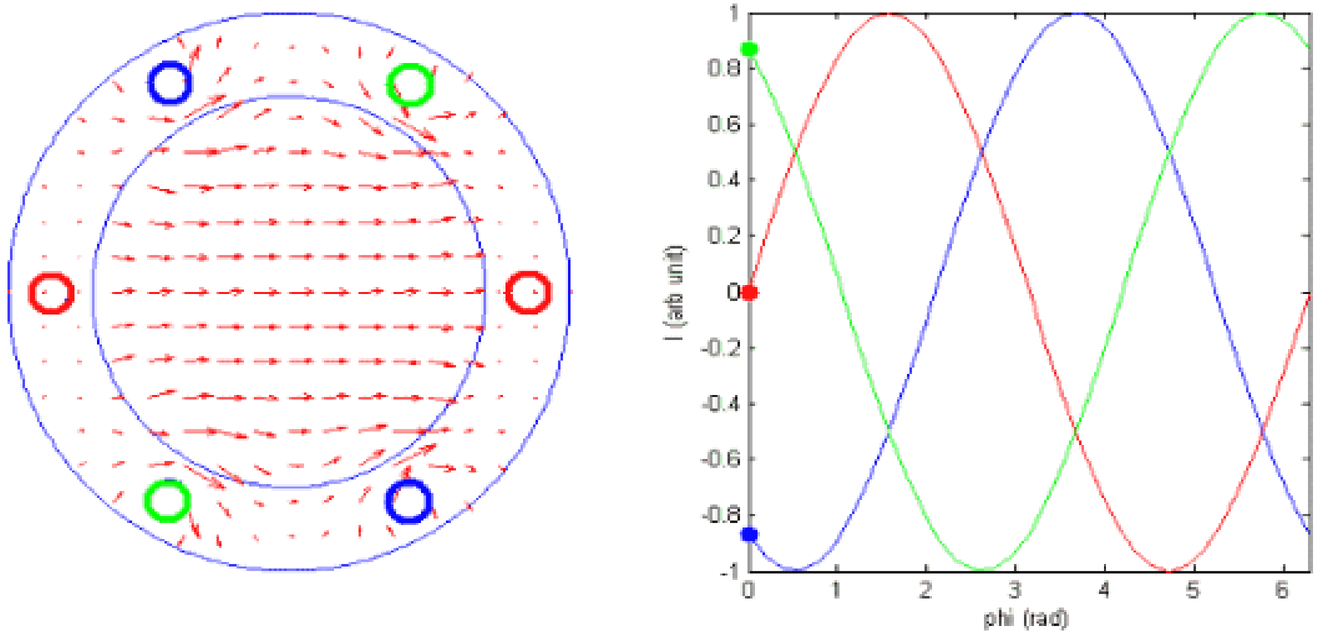
Learning Outcomes

- DC Motor
  - **Revision** of all motors studied so far – Induction, Stepper
  - **Operation** of a **Simple DC Motor**
  - **Why** use a DC Motor?
- **Boolean Algebra**
  - **Revision of Digital Electronics**
  - **Addition** (OR), **multiplication** (AND), **complement** (NOT)
  - **Laws**

Induction Motor



https://axljoann.blogspot.com/2021/05/3-phase-induction-motor-hitachi-three.html



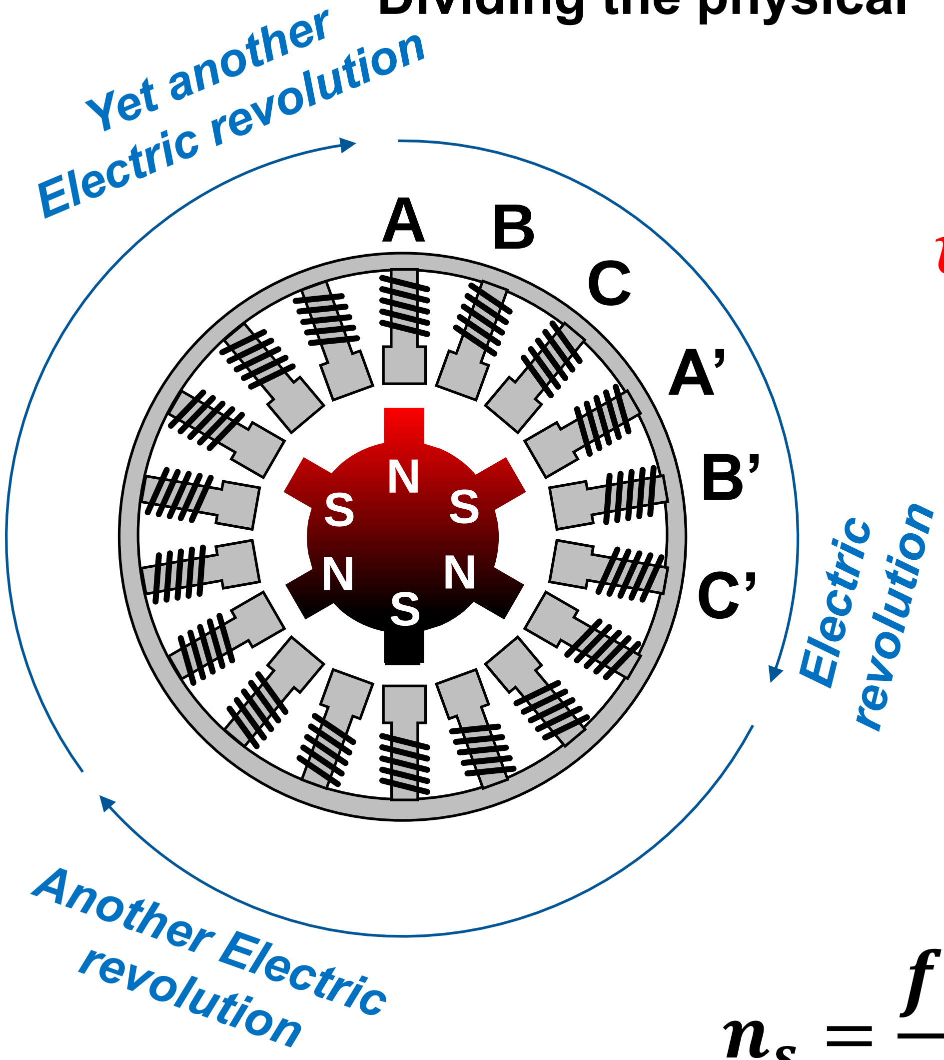
https://medium.com/@abhisheksingh73017/how-an-induction-motor-starts-real-answer-from-an-engineer-65f2fd7fa5b1

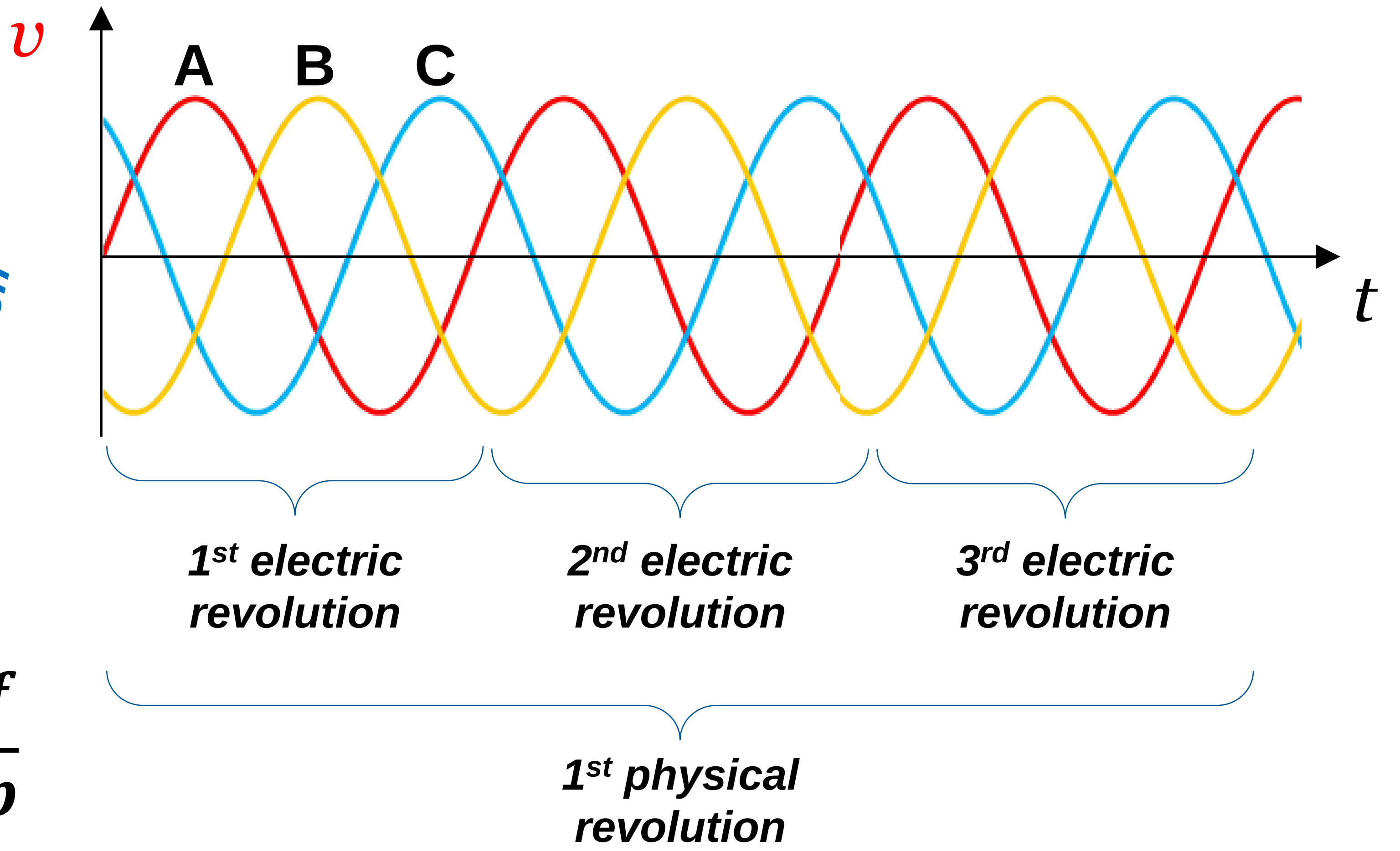
The speed of rotation is called “synchronous speed” which is nothing but the 3-phase AC frequency!

$$n_s(Hz) = f \text{ or } n_s(RPM) = 60 \times f$$

Induction Motor

Dividing the physical "angular space" into multiple sets of 3-phase

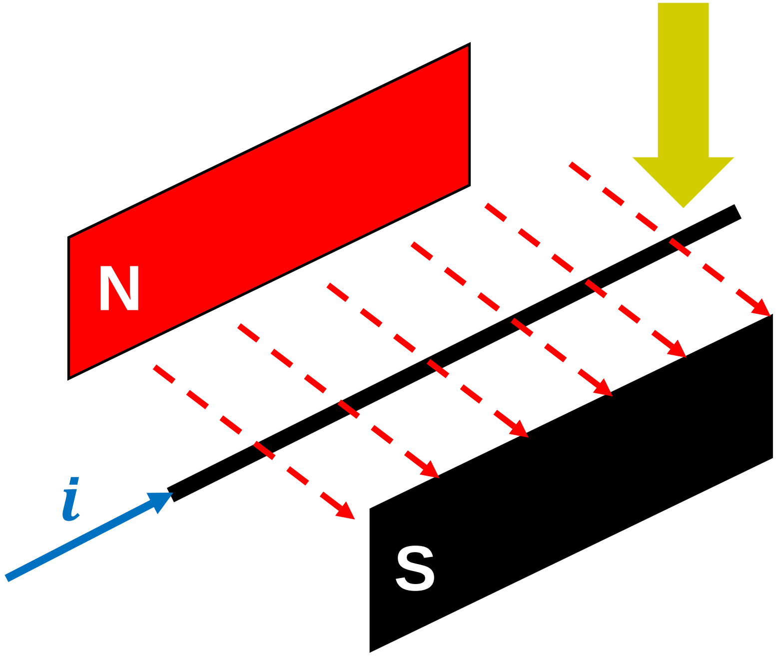


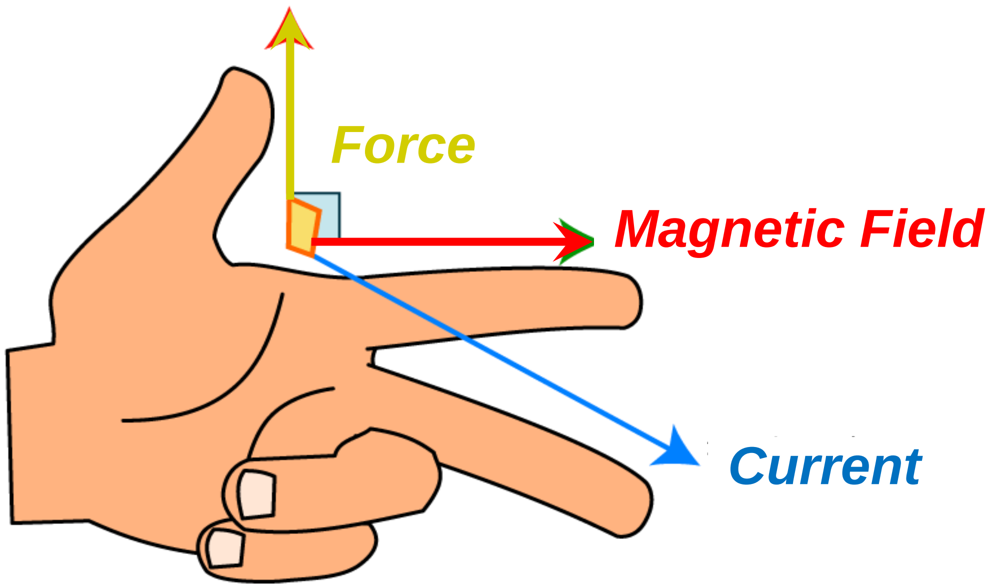


$$n_s = \frac{f}{p}$$

Induction Motor

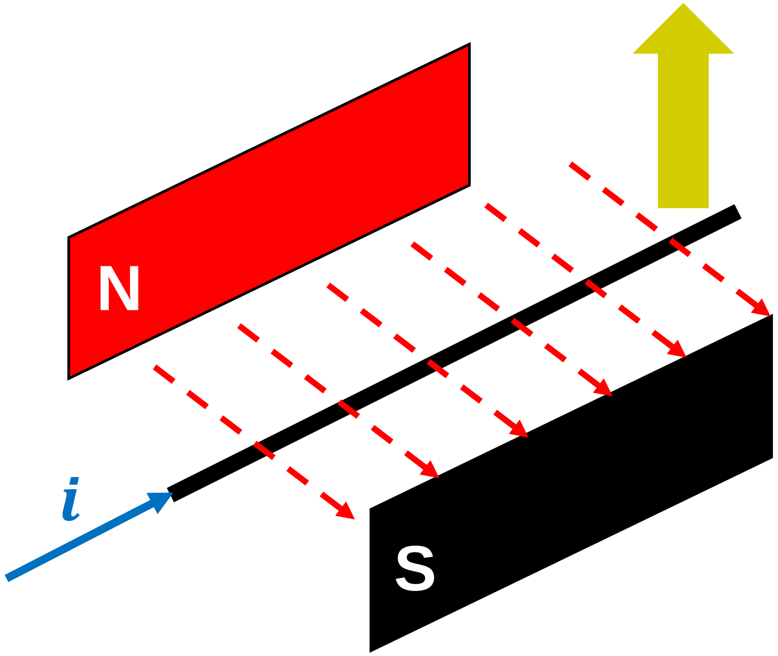
Left-Hand Rule (Motors)

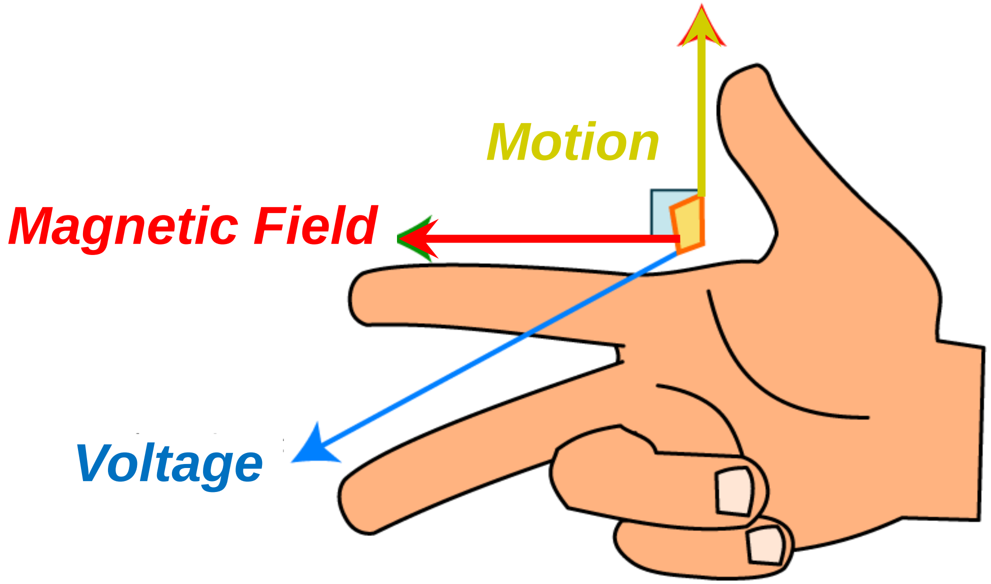




A current-carrying conductor in a magnetic field experiences a force/thrust

Right-Hand Rule (Generators)

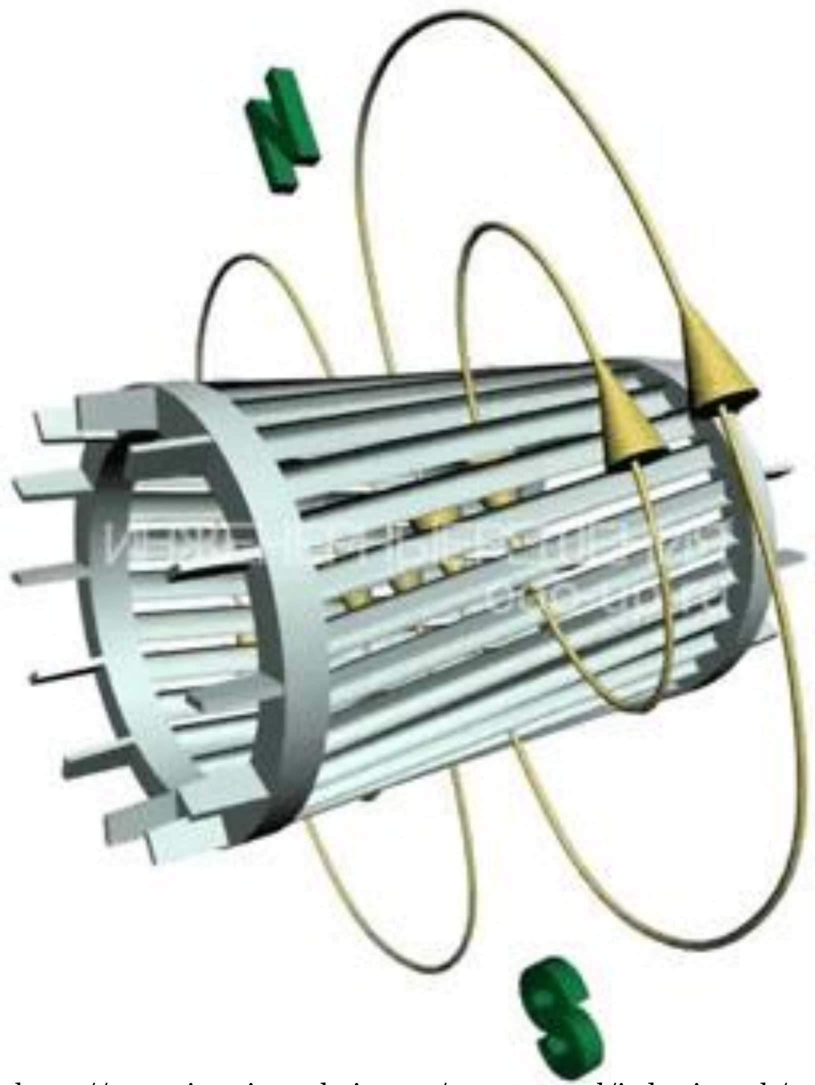




A conductor moving in a magnetic field generates a voltage across itself (current produced if circuit was to be completed)

University of Nottingham
UK | CHINA | MALAYSIA

Induction Motor



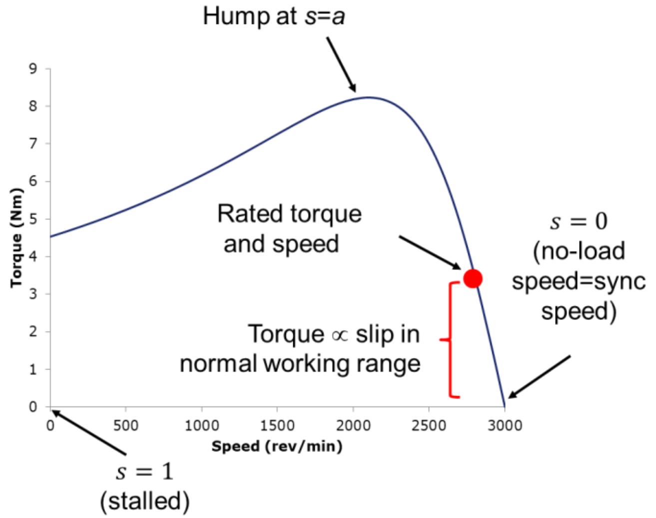
https://en.engineering-solutions.ru/motorcontrol/induction3ph/

- Rotating **Magnetic Field** produced by the stator is continually **cutting a conductor**
  - **Synchronous** Speed = Speed of the **rotating magnetic field**, i.e., **stator** field, i.e., **input supply**
- An **EMF gets generated** (RH Rule). In a squirrel cage rotor, everything is shorted! Hence, **current flows**
- Now the conductor is a **current-carrying** conductor. Current-carrying conductor experiences a **force** in the **magnetic field** (LH Rule)
- **Rotor needs to slip** (allowing the cutting) to produce any torque
- **Higher slip = higher torque**

Induction Motor

$$T = \frac{3p}{2\pi f} \times \frac{V^2 as}{X_R(a^2 + s^2)}$$

- T –Torque in star-connected motor
- p – Pole pairs per phase
- f – Supply frequency
- V – Supply phase voltage
- $a = \frac{R_R}{X_R}$ – Resistance-to-reactance ratio of rotor
- $s = \frac{n_s - n}{n_s}$ – Per-Unit slip (n_s – Sync Speed)
- n – Actual speed of rotor (same unit as sync speed)
- X_R – Reactance of Rotor (as seen from stator – referred impedance – remember Transformer?)

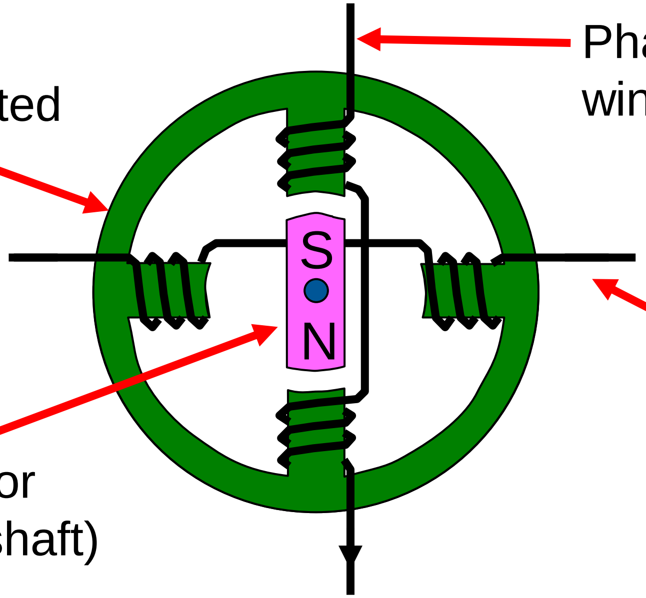


- No-load speed = synchronous speed
- Torque ∝ slip (approx.) for small torques
- Torque-speed characteristic has "hump" at $s = \frac{R_R}{X_R} = a$
- Under running conditions slip is small e.g. 5%
- By setting $\frac{dT}{ds} = 0$, can show that maximum ("pull-out") torque is

$$T_{max} = \frac{3p}{4\pi f} \frac{V^2}{X_R}$$

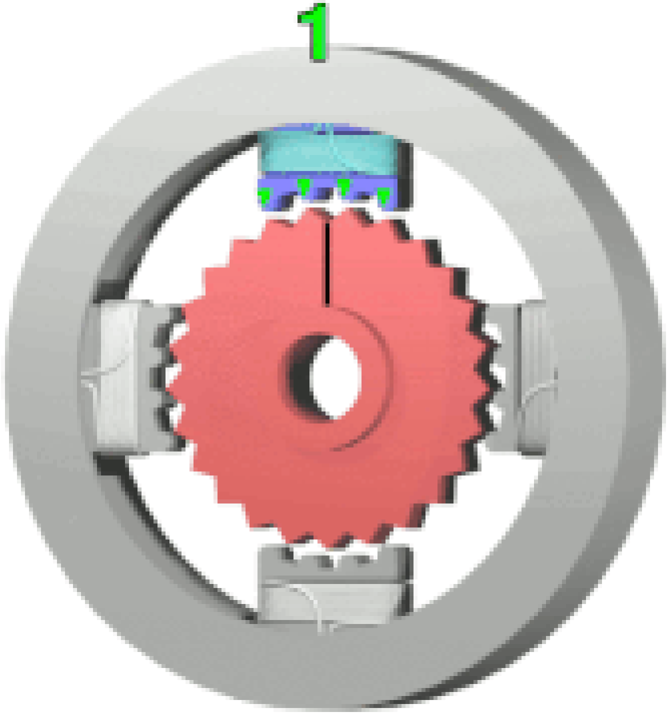
- Motor stalls if load torque T reaches $T_{\max}$

Stepper Motor



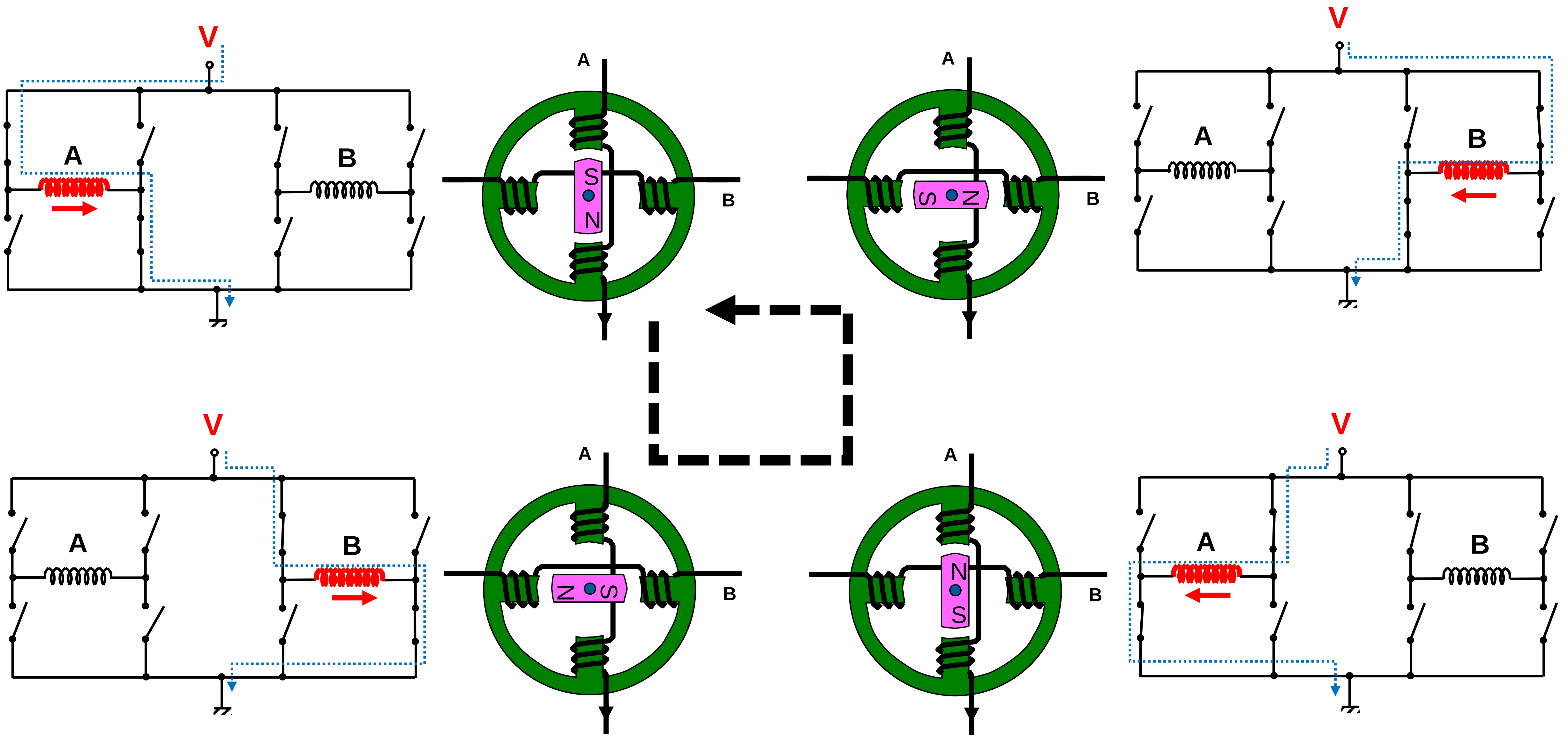
- Rotor is (usually) **permanently magnetised**
- Attracted to a **different pair of poles at each step**
- Moves from **pole to pole** as each pair of poles is energised
- So it moves in a **series of steps**

You can imagine, a motor design on left would make the motor spin in a **jerky** fashion. In real world, the motor looks like below. Each “**tooth**” is a magnet pole.

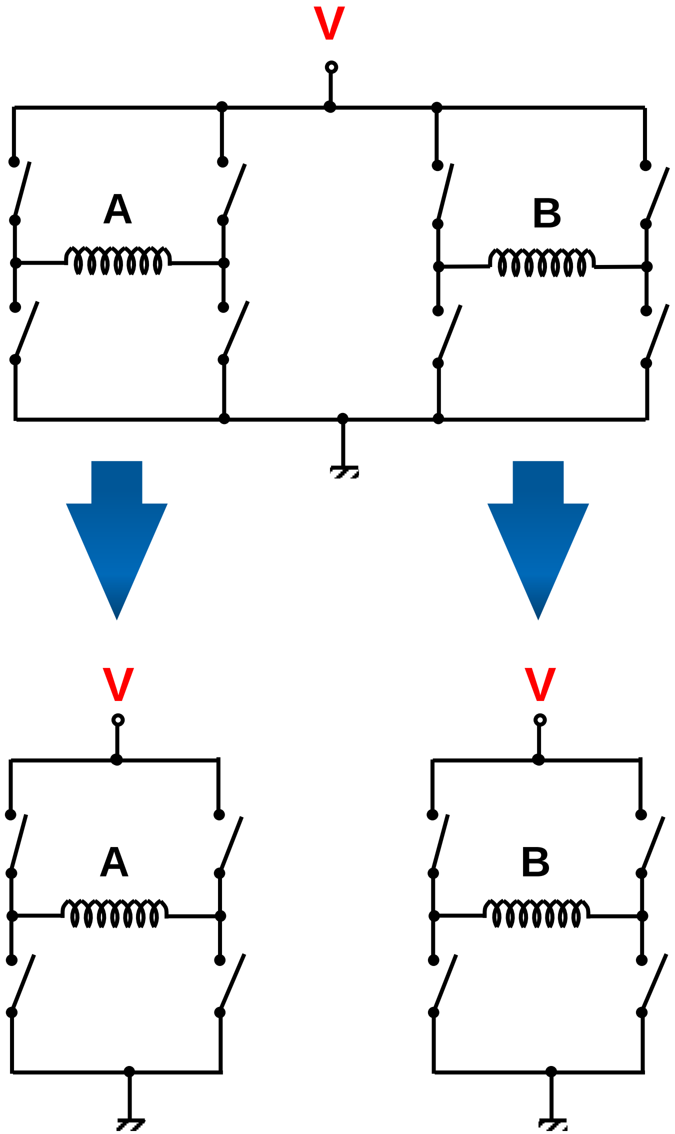


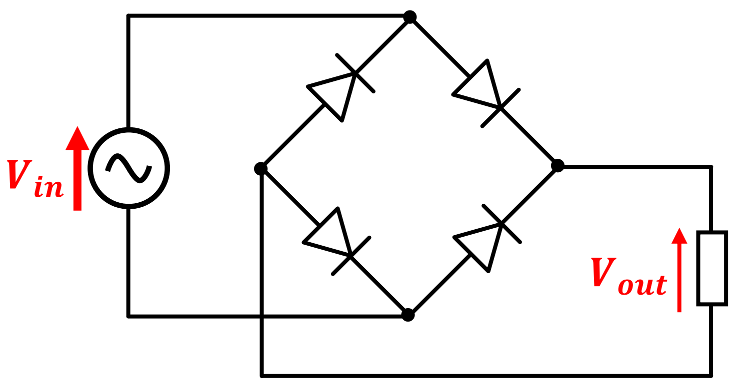
https://en.wikipedia.org/wiki/Stepper_motor

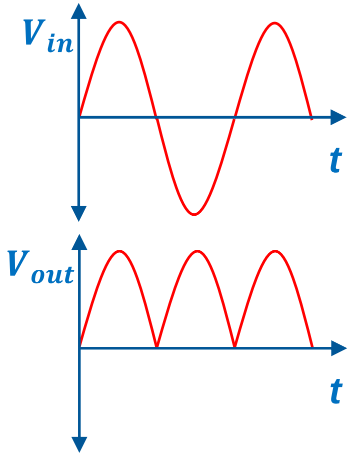
Stepper Motor



Stepper Motor







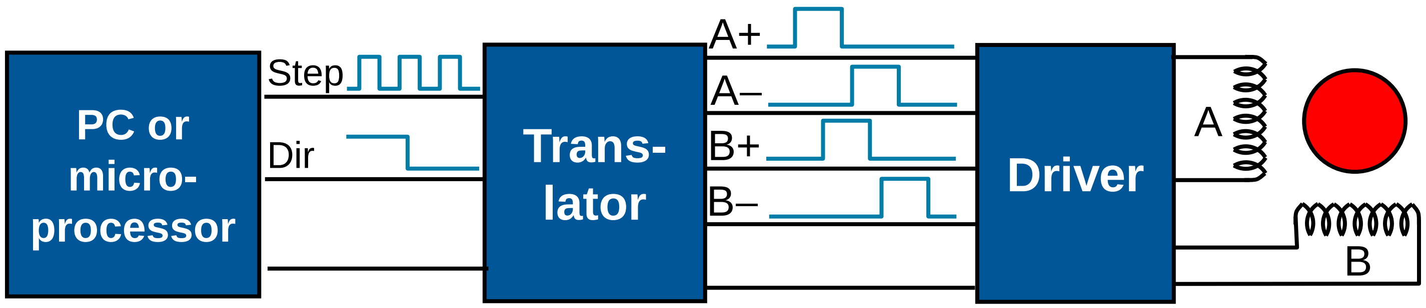
This circuit is more specifically called the **Diode Bridge**

H-Bridge

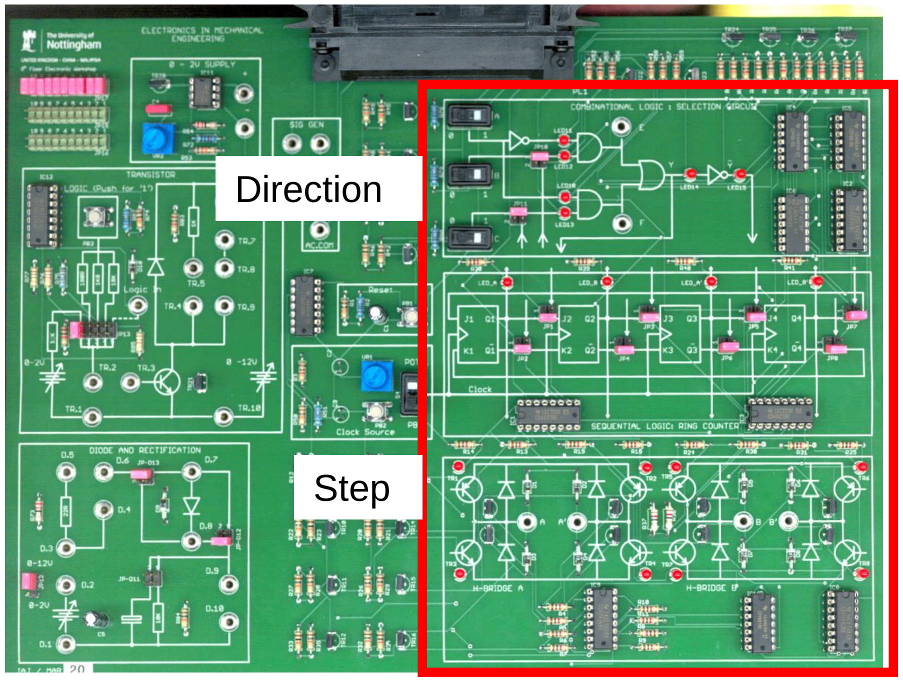
An H-Bridge is a circuit that allows polarity inversion across a load – basically allows current to flow in both direction by the application of switches (transistors) or diodes

Universally used circuit for **Rectification** and **Motor Control**

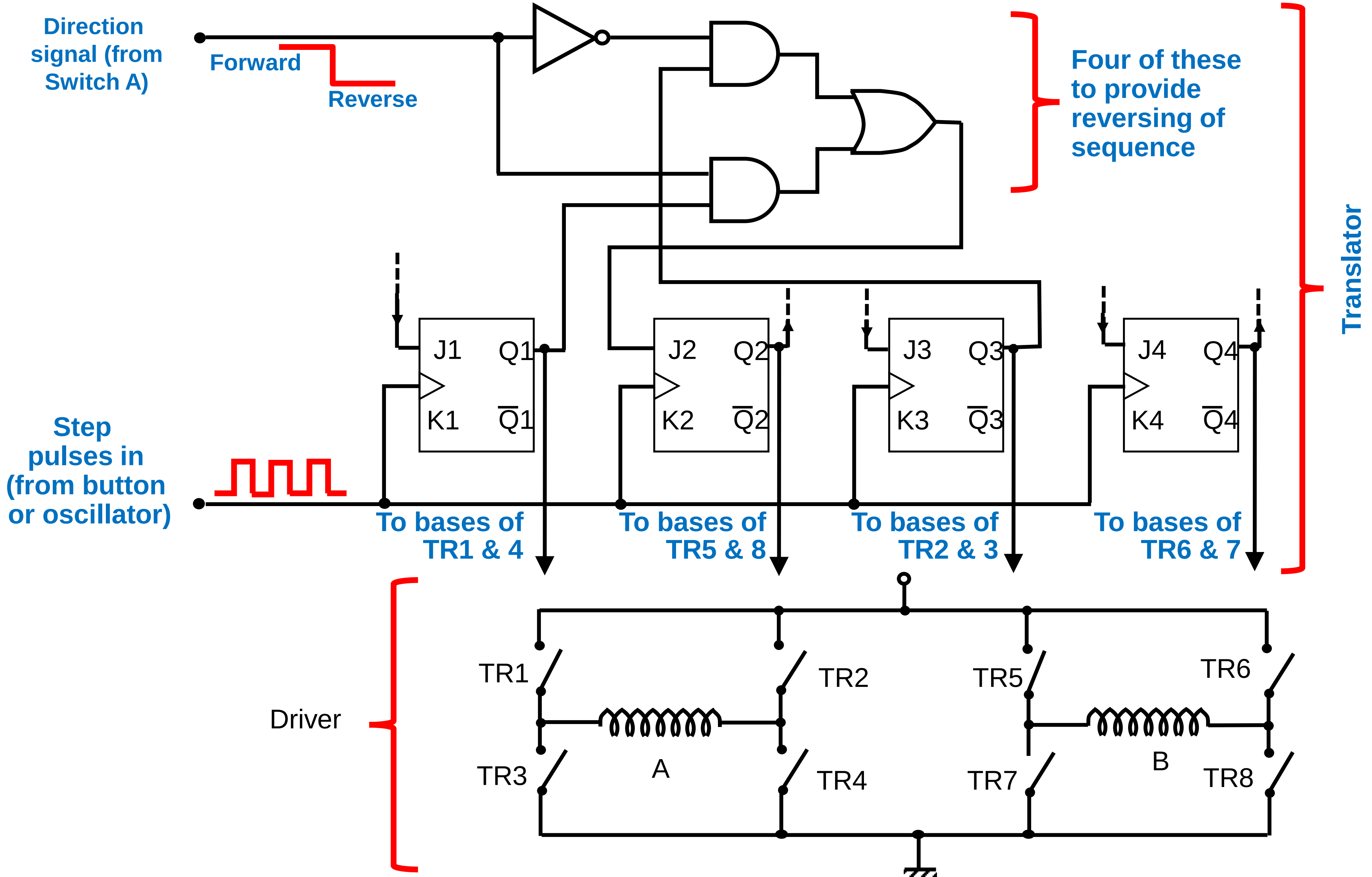
Stepper Motor



- **Sequential logic** (interprets "step" signals)
- **Combinational logic** (interprets "direction")
- **Transistors** (these are the switches which connect and disconnect the windings)

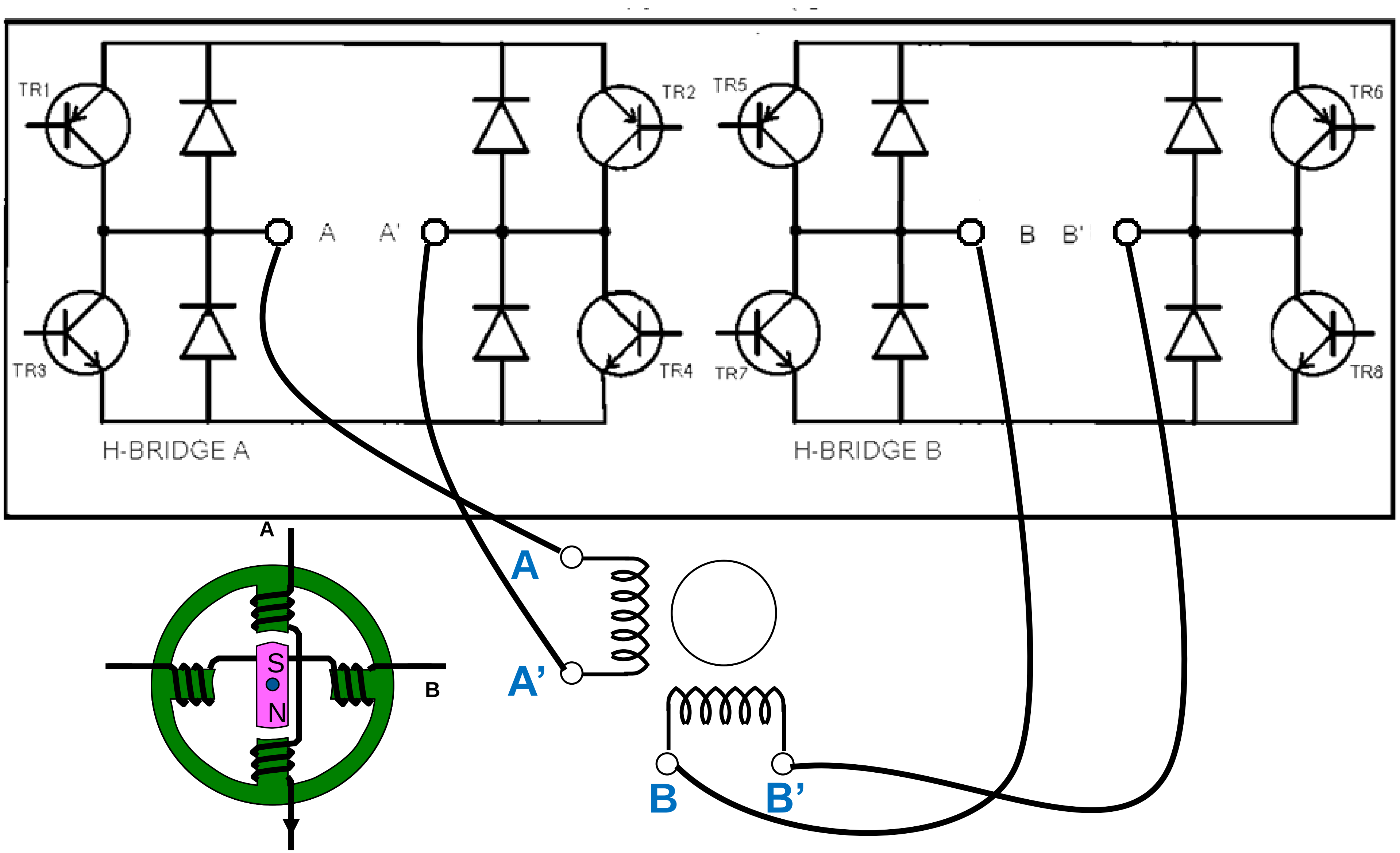


Stepper Motor



A	B	C	Y
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	1

Stepper Motor



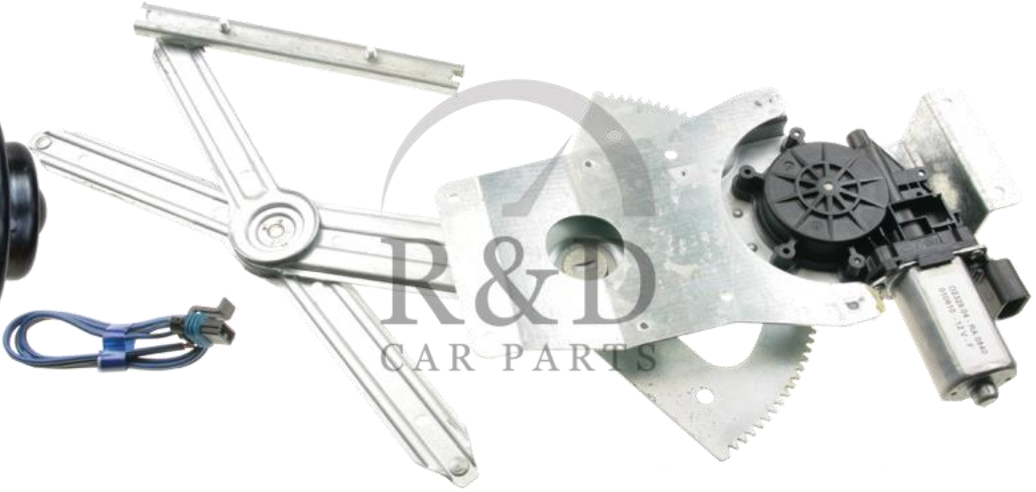
Simple DC Motor

“Simple” DC Motor are one of the oldest motor inventions that are still being used to this day – they are very simple from an engineering point of view

They are largely superseded now by “**electronically commutated**” DC Motors (hence the usage of “Simple” in this design)

Car Wiper Motor

https://ladaworld.com/en/electric/68-lada-wiper-motor-2103-3730000.html



Power Window Motor

https://www.rendcarparts.com/saab/parts/4328415-5184874/rd5184874a

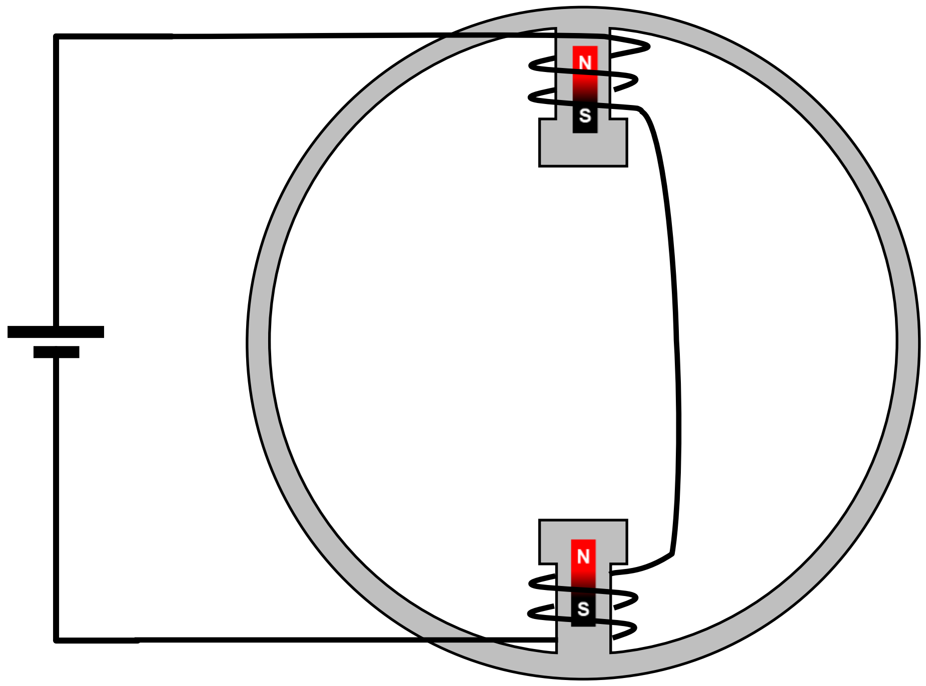
Electric Wheelchair

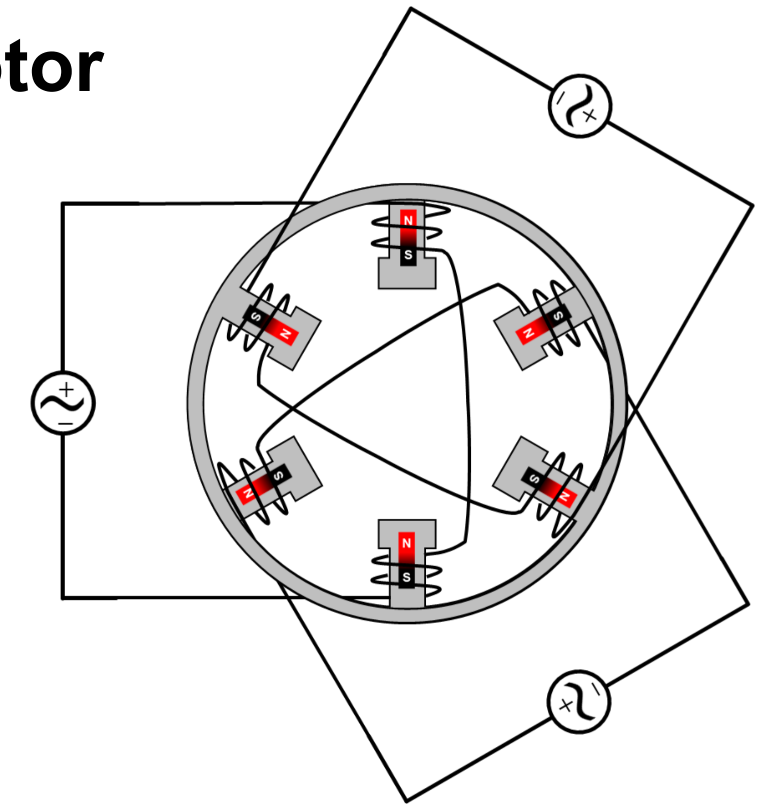
https://www.karmamedical.com/featured_item/efl/

"Simple" DC Motor

The stator is either **permanent magnet** or **wire-wound with DC voltage applied**

Effect is the same – constant magnetic field





Remember Induction motor stator? It is very similar, but much simpler:

- **No AC**
- **Only single phase**

“Simple” DC Motor

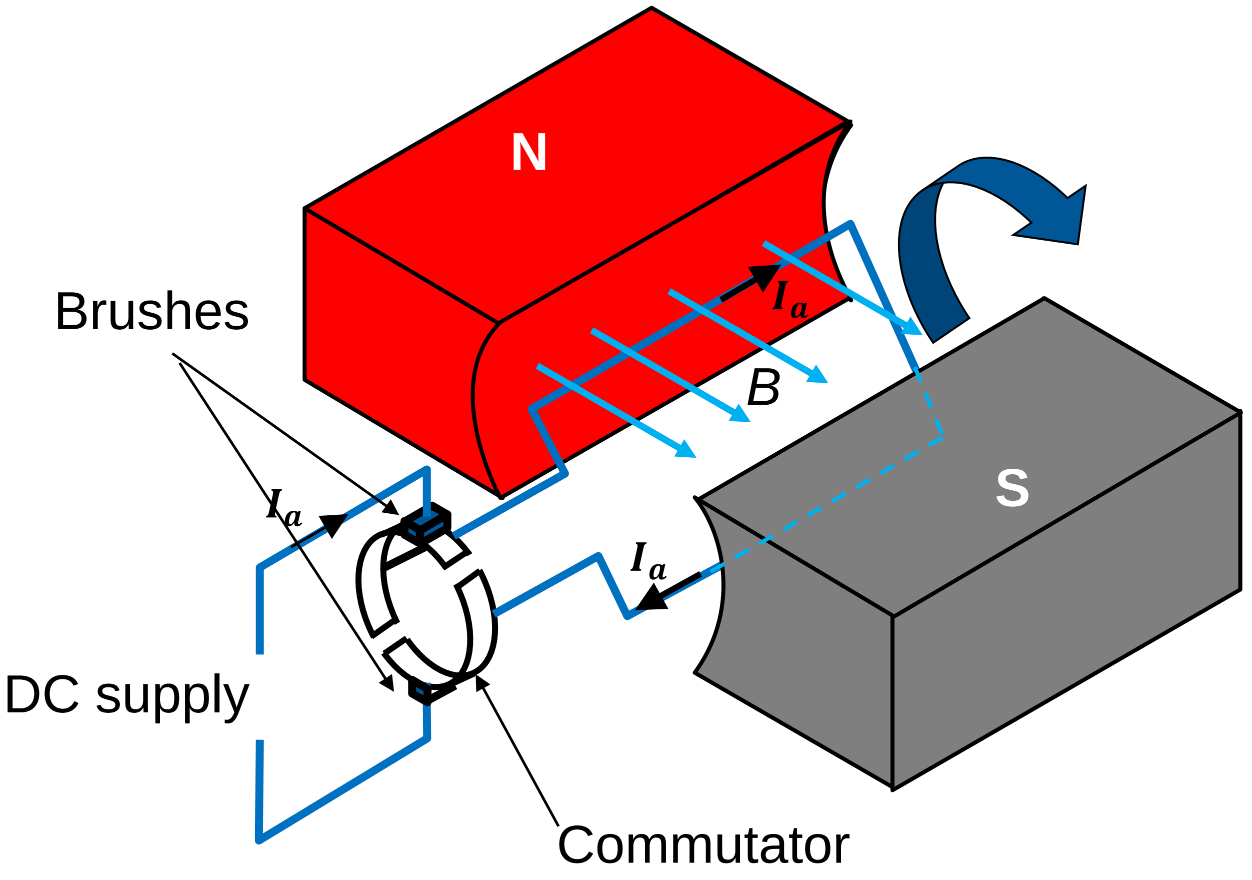
The rotor is simply a **coil** with current flowing through it via **another** DC voltage supply

Interesting bit here is the **commutator/brush pair** – this allows to flip the voltage polarity every half revolution

Hence, the current flow direction also flips

Now let us see how the motor operates!

Hint: Fleming’s LH and RH Rules.



Simple DC Motor

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

"Simple" DC Motor

Fleming's Left Hand Rule says a current-carrying conductor in a magnetic field experiences force/thrust

Lorentz Law says:

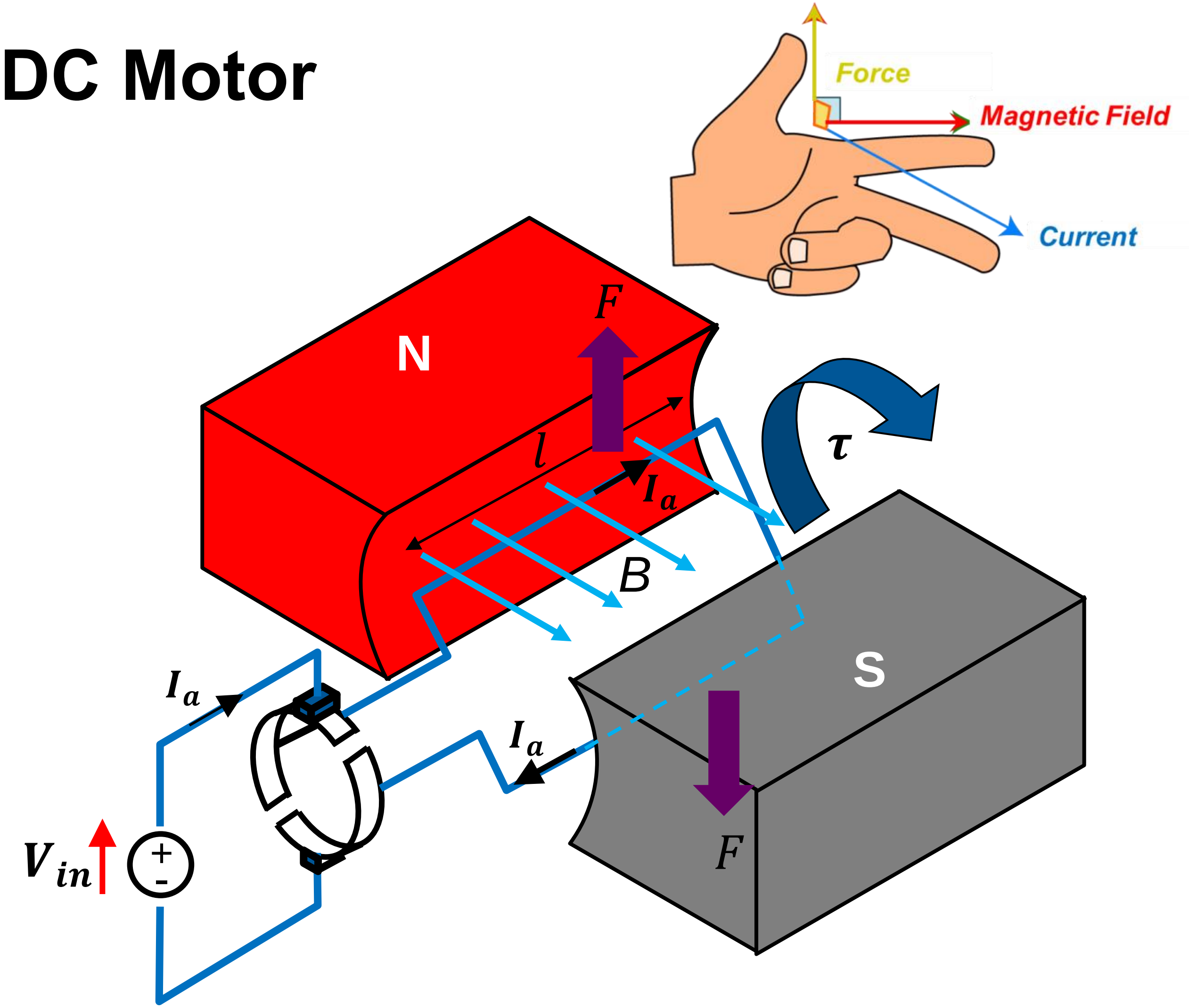
$$F = B.I_a l$$

Torque is a function of the force and radius (which is fixed):

$$\tau = r \times F$$

Hence, Torque is linearly proportional to the Armature Current:

$$\tau = KI_a$$



"Simple" DC Motor

Fleming's Right Hand Rule says a moving conductor in a magnetic field generates a voltage across it

Lorentz Law says:

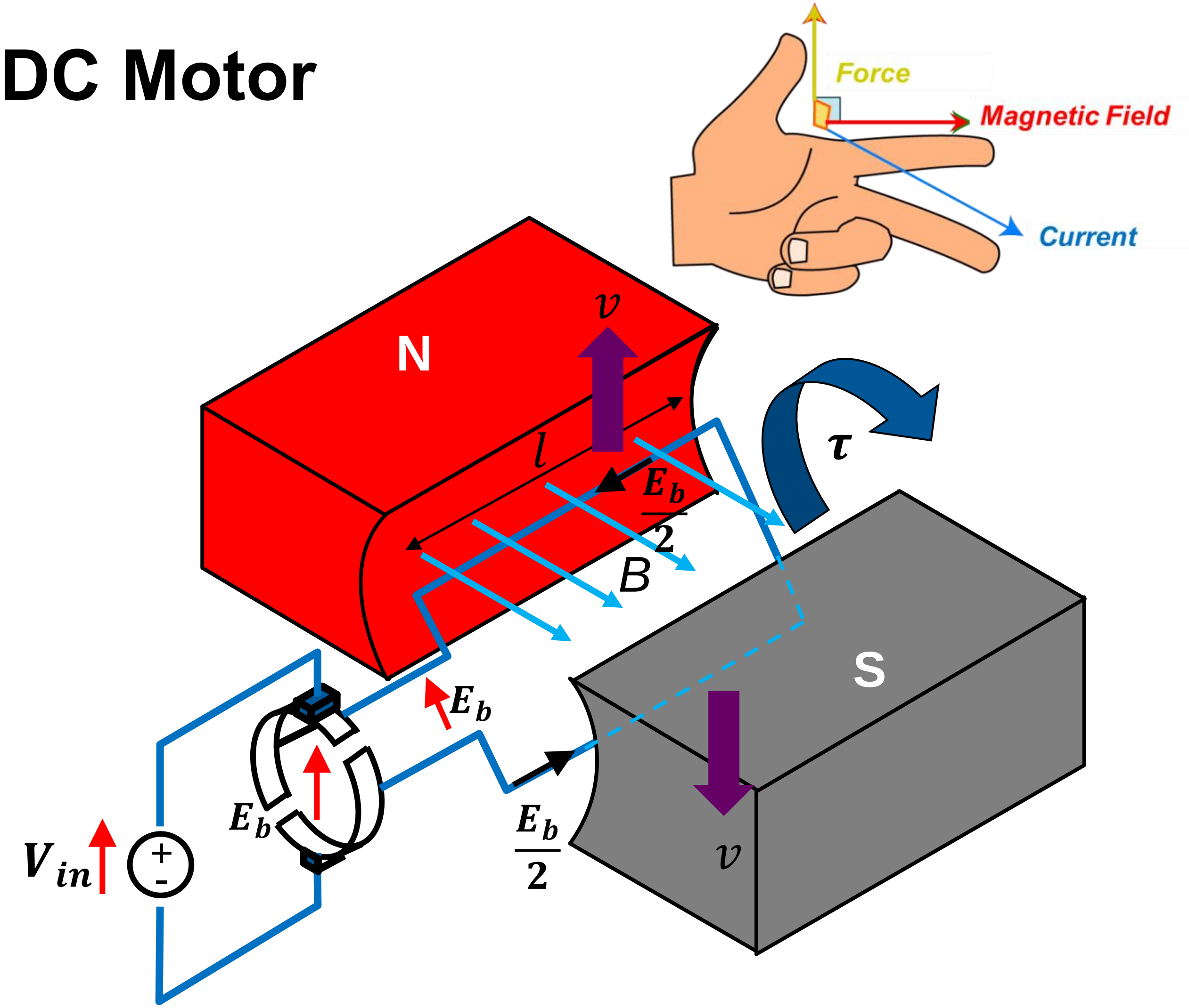
$$E_b = B \times vl$$

Angular velocity is linearly related to speed:

$$\omega = \frac{v}{r}$$

- Hence, Back EMF is linearly proportional to the Angular Speed:

$$E_b = K\omega$$



"Simple" DC Motor

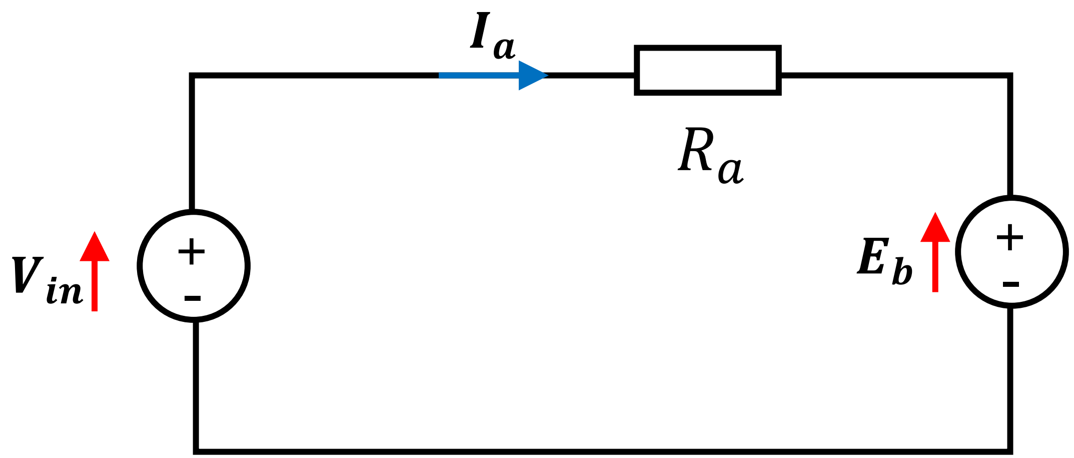
So the two equations to pay heed to are:

$$\tau = KI_a$$

$$E_b = K\omega$$

You can find out mathematically (using the original equations in previous two slides) that the value of K is same in both equations!

Equivalent electrical circuit can be used to visualise how the motor works



$$V_{in} = E_b + I_a R_a$$

$$V_{in} = K\omega + \frac{\tau}{K} R_a$$

"Simple" DC Motor

We can also plot the Torque-Speed curve of the "Simple" DC motor using this equation

At zero speed (stall):

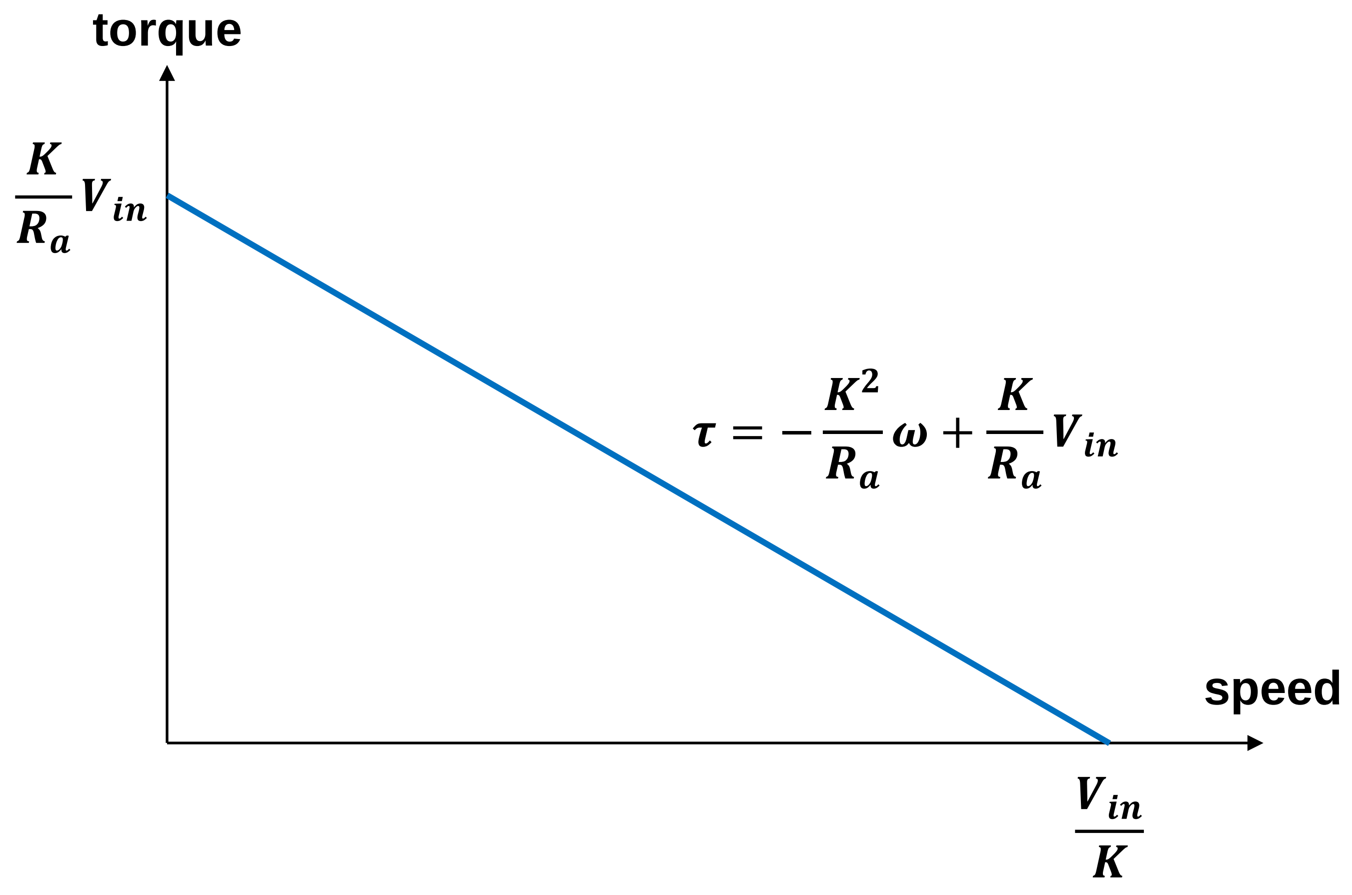
$$V_{in} = K(0) + \frac{\tau}{K} R_a$$

$$\boldsymbol{\tau = \frac{K}{R_a} V_{in}}$$

At zero torque (no load):

$$V_{in} = K\omega + \frac{(0)}{K} R_a$$

$$\boldsymbol{\omega = \frac{V_{in}}{K}}$$



"Simple" DC Motor

We can also plot the Torque-Speed curve of the "Simple" DC motor using this equation

At zero speed (stall):

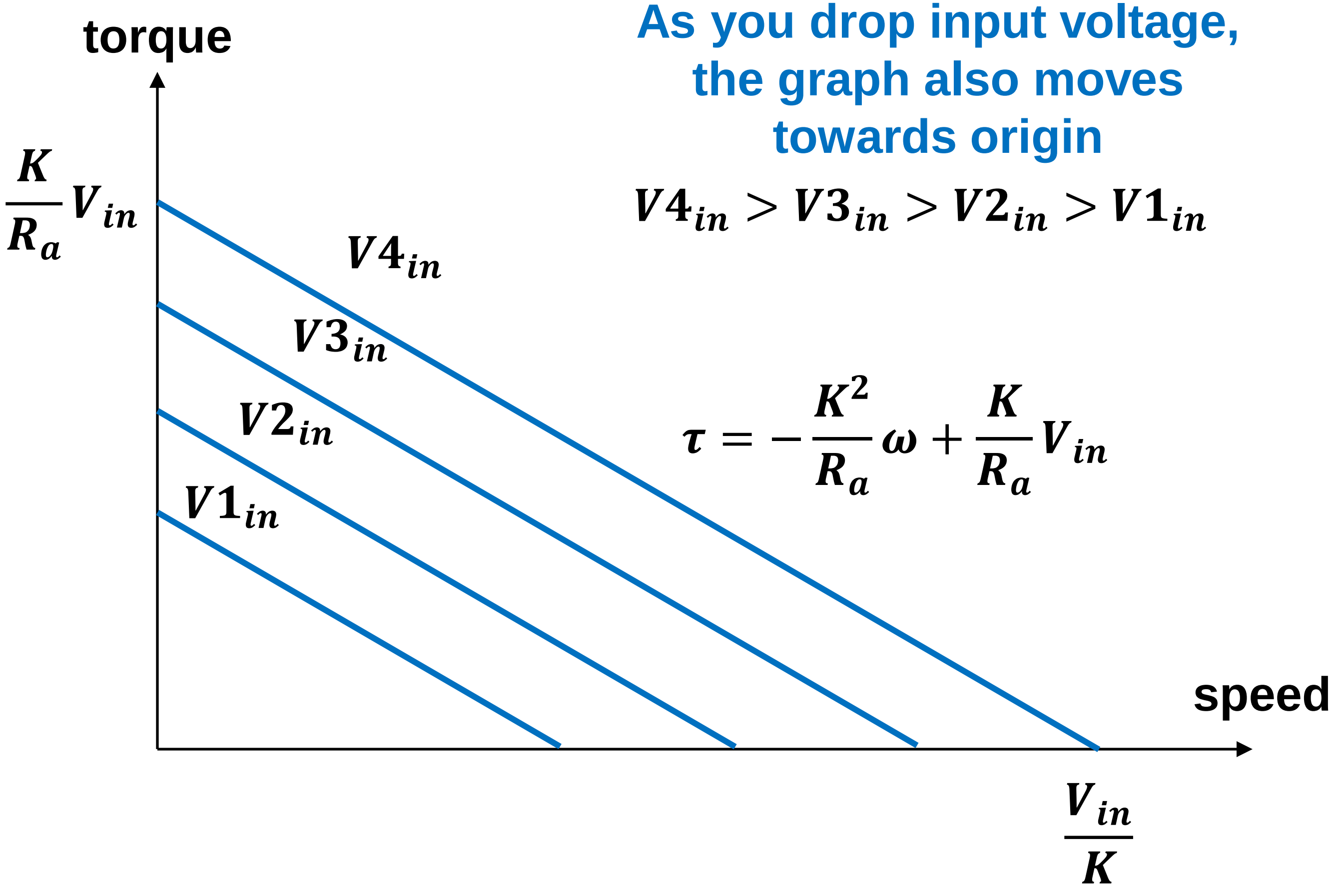
$$V_{in} = K(0) + \frac{\tau}{K} R_a$$

$$\boldsymbol{\tau = \frac{K}{R_a} V_{in}}$$

At zero torque (no load):

$$V_{in} = K\omega + \frac{(0)}{K} R_a$$

$$\boldsymbol{\omega = \frac{V_{in}}{K}}$$



As you drop input voltage, the graph also moves towards origin

$$V4_{in} > V3_{in} > V2_{in} > V1_{in}$$

$$\tau = -\frac{K^2}{R_a}\omega + \frac{K}{R_a} V_{in}$$

Worked Example 1

A motor has a constant of $\mathbf{0.025\frac{Vs}{rad}}$ and an armature resistance of $\mathbf{0.5\Omega}$. Find the torque which is produced when supplying the motor from 16 V and running at a speed of 5000 RPM.

$$V_{in} = E_b + I_a R_a = K\omega + I_a R_a$$

$$I_a = \frac{V_{in} - K\omega}{R_a}$$

Here,

$$V_{in} = 16\,V$$

$$K = 0.025$$

$$R_a = 0.5\,\Omega$$

$$n = 5000\,RPM$$

So,

$$\omega = 2\pi \times \frac{5000}{60} = \mathbf{523.6\frac{rad}{s}}$$

$$I_a = \frac{V_{in} - K\omega}{R_a}$$

$$I_a = \frac{16 - 0.025 \times 523.6}{0.5}$$

$$I_a = 5.82\,A$$

And,

$$\boldsymbol{\tau} = K \times I_a$$

$$\tau = 0.025 \times 5.82$$

$$\boldsymbol{\tau = 0.1455\,Nm}$$

Worked Example 2

A DC motor (the "Torpedo 850") is used for small electric drills and model boats. Its no-load speed (ignore frictional effects) is given as 9778 RPM when running from $12\ V$. It draws a current of $10.8\ A$ at $12\ V$ at a speed of 8311 RPM.

Find motor constant and armature resistance.

Find current, speed and mechanical power output at $12\ V$ and torque of $0.05\ Nm$.

$$V_{in} = E_b + I_a R_a = K\omega + I_a R_a$$

Motor constant: assume that under no-load condition there really is no torque so current is zero, so:

$$V_{in} = E_b = K\omega$$

$$K = \frac{V_{in}}{\omega} = \frac{12}{2\pi \times \frac{9778}{60}}$$

$$\boldsymbol{K = 0.0117 \frac{Vs}{rad}}$$

At 8311 RPM, current is 10.8 A

$$V_{in} = K\omega + I_a R_a$$

$$R_a = \frac{V_{in} - K\omega}{I_a}$$

$$R_a = \frac{12 - 0.0117 \times 2\pi \times \frac{8311}{60}}{10.8}$$

$$\boldsymbol{R_a = 0.168\ \Omega}$$

Worked Example 2

A DC motor (the "Torpedo 850") is used for small electric drills and model boats. Its no-load speed (ignore frictional effects) is given as 9778 RPM when running from $12\ V$. It draws a current of $10.8\ A$ at $12\ V$ at a speed of 8311 RPM.

Find motor constant and armature resistance.

Find current, speed and mechanical power output at $12\ V$ and torque of $0.05\ Nm$.

$$\tau = KI_a$$

$$I_a = \frac{\tau}{K}$$

$$I_a = \frac{0.05}{0.0117}$$

$$\boldsymbol{I_a = 4.27\ \mathrm{A}}$$

At 8311 RPM, current is 10.8 A

$$V_{in} = K\omega + I_a R_a$$

$$\omega = \frac{V_{in} - I_a R_a}{K}$$

$$\omega = \frac{12 - 4.27 \times 0.168}{0.0117}$$

$$\boldsymbol{\omega = 964 \frac{rad}{s}}$$

$$\boldsymbol{\omega = 9205\ RPM}$$

Mechanical output

$$W = \tau\omega = 0.05 \times 964 = \boldsymbol{48.2\ W}$$

Learning Outcomes

- DC Motor
  - **Revision** of all motors studied so far – Induction, Stepper
  - **Operation** of a **Simple DC Motor**
  - **Why** use a DC Motor?
- **Boolean Algebra**
  - **Revision of Digital Electronics**
  - **Addition** (OR), **multiplication** (AND), **complement** (NOT)
  - **Laws**

Digital v Analog

Digital

Information in form of **discrete** symbols, or **levels**

Variable can be only 1 out of a **finite number of options**

Humans interpret physical values in discrete levels

- **Alphabets**
- **Binary number**
- **Logic state**
- **Answer to the question** – “*Are you* enjoying this module?”

Analog

Information in form of **continuous** and **real-valued levels**

Variable can be only 1 out of an **infinite number of options**

The physical values exist naturally in continuous spectrum levels

- **Air pressure in this room**
- **Volume of my voice**
- **Battery voltage in your laptop**
- **Answer to the question** – “*How much* are you enjoying this module?”

Digital Methods of Information Exchange

Language – using letters

There are 26 alphabets in the English language – digital!

Numbers

Every number that we use, uses a distinct number of symbols (including the decimal point)

Binary

e.g.,
11100100

Octal

e.g.,
344

Decimal

e.g.,
228

Hexadecimal

e.g.,
E4

How does this actually relate to “numbers”?

Let us look at a number in the “Decimal” number-format, the one that we have grown up with.

Weight of the Formula Student 2021 car is

301 kg

MSB – Most Significant Bit

LSB – Least Significant Bit

$3 \times 10^{2} = 300$

$0 \times 10^{1} = 0$

$1 \times 10^{0} = 1$

How does this actually relate to “numbers”?

The same number in the Hexadecimal format will be

Weight of the Formula Student 2021 car is

12D kg

MSB – Most Significant Bit

LSB – Least Significant Bit

$1 \times 16^{2} = 256$	$2 \times 16^{1} = 32$	$D \times 16^{0} = 13$

How does this actually relate to “numbers”?

How about in Binary?

Weight of the Formula Student 2021 car is

MSB – Most Significant Bit

0001 0010 1101 kg

LSB – Least Significant Bit

- $0 \times 2^{11} = 0$
- $0 \times 2^{10} = 0$
- $0 \times 2^{9} = 0$
- $1 \times 2^{8} = 256$

256

- $0 \times 2^{7} = 0$
- $0 \times 2^{6} = 0$
- $1 \times 2^{5} = 32$
- $0 \times 2^{4} = 0$

32

- $1 \times 2^{3} = 8$
- $1 \times 2^{2} = 4$
- $0 \times 2^{1} = 0$
- $1 \times 2^{0} = 1$

13

Information in Electronics

We use **binary number system** in logical circuits in electronics

This aligns with computer/software engineering – binary system used

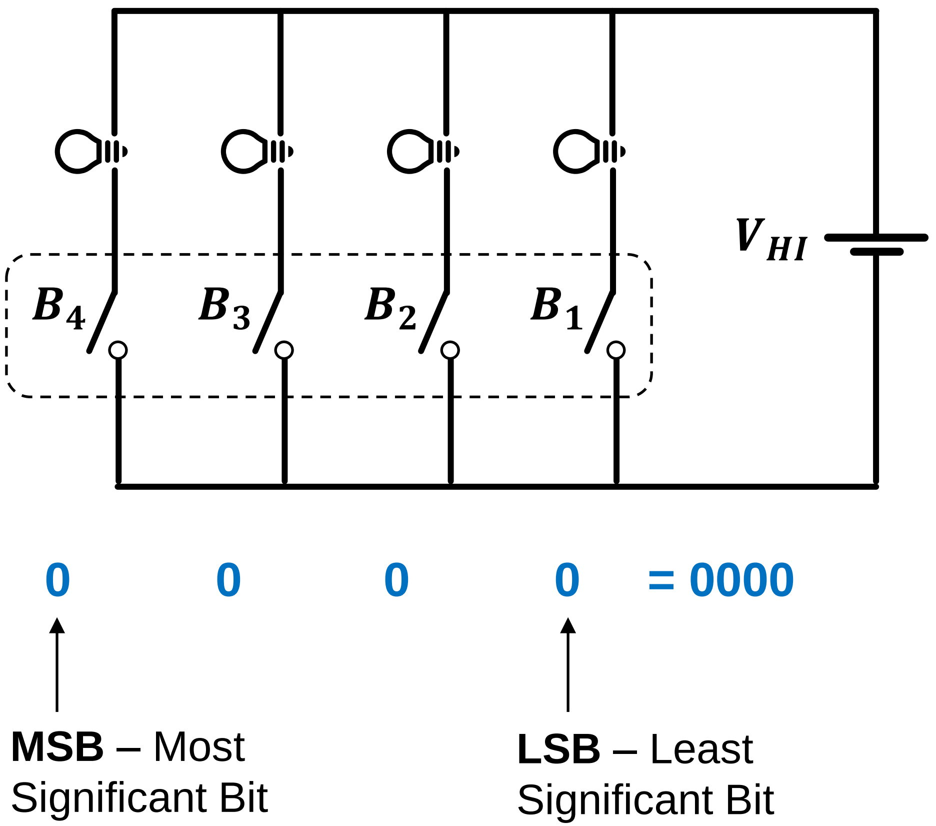
Logic – TRUE/FALSE

We said that **301** (weight of the FS21 in kg) is represented in binary as

0 0 0 1 0 0 1 0 1 1 0 1

How is this actually done in reality?

Two voltage levels – **High & Low**



Information in Electronics

We use **binary number system** in logical circuits in electronics

This aligns with computer/software engineering – binary system used

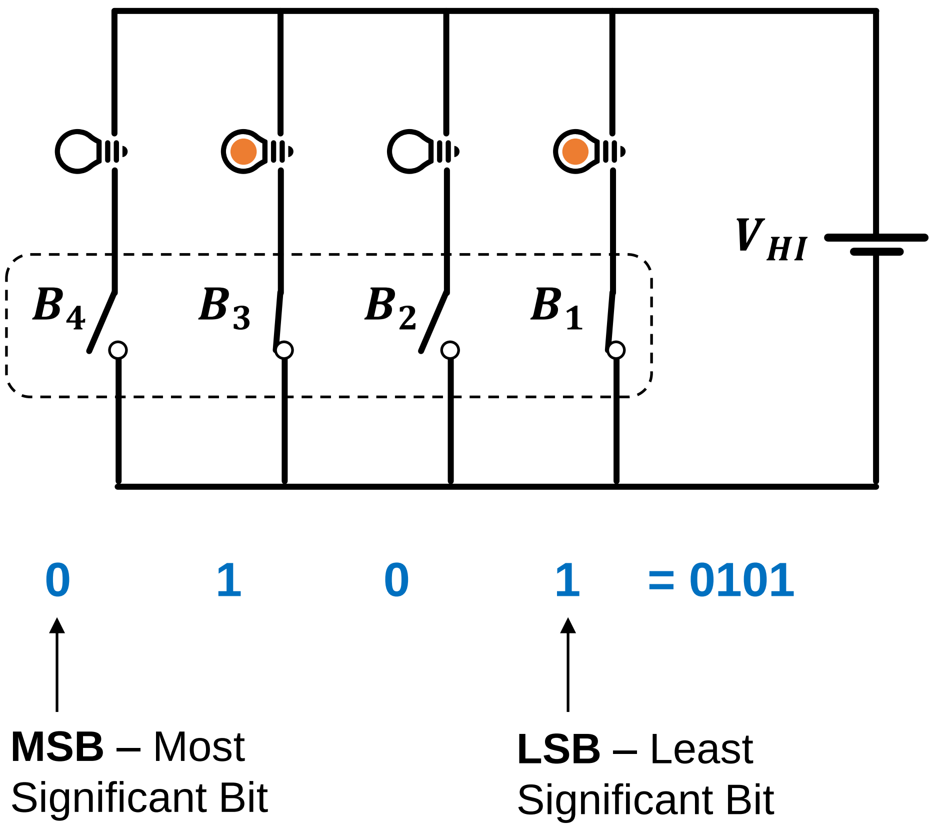
Logic – TRUE/FALSE

We said that **301** (weight of the FS21 in kg) is represented in binary as

0 0 0 1 0 0 1 0 1 1 0 1

How is this actually done in reality?

Two voltage levels – **High & Low**



Information in Electronics

We use **binary number system** in logical circuits in electronics

This aligns with computer/software engineering – binary system used

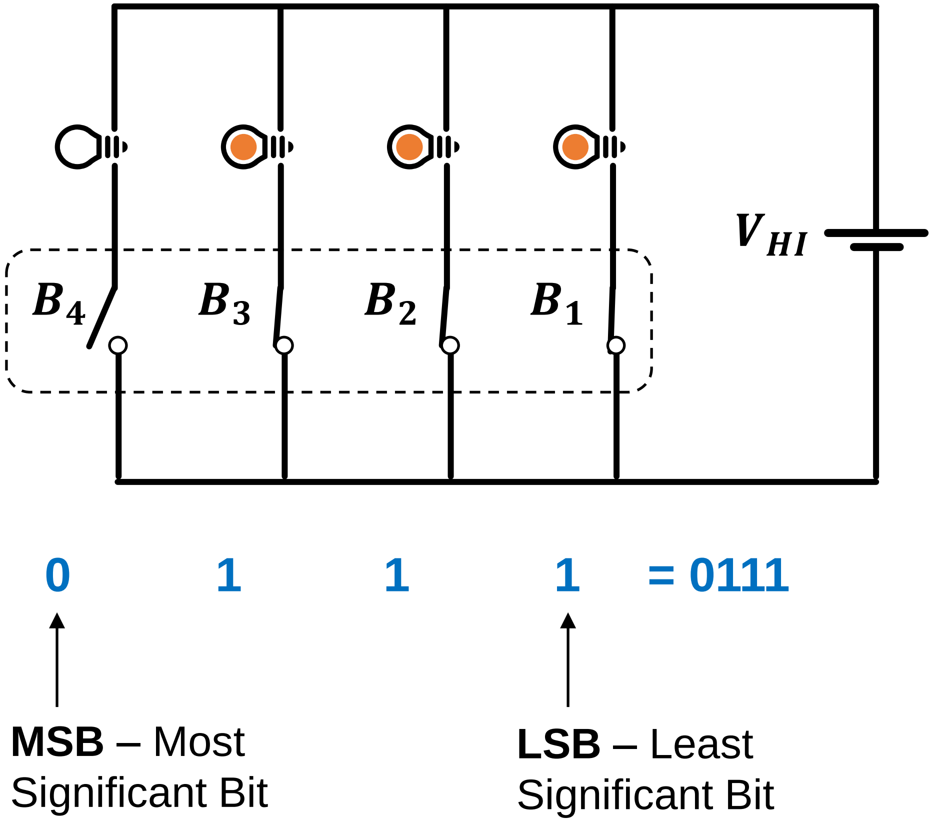
Logic – TRUE/FALSE

We said that **301** (weight of the FS21 in kg) is represented in binary as

0 0 0 1 0 0 1 0 1 1 0 1

How is this actually done in reality?

Two voltage levels – **High & Low**



Information in Electronics

We use **binary number system** in logical circuits in electronics

This aligns with computer/software engineering – binary system used

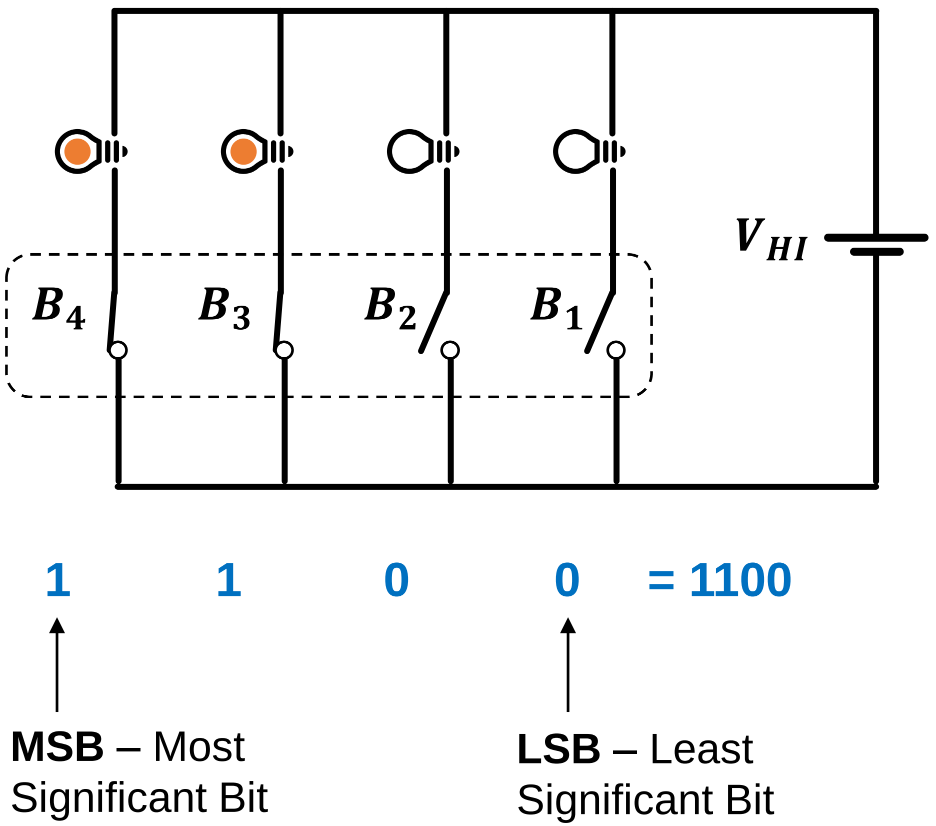
Logic – TRUE/FALSE

We said that **301** (weight of the FS21 in kg) is represented in binary as

0 0 0 1 0 0 1 0 1 1 0 1

How is this actually done in reality?

Two voltage levels – **High & Low**



How to Add/Subtract Binary Numbers?

Just the same way you do for decimal numbers!

Decimal	Binary
1	1 1 1 1 1
124	0111 1100
+229	+1110 0101
353	**1 0110 0001**
124	0111 1100
- 47	+0010 1111
77	**0100 1101**

We don't normal do **multiplication** and **division** operations on binary numbers

We shall study **Binary Algebra** later

4-bit Binary Number Range

Decimal	B_4	B_2	B_2	B_1	Binary
0	0	0	0	0	0000
1	0	0	0	1	0001
2	0	0	1	0	0010
3	0	0	1	1	0011
4	0	1	0	0	0100
5	0	1	0	1	0101
6	0	1	1	0	0110
7	0	1	1	1	0111
8	1	0	0	0	1000
9	1	0	0	1	1001
10	1	0	1	0	1010
11	1	0	1	1	1011
12	1	1	0	0	1100
13	1	1	0	1	1101
14	1	1	1	0	1110
15	1	1	1	1	1111

We would call this a 4-bit binary number – it is made of 4 bits

Maximum number we can count up to for a binary number is given by $\mathbf{2^n - 1}$

$$1\ byte\ =\ 8\ bits$$

Modern computers use **32-bit** or **64-bit** numbers in its operating system

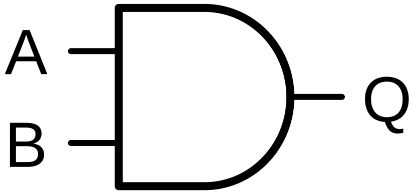
Remember the numeric data types you learnt in MATLAB last year?

- **Single** – 4 bytes
- **Double** – 8 bytes
- **Int8** – 1 byte

Logic Gates

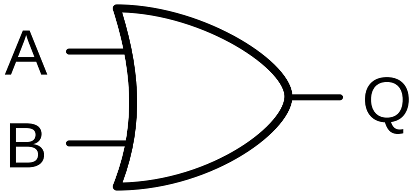
Truth Table

AND



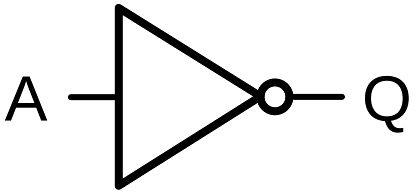
A	B	Q	Remark
0	0	**0**	HI if all inputs are HI
0	1	**0**	
1	0	**0**	
1	1	**1**	

OR



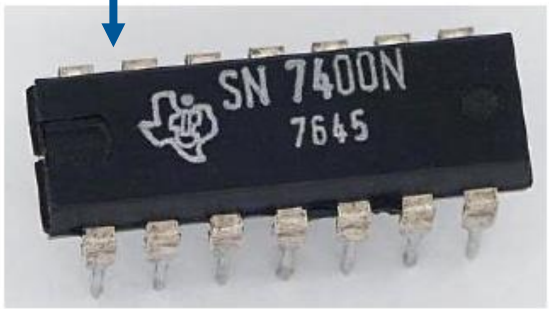
A	B	Q	Remark
0	0	**0**	HI if any input is HI
0	1	**1**	
1	0	**1**	
1	1	**1**	

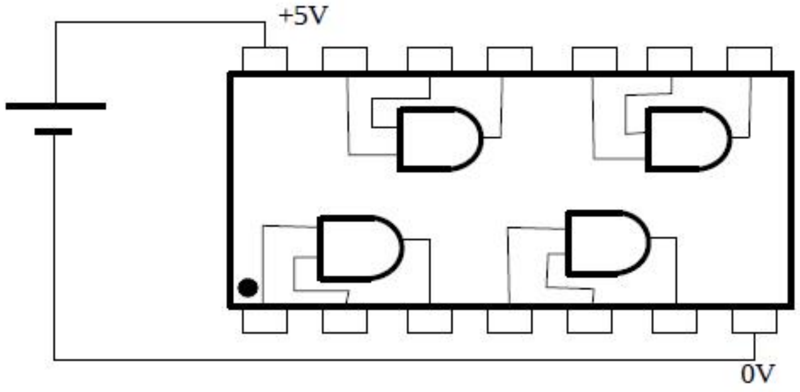
NOT



A	Q	Remark
0	**1**	Bit inversion
1	**0**	

This is an Integrated Circuit, or IC!

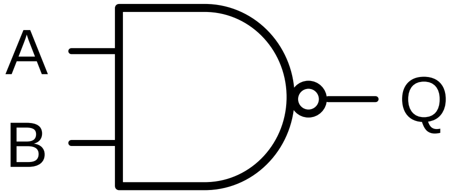




Logic Gates

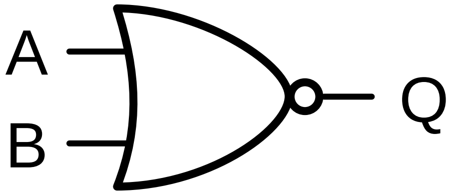
Truth Table

NAND



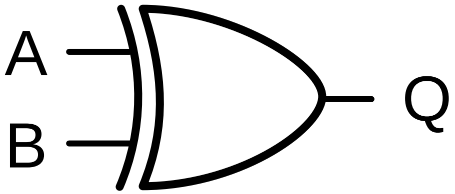
A	B	Q	Remark
0	0	**1**	LO if all inputs are HI
0	1	**1**	
1	0	**1**	
1	1	**0**	

NOR



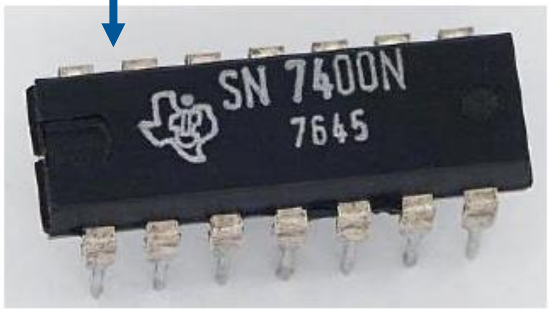
A	B	Q	Remark
0	0	**1**	LO if any input is HI
0	1	**0**	
1	0	**0**	
1	1	**0**	

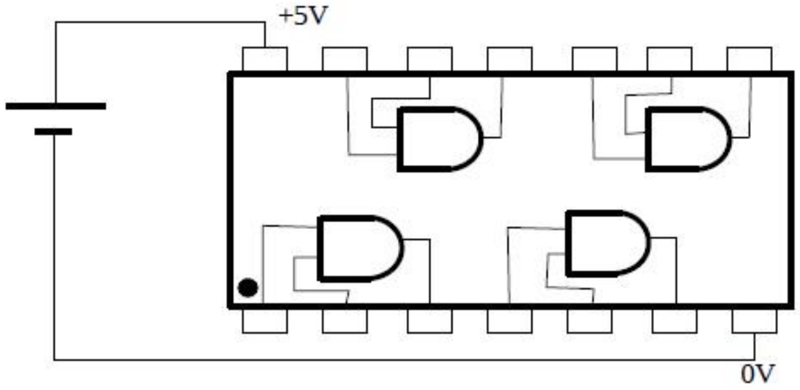
XOR



A	B	Q	Remark
0	0	**0**	HI if at least one input is HI and one is LO
0	1	**1**	
1	0	**1**	
1	1	**0**	

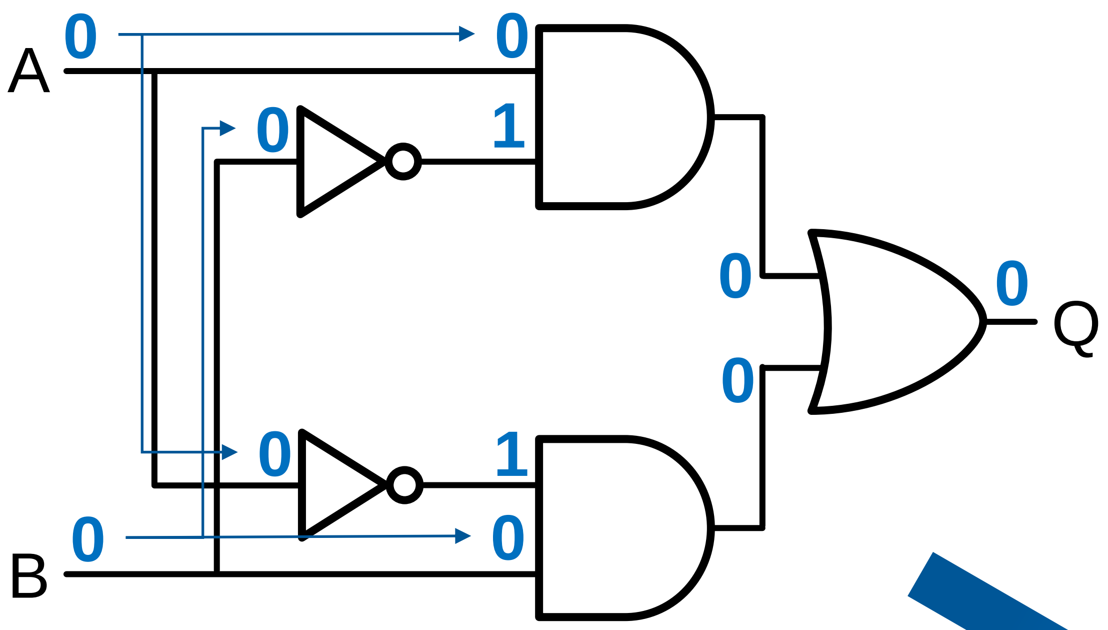
This is an Integrated Circuit, or IC!





Don't need to study this for exam

Example 3

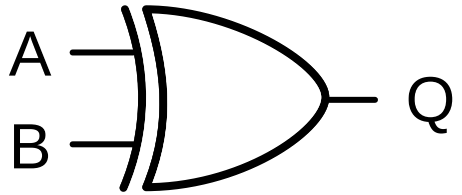


Total inputs = 2

Total combinations possible = $2^n = 4$

4 rows in truth table

A	B	Q	Remark
0	0	**0**	This is XOR gate
0	1	**1**	
1	0	**1**	
1	1	**0**	



~~**Step 1 – Identify how many inputs there are**~~

~~**Step 2 – Draw a truth table with as many number of rows as possible combinations of input bits**~~

~~**Step 3 – Try each input combination in the logic gate**~~

~~**Step 4 – Propagate the “logic” all the way to output**~~

~~**Step 5 – Fill the truth table row by row**~~

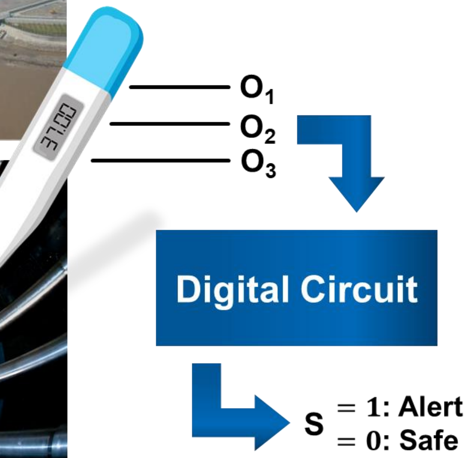
Example 4

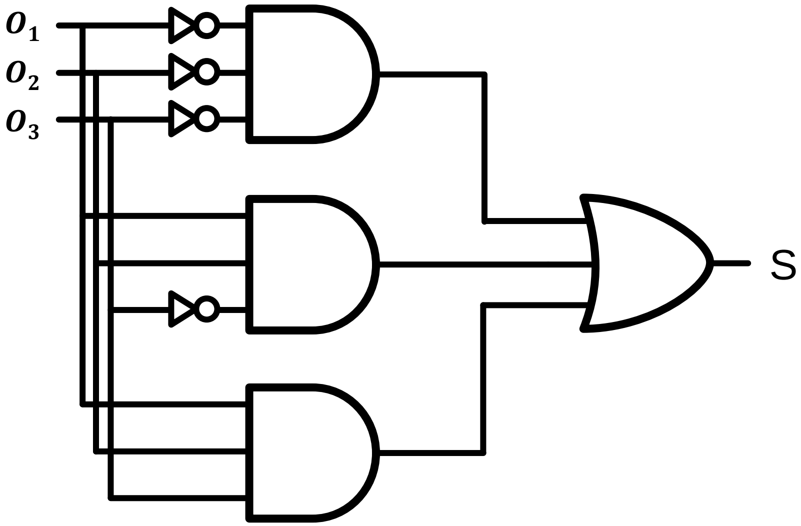
- Imagine you are designing a circuit to monitor a digital thermometer embedded in a nuclear reactor
- You want to automatically shut off the reactor when the cooling fluid rises above 50°C
- It would also be bad if the coolant froze – shut down the reactor!
- Thermometer gives a 3-bit binary output in 10°C steps –
  - $2^3 = \mathbf{8}$ **levels**
  - Count from 0 to $2^3 - 1 = \mathbf{7}$
  - **0°C** to **80°C** range of output

S=1 (as we said solving for HI) if:

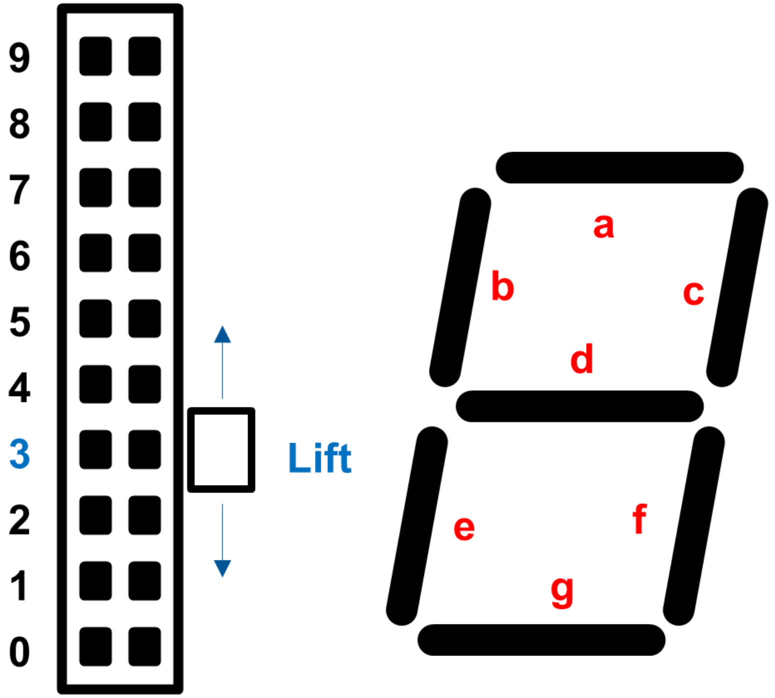
- $O_1 = 0$ AND $O_2 = 0$ AND $O_3 = 0$ **OR**
- $O_1 = 1$ AND $O_2 = 1$ AND $O_3 = 0$ **OR**
- $O_1 = 1$ AND $O_2 = 1$ AND $O_3 = 1$

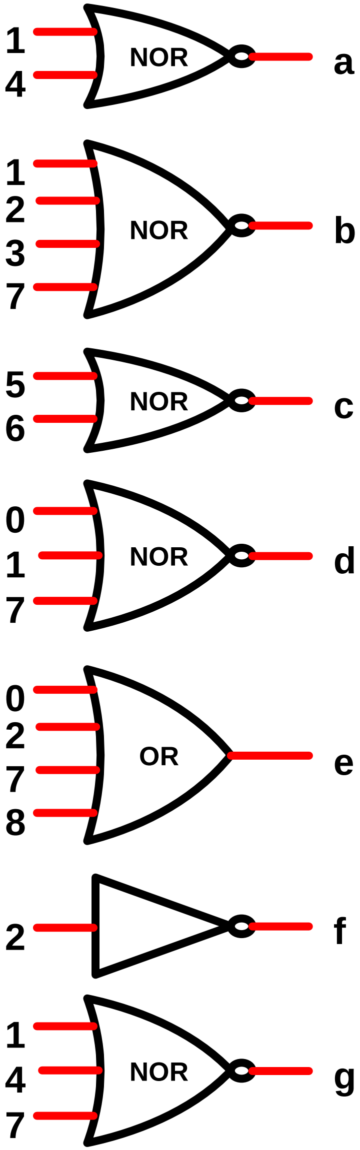
O_1	O_2	O_3	Dec	Temp	S
0	0	0	0	0°C	1
0	0	1	1	10°C	0
0	1	0	2	20°C	0
0	1	1	3	30°C	0
1	0	0	4	40°C	0
1	0	1	5	50°C	0
1	1	0	6	60°C	1
1	1	1	7	70°C	1





Example 5





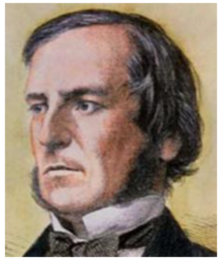
	a	b	c	d	e	f	g
0	1	1	1	0	1	1	1
1	0	0	1	0	0	1	0
2	1	0	1	1	1	0	1
3	1	0	1	1	0	1	1
4	0	1	1	1	0	1	0
5	1	1	0	1	0	1	1
6	1	1	0	1	1	1	1
7	1	0	1	0	0	1	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

Boolean Algebra

Boolean Algebra is like regular algebra – but instead of numbers, it operates with logical variables true and false, also denoted by 1 and 0 respectively

This is used to simplify logical circuits mathematically, i.e., instead of solving a complicated digital logic circuit via propagation of signal, you can simplify it mathematically using certain laws

There are three operators:



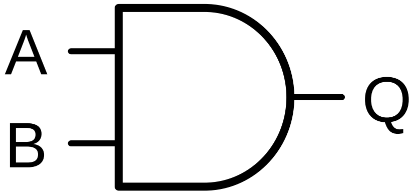
George Boole, self-taught English mathematician wrote the book *The Laws of Thoughts* (1854) and introduced Boolean Algebra

Operation	Symbol	Logic Gate
Conjunction	∧ or .	AND
Disjunction	∨ or +	OR
Negation	¬ or ‘ or overhead bar	NOT

Boolean Algebra

Truth Table

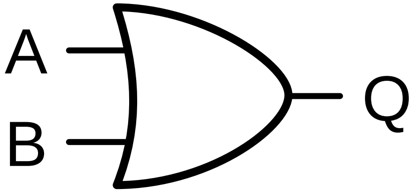
AND



A	B	Q	Remark
0	0	**0**	HI if all inputs are HI
0	1	**0**	
1	0	**0**	
1	1	**1**	

$$Q = A.B = A \wedge B$$

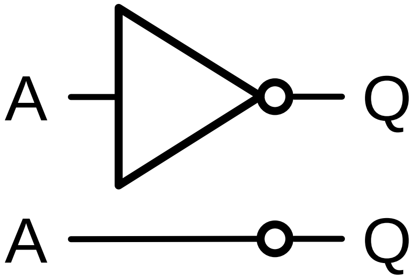
OR



A	B	Q	Remark
0	0	**0**	HI if any input is HI
0	1	**1**	
1	0	**1**	
1	1	**1**	

$$Q = A + B = A \vee B$$

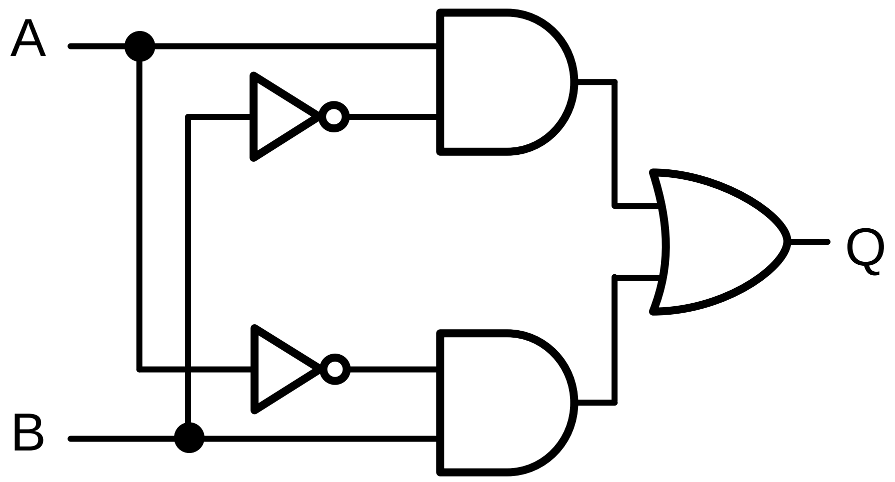
NOT



A	Q	Remark
0	**1**	Bit inversion
1	**0**	

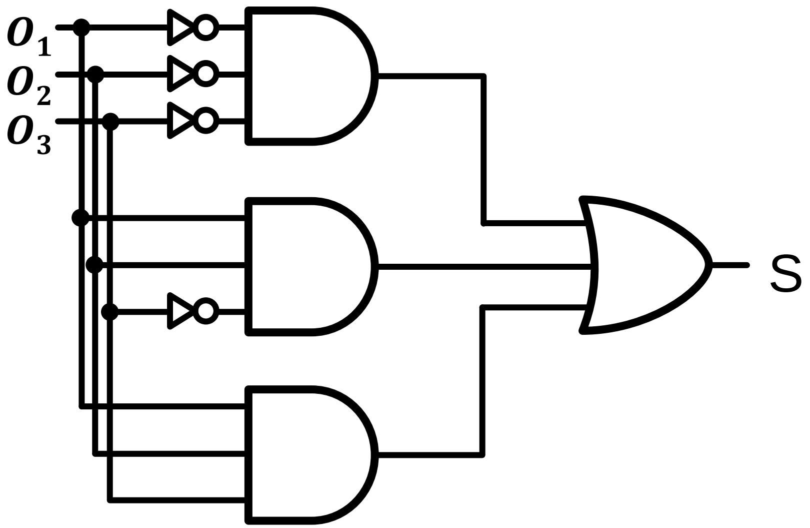
$$Q = A' = \neg A = \bar{A}$$

Boolean Algebra



$$Q = A.B' + A'.B$$

$$Q = A.\overline{B} + \overline{A}.B$$



$$S = O_1'.O_2'.O_3' + O_1.O_2.O_3' + O_1.O_2.O_3$$

Boolean Algebra

Boolean Law	Example 1	Example 2	Example 3
Annulment	$A.0 = 0$	$A + 1 = 1$	
Identity	$A.1 = A$	$A + 0 = A$	
Idempotent	$A.A = A$	$A + A = A$	
Complement	$A.A' = 0$	$A + A' = 1$	
Commutative	$A.B = B.A$	$A + B = B + A$	
Associative	$A.(B.C) = (A.B).C$	$A + (B + C) = (A + B) + C$	
Distributive	$A.(B + C) = A.B + B.C$	$A + (B.C) = (A + B).(A + C)$	
Absorption	$A.(A + B) = A$	$A + (A.B) = A$	$A + A'.B = A + B$
Involution	$(A')' = A$		
De Morgan's	$(A + B)' = A'.B'$	$(A.B)' = A' + B'$	

Summary

- DC Motor
  - **Revision** of all motors studied so far – Induction, Stepper
  - **Operation** of a **Simple DC Motor**
  - **Why** use a DC Motor?
- **Boolean Algebra**
  - **Revision of Digital Electronics**
  - **Addition** (OR), **multiplication** (AND), **complement** (NOT)
  - **Laws**

Attendance

UP|PHY|B1-MMME2051EMD