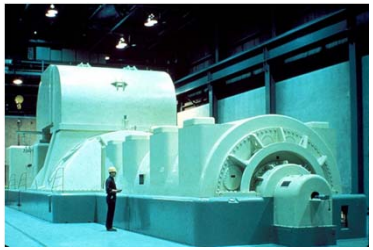
EG1108: Electrical Engineering

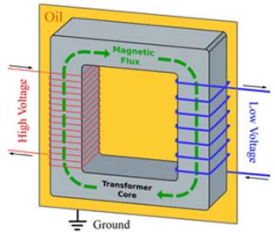
Part 2: Application Examples

Ben M. Chen

Professor of Electrical and Computer Engineering
National University of Singapore
Office: E4-06-08 Phone: 6516-2289
Email: bmchen@nus.edu.sg ~ http://www.bmchen.net

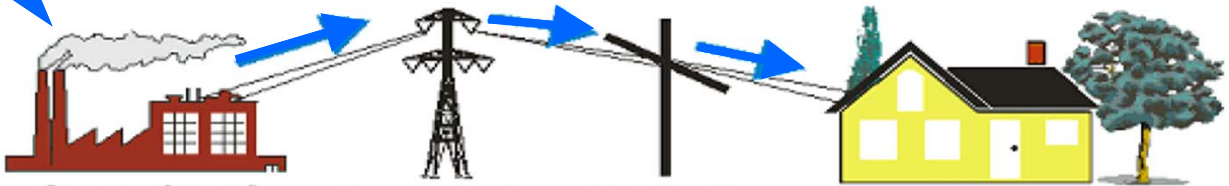
What to be covered in this 2nd part?





Magnetic Circuits and Transformers

Electric Generator



EG1108 Part 2

DC Motor

DC Power Supply

Digital Logic Circuits

Course Outline

1. **Introduction to Electrical Engineering**

   Introduction to some practical electrical engineering examples.

2. **Magnetic Circuits and Transformers**

   Principles of mutual inductance and transformers.

3. **DC Power Supply**

   Diode characteristics. Rectifier circuits. Bridge rectifiers.

4. **Brief Introduction to DC Motors and Electric Generators**

   Basic principles of operation of DC motors and electric generators.

5. **Digital Logic Circuits**

   Digital logic. Boolean algebra. Combinational logic. Logic circuit design.

Textbook & References

1. Basic Circuit Analysis for Electrical Engineering

   By C. C. Ko and B. M. Chen, 2nd Ed., Prentice Hall, 1998

2. Electrical Engineering

   By S. Elangovan and D. Srivivasan, Prentice Hall, 2005

3. Principles of Electrical Engineering

   By P. Z. Peebles, Jr. and T. A. Giuma, McGraw Hill, 1991

4. Electrical Engineering Principles and Applications

   By A. R. Hambley, Prentice Hall, 2011

5. Principles and Applications of Electrical Engineering

   By G. Rizzoni, McGraw Hill, 2007

Lectures

☺ **Attendance is essential** ☹

Any question at any time during the lecture is welcome!

Tutorials

As in Part 1, there will be **4 tutorial sets** for this part.

- **Week 8:** Free tutoring session. You should take the opportunity to clarify any doubts and problems you have.
- **Week 9:** Free tutoring session. You should take the opportunity to clarify any doubts and problems you have.
- **Week 10:** Tutorial 5
- **Week 11:** Tutorial 6
- **Week 12:** Tutorial 7
- **Week 13:** Tutorial 8

You are free to discuss among your classmates and/or to consult me or your tutors if you have met any difficulties in attempting the tutorial problems.

Office Hours

4:30–7:00pm

Every Thursday up to Reading Week

@ My Office ☺

Introduction to Electrical Engineering

Disciplines and Examples

Electrical Engineering Disciplines…

1. **Power Systems**

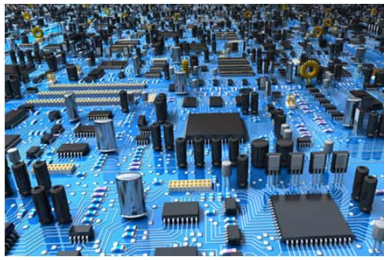
   The oldest specialty within the field deals with the generation and transmission of electricity from one location to another.

2. **Electric Machinery**

   It deals with conversion of energy to and from electrical form, and studies the design and operation of devices such as motors and generators.

3. **Electronics**

   This covers study and application of materials, devices and circuits used in amplifying and switching electrical signals.



4. **Computer Systems**

   These process and store information in digital form. It includes design and development of computer hardware systems and the computer programs (software) that control them.

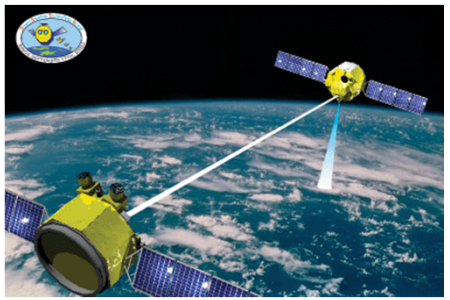
Electrical Engineering Disciplines…

5. Control Systems

These are a very important class of systems that gather information with sensors and use electrical energy to control a physical process.

6. Communications Systems

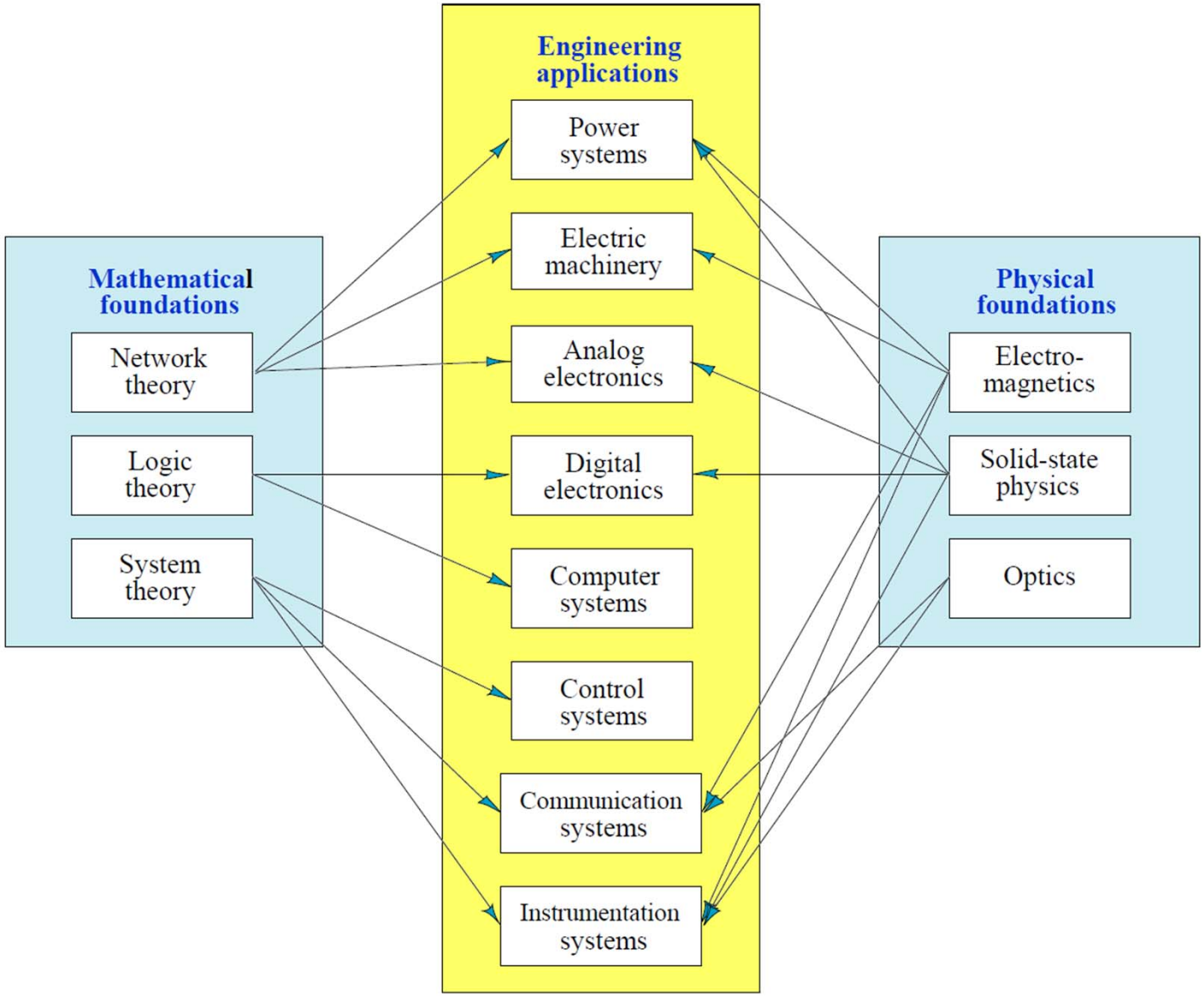
These systems transport information in electrical form by encoding information on an electrical signal. Some examples of such systems include cellular phone, radio, satellite television, and the Internet.



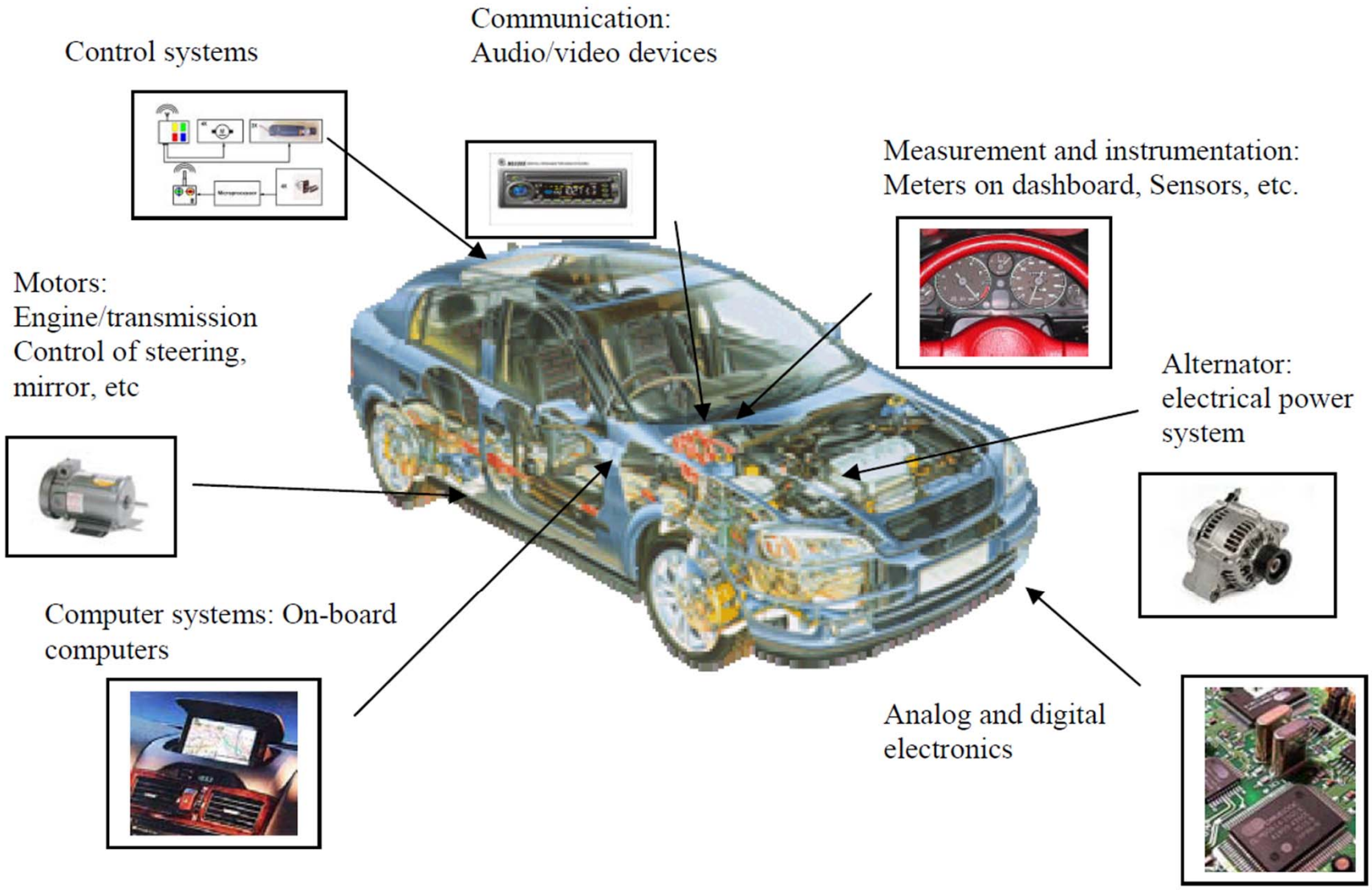
7. Instrumentation Systems

They include sensors and instruments commonly used in engineering systems. Modern instrumentation systems typically use electronic amplifiers and converters.

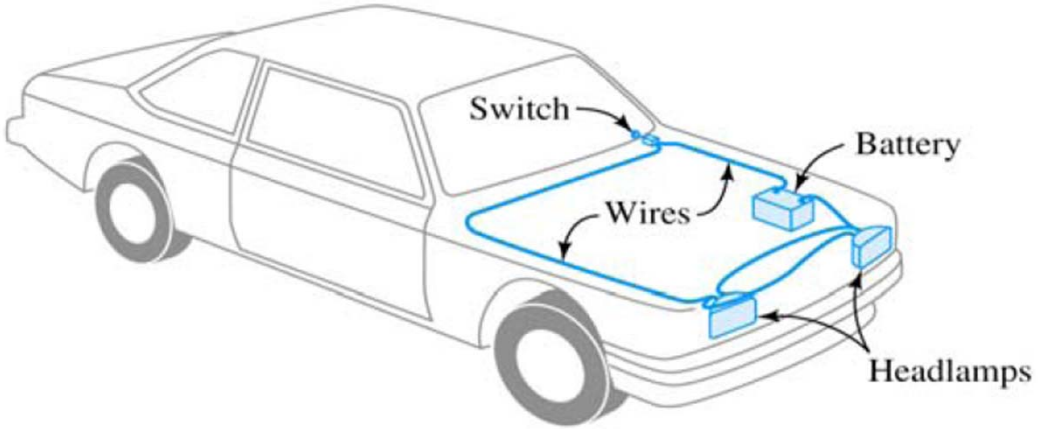
Electrical Engineering Disciplines…



Example: A Passenger Automobile…



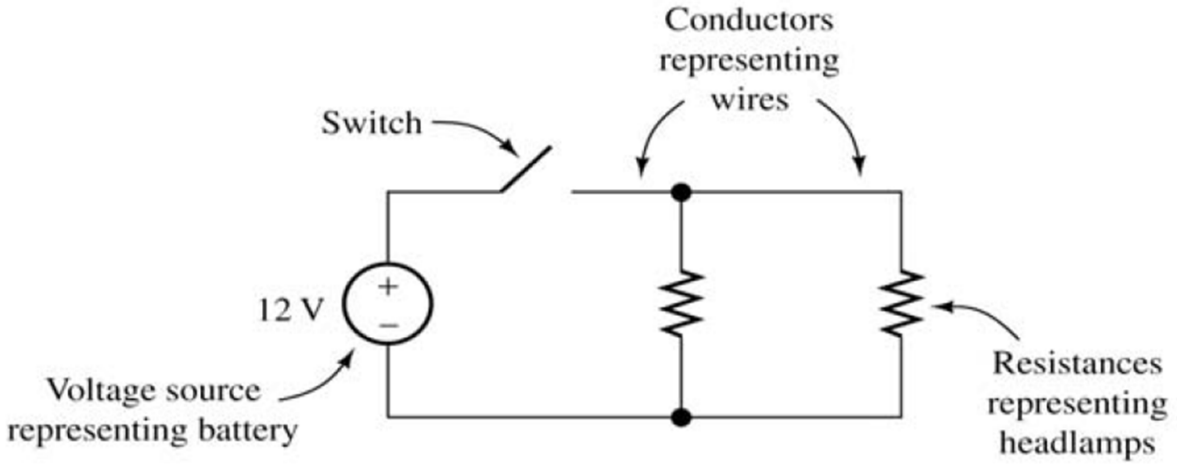
An Electric Circuit in an Automobile



(a) Physical configuration

Headlight System

Electric Circuit

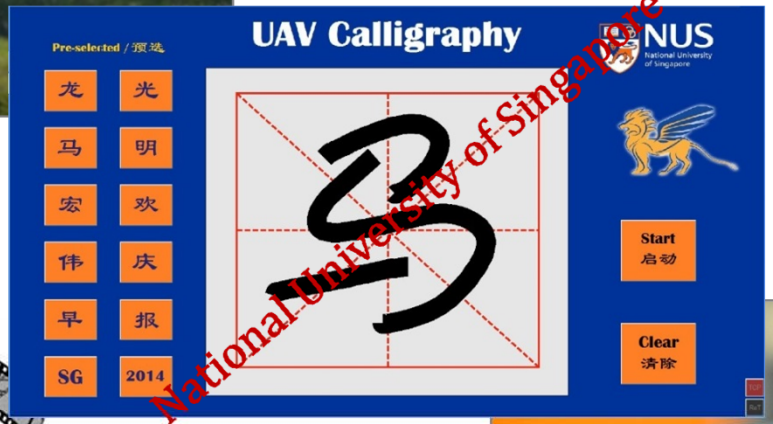


(b) Circuit diagram

Example: Unmanned Helicopters…

Actual Flight Tests

mechanical engineering?



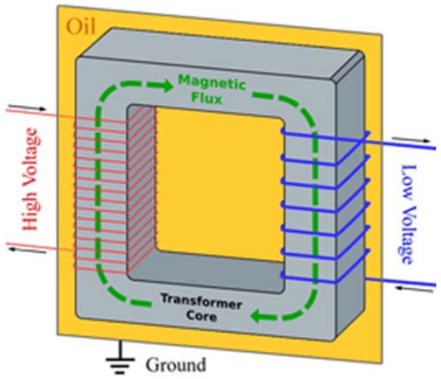
engineering and arts?

material engineering?

Magnetic Circuits & Transformers

Basic Principles

Examples of Transformers

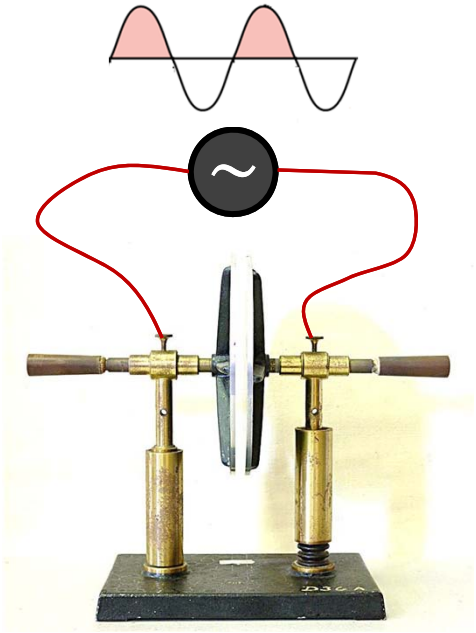


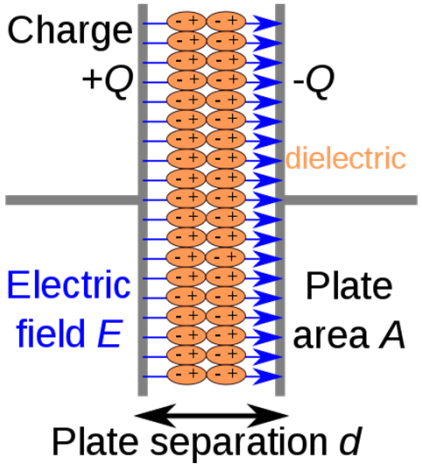
Magnetic Field and Material

In electrostatic, an electric field is formed by static charges. It is described in terms of the electric field intensity. The **permittivity** is a measure of how easy it is for the field to be established in a medium given the same charges.

The permittivity of free space or vacuum permittivity or electric constant is given by

$$\varepsilon_0 = 8.8542 \times 10^{-12} \text{ F(arad)/m}$$



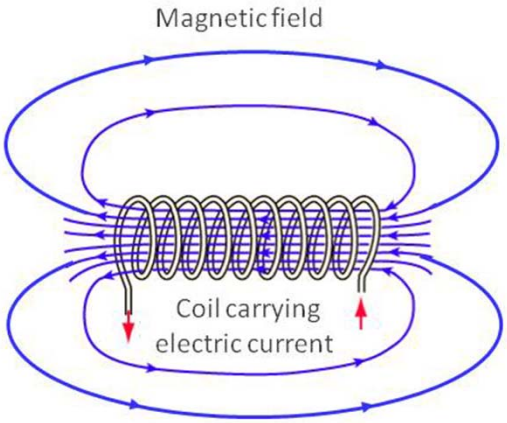


If the insulator is free space (or a piece of paper), the resulting capacitance

$$C = \frac{Q}{V} = \varepsilon_0 \cdot \frac{A}{d}$$

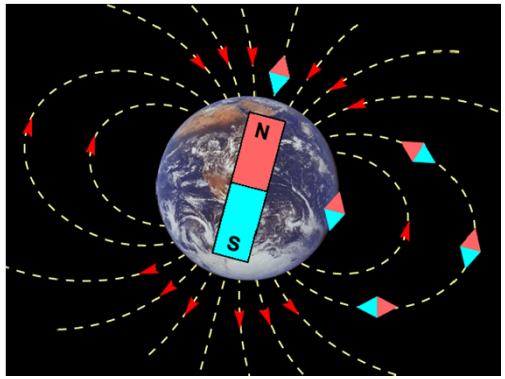
Permeability

Similarly, a **magnetic field** is formed by moving charges or electric currents. It is described in terms of the magnetic **flux density** *B*, which has a unit of tesla (T) = (N/A)m. The **permeability** μ is a measure of how easy it is for a magnetic field to be formed in a material. The higher the μ, the greater the *B* for the same currents.



In free space, μ is $\mu_0 = 4\pi \times 10^{-7}\ \mathrm{H/m}$ and the **relative permeability** μ_r is defined as

$$\mu_r = \frac{\mu}{\mu_0}$$

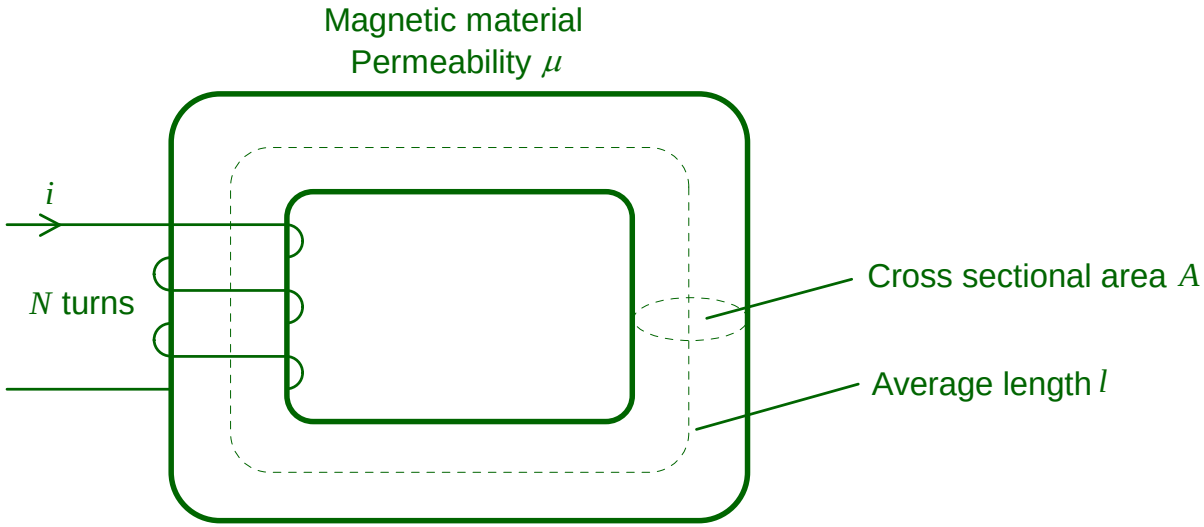


Most "non-magnetic" materials such as air and wood have a $\mu_r \approx 1$. However, "magnetic materials" such as iron may have $\mu_r \approx 5000$.

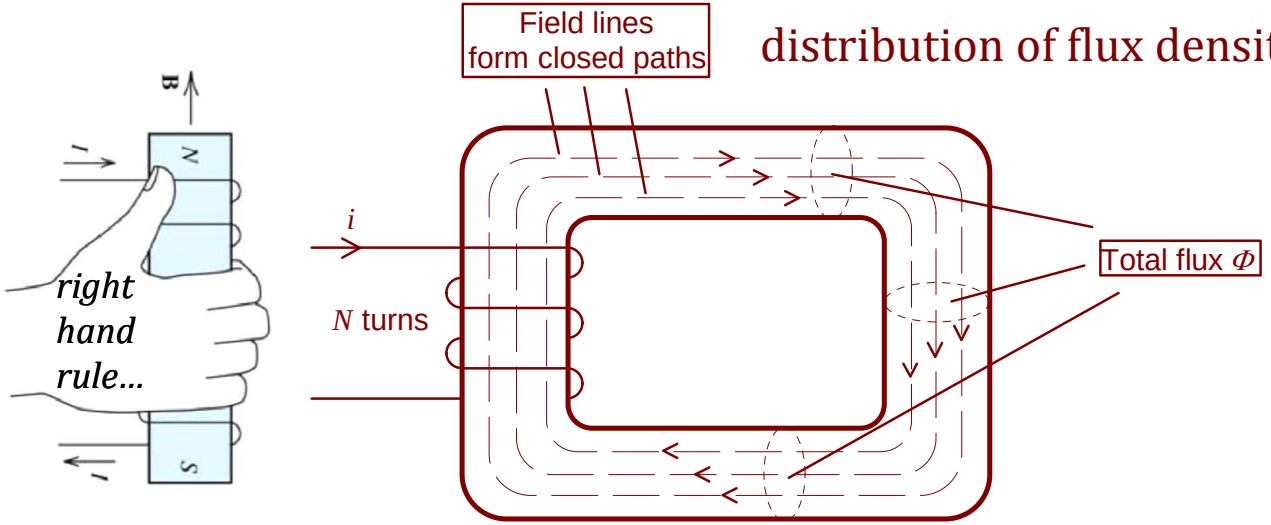
pure iron	$\mu_r = 5,000$
silicon GO steel	$\mu_r = 40,000$
supermalloy	$\mu_r = 1,000,000$

Magnetic Flux

Consider the following magnetic system:



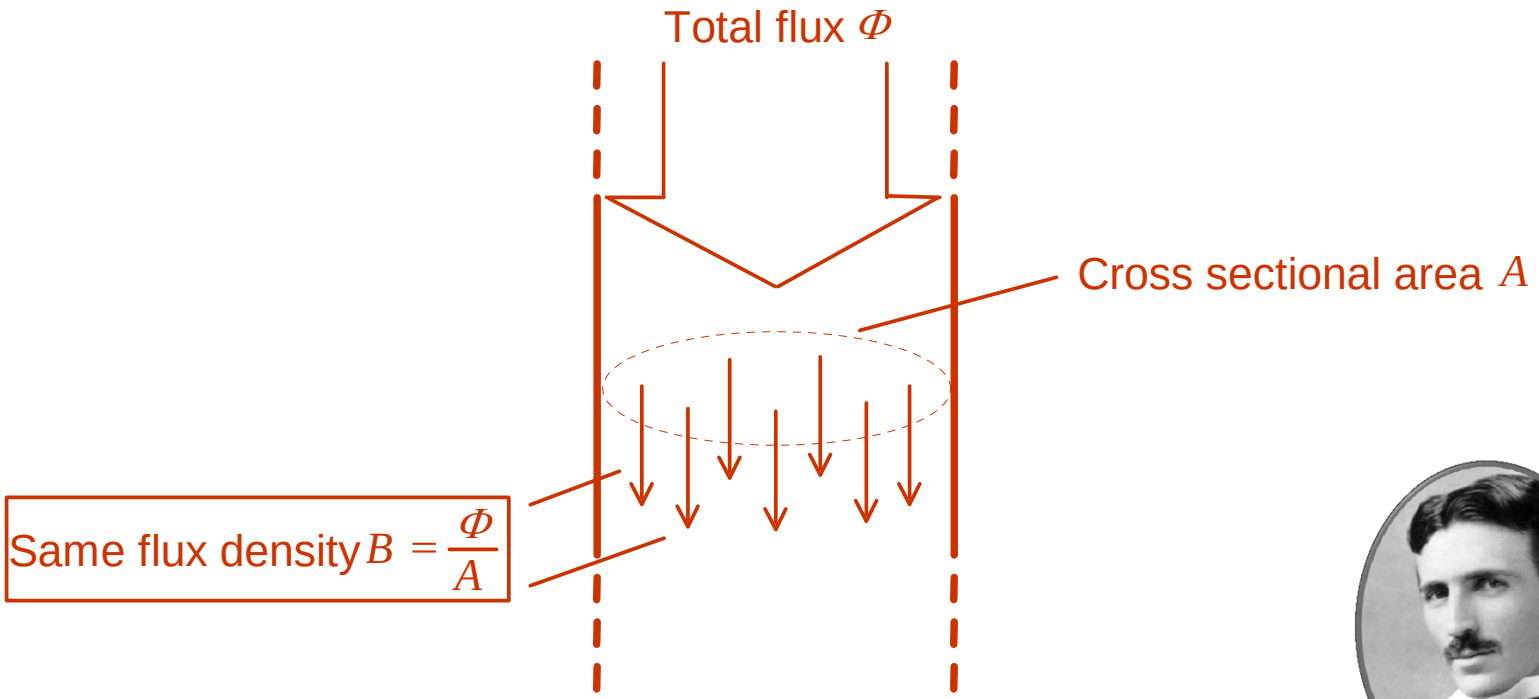
If μ is large, almost the entire magnetic field will be concentrated inside the material and there will be no flux leakage.



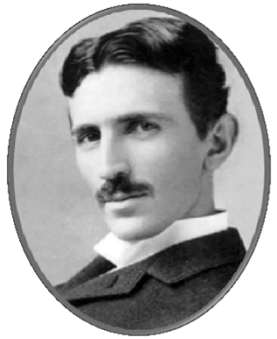
distribution of flux density: Since the field lines form closed paths and there is no leakage, the total flux Φ passing through any cross section of the material is the same.

Flux Density

Assuming the flux to be uniformly distributed so that the flux density B have the same value over the entire cross sectional area A:



$$B = \frac{\Phi}{A} \qquad \text{with units tesla (T)} = \frac{\text{weber (Wb)}}{\text{m}^2}$$

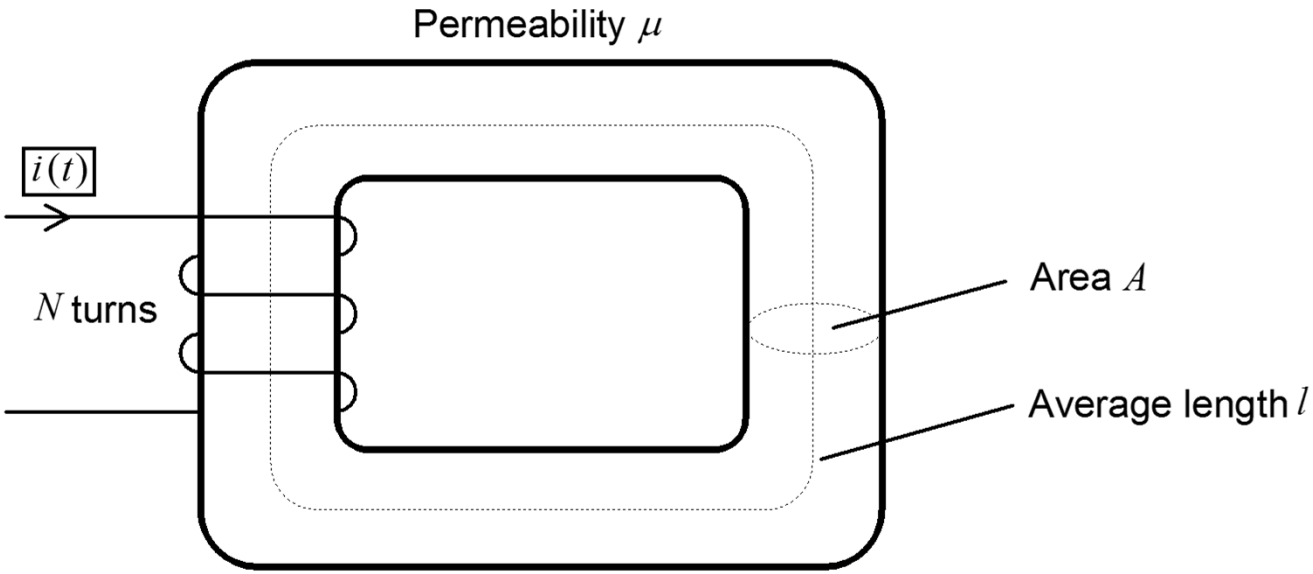


Nikola Tesla
Serbian American
1856–1943

Ampere's Law

The values of Φ or B can be calculated using **Ampere's law**:

Line integral of $\frac{B}{\mu}$ along any closed path = current enclosed by path



Andre-Marie Ampere
French
1775–1836

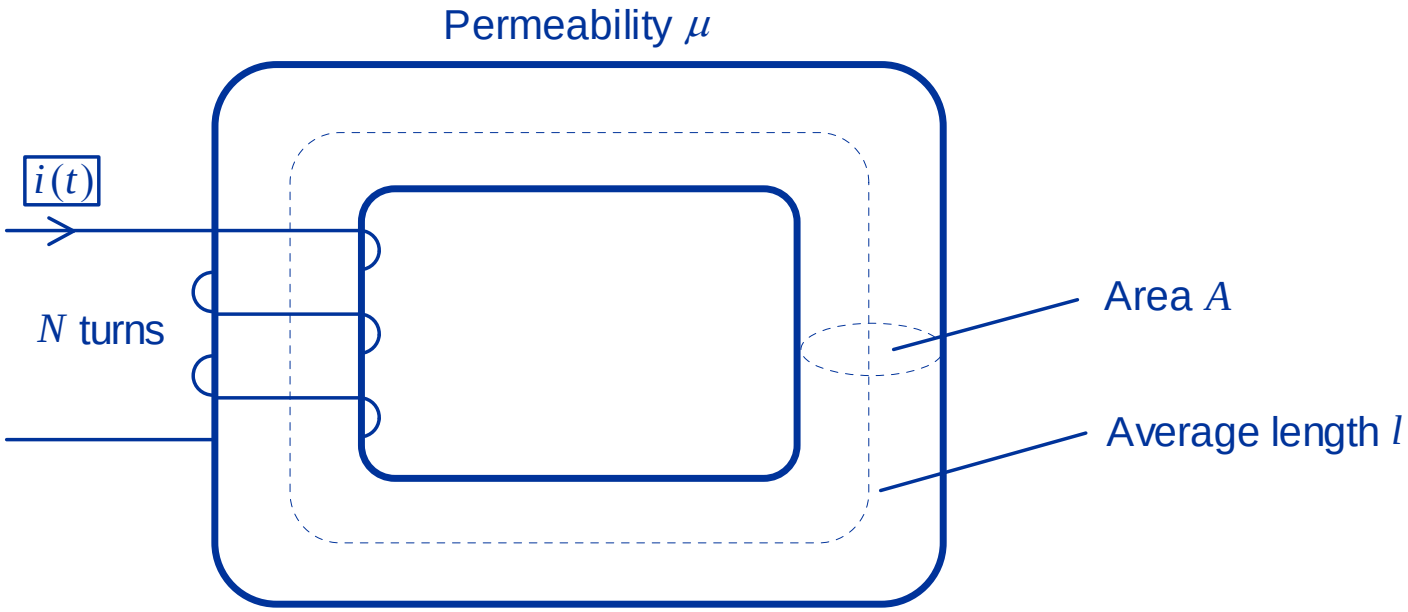
Line integral of $\frac{B}{\mu}$ along dotted path $= \frac{Bl}{\mu} = \left(\frac{l}{\mu A}\right)\Phi$

$$Ni = \left(\frac{l}{\mu A}\right)\Phi$$

= Current enclosed by dotted path $N\,i$

Inductance

Consider the magnetic circuit:

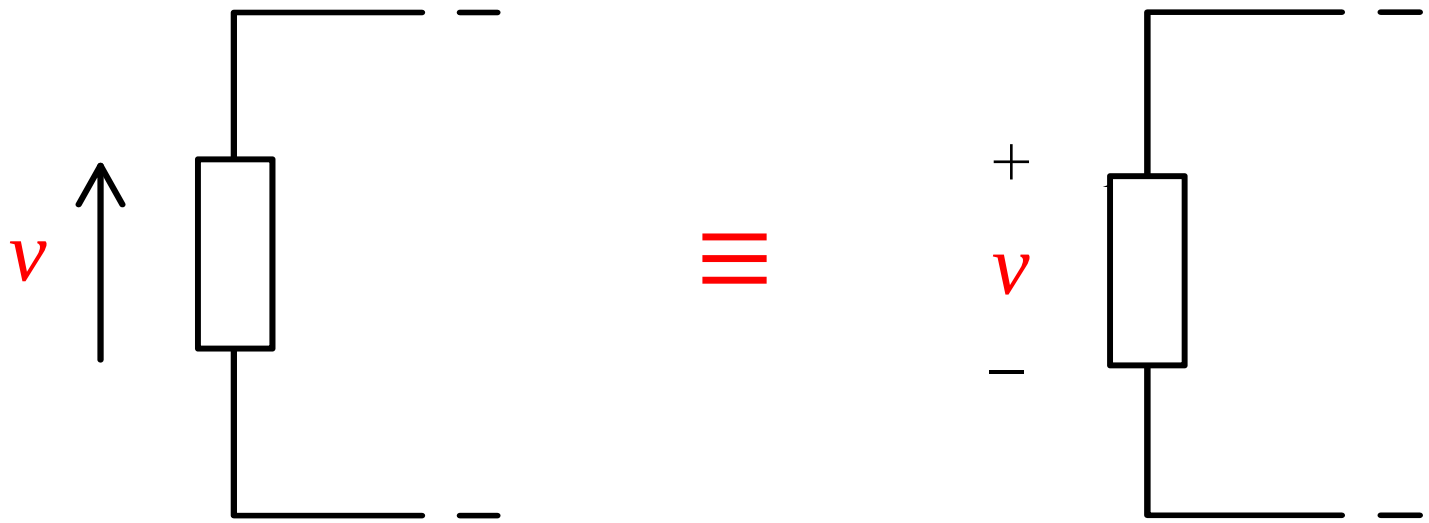


Assuming no flux leakage and uniform flux distribution, the reluctance and the flux linking or enclosed by the winding are

$$\Re = \frac{l}{\mu A} \quad \text{and} \quad \Phi(t) = \frac{Ni(t)}{\Re}$$

Side Note – Notation for a branch voltage

For this second part, the following symbols are identical



The arrow points to the 'positive' potential of the circuit element even though it can be physically negative depending on the actual value of v.

Faraday's Law of Inductance

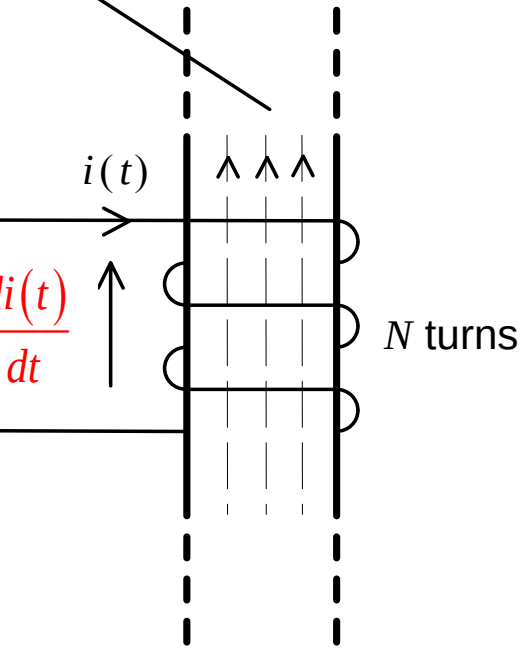
From **Faraday's law of induction**, a voltage will be induced in the winding if the flux linking the winding changes as a function of time. This induced voltage, called the **back emf (electromotive force)** will attempt to oppose the change and is given by

$$v(t) = N\frac{d\Phi(t)}{dt} = \frac{N^2}{\Re}\cdot\frac{di(t)}{dt}$$

Michael Faraday
English
1791–1867

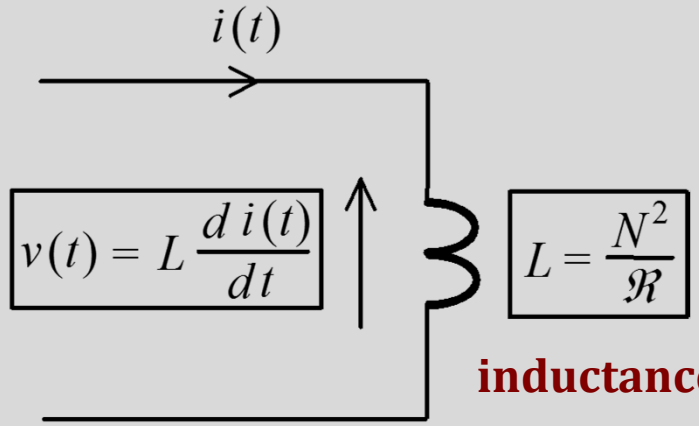
Flux linking winding

$$\Phi(t) = \frac{Ni(t)}{\Re}$$



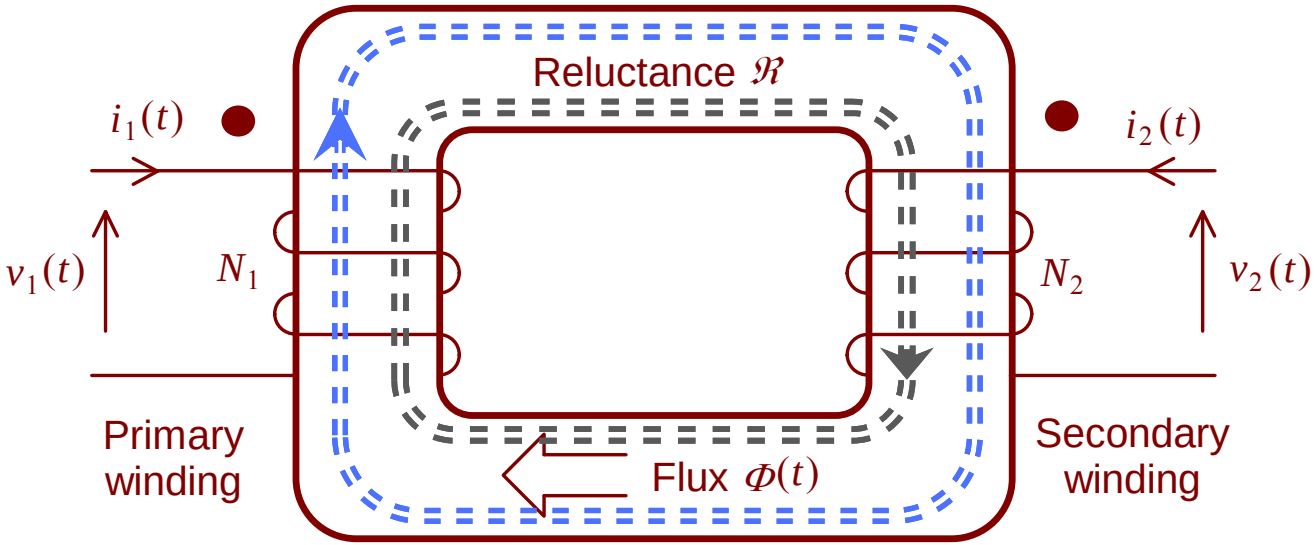
$$v(t) = N\frac{d\Phi(t)}{dt} = \frac{N^2}{\Re}\frac{di(t)}{dt}$$

An equivalent inductor:



Mutual Inductor

The two **dots** are associated with the directions of the windings. The fields produced by the two windings will be constructive if the currents going into the dots have the same sign.



$$\text{Total flux} = \Phi(t) = \frac{N_1 i_1(t) + N_2 i_2(t)}{\mathfrak{R}}$$

$$v_1(t) = N_1 \frac{d\Phi(t)}{dt} = \left(\frac{N_1^2}{\mathfrak{R}}\right)\frac{di_1(t)}{dt} + \left(\frac{N_1 N_2}{\mathfrak{R}}\right)\frac{di_2(t)}{dt}$$

$$\frac{N_1 N_2}{\mathfrak{R}} = \sqrt{\frac{N_1^2}{\mathfrak{R}} \cdot \frac{N_2^2}{\mathfrak{R}}}$$

$$v_2(t) = N_2 \frac{d\Phi(t)}{dt} = \left(\frac{N_1 N_2}{\mathfrak{R}}\right)\frac{di_1(t)}{dt} + \left(\frac{N_2^2}{\mathfrak{R}}\right)\frac{di_2(t)}{dt}$$

Mutual Inductance

Inductance of primary winding on its own $= L_1 = \dfrac{N_1^{\,2}}{\mathfrak{R}}$

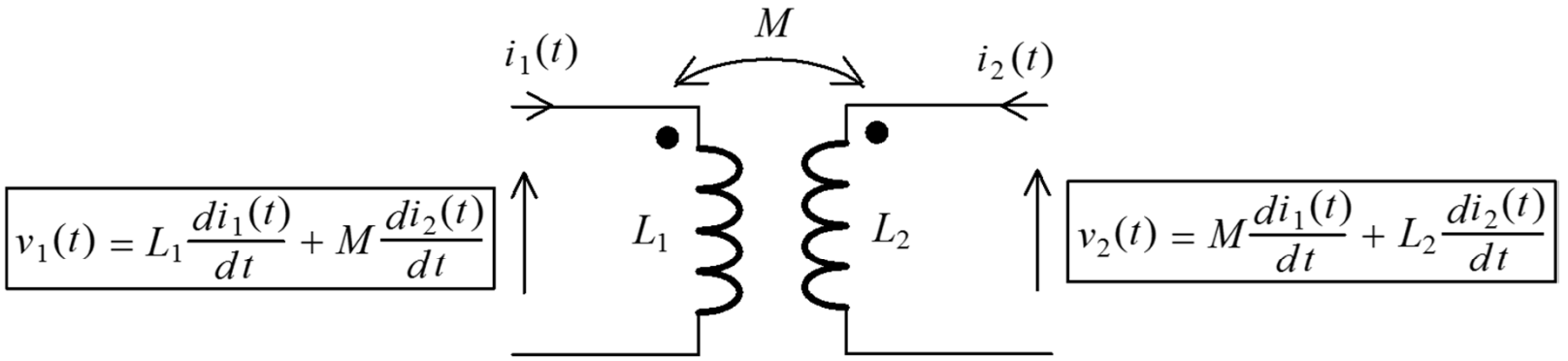
Inductance of secondary winding on its own $= L_2 = \dfrac{N_2^{\,2}}{\mathfrak{R}}$

$$v_1(t) = L_1 \frac{di_1(t)}{dt} + \sqrt{L_1 L_2}\,\frac{di_2(t)}{dt} = L_1 \frac{di_1(t)}{dt} + M \frac{di_2(t)}{dt}$$

$$v_2(t) = \sqrt{L_1 L_2}\,\frac{di_1(t)}{dt} + L_2 \frac{di_2(t)}{dt} = M \frac{di_1(t)}{dt} + L_2 \frac{di_2(t)}{dt}$$

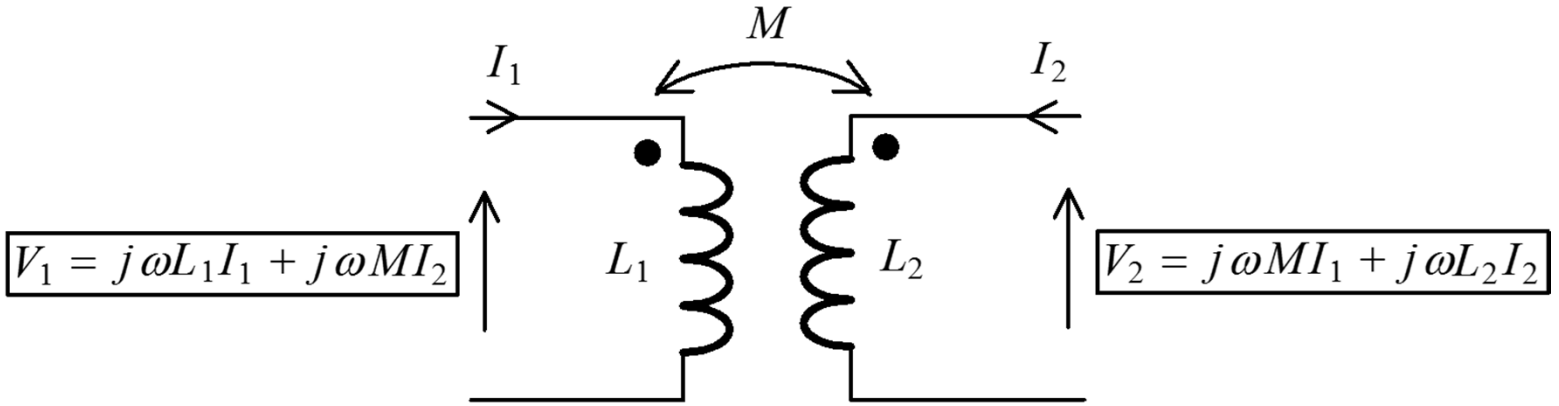
where $M = \sqrt{L_1 L_2}$ is called the **mutual inductance** between the two windings.

AC Environment



$L_1 = \frac{N_1^2}{\mathcal{R}}$, $L_2 = \frac{N_2^2}{\mathcal{R}}$, $M^2 = L_1 L_2$ for no flux leakage and perfect coupling

Why? How?.....



Why? How?..... It follows from the phasor technique

By using phasors, a time-varying ac voltage

$$v(t) = \sqrt{2}\, r \cos(\omega t + \theta) = \mathrm{Re}\left[\left(re^{j\theta}\right)\left(\sqrt{2}e^{j\omega t}\right)\right]$$

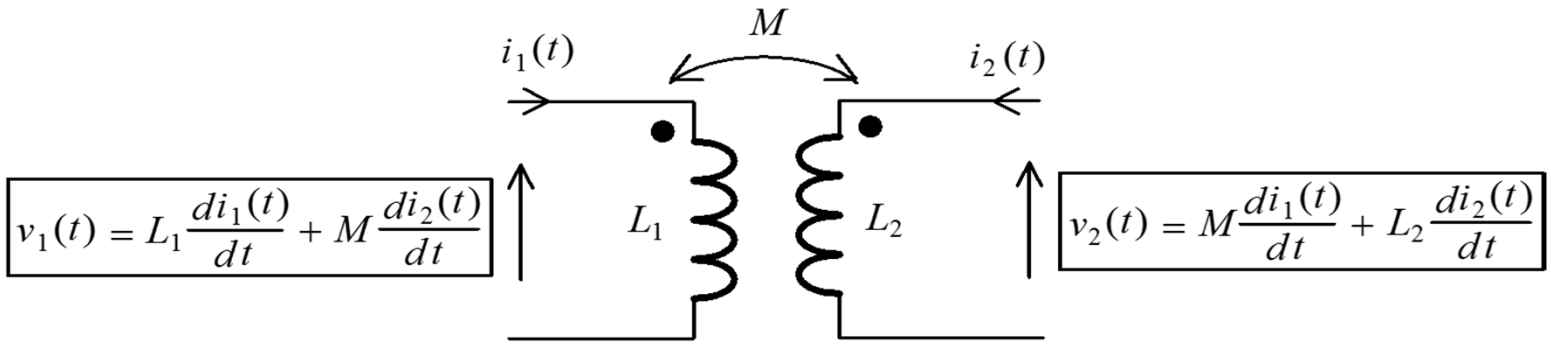
becomes a simple complex time-invariant number/voltage

$$V = r\, e^{j\theta} = r\angle\theta$$

where $r = |V|$ = magnitude of V and θ = Arg [V] = phase of V.

Using phasors, all time-varying AC quantities become complex DC quantities and all DC circuit analysis techniques can be employed for AC circuits without virtually any modification!

Why? How?..... It follows from the phasor technique



Let $i_1(t) = \sqrt{2}\,r_1\cos(\omega t + \theta_1) \Rightarrow I_1 = r_1 e^{j\theta_1}$ | $i_2(t) = \sqrt{2}\,r_2\cos(\omega t + \theta_2) \Rightarrow I_2 = r_2 e^{j\theta_2}$

$$\Rightarrow v_1(t) = L_1\frac{di_1(t)}{dt} + M\frac{di_2(t)}{dt} = L_1\frac{d\left(\sqrt{2}\,r_1\cos(\omega t+\theta_1)\right)}{dt} + M\frac{d\left(\sqrt{2}\,r_2\cos(\omega t+\theta_2)\right)}{dt}$$

$$= -L_1\sqrt{2}\,r_1\omega\cdot\sin(\omega t+\theta_1) - M\sqrt{2}\,r_2\omega\cdot\sin(\omega t+\theta_2)$$

$$= \sqrt{2}(r_1\omega L_1)\cos\left(\omega t+\theta_1+\frac{\pi}{2}\right) + \sqrt{2}(r_2\omega M)\cos\left(\omega t+\theta_2+\frac{\pi}{2}\right)$$

$$\Rightarrow V_1 = (r_1\omega L_1)e^{j\left(\theta_1+\frac{\pi}{2}\right)} + (r_2\omega M)e^{j\left(\theta_2+\frac{\pi}{2}\right)} = r_1\omega L_1 e^{j\theta_1}e^{j\frac{\pi}{2}} + r_2\omega M e^{j\theta_2}e^{j\frac{\pi}{2}}$$

$$= (j\omega L_1)\,r_1 e^{j\theta_1} + (j\omega M)\,r_2 e^{j\theta_2} = j\omega L_1 I_1 + j\omega M I_2$$

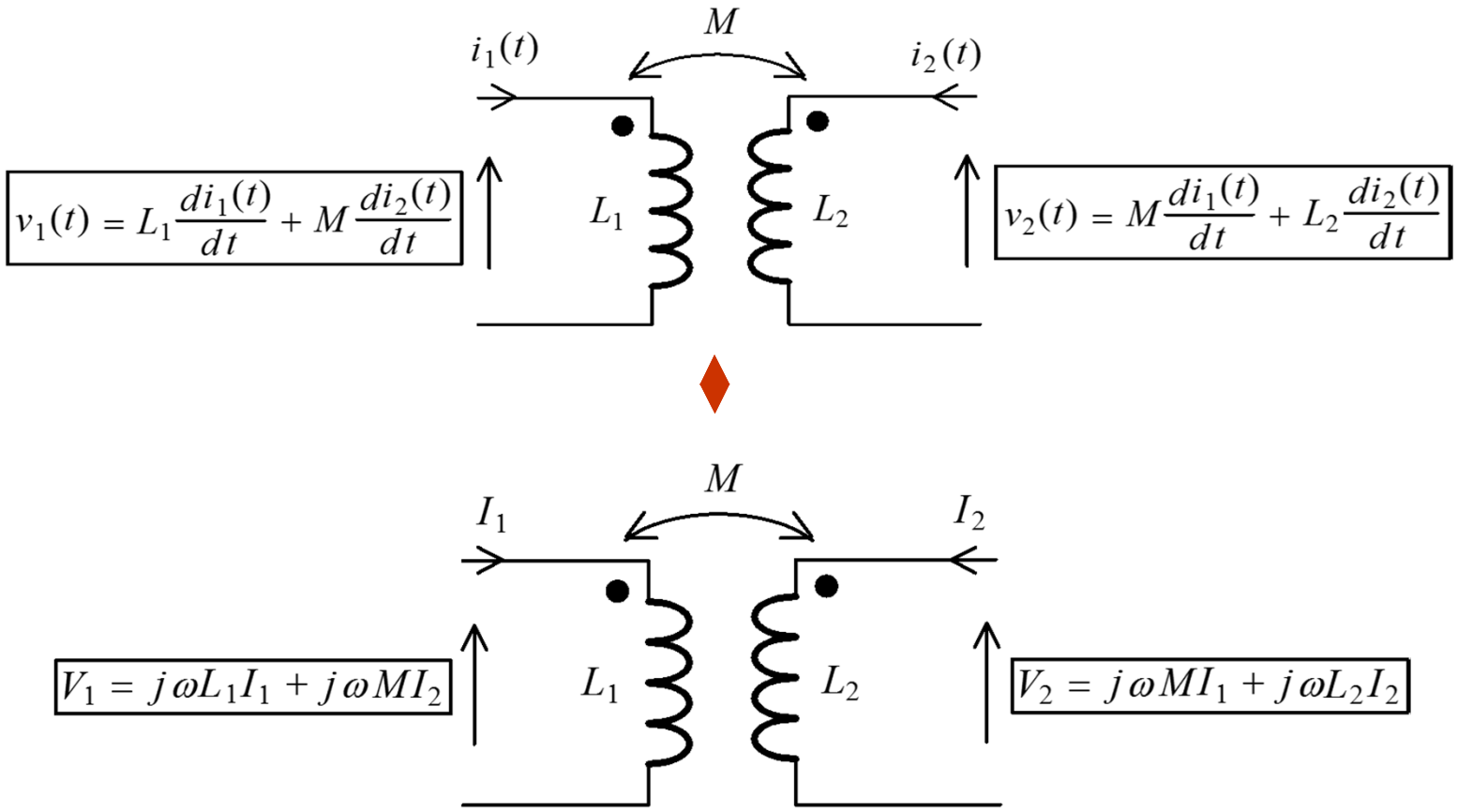
$$e^{j\frac{\pi}{2}} = \cos\frac{\pi}{2} + j\sin\frac{\pi}{2} = j$$

Why? How?..... It follows from the phasor technique

Similarly, one can derive

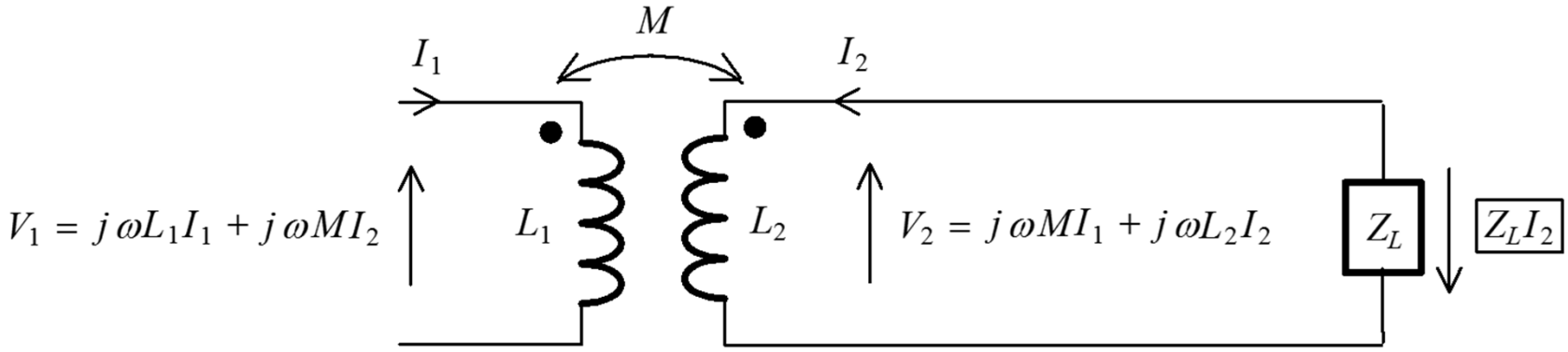
$$V_2 = j\omega M\, I_1 + j\omega L_2\, I_2$$

Thus, for an AC environment,



Transformer

Now consider connecting a mutual inductor to a load with impedance Z_L



by KVL

$$0 = V_2 + Z_L I_2 = j\omega M I_1 + j\omega L_2 I_2 + Z_L I_2$$

$$\frac{V_2}{V_1} = \frac{j\omega(MI_1 + L_2 I_2)}{j\omega(L_1 I_1 + MI_2)} = \frac{\sqrt{L_1 L_2}\, I_1 + L_2 I_2}{L_1 I_1 + \sqrt{L_1 L_2}\, I_2}$$

$$= \frac{\sqrt{L_2}\left(\sqrt{L_1} I_1 + \sqrt{L_2} I_2\right)}{\sqrt{L_1}\left(\sqrt{L_1} I_1 + \sqrt{L_2} I_2\right)}$$

$$= \sqrt{\frac{L_2}{L_1}} = \sqrt{\frac{N_2^2/\Re}{N_1^2/\Re}} = \frac{N_2}{N_1} = n, \text{ turn ratio}$$

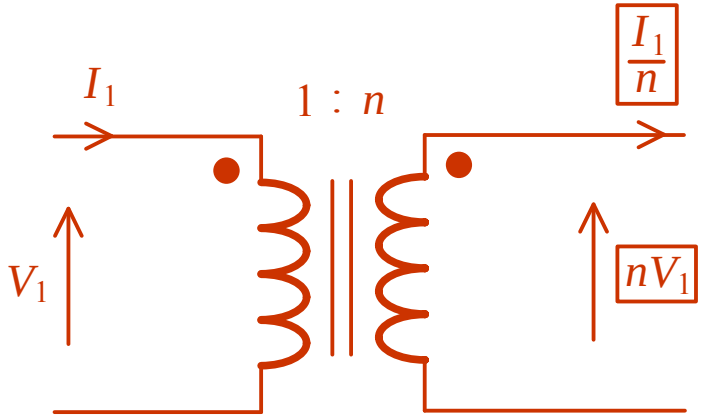
$$\frac{I_1}{I_2} = -\frac{j\omega L_2 + Z_L}{j\omega M}$$

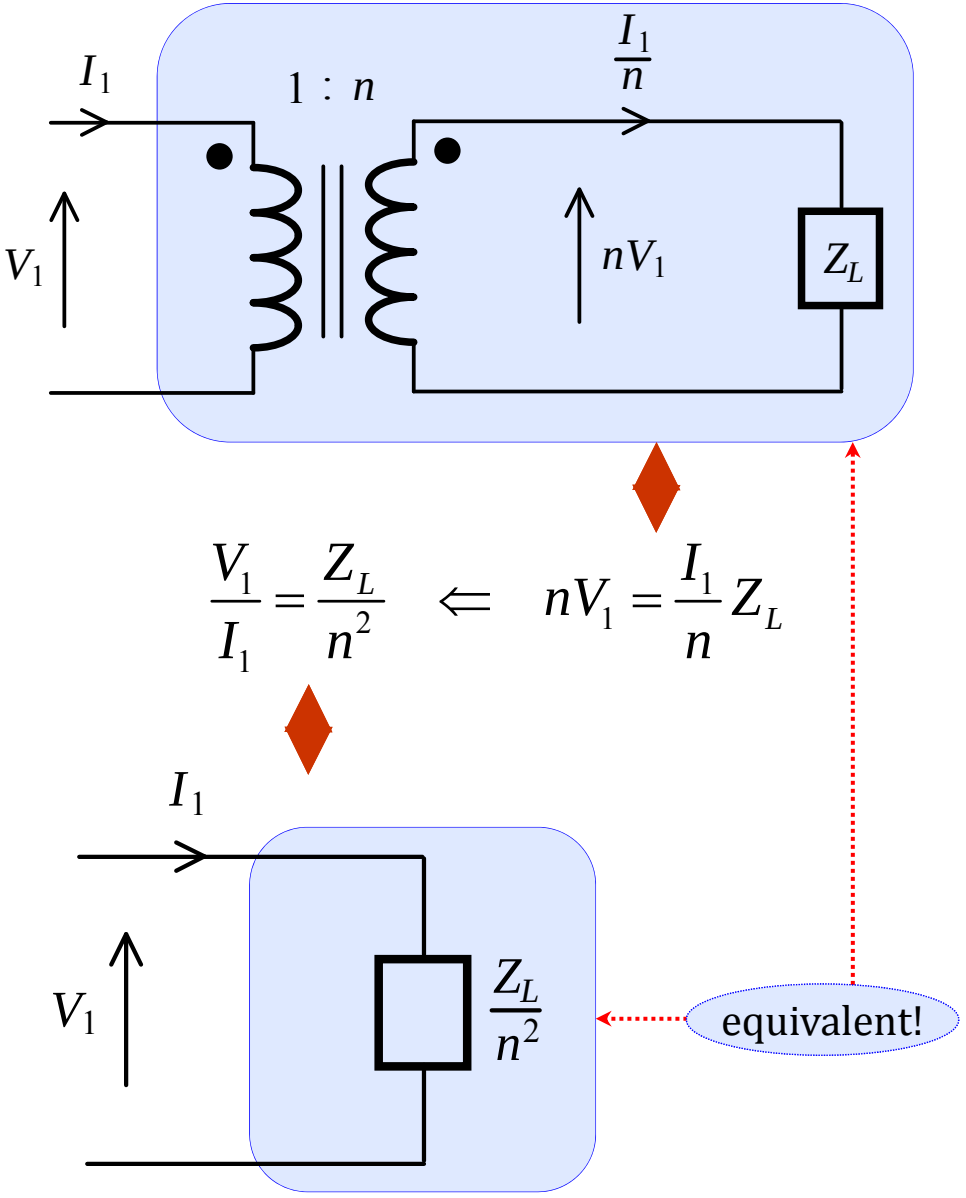
$$\approx -\frac{j\omega L_2}{j\omega M} \quad \text{if } |Z_L| << |\, j\omega L_2\,|$$

$$= -\frac{L_2}{\sqrt{L_1 L_2}} = -\sqrt{\frac{L_2}{L_1}} = -n$$

Equivalent Load

Voltages and currents of the primary and secondary windings of the ideal transformer with $|j\omega L_2| >> |Z_L|$

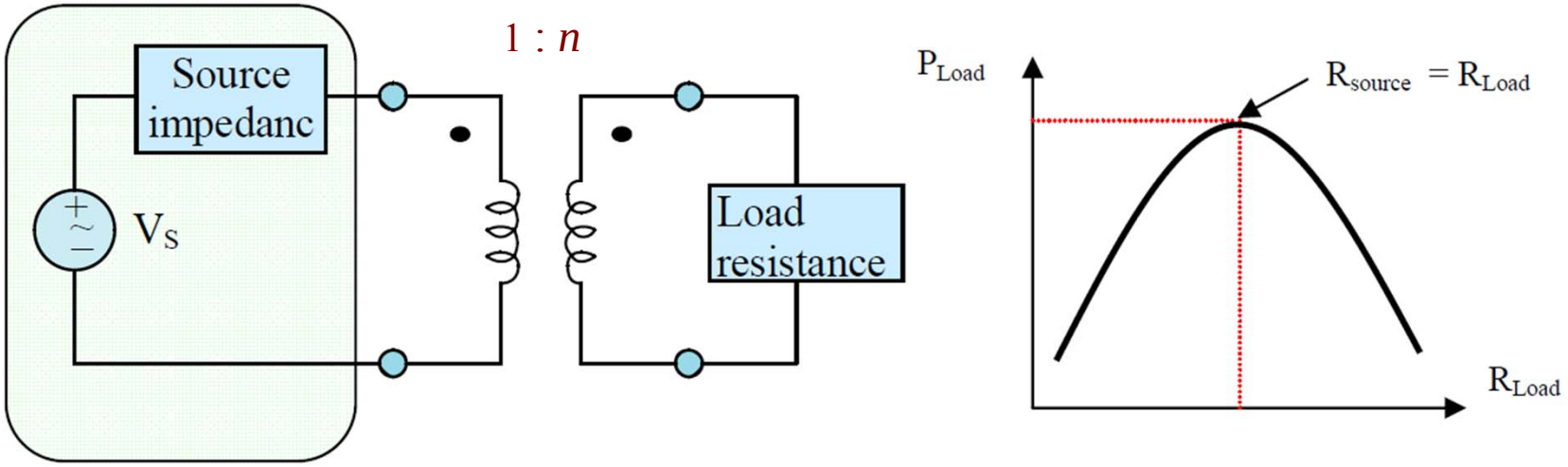




Equivalent Load: A load connected to the secondary of a transformer can be replaced by an equivalent load directly connected to the primary.

Application: Impedance Matching

A very important application of transformers is as an impedance matching device using the concept of equivalent load. Recall that the maximum power transfer theorem states that a power source delivers maximum power to the load when the load resistance is equal to the internal resistance of the source. This can be accomplished by using a transformer to match the two resistances.



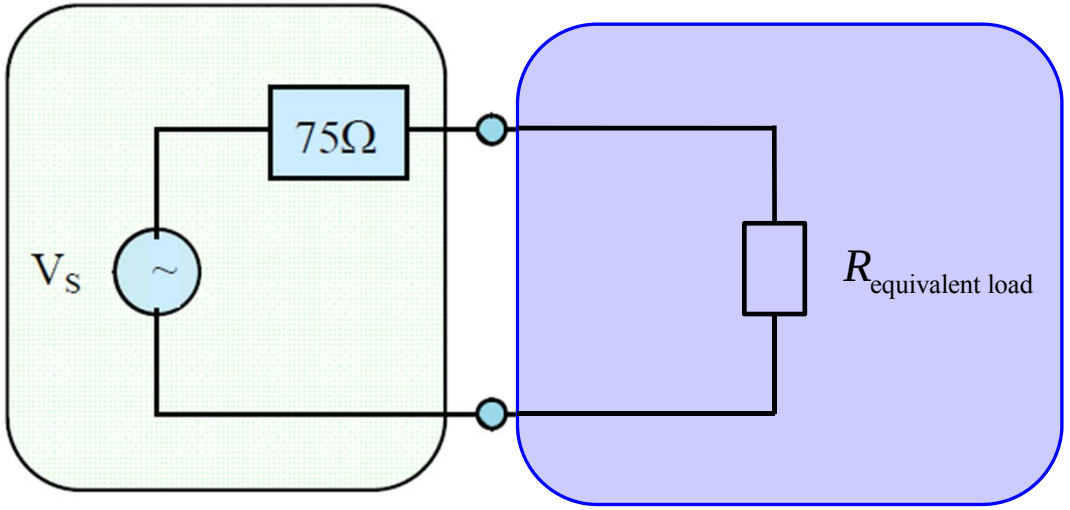
By choosing an appropriate transformer turn ratio n, the effective load resistance R_L (or effective load impedance Z_L) can be made equal to the internal resistance (or impedance) of the source. Such a process is called **impedance matching**.

Example 1

A sound system with a loudspeaker can be represented in a circuit diagram below. If the internal resistance of the source is 75Ω, and the resistance of the load is 300Ω, find an appropriate transformer turns ratio, which results in impedance matching.

Solution: The equivalent load resistance seen by the source (or the primary winding) is given by

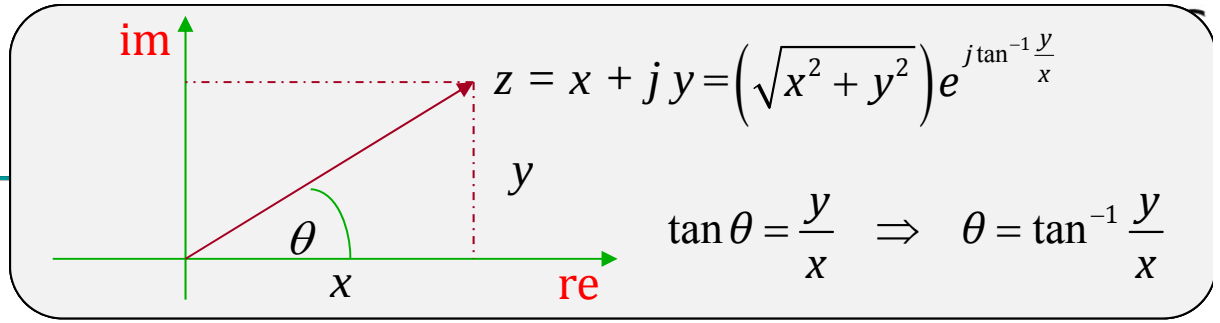
$$R_{\text{equivalent load}} = \frac{300}{n^2}$$



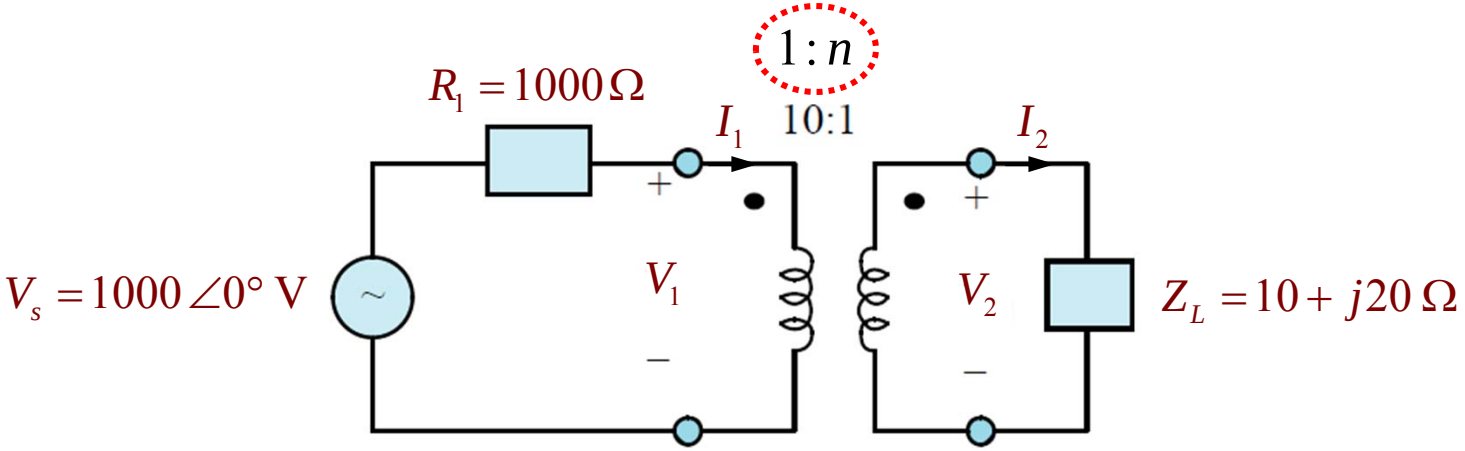
To match it with the source internal resistance, we set

$$R_{\text{equivalent load}} = \frac{300}{n^2} = 75 \quad \Rightarrow \quad n^2 = \frac{300}{75} = 4 \quad \Rightarrow \quad n = 2 \quad \leftarrow \text{ the required turns ratio}$$

Example 2



For the given circuit, find the phasor currents and voltages, and the power delivered to the load.

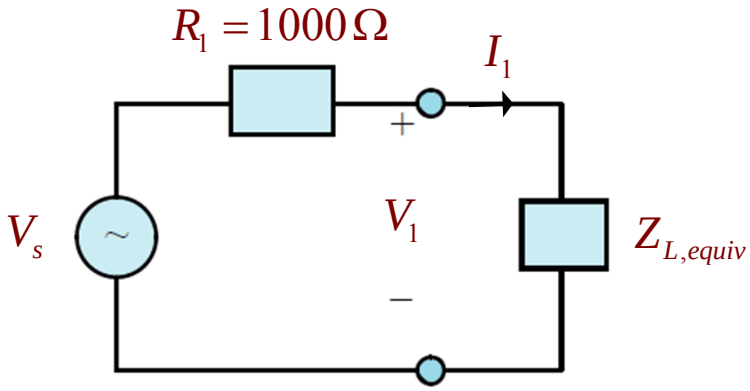


Solution: The equivalent load impedance seen by the primary side is given by

$$Z_{L,equiv} = \frac{Z_L}{n^2} = \frac{10+j20}{(1/10)^2} = 100(10+j20)\,\Omega$$

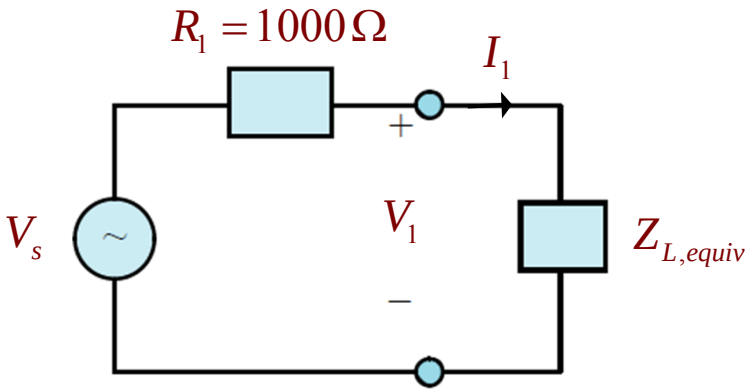
The total impedance seen by the source is:

$$Z_{total} = R_1 + Z_{L,equiv}$$
$$= 1000 + 1000 + j2000 = 2828\angle 45°\,\Omega$$



Example 2 (cont.)

The primary side current and voltage can be calculated as:



$$I_1 = \frac{V_s}{Z_{total}} = \frac{1000\angle 0°}{2828\angle 45°} = 0.3536\angle -45° \text{ A}$$

and

$$V_1 = I_1 Z_{L,equiv} = 0.3536\angle -45° \times (1000 + j2000) = 790.6\angle 18.43° \text{ V}$$

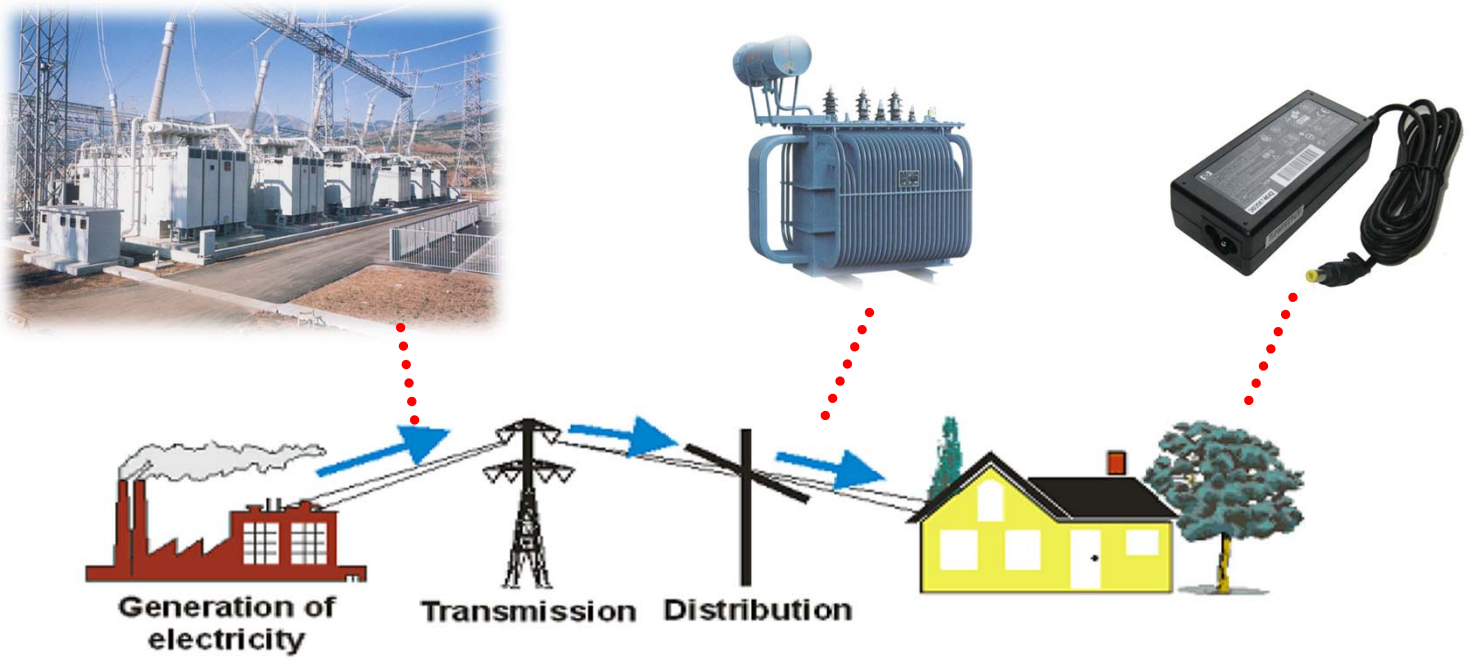
The secondary side current and voltage can be calculated as:

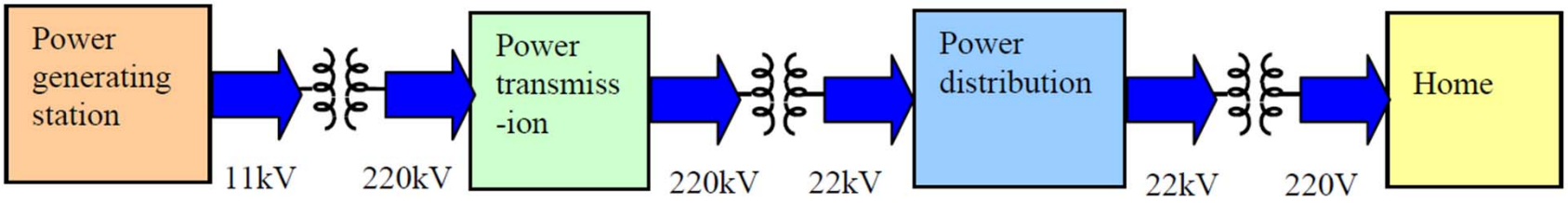
$$I_2 = \frac{I_1}{n} = \frac{0.3536\angle -45°}{1/10} = 3.536\angle -45° \text{ A}$$

$$V_2 = nV_1 = \frac{1}{10} \times 790.6\angle 18.43° = 79.06\angle 18.43° \text{ V}$$

The power delivered to the load: $P_L = |I_2|^2 R_L = 3.536^2 \times 10 = 125 \text{ W}$

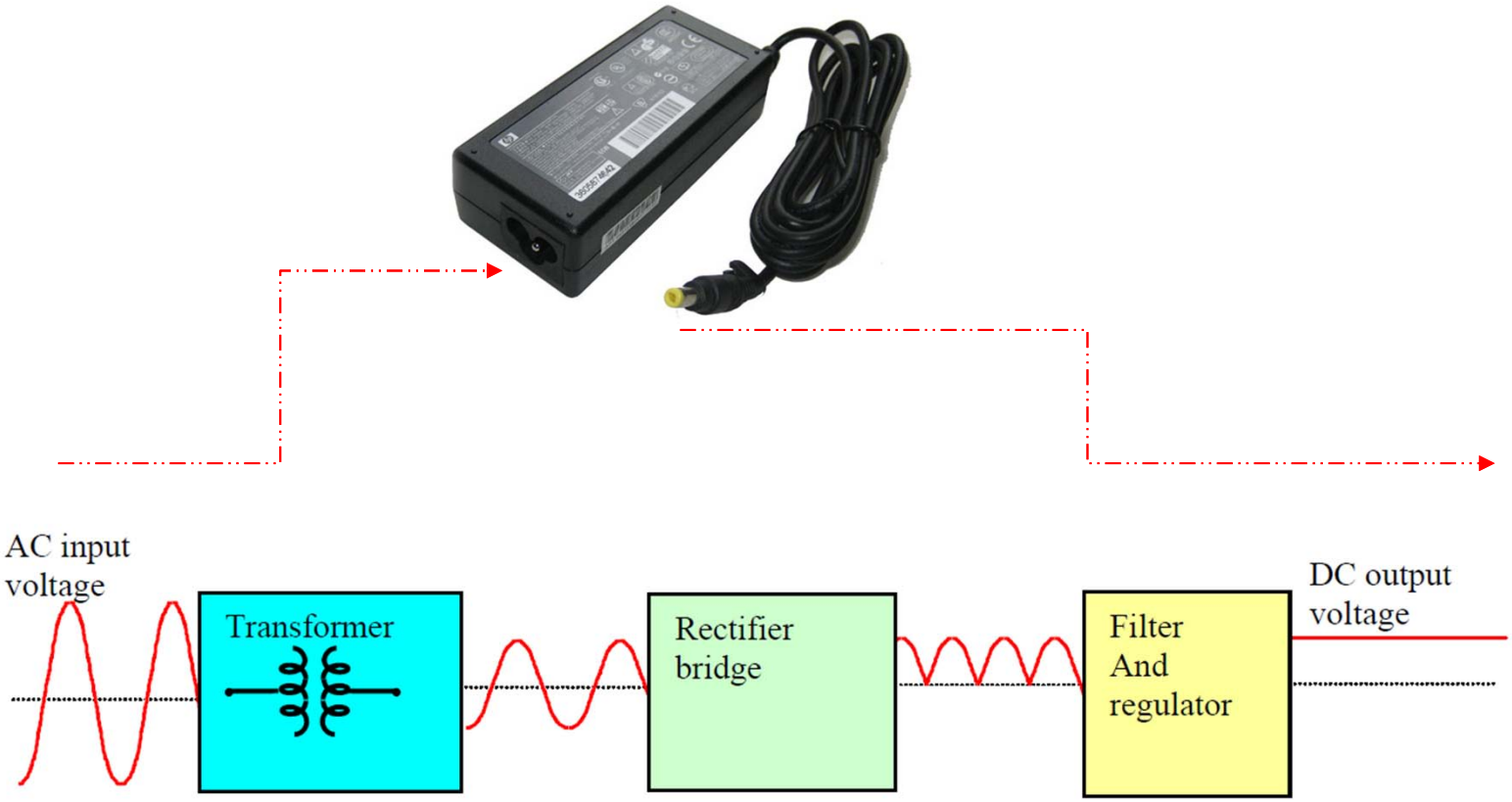
Transformers Used in Power Transmission





Transformers used to raise/lower voltages in electrical energy distribution

Application 2: Power Supply



Transformer used in DC power supply

DC Power Supply

Rectifier Circuits

Examples of DC Power Supply

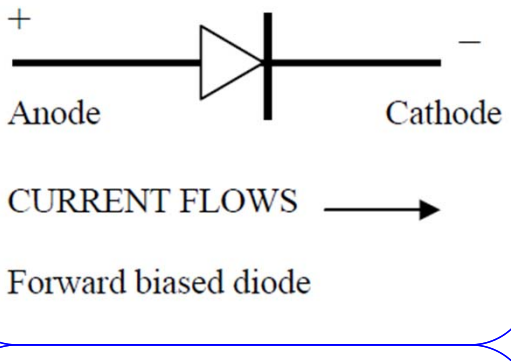
Learning Objectives

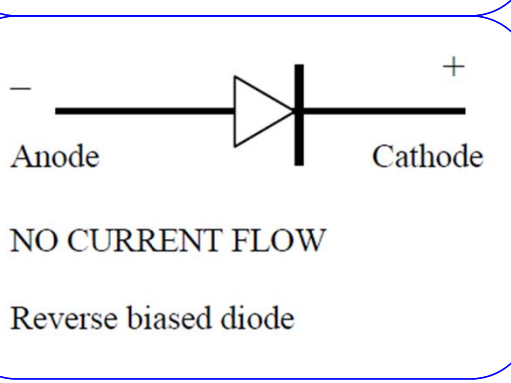
The main learning objectives for this topic are as follows:

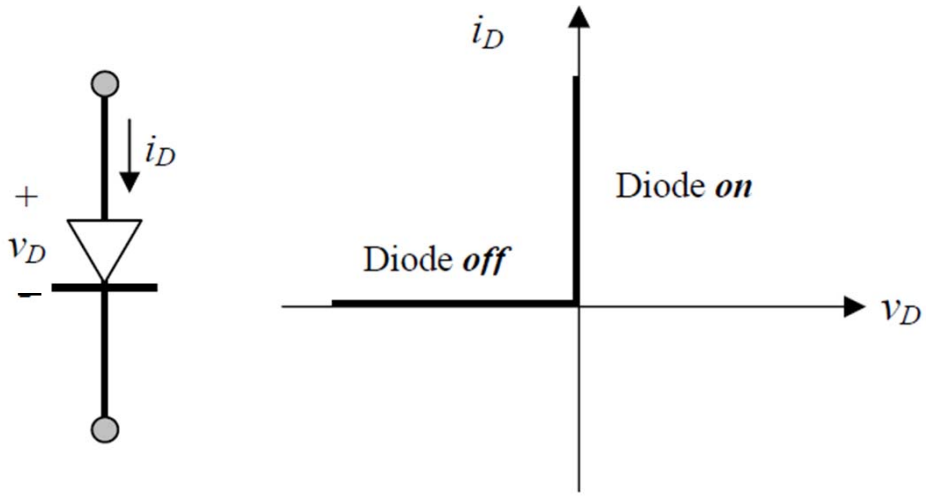
1. To understand voltage-current characteristics of a diode
2. To understand operation of half-wave and full-wave rectifier circuits
3. To determination of output voltages and currents
4. To analyze operation of rectifier circuit with capacitor filter
5. To calculate peak inverse voltages for rectifier circuits

Diode

Diodes allow electricity to flow in only one direction. Diodes are the electrical version of a valve and early diodes were actually called valves. The schematic symbol of a diode is shown below. The arrow of the circuit symbol shows the direction in which the current can flow. The diode has two terminals, a cathode and an anode as shown in the figures.

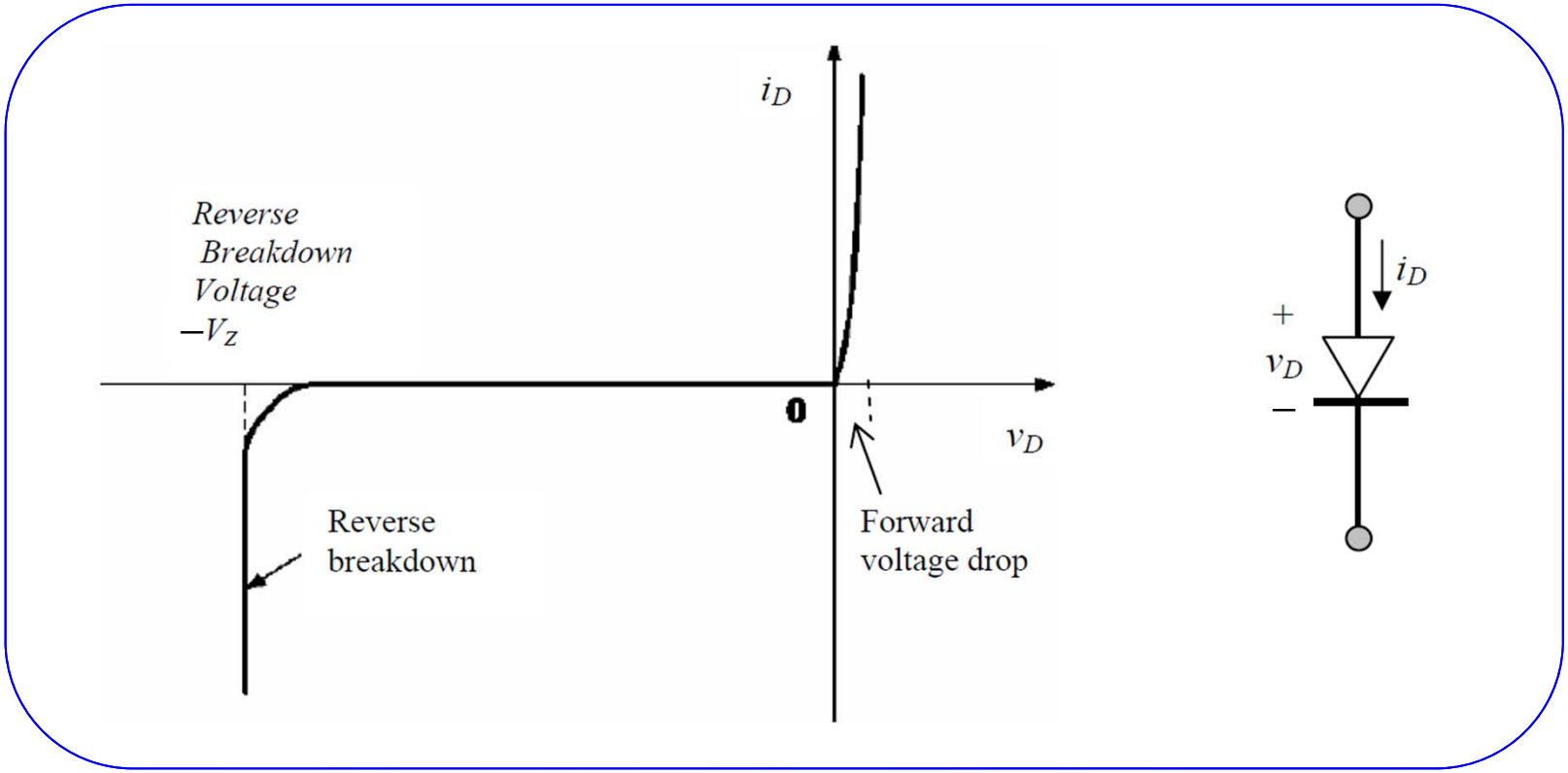






Characteristics of an ideal diode

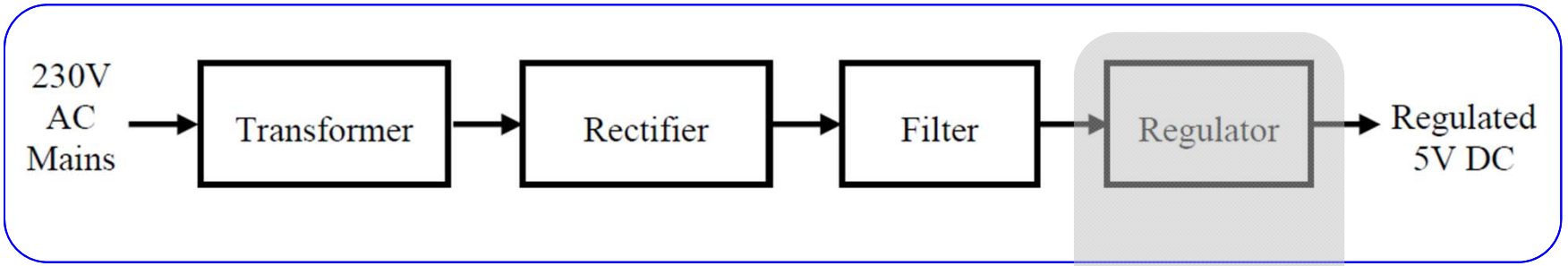
Characteristics of Silicon Diodes



The actual characteristics of a silicon diode is slightly different from the ideal one. In particular, if the voltage v_D is more negative than the Reverse Breakdown voltage, the diode conducts again, but in a reverse direction.

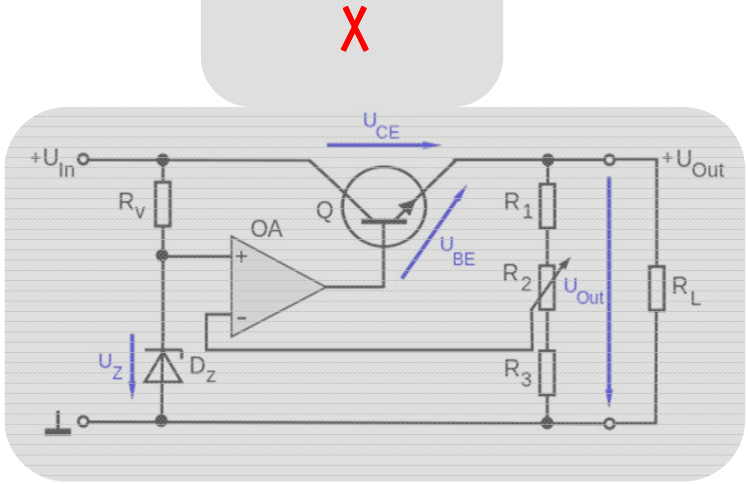
Diode Rectifier Circuits

One of the important applications of a semiconductor diode is in rectification of AC signals to DC. Diodes are very commonly used for obtaining DC voltage supplies from the readily available AC voltage.

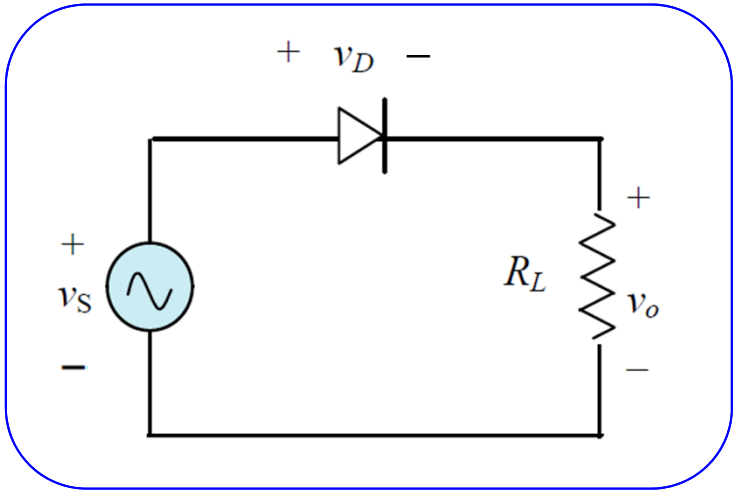


There are many possible ways to design rectifier circuits using diodes, e.g.,

- Half Wave Rectifier
- Full Wave Rectifier
- Bridge Rectifier



Simple Half Wave Rectifier

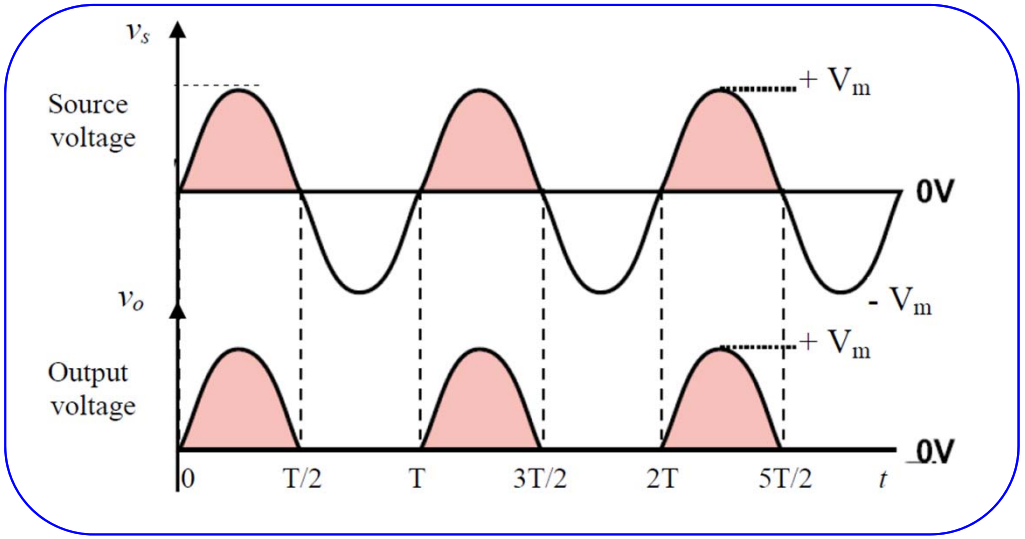


The AC power supply is given by

$$v_s = V_m \sin(\omega t) = V_m \sin(2\pi f t) = V_m \sin\left(2\pi \frac{1}{T} t\right)$$

We are interested in obtaining DC voltage across the "load resistance" R_L.

During the positive half cycle, the diode is on. The source voltage is directly connected across the load. During the negative half cycle, the diode is off. The source voltage is disconnected from the load. The waveforms for v_s and v_o are shown in figure on the right…



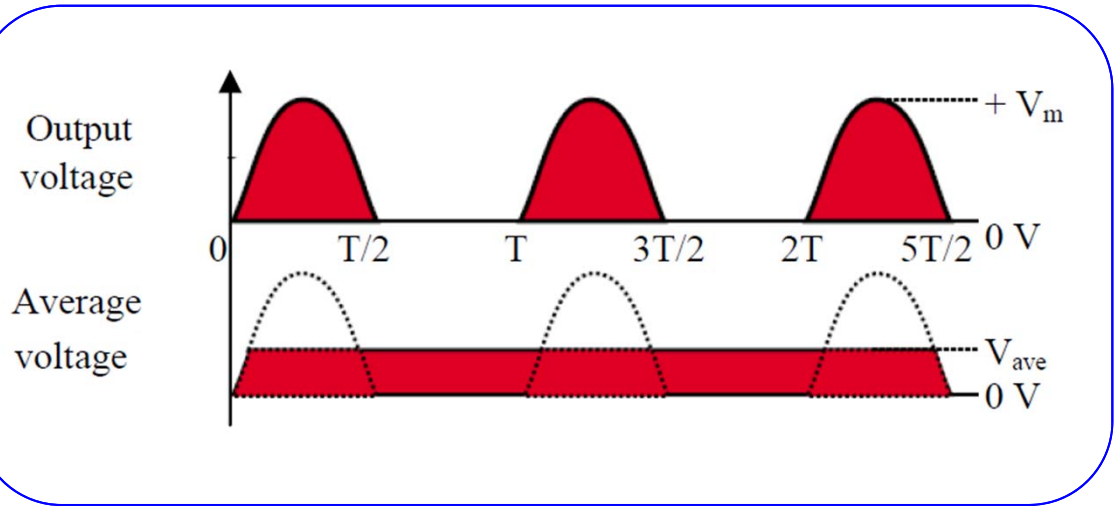
¤ v_o varies between the peak V_m and 0 in each cycle. This variation is called **ripple**, and the corresponding voltage is called the peak-to-peak ripple voltage, $V_{p\text{-}p}$.

Average Load Voltage & Current

If a DC voltmeter is connected to measure the output voltage of the half-wave rectifier (i.e., across the load resistance), the reading obtained would be the **average load voltage** V_{ave}, also called the DC output voltage. The meter averages out the pulses and displays the following average:

$$\omega = 2\pi f = 2\pi \frac{1}{T}$$

$$V_{ave} = \frac{1}{T}\int_0^T v_o(t)dt = \frac{1}{T}\left[\int_0^{T/2} V_m \sin(\omega t)\,dt + 0\right] = \frac{V_m}{\omega T}\left[\cos 0 - \cos\frac{\omega T}{2}\right] = \frac{V_m}{2\pi}\left[\cos 0 - \cos\pi\right] = \frac{V_m}{\pi}$$

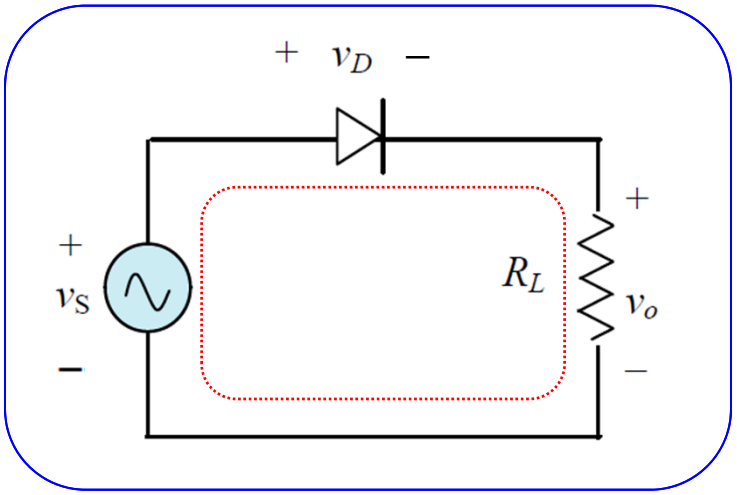


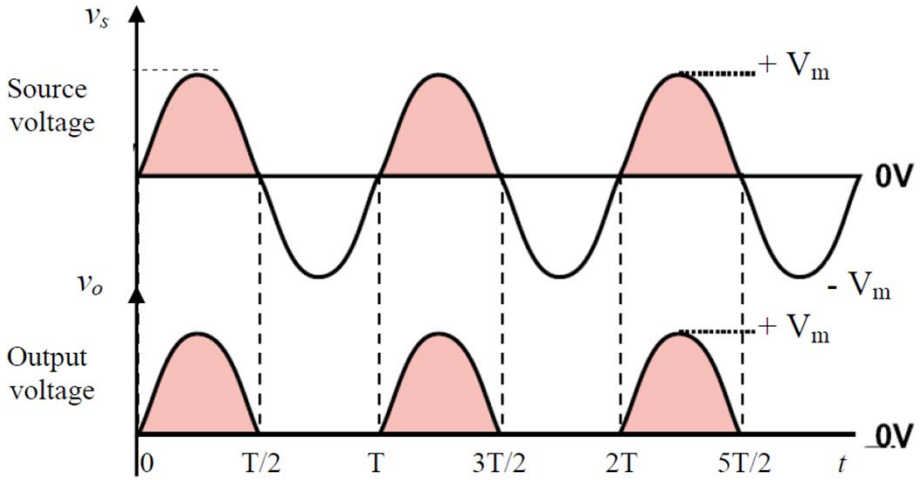
Average load current:

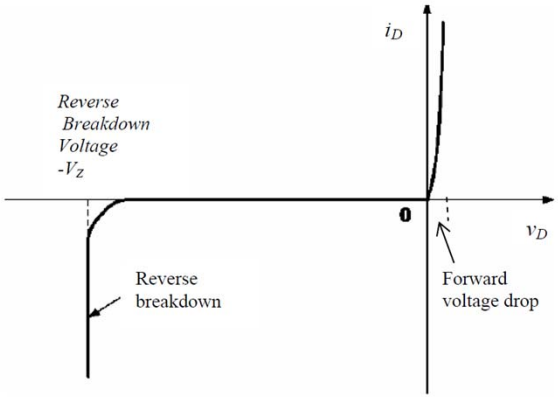
$$I_L = \frac{V_{ave}}{R_L} = \frac{V_m}{\pi R_L}$$

¤ Truly speaking, the output voltage & current are not so 'DC'…

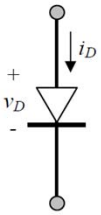
Reverse Voltage of the Diode

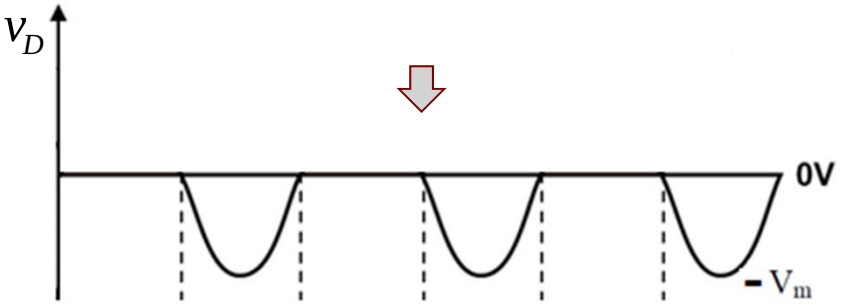






By KVL: $v_S = v_D + v_O \quad \Rightarrow \quad v_D = v_S - v_O$





Clearly, the maximum reverse voltage across the diode is V_m. Note that the diode will function properly so long as V_m is smaller than its reverse breakdown voltage (i.e., V_z).

Example 1

A 50Ω load resistance is connected across a half wave rectifier. The input supply voltage is 230V (rms) at 50 Hz. Determine the DC output (average) voltage, peak-to-peak ripple in the output voltage ($V_{p\text{-}p}$), and the output ripple frequency (f_r).

Solution: Peak amplitude of the source voltage is: $V_m = 230 \times \sqrt{2} = 325.3$ V

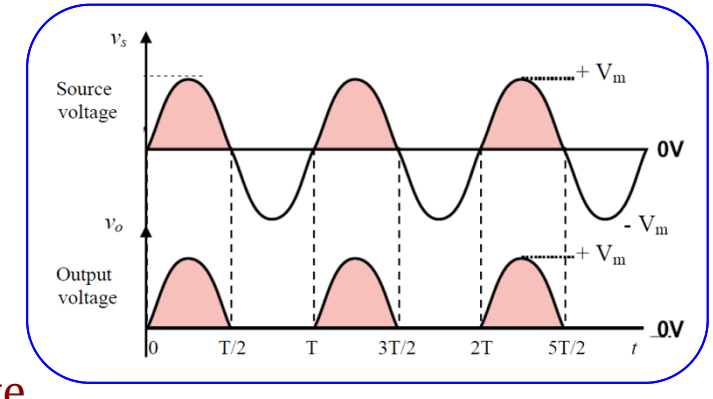
Output DC voltage: $V_{ave} = \dfrac{V_m}{\pi} = \dfrac{325.3}{3.14} = 103.5$ V

Peak-to-peak ripple voltage is the difference between max and min in v_o.

$$V_{p\text{-}p} = V_{\max} - V_{\min} = V_m - 0 = 325.3 \text{ V}$$

Percentage ripple = $(V_{p\text{-}p}/V_{ave}) \times 100 = 314\%$

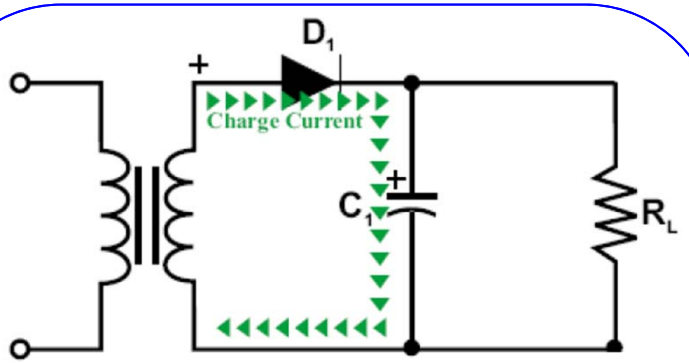
The ripple frequency is also 50 Hz, i.e., $f_r = 50$ Hz.



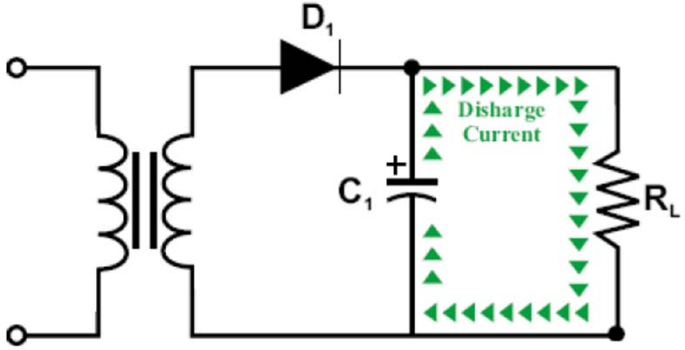
Note that the peak-to-peak ripple and the percentage ripple value are the measure of the smoothness of the output 'DC' voltage…

Half Wave Rectifier with Capacitor

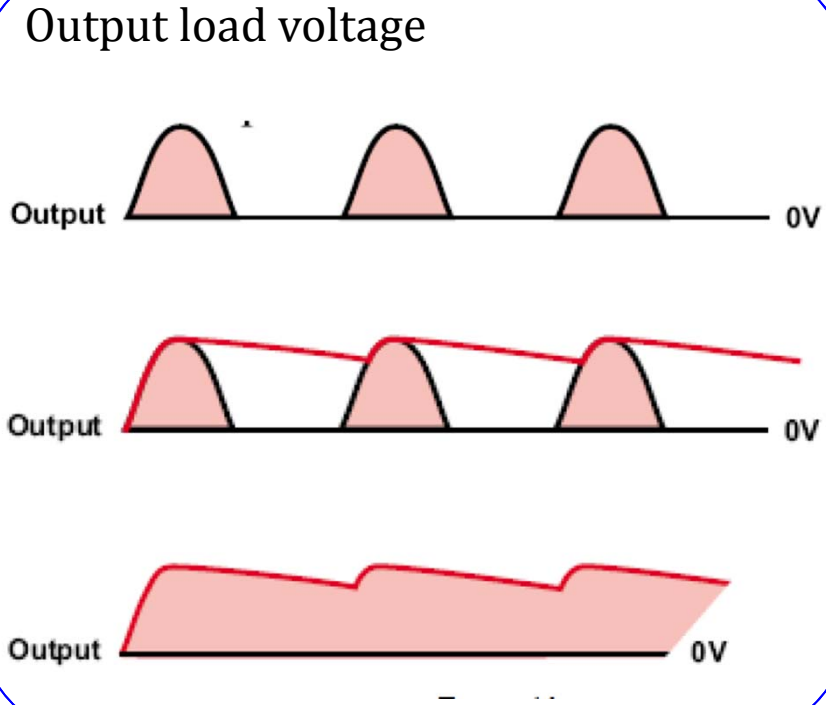
We have observed that the simple half wave rectifier does not really provides DC voltage and current. To get a better load voltage and current, we add an additional capacitor to the circuit…



positive half cycle



negative half cycle

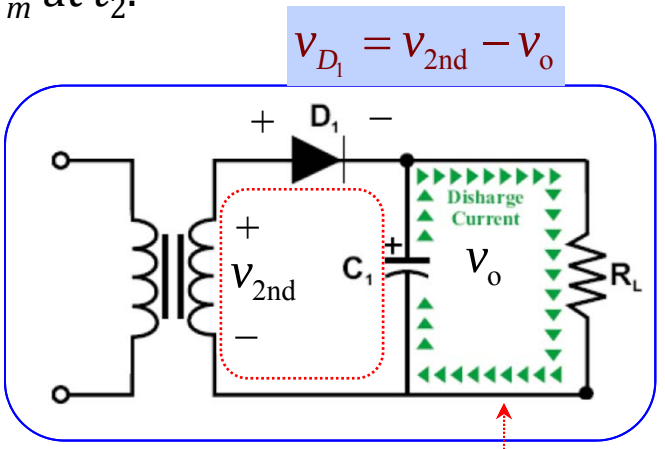


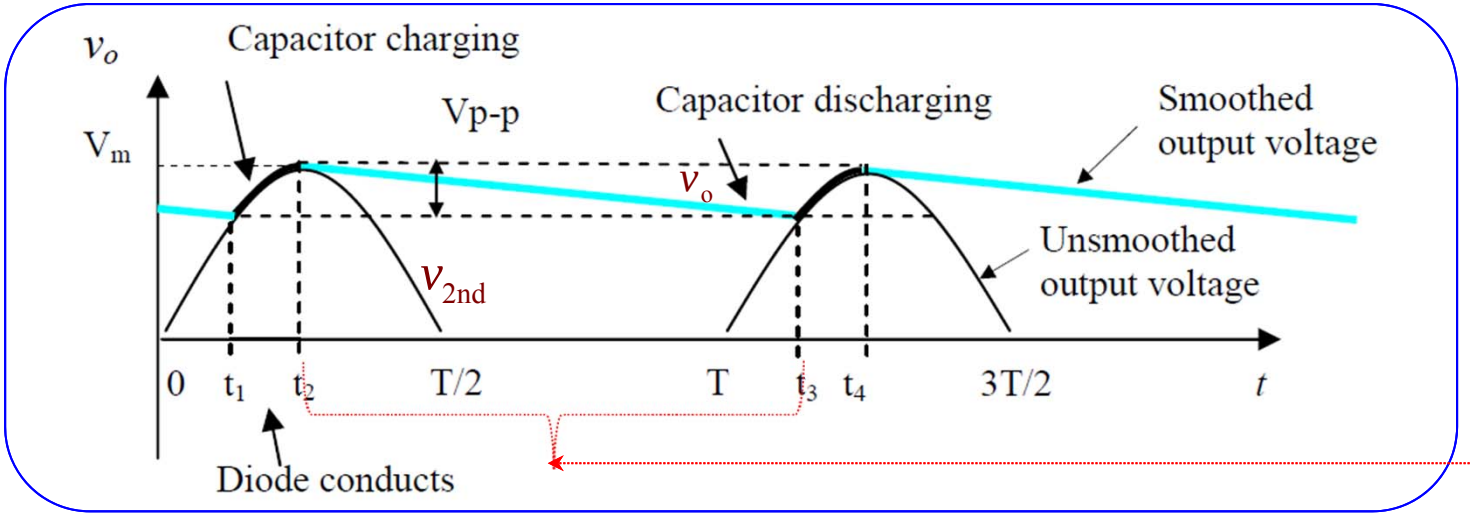
¤ The output voltage is pretty 'DC'…

Operational Analysis

During each positive half cycle, the capacitor charges during the interval t_1 to t_2. During this interval, the diode will be ON. Due to this charging, the voltage across the capacitor v_o will be equal to the AC peak voltage V_m at t_2.

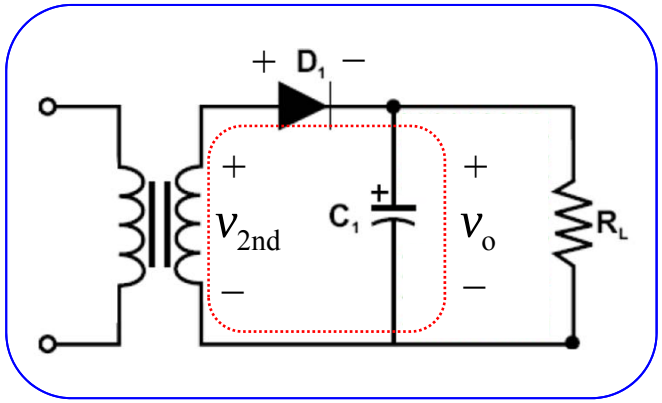
The capacitor will supply current to load resistor R_L during time interval t_2 to t_3. During this interval, diode will OFF since the AC voltage is less than v_o. Due to the large energy stored in the capacitor, the capacitor voltage will not reduce much during t_2 to t_3, and v_o will remain close to the peak value.

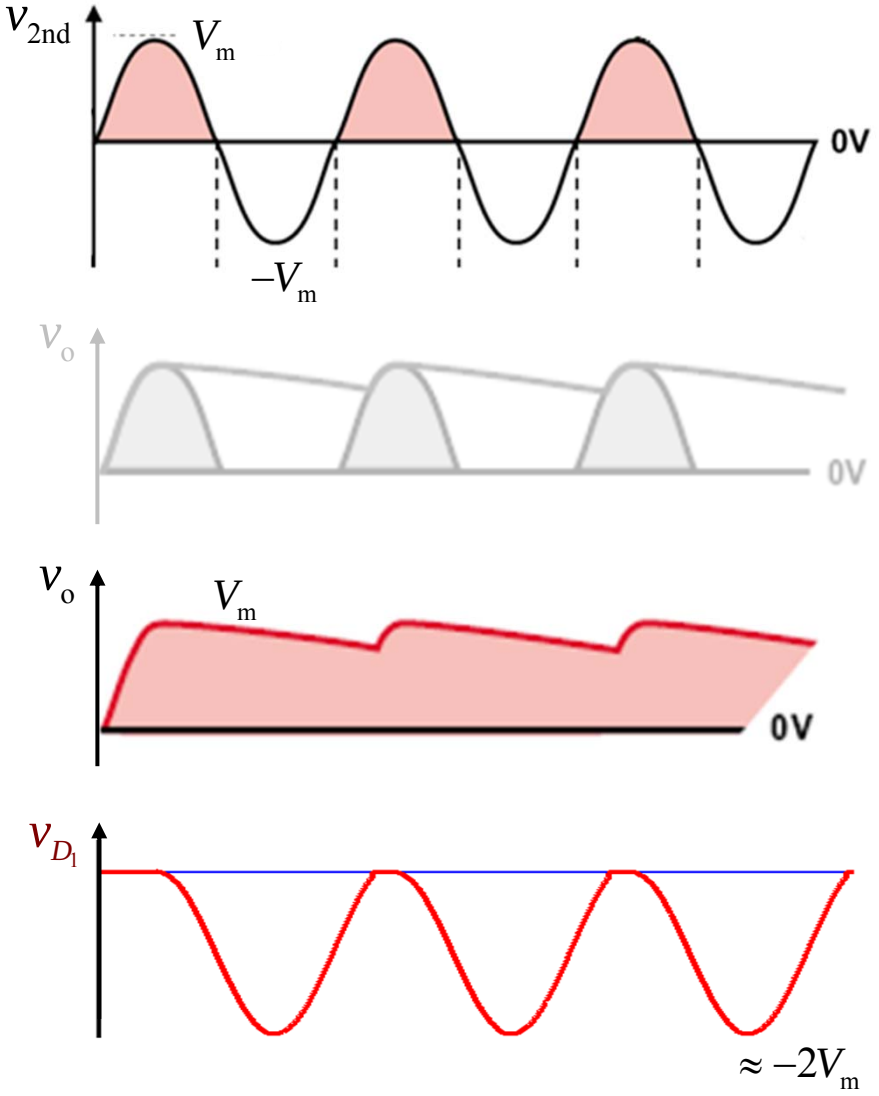




discharging

Reverse Voltage of the Diode





By KVL:

$$v_{2\text{nd}} = v_{D_1} + v_{\text{o}} \quad \Rightarrow \quad v_{D_1} = v_{2\text{nd}} - v_{\text{o}}$$

The peak voltage across the diode is

$$v_{D_1} \approx -V_{\text{m}} - V_{\text{m}} = -2V_{\text{m}}$$

The maximum reverse diode voltage is approximately $2V_{\text{m}}$.

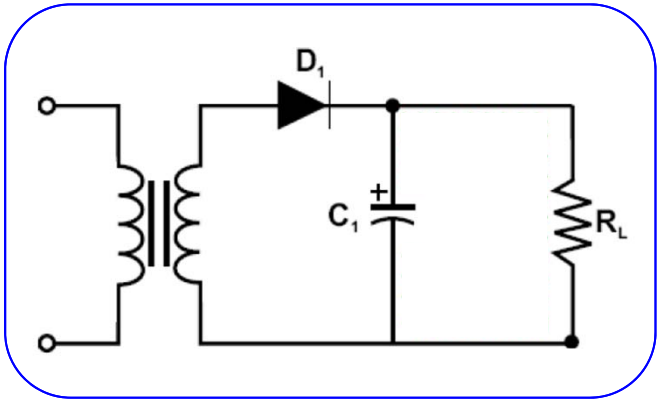
Average Load Voltage & Current

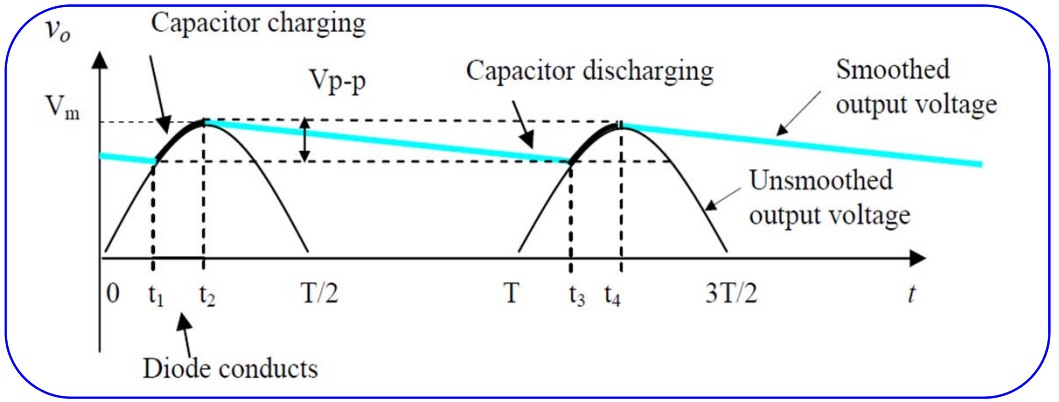
In practical applications, a very large capacitor is used so that the output voltage is close to the peak value. The **average voltage** (also called DC output voltage) across the load can therefore be approximated to:

$$V_{ave} \cong V_m$$

The **average load current** is then given by

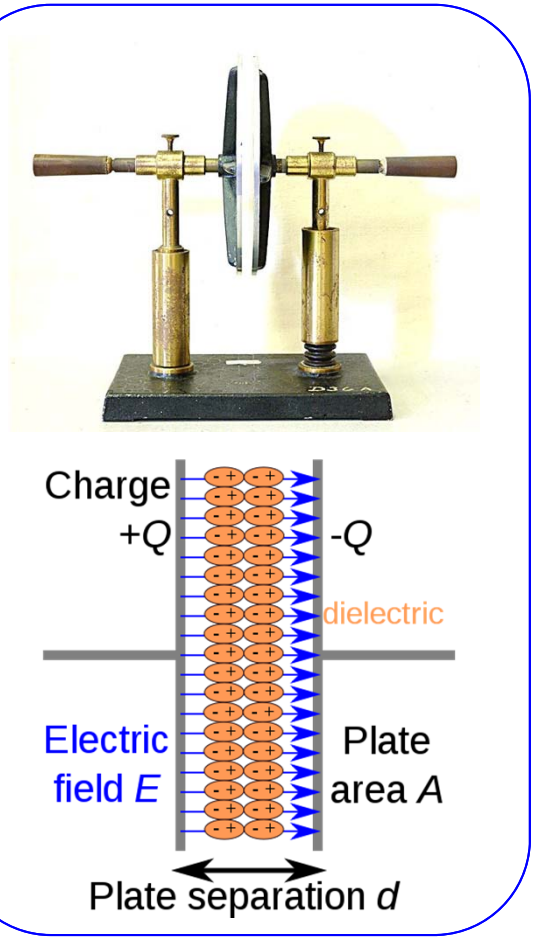
$$I_L \cong \frac{V_m}{R_L}$$





Capacitor and Capacitance

A **capacitor** is an electrical component used to store energy in an electric field. The forms of practical capacitors vary, but all contain at least two electrical conductors separated by a dielectric (insulator). The capacitance is defined as



$$C = \frac{Q}{V}$$

For general situations, we have capacitor voltage

$$v(t) = \frac{q(t)}{C} \quad \Rightarrow \quad \boxed{\Delta V = \frac{\Delta Q}{C}}$$

Noting that (the definition of electric current)

$$i(t) = \frac{dq(t)}{dt} \quad \Rightarrow \quad \boxed{I = \frac{\Delta Q}{\Delta T}}$$

we have

$$\frac{dv(t)}{dt} = \frac{1}{C}\frac{dq(t)}{dt} = \frac{1}{C} \cdot i(t) \quad \Rightarrow \quad \boxed{i(t) = C\frac{dv(t)}{dt}}$$

Peak-to-Peak Ripple Voltage & Required Capacitance

The voltage waveforms show a small AC component called "ripple" present in the output voltage. This ripple can be minimized by choosing the largest capacitance value that is practical. We can calculate the **required capacitance** as follows.

The charge removed from the capacitor during the discharge cycle (i.e., t_2 to t_3) is:

$$\Delta Q \cong I_L T \qquad \Leftarrow \quad I = \frac{\Delta Q}{\Delta T}$$

where I_L is the average load current and T is the period of the AC source. As the interval t_1 to t_2 is very small, the discharge time can be approximated by T.

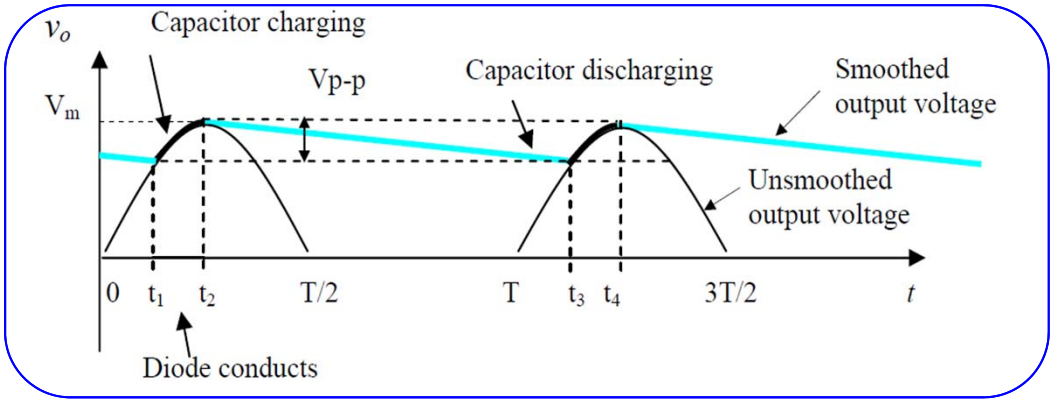
If $V_{p\text{-}p}$ is the peak-to-peak ripple voltage and C_1 is the capacitance, The charge removed from the capacitor can also be expressed as

$$\Leftarrow \quad \Delta V = \frac{\Delta Q}{C}$$

$$\Delta Q \cong V_{p\text{-}p} C_1 \qquad \left(\Leftarrow \quad V_{p\text{-}p} = \frac{\Delta Q}{C_1} \right)$$

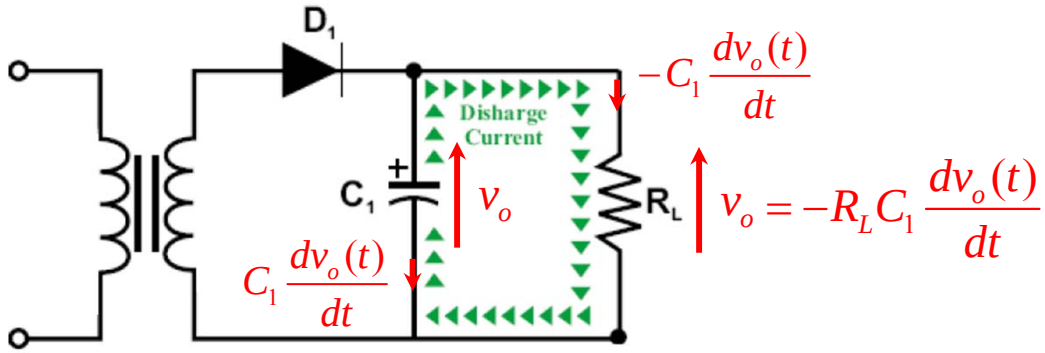
Required capacitance is given by

$$C_1 = \frac{I_L T}{V_{p\text{-}p}} = \frac{V_m T}{R_L V_{p\text{-}p}} \quad \text{F (Farads)}$$



An Alternative Way for Finding Required Capacitance

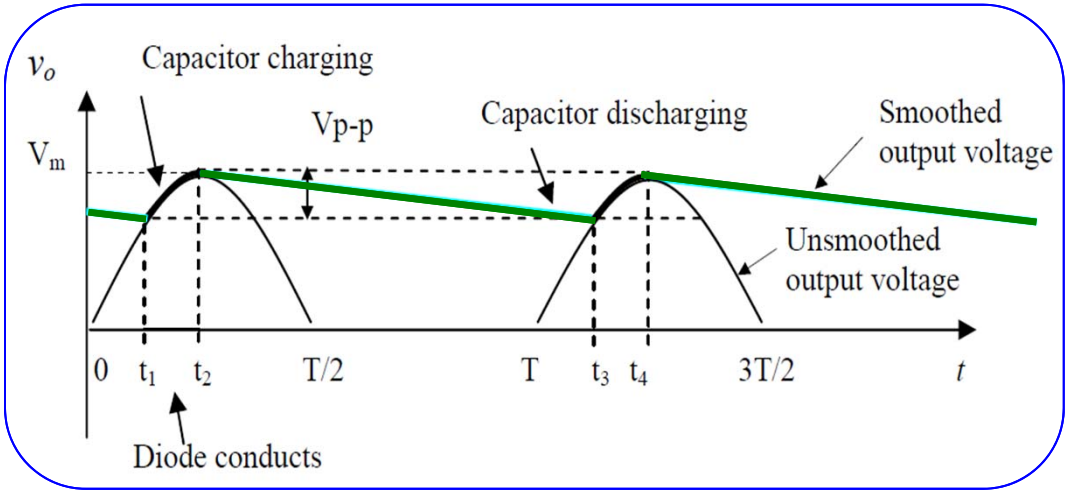
During the capacitor discharging period (the **green** curves), the **green** portion of the circuit below is governed by



$$R_L C_1 \frac{dv_o(t)}{dt} + v_o(t) = 0$$

$$v_o(t) = V_m e^{-t/R_L C_1}$$

Transient Response...



As the discharge time can be approximated by T, we have

$$v_o(T) = V_m e^{-T/R_L C_1} = V_m - V_{p\text{-}p}$$

$$e^x = 1 + x + x^2/2! + \cdots$$

$$V_m\left(1 - \frac{T}{R_L C_1}\right) \approx V_m - V_{p\text{-}p}$$

$$\frac{V_m T}{R_L C_1} \approx V_{p\text{-}p}$$

Required Capacitance

$$\frac{I_L T}{V_{p\text{-}p}} = \frac{V_m T}{R_L V_{p\text{-}p}} = C_1$$

Example 2

In the circuit of Example 1, a 10000μF filter capacitor is added across the load resistor. The voltage across the secondary terminals of the transformer is 230V (rms). Determine the DC output voltage (i.e. average voltage), load current, peak-to-peak ripple in the output voltage, and the output ripple frequency.

Solution: Output DC voltage: $V_{ave} \cong V_m = 325.3$ V

Average load current: $I_L = \dfrac{V_{ave}}{R_L} \cong \dfrac{325.3}{50} = 6.51$ A . This current discharges the capacitor during the interval t_2 to t_3.

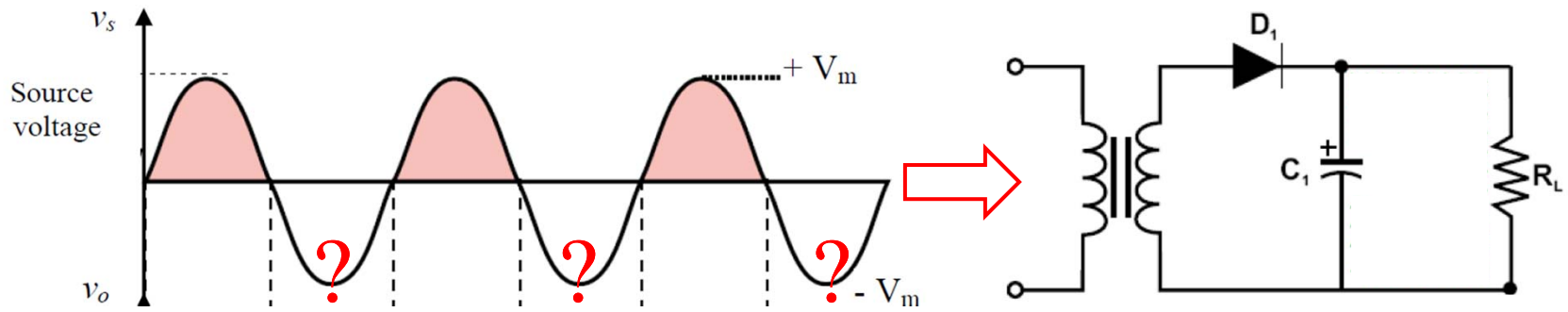
The time period of the AC source = 20 ms (for frequency of 50 Hz). Thus, the peak-to-peak ripple in the output voltage is given by

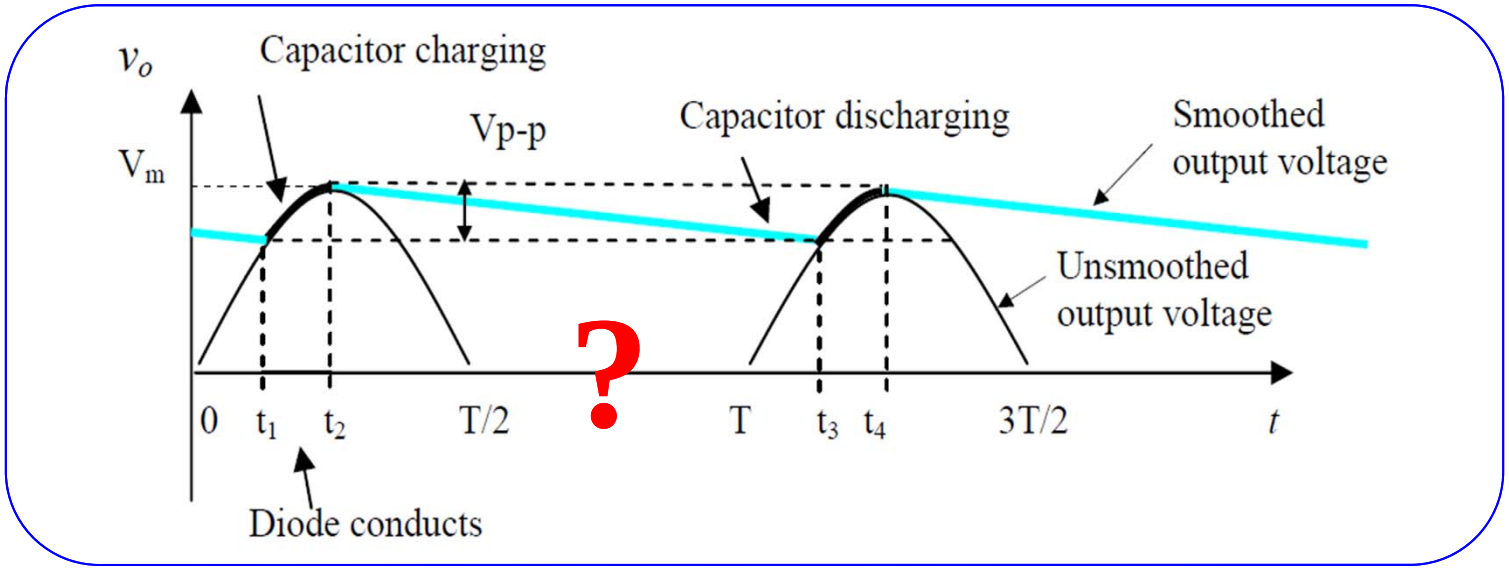
$$\frac{I_L T}{V_{p\text{-}p}} = \frac{V_m T}{R_L V_{p\text{-}p}} = C_1 \quad \Rightarrow \quad V_{p\text{-}p} = \frac{I_L T}{C_1} = \frac{6.51 \times 20 \times 10^{-3}}{10000 \times 10^{-6}} = 13.02 \text{ V}$$

The ripple voltage is only 4% (= 13.02/325.3) now, its frequency remains at 50 Hz.

Full Wave Rectifier

What is the shortage with the half-wave rectifier?

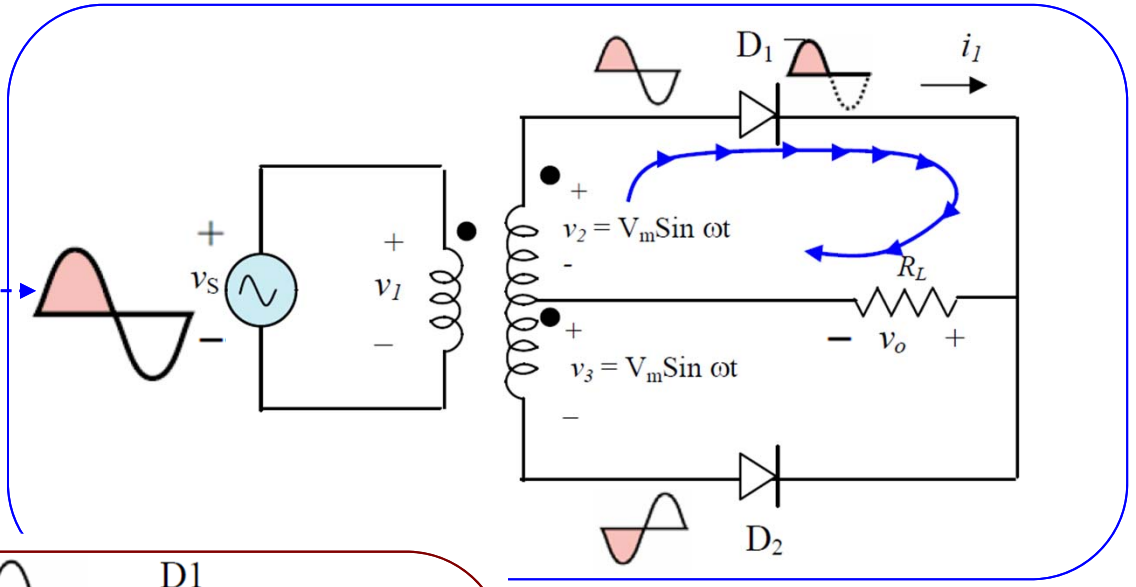


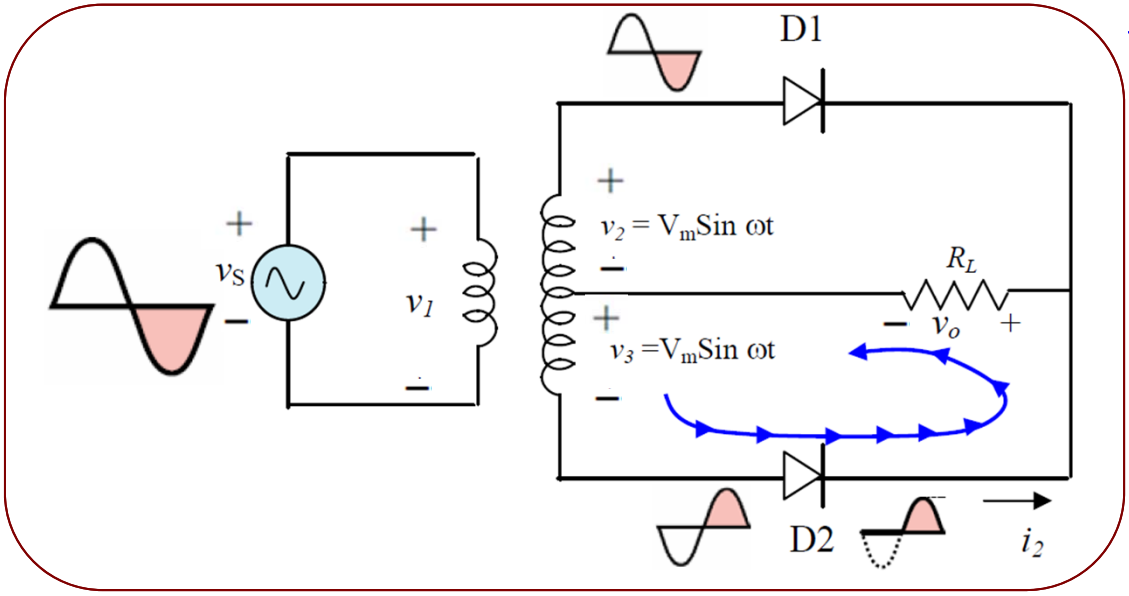


Solution…

The full wave rectifier…

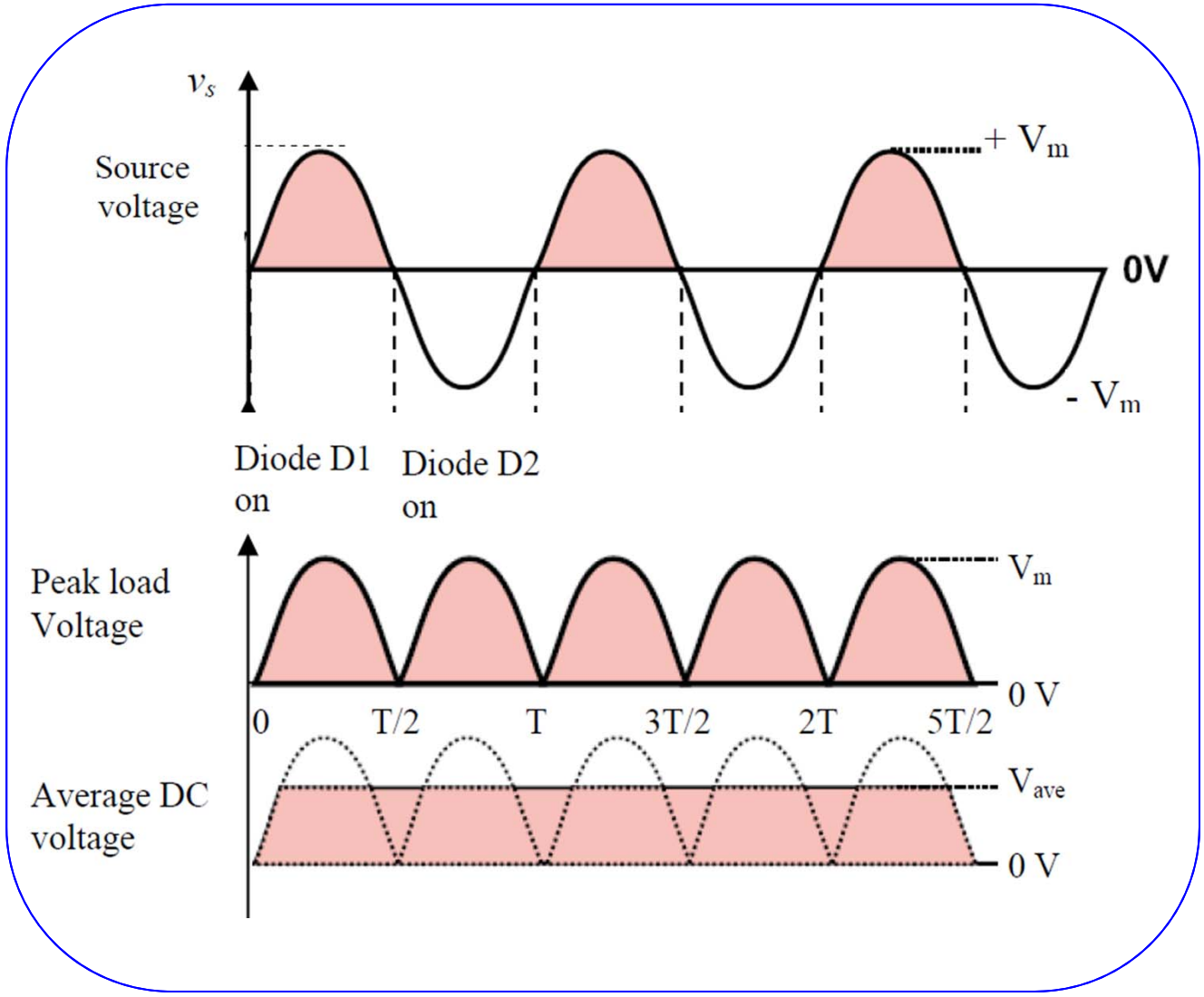
Circuit operation during the positive half cycle…





Circuit operation during the negative half cycle

Average Load Voltage & Current

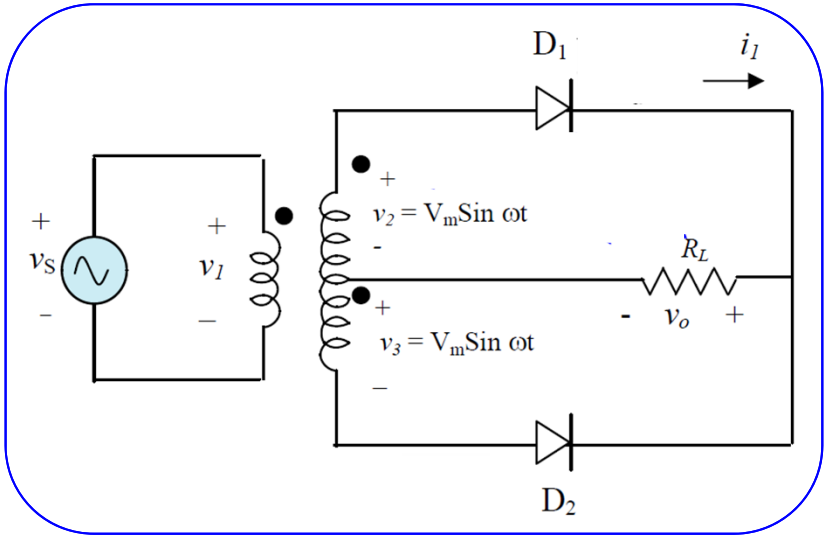


$$V_{ave} = \frac{2V_m}{\pi}$$

$$I_L = \frac{V_{ave}}{R_L} = \frac{2V_m}{\pi R_L}$$

Example 3

The transformer in the full-wave rectifier circuit has a turns ratio of 1:2. Its primary winding is connected across an AC source of 230V (rms), 50 Hz. The load resistor is 50 Ω. Determine the DC output voltage, peak-to-peak ripple in the output voltage, and output ripple frequency.



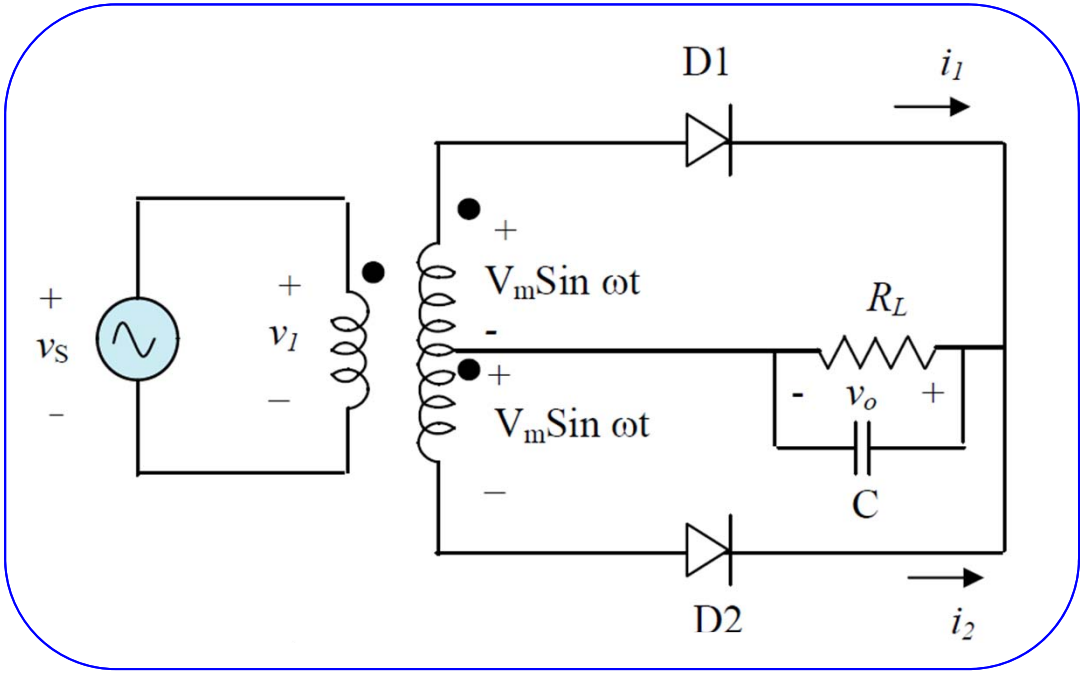
Solution: The rms value of the secondary voltage = 460 V. Thus, the rms value of v_2 (and v_3) = 230 V. The peak value of v_2 (and v_3): $V_m = \sqrt{2} \times 230 = 325.3$ V

DC output voltage (average load voltage): $V_{ave} = \dfrac{2V_m}{\pi} = 207$ V

Peak-to-peak ripple voltage: $V_{p\text{-}p} = V_m - 0 = 325.3$ V. Ripple frequency = 100 Hz (why?)

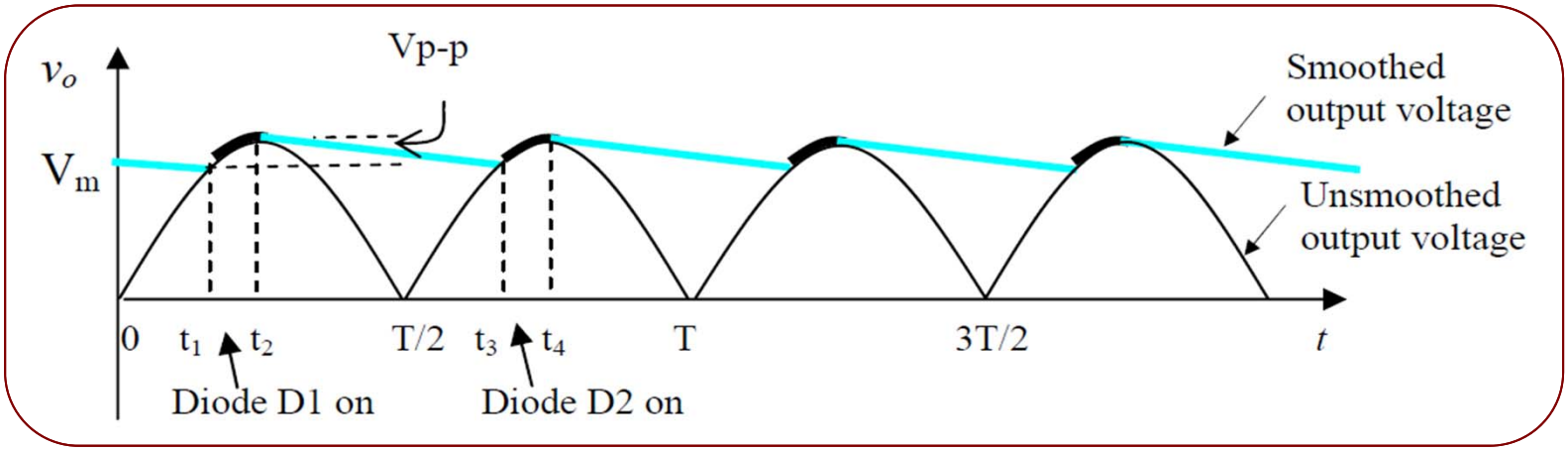
¤ The ripple is still very large, but its percentage is reduced to 157%...

Full Wave Rectifier with Capacitor

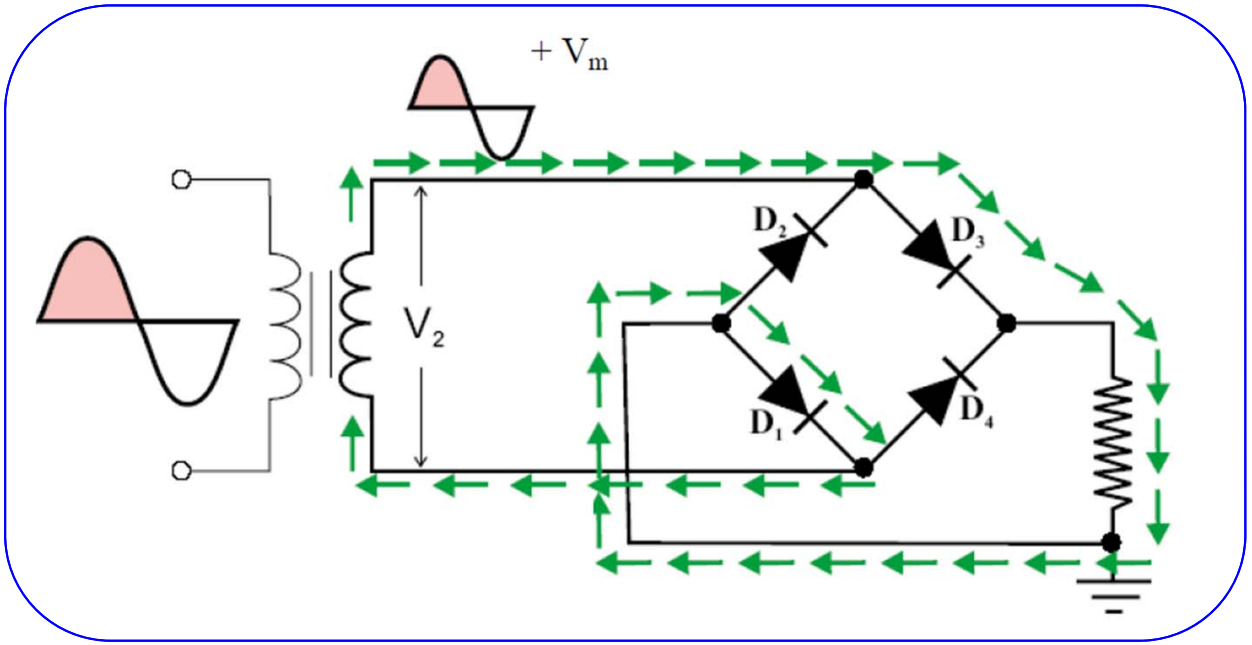


The required capacitance can be found to be (half of the value required for the half-wave rectifier):

$$C = \frac{I_L T}{2V_{p\text{-}p}} \quad \text{F}$$

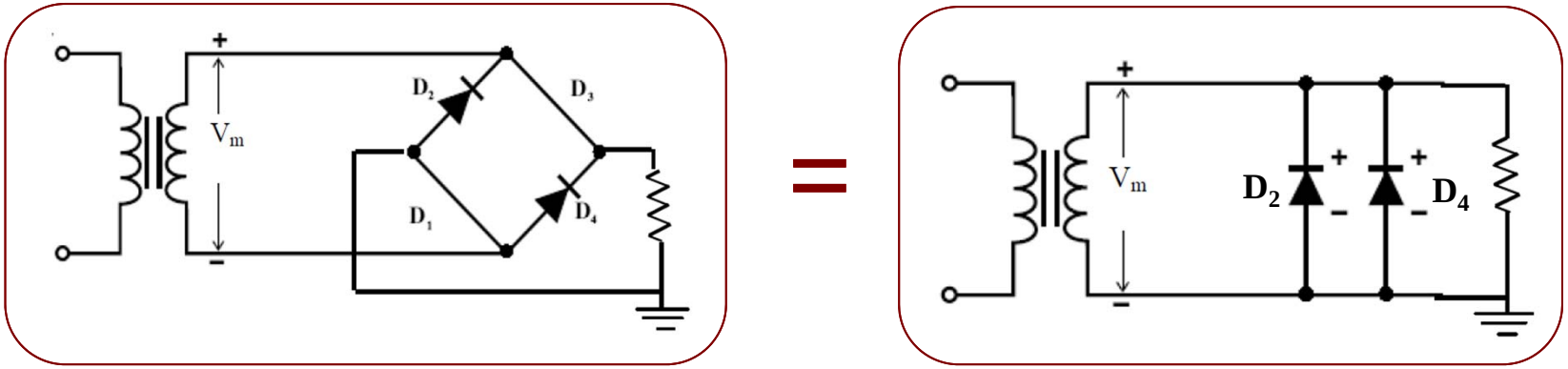


Full Wave Bridge Rectifier

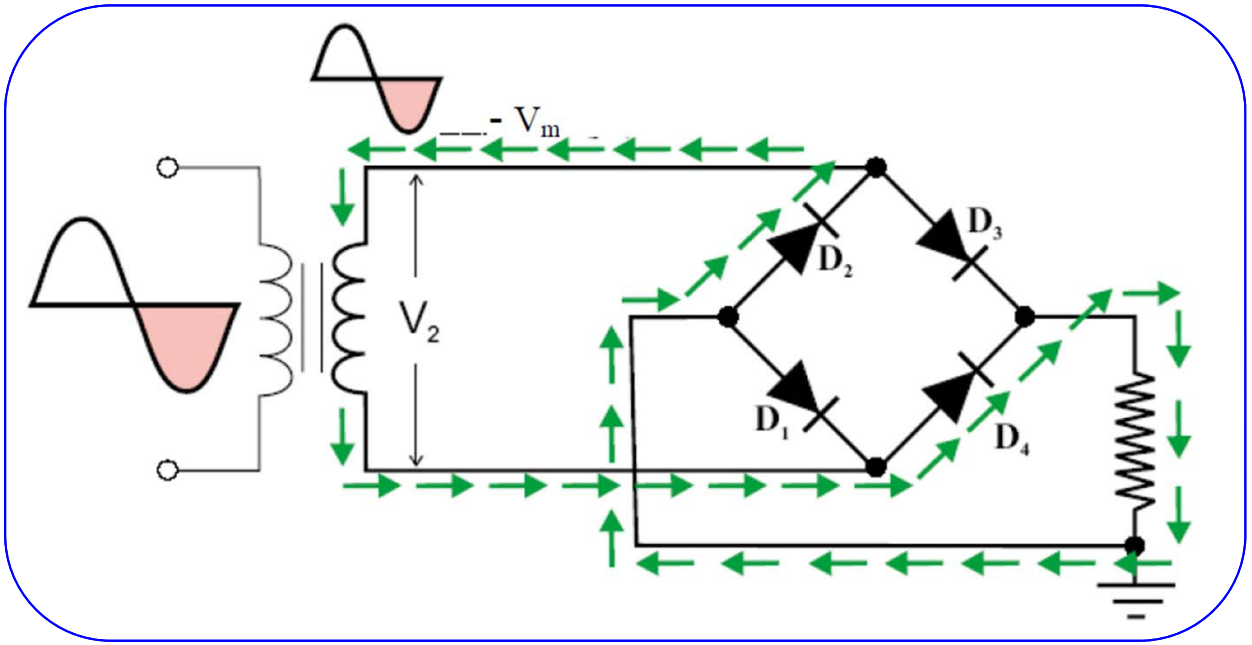


Circuit operation during the positive half cycle…

Equivalent circuit for positive half cycle: (What is the reverse voltage of D_2 and D_4?)

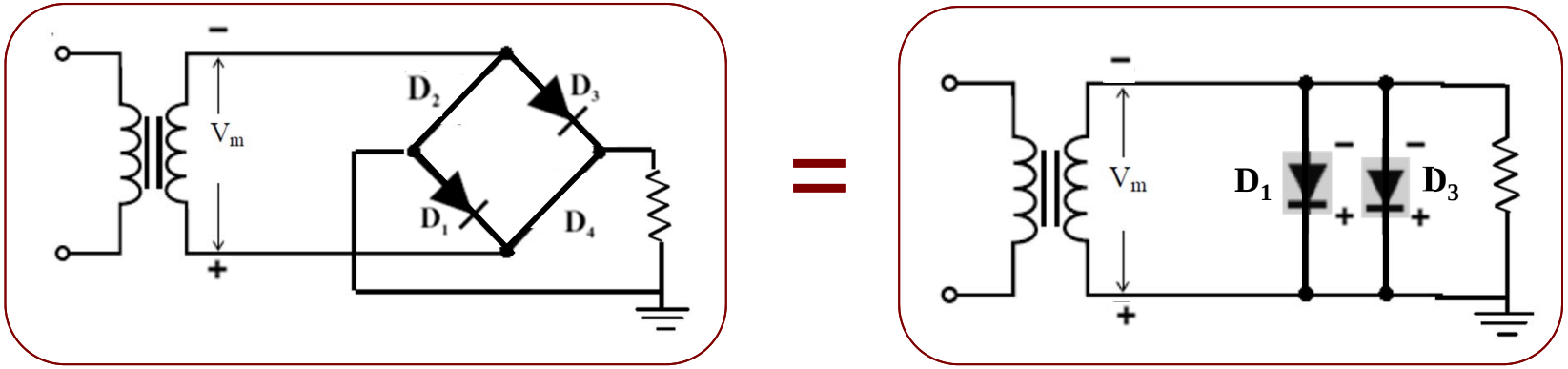


Operational Analysis



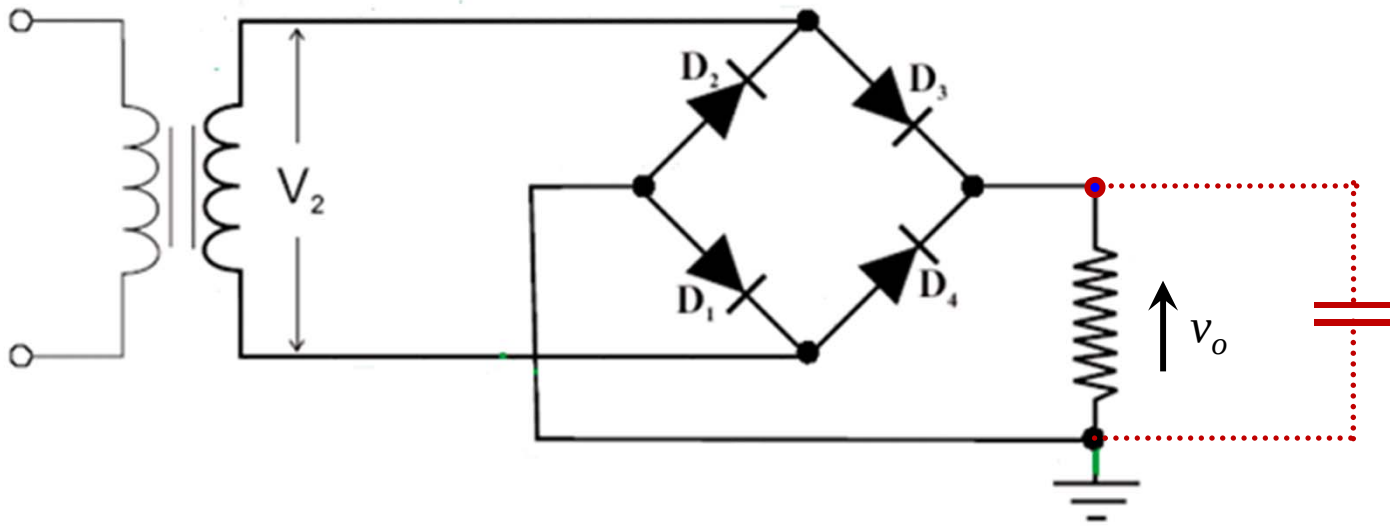
Circuit operation during the negative half cycle…

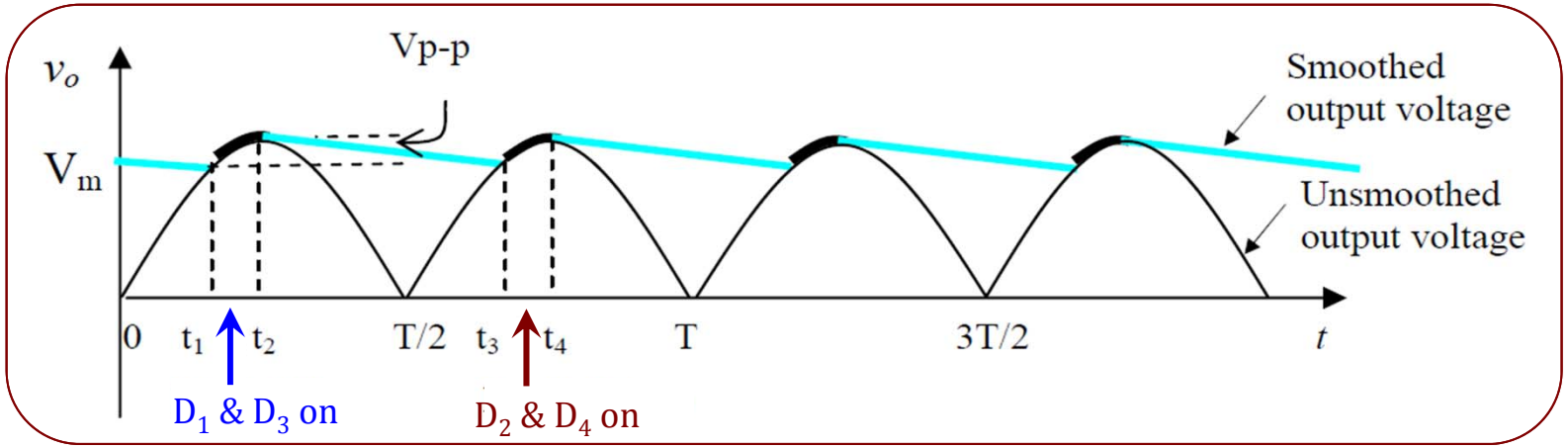
Equivalent circuit for negative half cycle: (What is the reverse voltage of D_1 and D_3?)



Full Wave Bridge Rectifier with Capacitor

As usual, in order to have a smoother output voltage, we can add a capacitor…





Video: How to Make a Real DC Power Supply?

DC Motors and Electric Generators

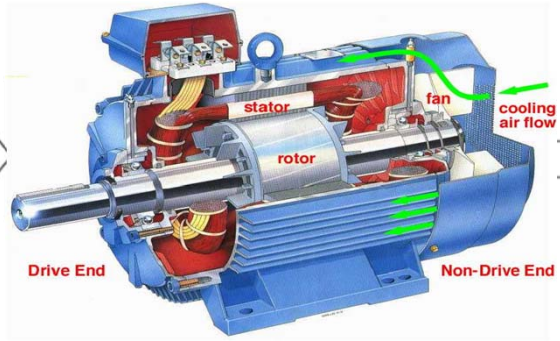
A Very Brief Introduction

Electromechanical Energy Conversion

Electric Motor

Electrical energy

Source



Mechanical energy

Load

Electric Generator

Mechanical energy

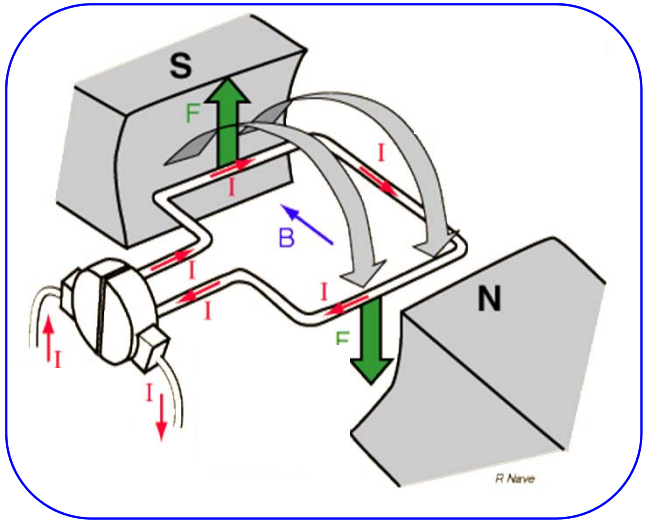
Source

Electrical energy

Load

Components of a DC Motor

DC motors consist of one set of coils, the armature winding, inside another set of coils or a set of permanent magnets, called the stator. Applying a voltage to the coils produces a torque in the armature, resulting in motion.



Stator

- The stator is the stationary outside part of a motor.
- The stator of a permanent magnet dc motor is composed of two or more permanent magnet pole pieces.
- The magnetic field can alternatively be created by an electromagnet. In this case, a DC coil is wound around a magnetic material that forms part of the stator.

Rotor

- The rotor is the inner part which rotates.
- The rotor is composed of windings (called armature windings) which are connected to the external circuit through a mechanical commutator.
- Both stator and rotor are made of ferromagnetic materials.

DC Motor Basic Principles

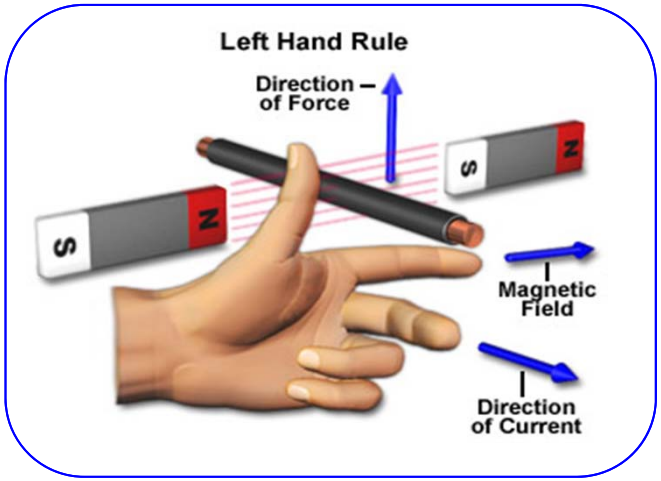
If electric energy is supplied to a conductor lying perpendicular to a magnetic field, the interaction of current flowing in the conductor and the magnetic field will produce magnetic **force**, which attempts to move the conductor in a direction perpendicular to the magnetic field, and is given by

$$F = B \cdot I \cdot L \quad \text{(Newton)}$$

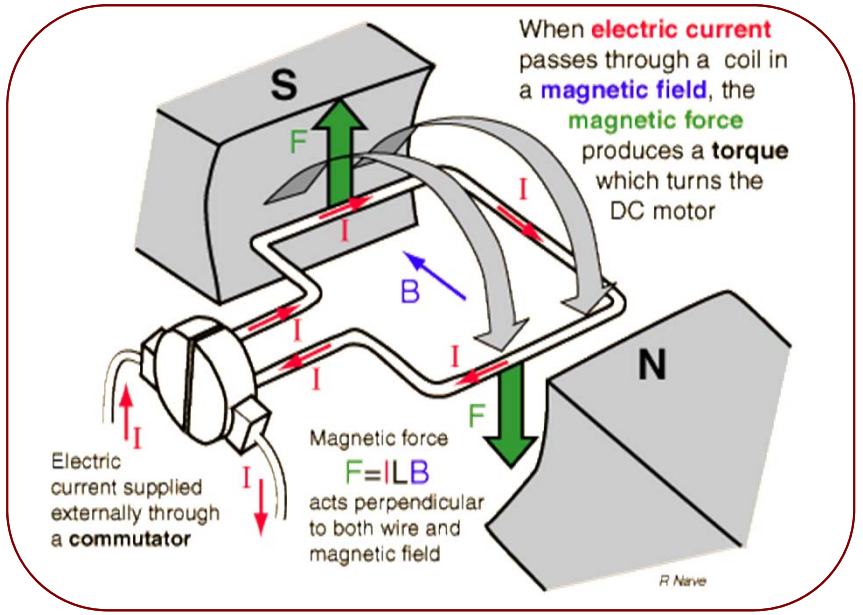
B : magnetic flux density; L : conductor length

I : current flowing in the conductor

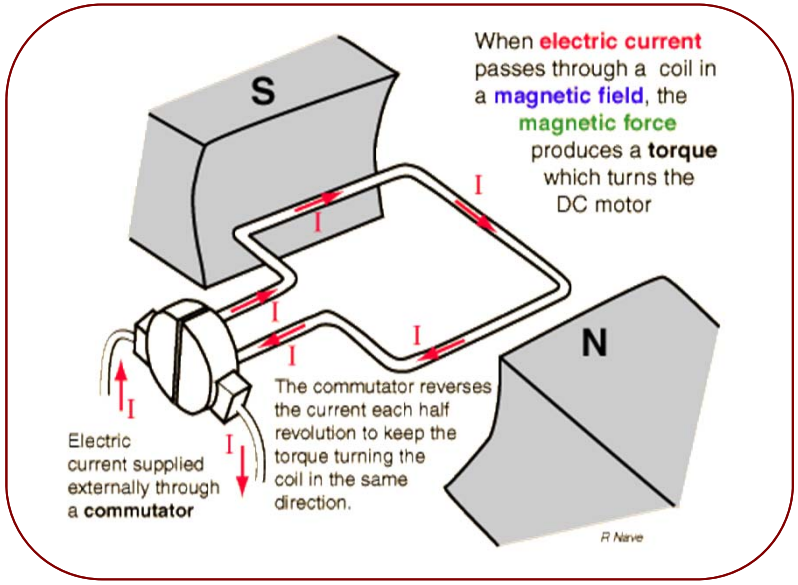
John A. Fleming
English
1849–1945

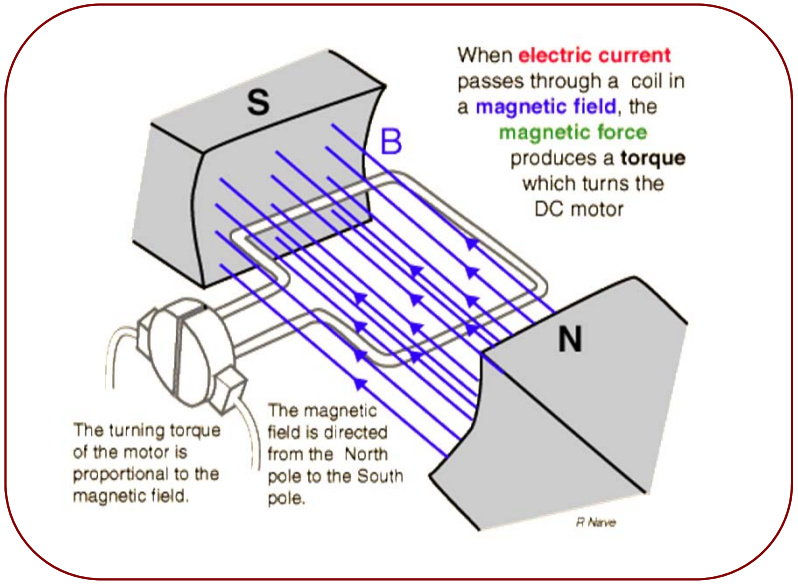


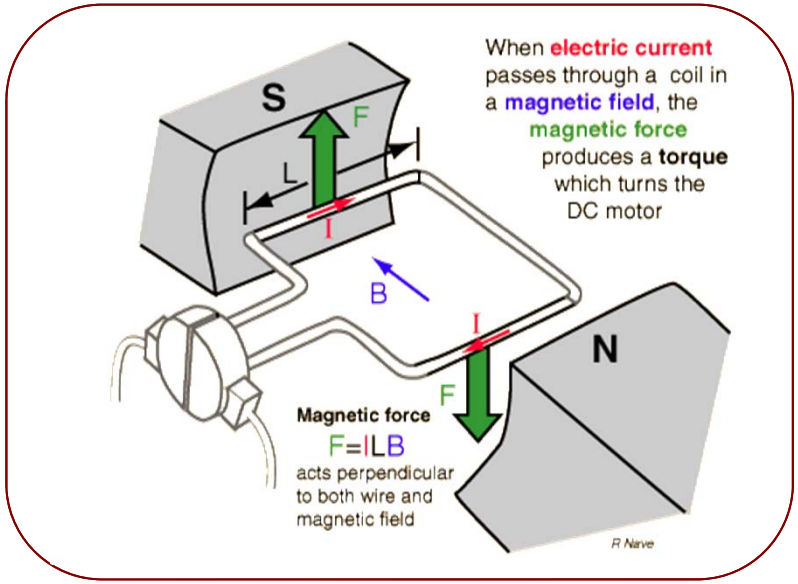
Fleming's Left Hand Rule

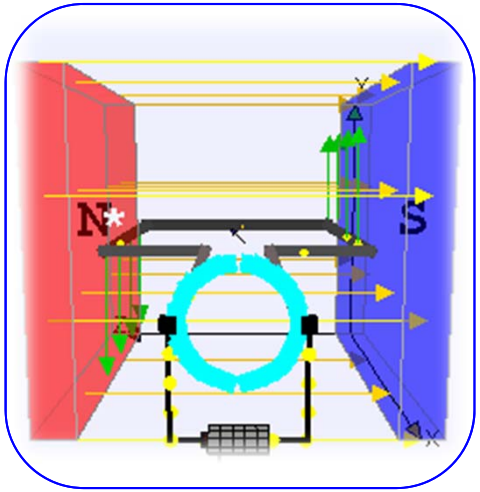


DC Motor Operational Principle





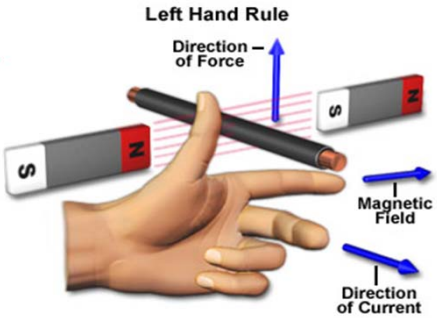


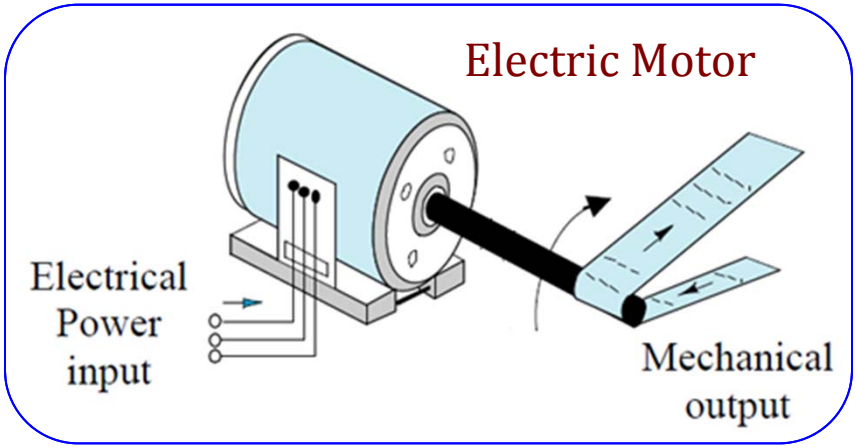


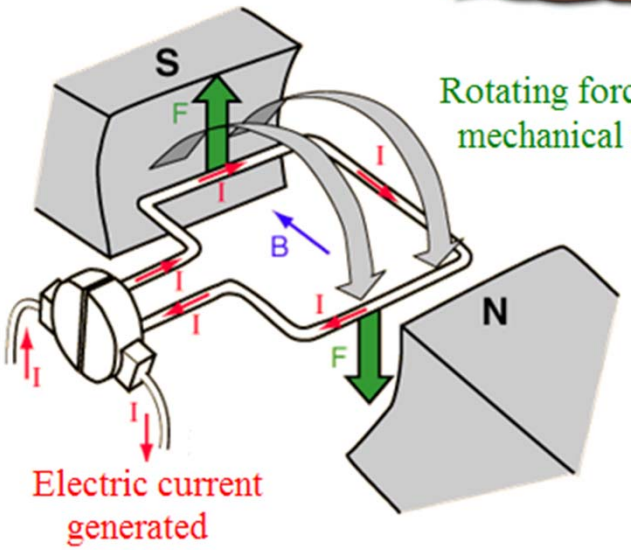
An actual DC motor might contain many turns of armature windings in its rotor…

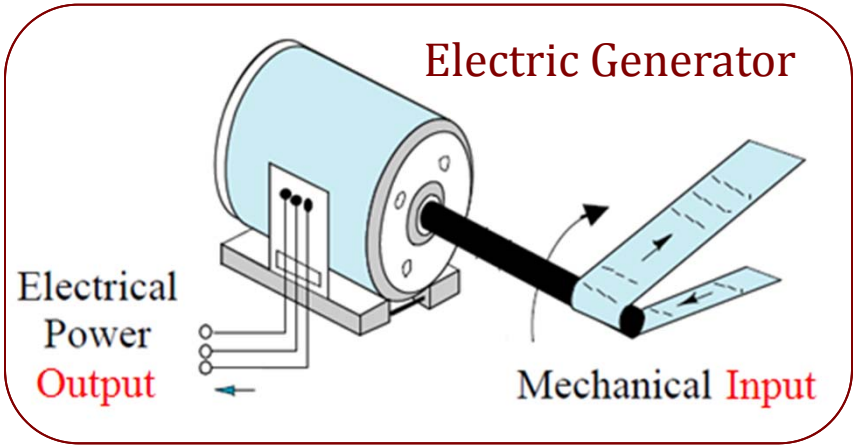
Electric Generators

An **electric generator** is a device converting mechanical energy to electrical energy. Electric motors and generators have many similarities. In fact, many motors can be mechanically driven to generate electricity…









$$F = B \cdot I \cdot L$$

F : the force; B : flux density

L : conductor length

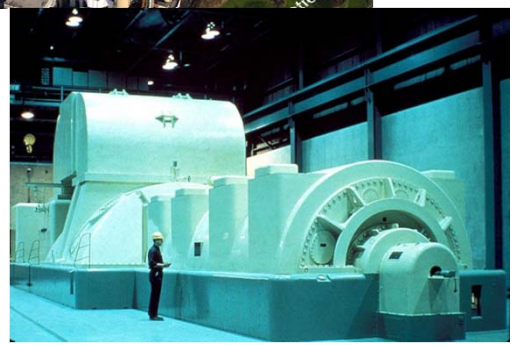
I : current flowing in the conductor

Video: Motors and Generators

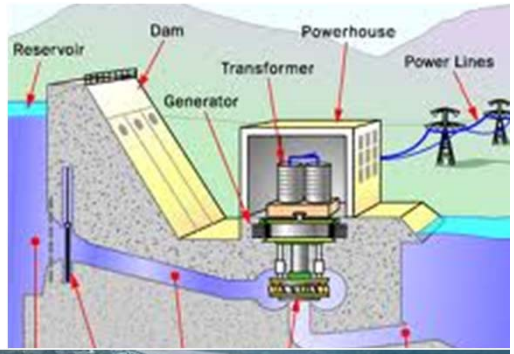
Examples of Electric Generators

The source of mechanical energy used in real situations for generating electricity may be a reciprocating or turbine steam engine, water falling through a turbine or waterwheel, an internal combustion engine, a wind turbine, a hand crank, compressed air or any other source of mechanical energy.

power plant and steam turbine generator



hydro power and water dam



Digital Logic Circuits

Introduction

Many scientific, industrial and commercial advances have been made possible by the advent of computers. Digital Logic Circuits form the basis of any digital system, such as computers, laptops and smart phones. In this topic, we will study the essential features of digital logic circuits, which are at the heart of digital computers.

Digital Logic Circuits may be subdivided into Combinational Logic Circuits and Sequential Logic Circuits. In EG1108, our focus will be on Combinational Logic Circuits. These circuits can be easily analyzed/designed using Boolean Algebra, which is the mathematics associated with binary systems.

We will see how Combinational Logic Circuits can be designed and used for interesting practical problems. Circuit minimization techniques such as Karnaugh maps for simplification of combinatorial logic circuits will also be covered.

Some Devices Involved Digital Logic Circuits

Digital Watch

Aircraft

Personal Computer

Smart Phones

Super Computer

Traffic Light

Learning Objectives

The main learning objectives for this topic are as follows:

1. To understand basic terminology, types of logic gates
2. To understand the basic operations used in computers and other digital systems
3. To study basic rules of Boolean algebra, De Morgan's laws
4. To study Karnaugh maps for circuit minimization

Flowchart for Designing a Logic Circuit

Product or Design Specifications

Identify necessary input and output variables

Obtain a relationship between input and output variables

Determine a logical expression characterizing the input-output relationship

Implement the logical expression using logic gates

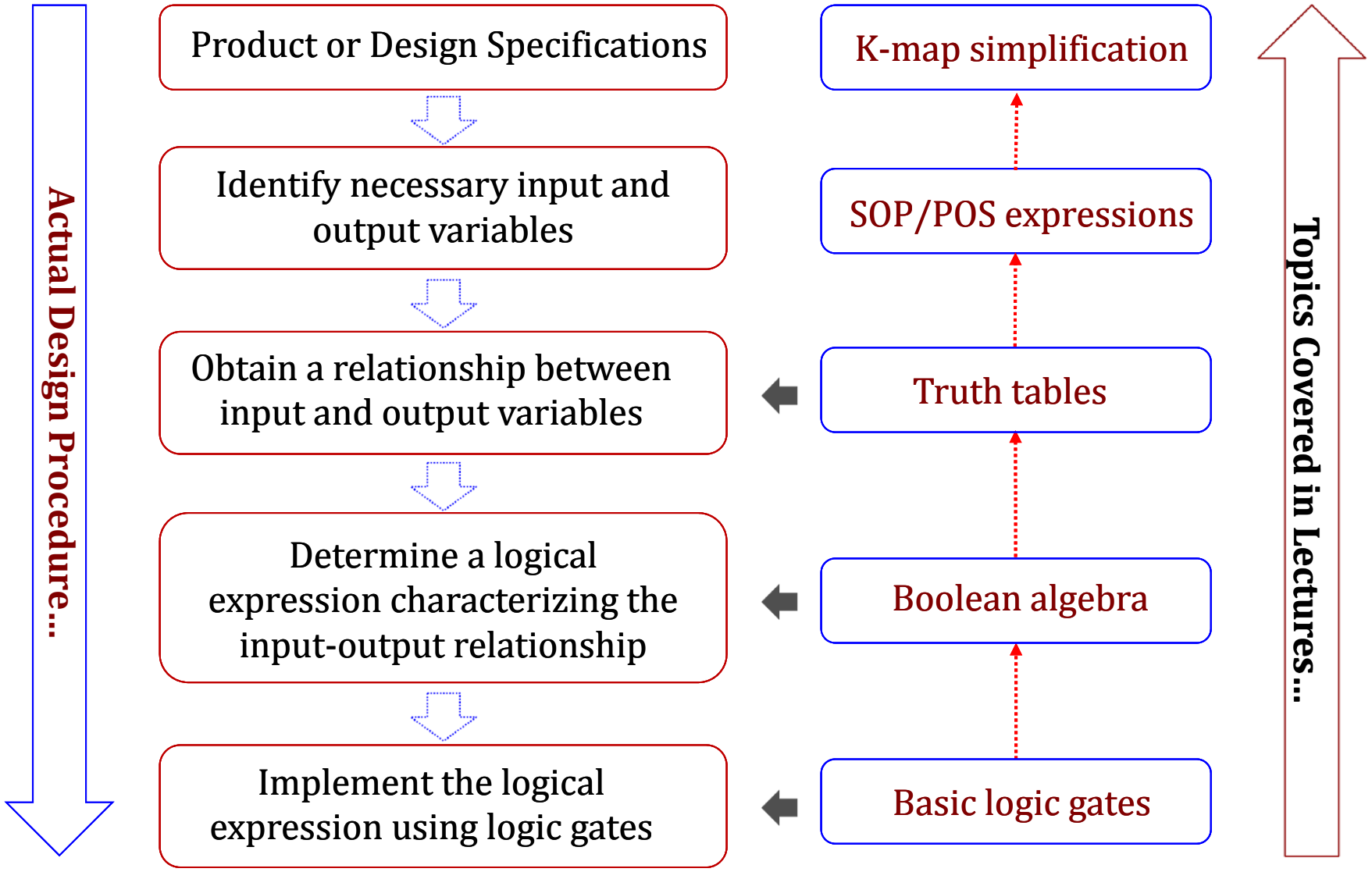
Example: Design an LED panel to display Letters ***E*** or ***C***

Two LED bars as shown in **R**ed and **B**lack on left are sufficient

To display Letter ***E***, both ***R*** and ***B*** are required to be on;
To display Letter ***C***, ***R*** is required to be on and ***B*** is to be off

$E = R \cdot B;\ C = R \cdot \overline{B}$

Flowchart for Designing a Logic Circuit (cont.)



Number Systems

A number system is an ordered set of symbols (digits) with relationships defined for addition, subtraction, multiplication and division. The base of the number system is the total number of digits in the system. For example, for the usual **decimal** system, the set of digits is $\{0,1,2,3,4,5,6,7,8,9\}$ and hence the base is **ten** (B = 10). In the **binary** system, the set of digits (bits) is $\{0,1\}$ and hence the base is **two** (B = 2).

There are two possible ways of writing a number in a given system: *positional notation* and the *polynomial representation*.

Positional Notation: A number N can be written in positional notation as follows:

$$N = \left(b_{n-1}\, b_{n-2} \cdots b_2\, b_1\, b_0 \cdot b_{-1}\, b_{-2} \cdots b_{-m}\right)_B \quad \text{for example} \quad N = \left(2536.47\right)_{10}$$

Polynomial Representation: The above number can also be written as a polynomial of the form

$$N = b_{n-1}B^{n-1} + b_{n-2}B^{n-2} + \cdots + b_2B^2 + b_1B^1 + b_0B^0 + b_{-1}B^{-1} + b_{-2}B^{-2} + \cdots + b_{-m}B^{-m}$$

For example, $N = \left(2536.47\right)_{10} = 2\times10^3 + 5\times10^2 + 3\times10^1 + 6\times10^0 + 4\times10^{-1} + 7\times10^{-2}$.

Binary-Number System

The binary number is a base 2 system with two distinct digits (bits), 1 and 0. It is expressed as a string of 0s and 1s and a binary point, if a fraction exists. To convert from the binary to decimal system, express the binary number in the polynomial form and evaluate this polynomial by using decimal-system addition.

The following are mappings of some binary numbers to their decimal counterparts:

Binary		*Polynomial Representation*		Decimal
000	=	$0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	=	0
001	=	$0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	=	1
010	=	$0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$	=	2
011	=	$0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$	=	3
100	=	$1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$	=	4
101	=	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$	=	5
110	=	$1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$	=	6
111	=	$1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$	=	7

Similarly,

Binary		Decimal
1000	=	8
1001	=	9
1010	=	10
1011	=	11
1100	=	12
1101	=	13
1110	=	14
1111	=	15

Boolean Algebra and Logic Gates

Boolean algebra is the mathematics of digital logic and is particularly useful in dealing with the binary-number system. Boolean algebra is used in the analysis and synthesis of logical expressions. Logical expressions are constructed using logical-variables and operators. The value of any logical expression boils down to any one of the two logical constants: **true** and **false.**

In Boolean algebra, a logical variable X may take any one of the two possible values 1 and 0. $X = 1$ and $X = 0$, which may represent respectively

- **truth** or **falsehood** of a statement
- **on** or **off** states of a switch
- **high** or **low** of a voltage level

George Boole
English
1815–1864

A **logical expression** is a finite combination of logical variables (which are also called **input** variables) that are well-formed according to the rules of Boolean algebra. The value of the logical expression is referred to as the **output** variable (which itself is also a logical variable).

Truth Tables

A **truth table** is a mathematical table used in logic to compute the functional values of logical expressions on each of their functional arguments, that is, on each combination of values taken by their logical variables. In particular, truth tables can be used to tell whether a propositional expression (output) is true or false for all possible combinations of input values. The following are the **examples** of the truth tables for 1, 2, 3 and 4 input variables, respectively:

Input	Output
A	Z
0	1
1	0

	Input Variables		Output
	A	B	W
0	0	0	1
1	0	1	1
2	1	0	1
3	1	1	0

	A	B	C	R
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

	A	B	C	D	Y
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0

Logic Circuits

Electrical circuits designed to represent logical expressions are popularly known as logic circuits. Such circuits are extensively used in industrial processes, household appliances, computers, communication devices, traffic signals and microprocessors to make important logical decisions. Logic circuits are usually represented by logic operations involving Boolean variables.

There are three basic logic operations as listed below:

- **NOT** operation
- **AND** operation
- **OR** operation

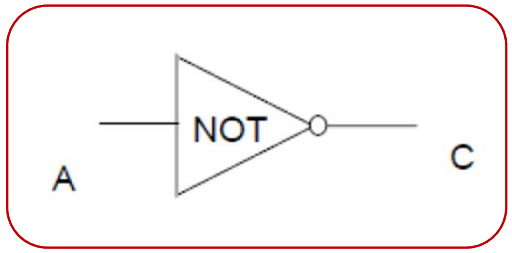
We will illustrate these basic logical operations in the following sections using Boolean variables A and B. A logic gate is an electronic circuit/device which makes the logical decisions based on these operations. Logic gates have one or more inputs and only one output. The output is active only for certain input combinations. Logic gates are the building blocks of any digital circuit. Logic gates are also called switches.

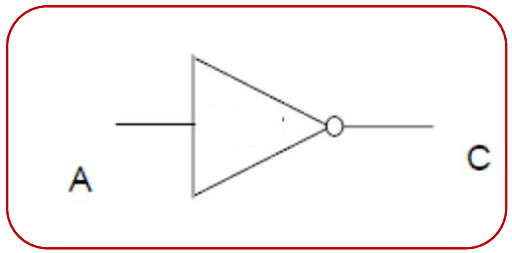
NOT Operation

NOT operation is represented by

$$C = \overline{A}$$

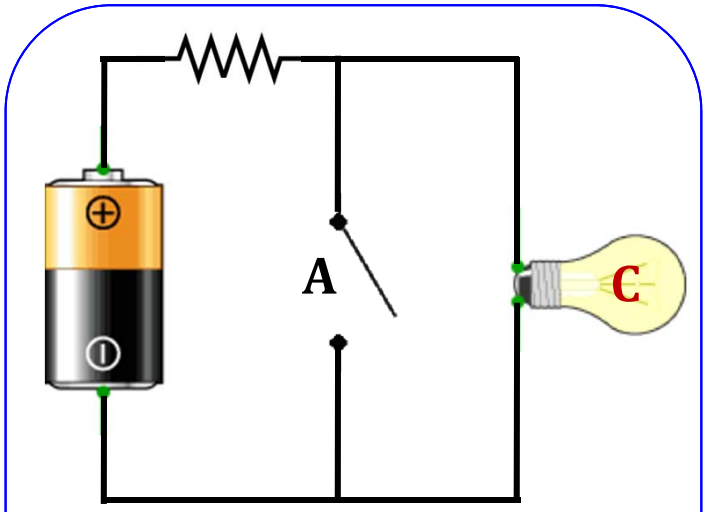
The NOT gate has only one input which is then inverted by the gate. Here, $\overline{A}$ is the 'complement' of A. The symbol and truth table for the operation are shown below:





A	C
0	1
1	0

Truth Table for NOT Gate

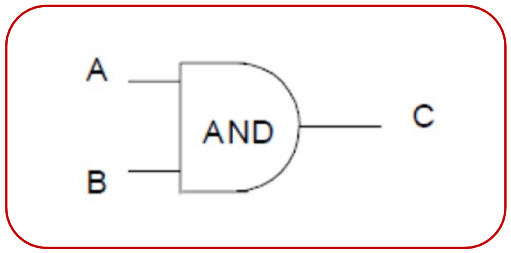


Switch Circuit for a NOT Gate

AND Operation

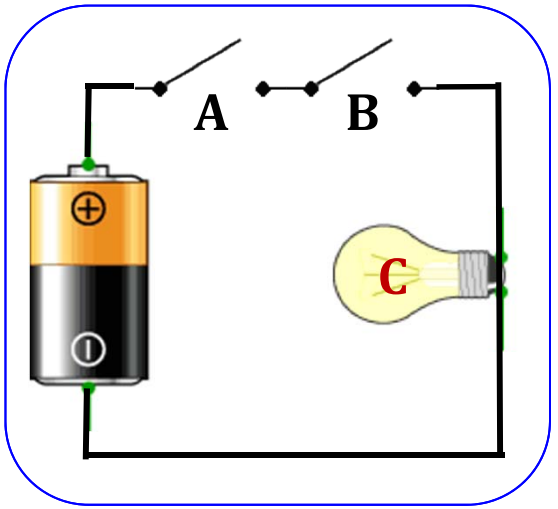
AND operation is represented by C = A • B

Its associated TRUTH TABLE is shown below. A truth table gives the value of output variable (here C) for all combinations of input variable values (here A and B). Thus in an **AND** operation, the output will be **1** (True) only if all of the inputs are **1** (True).



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

Truth Table for AND Gate



The following relationships can be easily derived:

$A \cdot A = A$ $\quad 1 \cdot A = A$ $\quad 0 \cdot A = 0$

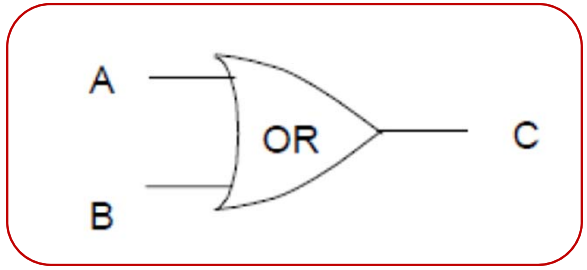
$A \cdot \bar{A} = 0$ $\quad A \cdot B = B \cdot A$

Note: The • sign can be omitted when indicating an AND operation. Thus, C = A • B and C = A B mean the same operation.

OR Operation

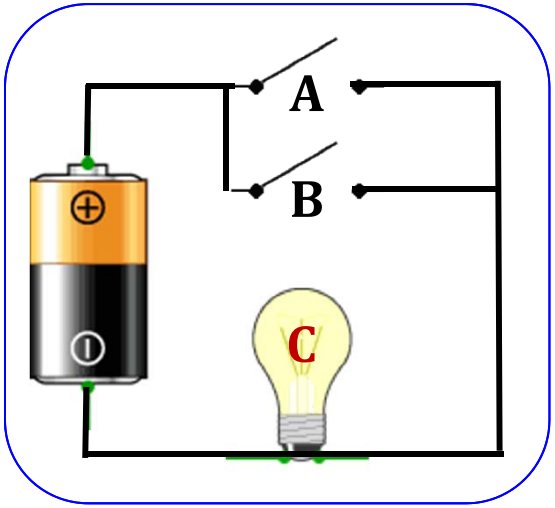
OR operation is represented by C = A + B

Here A, B and C are logical (Boolean) variables and the + sign represents the logical addition, called an OR operation. The symbol for the operation (called an OR gate) is shown below. Its associated TRUTH TABLE is shown below. Thus in an OR operation, the output will be **1** (True) if either of the inputs is **1** (True). If both inputs are **0** (False), only then the output will be **0** (False). Notice that though the symbol + is used, the logical addition described above does not follow the rules of normal arithmetic addition.



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

Truth Table for OR Gate



Switch Circuit for an OR Gate

The following relationships can be easily derived:

$A + \bar{A} = 1$ $\quad A + A = A$ $\quad 0 + A = A$

$1 + A = 1$

Multivariable Logic Gates

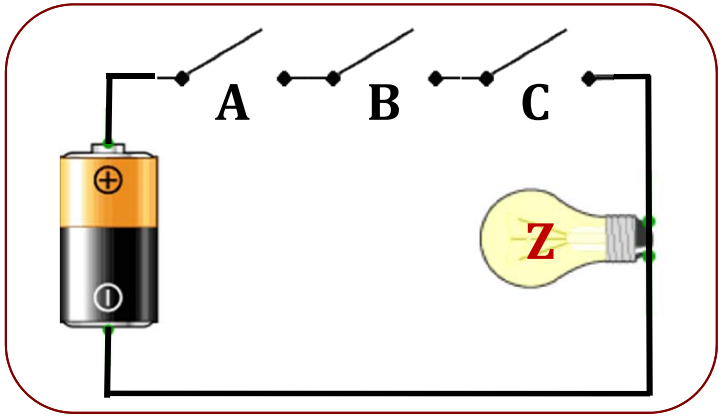
Many logic gates can be implemented with more than two inputs, and for reasons of space in circuits, usually multiple input, complex gates are made.

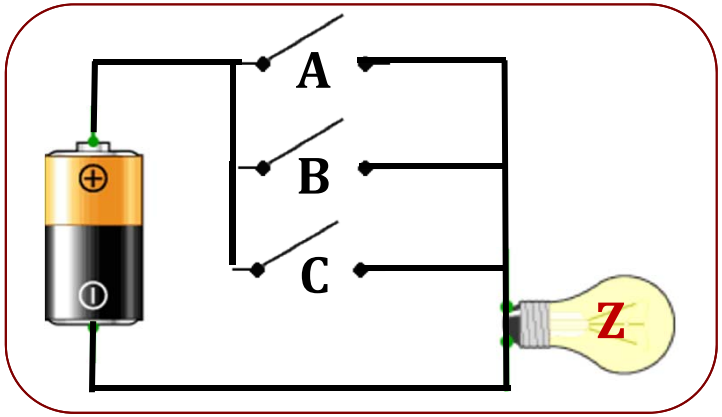
3-Input **AND** Gate Truth Table

Input			Output
A	B	C	Z = A·B·C
0	0	0	0
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	1

3-Input **OR** Gate Truth Table

Input			Output
A	B	C	Z = A+B+C
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1





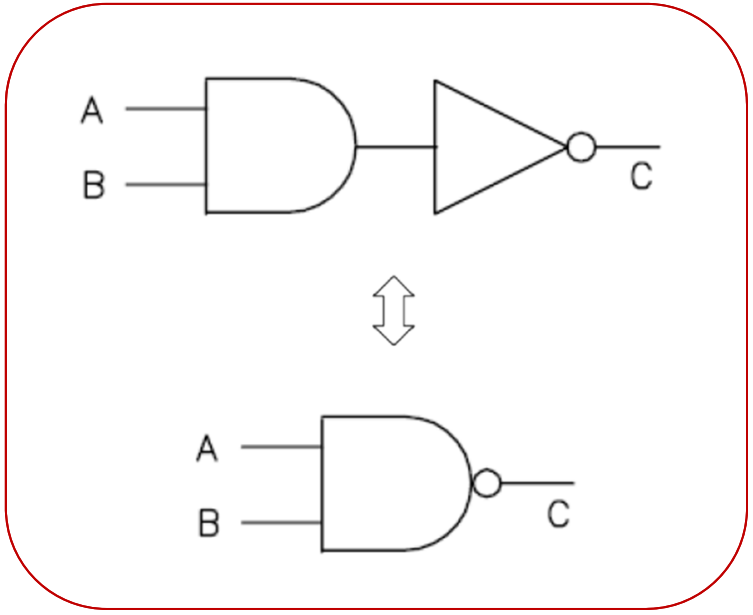
$A \bullet (B \bullet C) = (A \bullet B) \bullet C = A \bullet B \bullet C$

$(A + B) + C = A + (B + C) = A + B + C$

NAND Gate

We could combine AND and NOT operations together to form a NAND gate. Thus the logical expression for a NAND gate is $C = \overline{A \cdot B}$.

The symbol and truth table are given in the following figure. The NAND gate symbol is given by an AND gate symbol with a circle at the output to indicate the inverting operation.



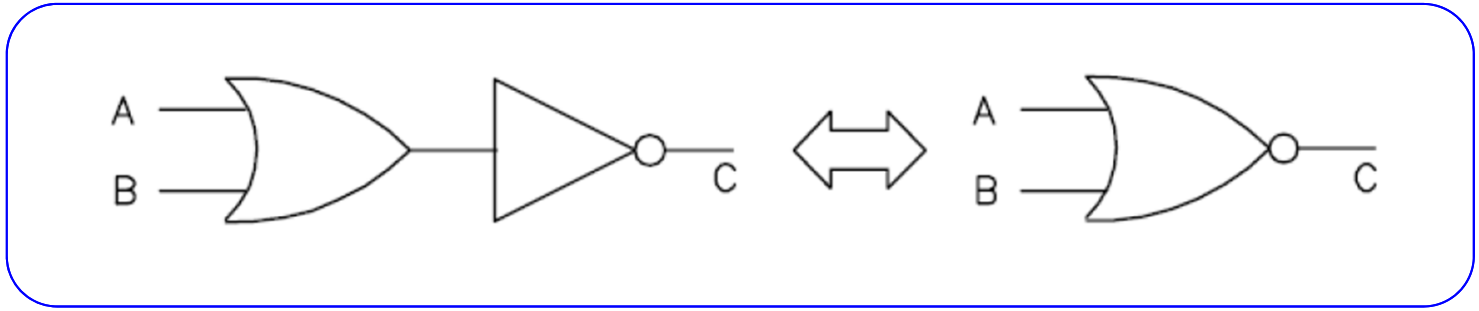
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

Truth Table for NAND Gate

NOR Gate

Similarly, OR and NOT gates could be combined to form a NOR gate.

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



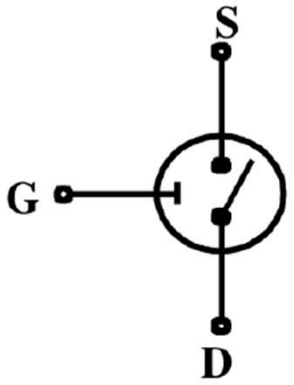
A	B	C
0	0	1
0	1	0
1	0	0
1	1	0

Truth Table for NOR Gate

Realization of Logic Gates by Ideal Transistors ⋆

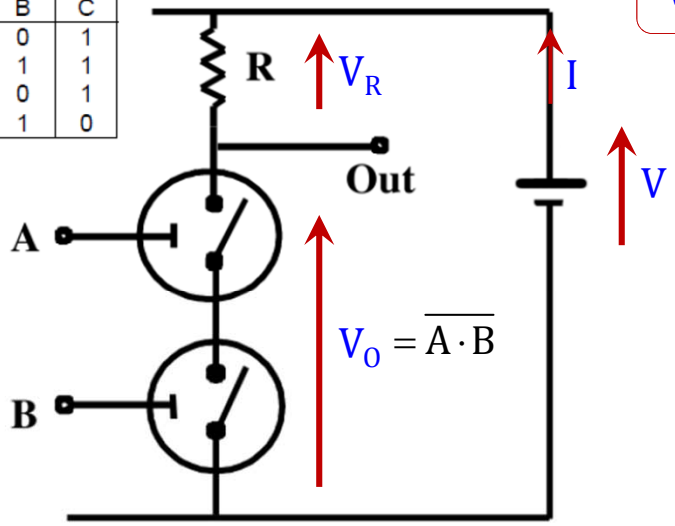
Logic gates can be realized using ideal transistors, which have the properties:

1. There is no connection between G and D and G and S
2. If the voltage between G and D (Vgd) is less than x volts the switch is open and there is no connection between D and S.
3. If the voltage Vgd is greater than x volts the switch is closed and D is connected directly to S.



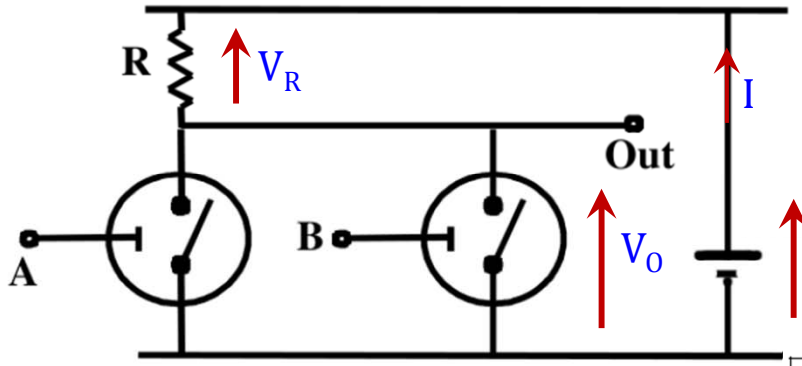
$\overline{A \cdot B}$

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



(a) NAND Gate

$V = V_O + V_R \rightarrow V_O = V$ when $V_R = 0$, i.e., $I = 0$

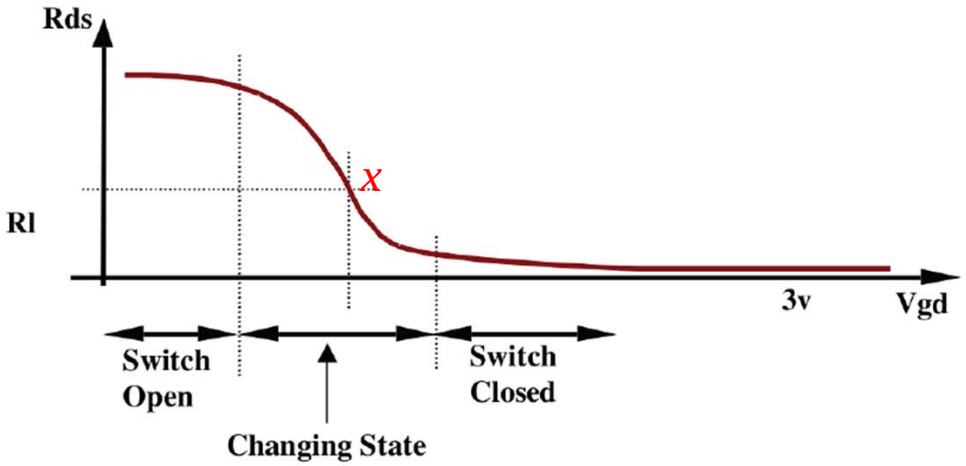


(b) NOR Gate

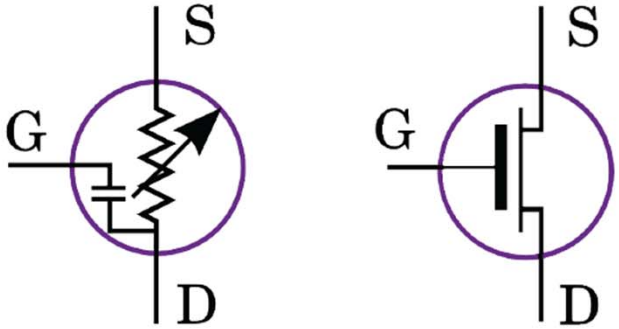
$\overline{A+B}$

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

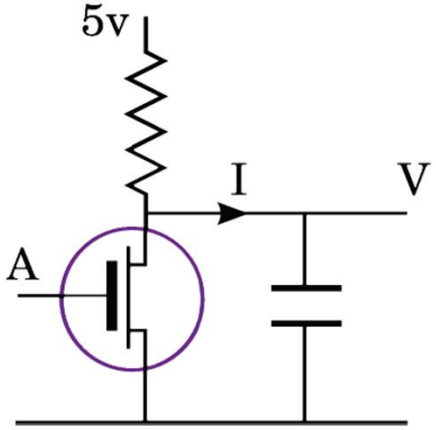
Properties of Practical Transistors ⋆

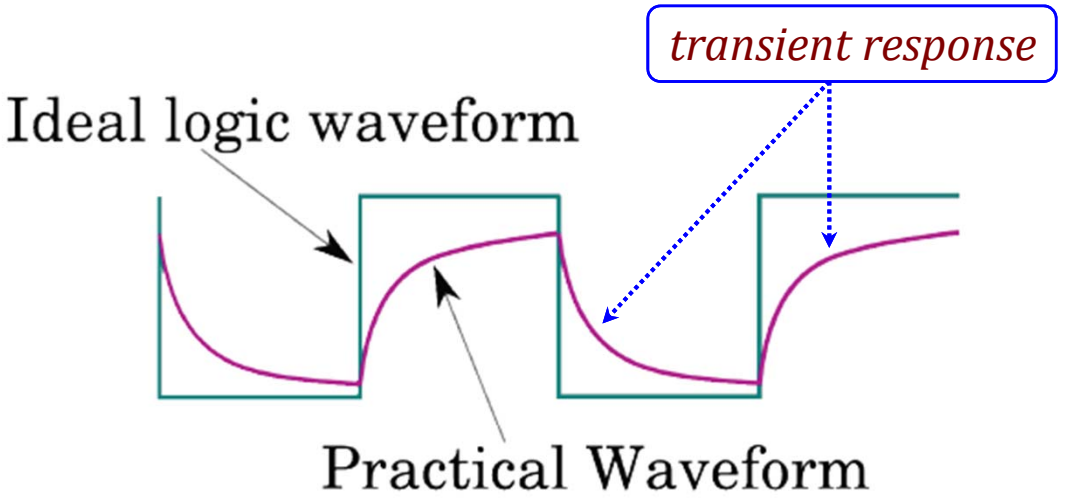


(a) Typical transistor characteristic



(b) NMOS transistor

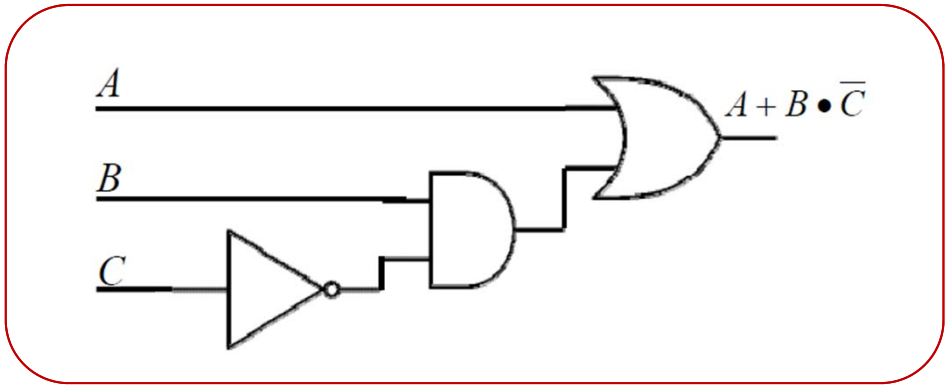




Elements of Boolean Algebra

A symbolic binary logic expression consists of binary variables and the operators AND, OR and NOT (e.g. A+ B • $\overline{C}$). The possible values for any Boolean expression can be tabulated in a **truth table.**

Boolean algebra expressions can be implemented by interconnection of AND gates, OR gates, and inverters.



As can be seen, the number of simple gates needed to implement an expression is equal to the number of operations in the Boolean expression. We could use the rules of Boolean Algebra or Karnaugh Maps to simplify a given Boolean expression. This would allow the given expression to be implemented using less number of gates.

Basic Laws of Boolean Algebra

Some of the basic rules of Boolean algebra that may be used to simplify the Boolean expressions are shown on the right. These rules may be proved using the truth tables. Essentially, we consider all combinations of inputs and show that in all cases the LHS expression and RHS expression lead to the same result. Such a method of proving logical equations is known as proof by perfect induction.

Rules:

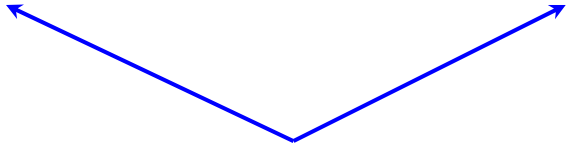
1: $0 + A = A$
2: $1 + A = 1$
3: $A + A = A$
4: $A + \overline{A} = 1$
5: $0 \cdot A = 0$
6: $1 \cdot A = A$
7: $A \cdot A = A$
8: $A \cdot \overline{A} = 0$
9: $A = \overline{\overline{A}}$
10: $A + B = B + A$
11: $A \cdot B = B \cdot A$
12: $A + (B + C) = (A + B) + C$
13: $(A \cdot B) \cdot C = A \cdot (B \cdot C)$
14: $A \cdot (B + C) = A \cdot B + A \cdot C$
15: $A + A \cdot B = A$
16: $A \cdot (A + B) = A$
17: $(A + B) \cdot (A + C) = A + B \cdot C$
18: $A + \overline{A} \cdot B = A + B$

Example 1

Show by perfect induction that $A + \overline{A} \cdot B = A + B$ [See Rule 18].

Proof: Let us create the following table and show the LHS = RHS for all the values of A and B.

		RHS			LHS
A	B	$A + B$	$\overline{A}$	$\overline{A} \cdot B$	$A + \overline{A} \cdot B$
0	0	0	1	0	0
0	1	1	1	1	1
1	0	1	0	0	1
1	1	1	0	0	1



identical ✓

De Morgan's Laws

These laws are very useful in simplifying Boolean expressions. According to De Morgan's theorem, we have

$$\overline{A+B} = \overline{A}\cdot\overline{B} \qquad \overline{A+B+C} = \overline{A}\cdot\overline{B}\cdot\overline{C}$$

$$\overline{A\cdot B} = \overline{A}+\overline{B} \qquad \overline{A\cdot B\cdot C} = \overline{A}+\overline{B}+\overline{C}$$

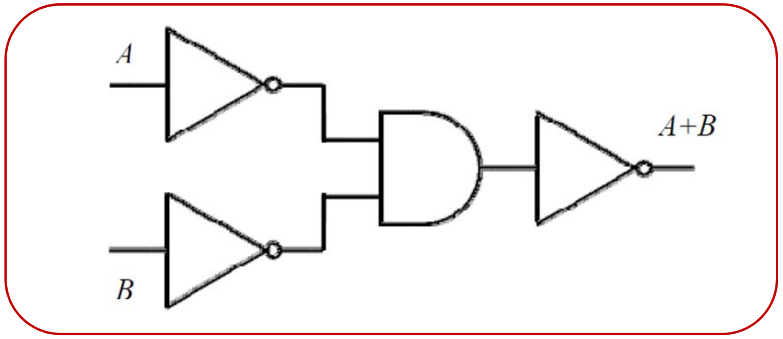
Notice that the De Morgan's Laws give the link between the OR operation and the AND operation. Application of De Morgan's theorem makes it easy to design logic circuits using NAND and NOR logic gates which we will soon see.

$$A+B = \overline{\overline{A+B}} = \overline{\overline{A}\cdot\overline{B}}$$

$$A\cdot B = \overline{\overline{A\cdot B}} = \overline{\overline{A}+\overline{B}}$$

Augustus De Morgan
English
1806–1871

Because of the above relationships, any logical function can be implemented by using only (i) AND and NOT gates or (ii) OR and NOT gates. Thus, an OR gate can be implemented with AND and NOT gates as shown on the right.



Some Side Notes on Boolean Algebra

In Boolean algebra, the equality of $A + B = A + C$ does not only imply that $B = C$.

This can be shown by a specific the example: Let $A = 1$, $B = 1$ and $C = 0$. Then, we have

$$A + B = A + C = 1$$

Obviously, $B \neq C$.

Similarly, in Boolean algebra, the equality of $A \cdot B = A \cdot C$ does not only imply that $B = C$ either.

This can be shown by letting $A = 0$, $B = 1$ and $C = 0$. Then, we have

$$A \cdot B = A \cdot C = 0$$

Again, it is obvious that $B \neq C$.

Lastly, we note that $A + B \cdot C \neq (A + B) \cdot C$.

Note: In Boolean algebra, we do not define logical 'subtraction' and 'division'.

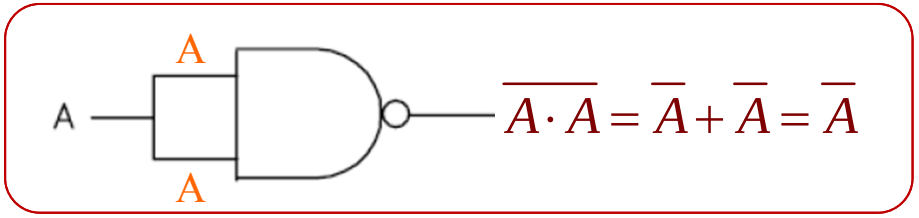
Universality of NAND and NOR Gates

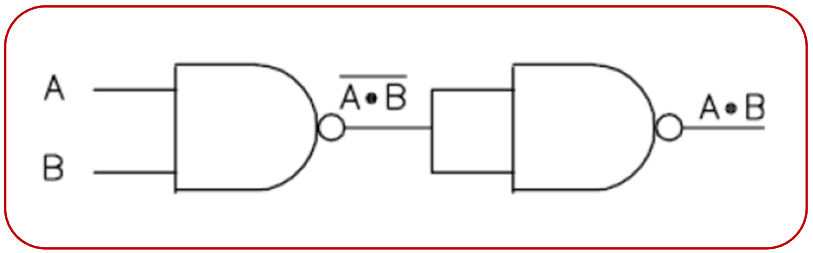
$$\overline{A+B} = \overline{A}\cdot\overline{B}$$
$$\overline{A\cdot B} = \overline{A}+\overline{B}$$

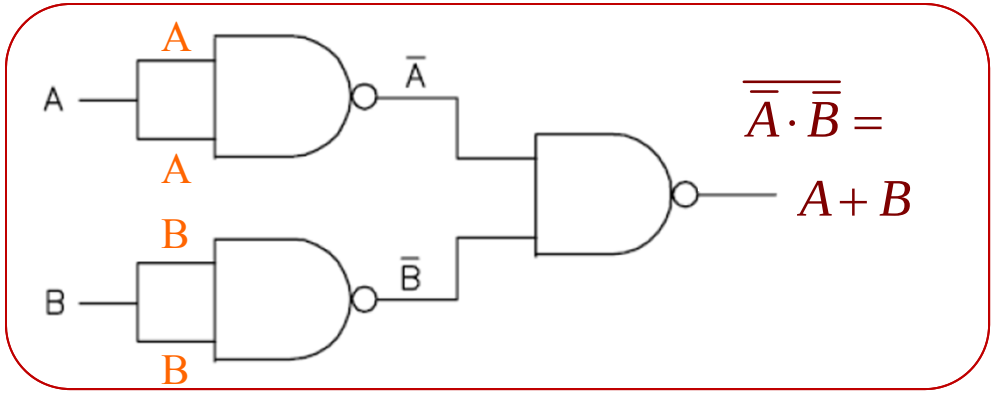
Universality of NAND Gates

NAND gates can be used to implement the functions of a NOT gate, an AND gate, and an OR gate, as illustrated from the figures on the right.

Thus, any given logic function can be implemented by using NAND gates alone. For this reason, NAND gate is said to be logically complete.







Example 2

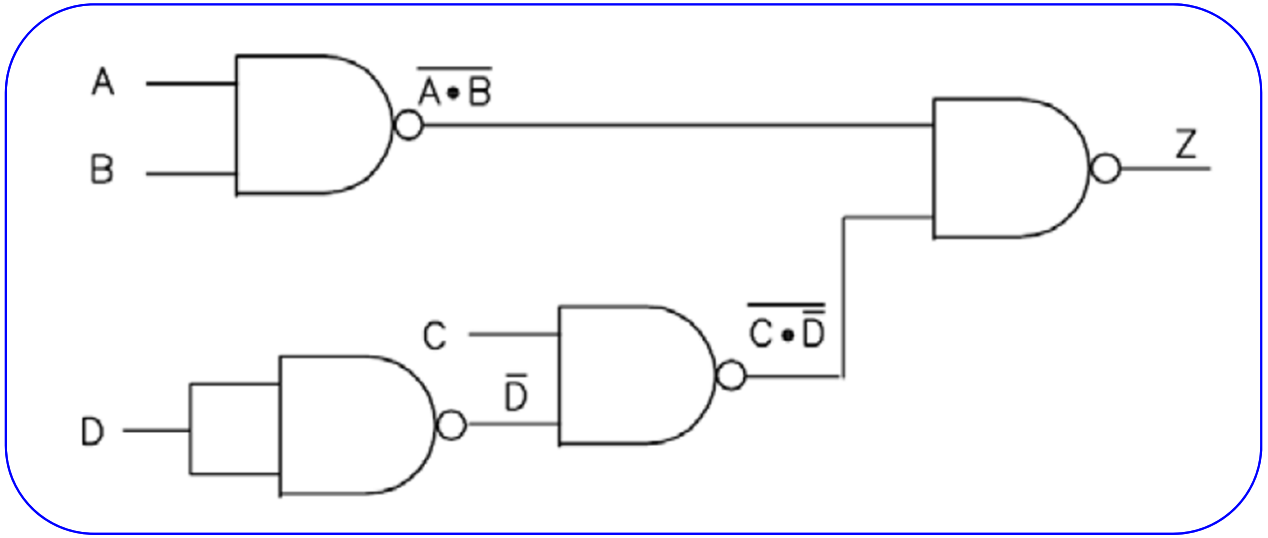
Implement $Z = A \cdot B + C \cdot \overline{D}$ using only NAND gates.

$$\overline{A+B} = \overline{A} \cdot \overline{B}$$
$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

Solution:

$$Z = A \cdot B + C \cdot \overline{D} = \overline{\overline{(A \cdot B) + (C \cdot \overline{D})}} = \overline{\overline{(A \cdot B)} \cdot \overline{(C \cdot \overline{D})}}$$

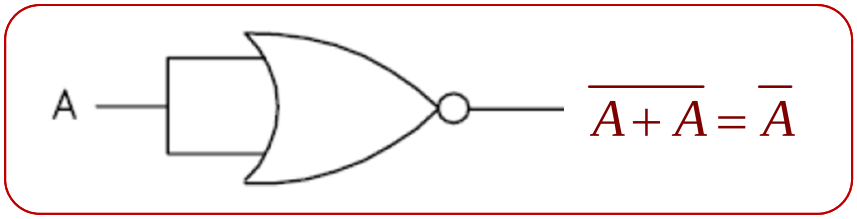
Circuit implementation using NAND gates:

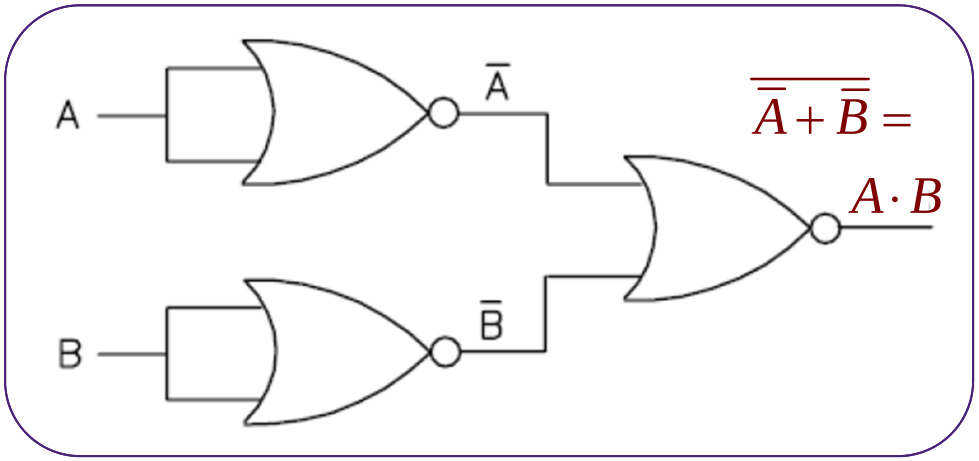


Universality of NOR Gates

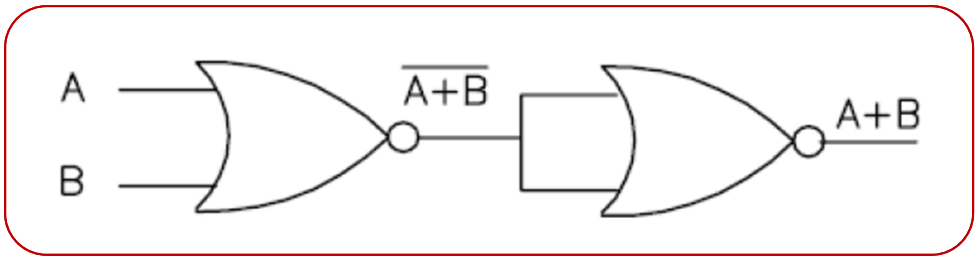
$$\overline{A+B} = \overline{A} \cdot \overline{B}$$
$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

Likewise, it can be shown that any logic function can be implemented using NOR gates alone. This is so because NOR gates can be used to implement the functions of a NOT gate, an AND gate and an OR gate, as shown in the figures on the right.





A NOR gate is functionally complete because AND, OR, and NOT gates can be implemented using NOR gates alone.



Example 3

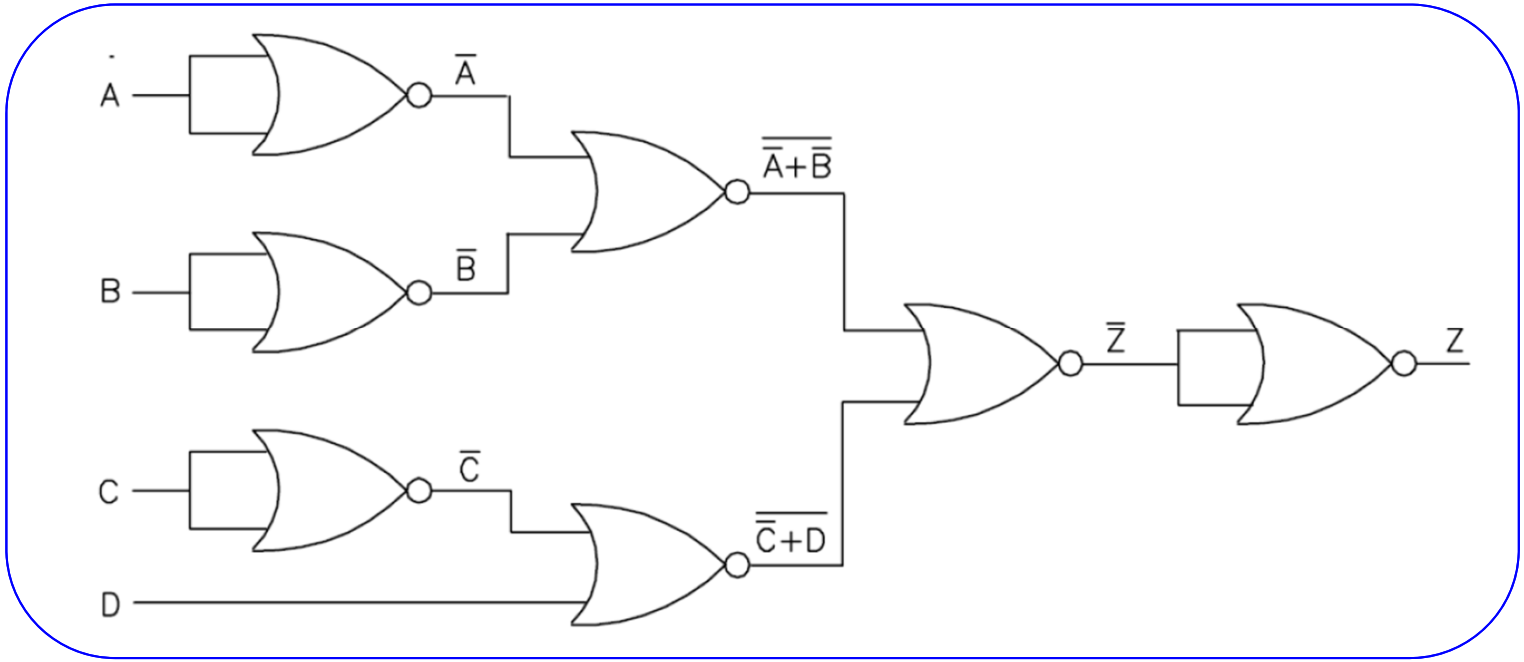
Implement $Z = A \cdot B + C \cdot \overline{D}$ using only NOR gates.

$$\overline{A+B} = \overline{A} \cdot \overline{B}$$
$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

Solution:

$$Z = A \cdot B + C \cdot \overline{D} = \overline{\overline{A \cdot B}} + \overline{\overline{C \cdot \overline{D}}} = \overline{\overline{A} + \overline{B}} + \overline{\overline{C} + D}$$

Circuit implementation using NOR gates:

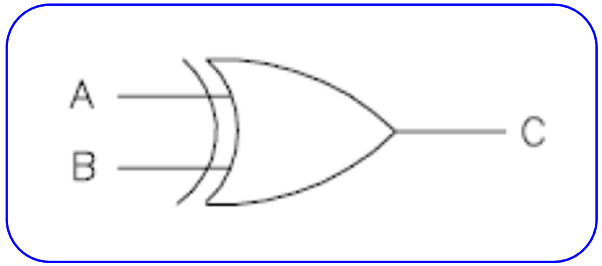


Exclusive OR and Exclusive NOR Gates

Exclusive OR operation is defined as

$$C = A \oplus B = A \cdot \overline{B} + \overline{A} \cdot B$$

(This is to be proven in Example 5)

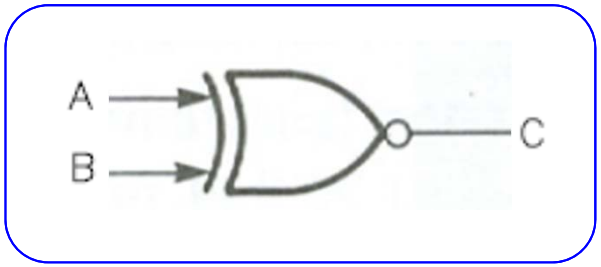


A	B	$C = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Truth Table for exclusive OR Gate

Exclusive NOR operation is defined as

$$C = \overline{A \oplus B} = \overline{A \cdot \overline{B} + \overline{A} \cdot B}$$
$$= \overline{(A \cdot \overline{B})} \cdot \overline{(\overline{A} \cdot B)}$$
$$= (\overline{A} + B) \cdot (A + \overline{B})$$

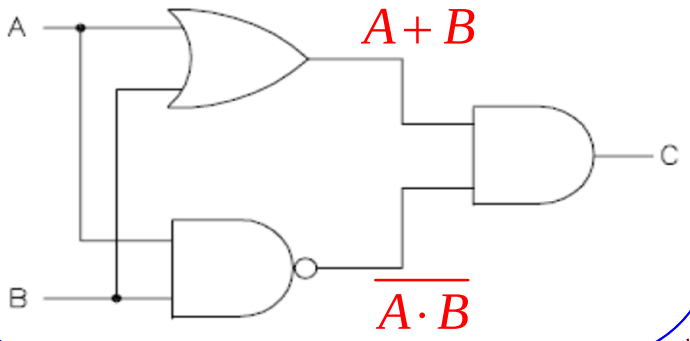


A	B	$C = \overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

Truth Table for exclusive NOR Gate

Example 4

Figure below shows a logical circuit that may be used to achieve exclusive OR operation. Determine the output C.



Solution: The output of each gate is labeled directly in the figure. It is easy to see that

$$C = (A+B) \cdot \overline{(A \cdot B)}$$

$$\Updownarrow$$

$$A \oplus B$$

? checked ✓

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

Truth Table for OR Gate

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

Truth Table for NAND Gate

A	B	$C = A \oplus B$	
0	0	0	✓
0	1	1	✓
1	0	1	✓
1	1	0	✓

Truth Table for exclusive OR Gate

Example 5

Simplify the expression for C in Example 4 and draw an equivalent logical circuit for exclusive OR operation.

Solution:

$$C = A \oplus B$$

$$= (A+B)\cdot(\overline{A\cdot B}) = (A+B)\cdot(\overline{A}+\overline{B})$$

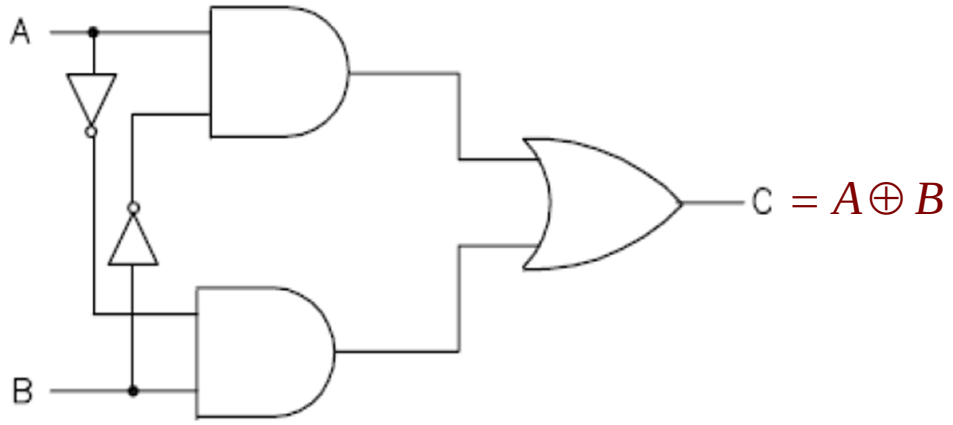
$$= A\cdot\overline{A} + A\cdot\overline{B} + B\cdot\overline{A} + B\cdot\overline{B}$$

$$= 0 + A\cdot\overline{B} + B\cdot\overline{A} + 0$$

$$= A\cdot\overline{B} + B\cdot\overline{A}$$

$$\overline{A+B} = \overline{A}\cdot\overline{B}$$

$$\overline{A\cdot B} = \overline{A}+\overline{B}$$



Logic Circuit Design – An Example

A majority detector has four input variables **A**, **B**, **C** and **D** and three output light indicators. **R**ed light will be on if majority of the input variables are equal to 0. Design an appropriate logic circuit for this application.

Design Procedure…

- Product or Design Specifications
- Identify necessary input and output variables
- Obtain a relationship between input and output variables — true table
- Determine a logical expression characterizing the input-output relationship — ?
- Implement the logical expression using logic gates

Logic Circuit Design – An Example

Construct a corresponding truth table for the red light output, for example…

	Inputs				Output
	A	B	C	D	R
# 0s > # 1s	0	0	0	0	1
	0	0	0	1	1
	0	0	1	0	1
	0	0	1	1	0
# 0s > # 1s	0	1	0	0	1
	0	1	0	1	0
	0	1	1	0	0
	0	1	1	1	0
# 0s > # 1s	1	0	0	0	1
	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	0

How to obtain a logical expression for R?

Logic Circuit Design

In designing digital circuits, the designer often begins with a truth table describing what the circuit should do. The design task is to determine what type of circuit will perform the function described in the truth table. Although, it may not always be obvious what kind of logic circuit would satisfy the truth table, two simple methods for designing such a circuit are found in standard form of Boolean expression called the **Sum-Of-Products** (or **SOP**) form and **Product-Of-Sums** (or **POS**) forms.

Based on the description of the problem, natural language is first translated into a truth table and Boolean expressions are found methodically using one of these two methods. The Boolean expression is then simplified using rules of Boolean algebra or **Karnaugh Maps** (which we will study later), so that it can be implemented using minimum number of logic gates for practical implementation.

Sum-of-Products Implementation

As you might suspect, a Sum-Of-Products Boolean expression is literally a set of Boolean terms added (summed) together, each term being a multiplicative (product) combination of Boolean variables:

$$\{\textit{sum-of-products-expression}\} = \{\textit{product term}\} + \cdots + \{\textit{product term}\}$$

Product terms that include all of the input variables (or their inverses) are called **minterms**. In a sum-of-products expression, we form a product of all the input variables (or their inverses) for each row of the truth table for which the result is logic 1. The output is the logical "sum" of these minterms Sum-Of-Products expressions are easy to generate from truth tables as shown in Example 6, by determining which rows of the table have an output of 1, writing one product term for each row, and finally summing all the product terms. This creates a Boolean expression representing the truth table as a whole.

Sum-Of-Products expressions lend themselves well to implementation as a set of AND gates (products) feeding into a single OR gate (sum).

Example 6.a

Obtain *C* for the exclusive OR operation from the truth table below in the Sum of Products (SOP) form.

Solution: ……

When the output *C* is true?

A	B	$C = A \oplus B$
0	0	0
0	1	X 1
1	0	Y 1
1	1	0

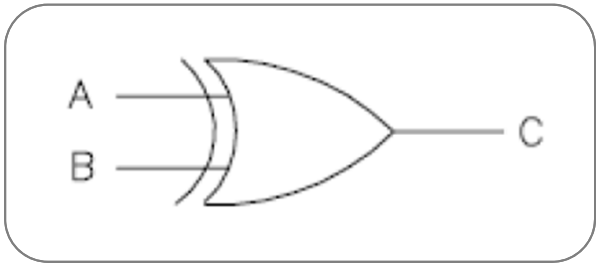
This term has a value of 1 if and only if $\overline{A} = 1, B = 1$

$\overline{A} \cdot B \rightarrow X$

$A \cdot \overline{B} \rightarrow Y$

The output variable *C* is true if and only if *X* is true or *Y* is true

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



This term has a value of 1 if and only if $A = 1, \overline{B} = 1$

Example 6

Obtain W from the truth table in the Sum of Products (SOP) form. Draw the logical circuit to implement it.

A	B	C	W
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Solution:

$\overline{A} \cdot \overline{B} \cdot C$

$\overline{A} \cdot B \cdot \overline{C}$

$\overline{A} \cdot B \cdot C$

$A \cdot \overline{B} \cdot C$

$A \cdot B \cdot \overline{C}$

This term would have a value of 1 if and only if $\overline{A}=1, \overline{B}=1, C=1$

This term would have a value of 1 if and only if $\overline{A}=1, B=1, \overline{C}=1$

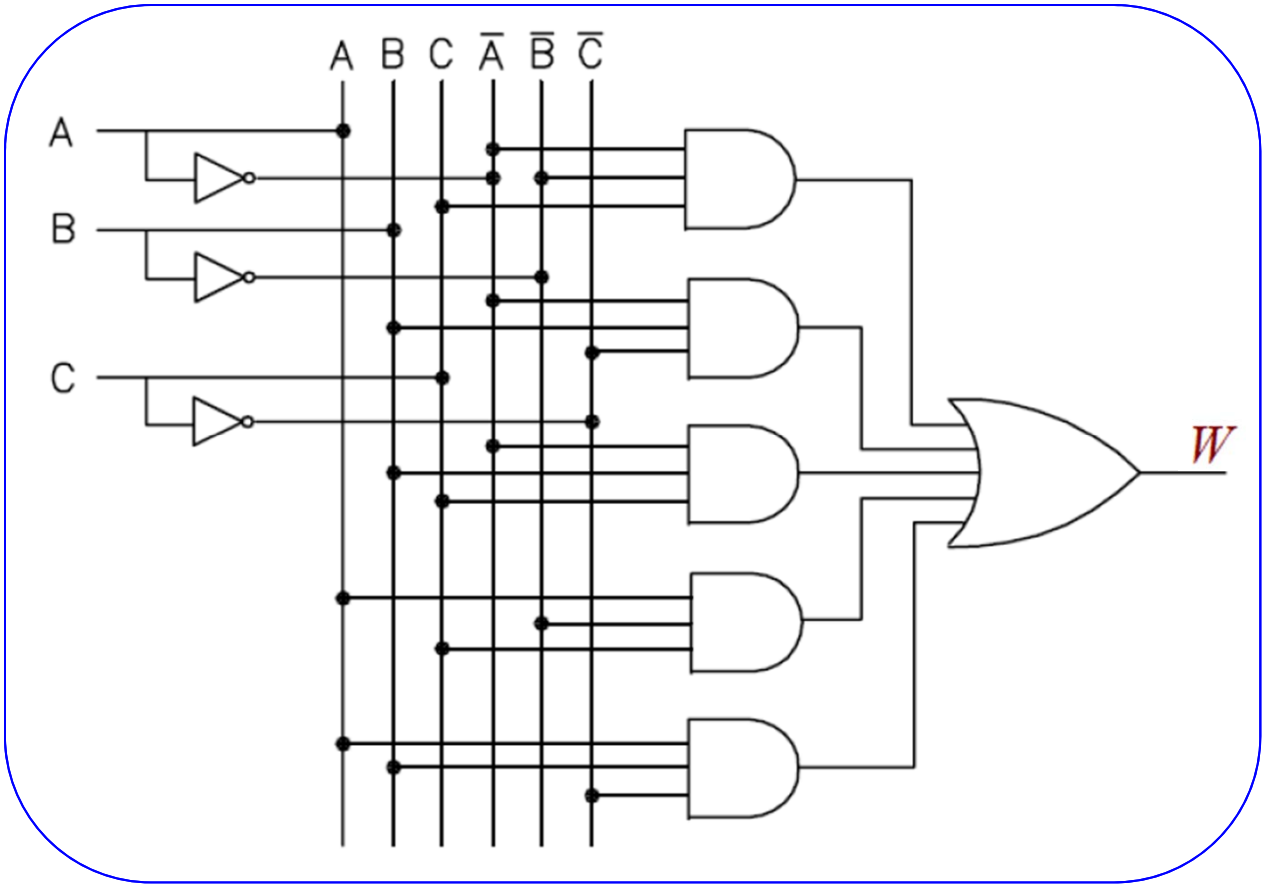
This term would have a value of 1 if and only if $\overline{A}=1, B=1, C=1$

This term would have a value of 1 if and only if $A=1, \overline{B}=1, C=1$

This term would have a value of 1 if and only if $A=1, B=1, \overline{C}=1$

$$W = \overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C + A \cdot \overline{B} \cdot C + A \cdot B \cdot \overline{C}$$

Example 6 (cont.)



$$W = \overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C + A \cdot \overline{B} \cdot C + A \cdot B \cdot \overline{C}$$

Example 7

Simplify the logical expression W obtained in Example 6 and show its implementation.

Solution: The Boolean expression can be simplified as follows

$$W = \overline{A}\cdot\overline{B}\cdot C + \overline{A}\cdot B\cdot\overline{C} + \overline{A}\cdot B\cdot C + A\cdot\overline{B}\cdot C + A\cdot B\cdot\overline{C}$$

$$= \left(\overline{A}\cdot\overline{B}\cdot C + A\cdot\overline{B}\cdot C\right) + \left(\overline{A}\cdot B\cdot\overline{C} + \overline{A}\cdot B\cdot C\right) + A\cdot B\cdot\overline{C}$$

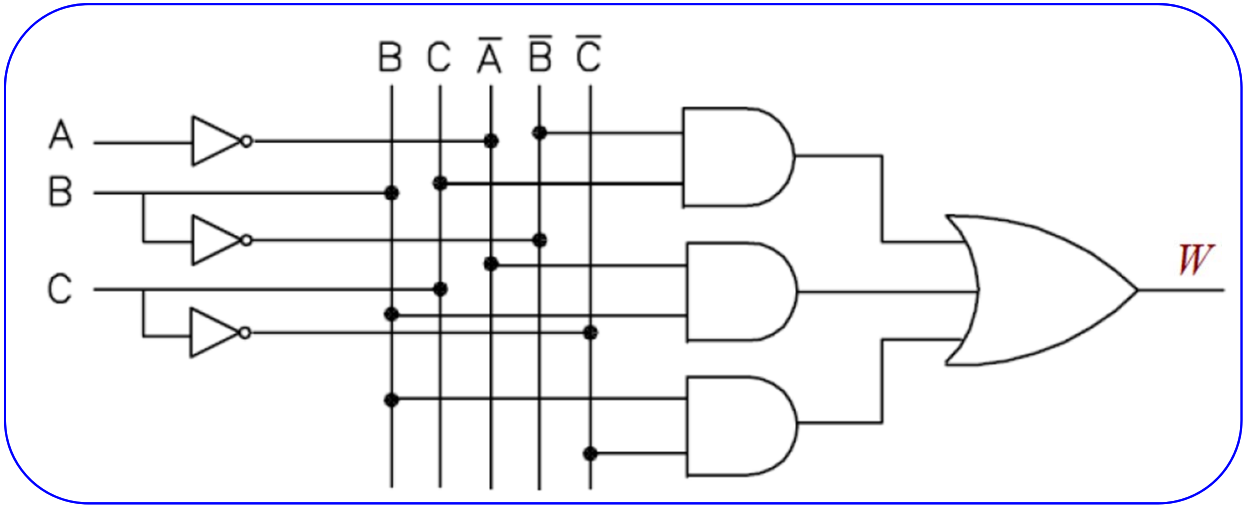
$$= \left(\overline{A} + A\right)\cdot\overline{B}\cdot C + \overline{A}\cdot B\cdot\left(\overline{C} + C\right) + A\cdot B\cdot\overline{C}$$

$$= \overline{B}\cdot C + \overline{A}\cdot B + A\cdot B\cdot\overline{C} = \overline{B}\cdot C + B\cdot\left(\overline{A} + A\cdot\overline{C}\right) = \overline{B}\cdot C + B\cdot\left(\overline{A} + \overline{C}\right)$$

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

$A + \overline{A}\cdot B = A + B$ **Rule 18**

Implementation of W after simplification:



Product-of-Sums Implementation

An alternative to generating a Sum-Of-Products expression to account for all the "high" (1) output conditions in the truth table is to generate a Product-Of-Sums, or POS, expression, to account for all the "low" (0) output conditions instead. POS Boolean expressions can be generated from truth tables quite easily, by determining which rows of the table have an output of **0**, writing one sum term for each row, and finally multiplying all the sum terms. This creates a Boolean expression representing the truth table as a whole.

$$\{\textit{product-of-sums-expression}\} = \{\textit{sum term}\} \bullet \cdots \bullet \{\textit{sum term}\}$$

These "sum" terms that include all of the input variables (or their inverses) are called **maxterms**. For POS implementation, the output variable is the logical product of maxterms. Product-Of-Sums expressions lend themselves well to implementation as a set of OR gates (sums) feeding into a single AND gate (product).

Example 8.a

Find Z in terms of A, B and C in product-of-sum (POS) form from the following truth table.

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

A	B	C	Z	$\overline{Z}$	
0	0	0	1	0	
0	0	1	1	0	
0	1	0	1	0	
0	1	1	1	0	
1	0	0	0	1	$\rightarrow A\cdot\overline{B}\cdot\overline{C}$
1	0	1	0	1	$\rightarrow A\cdot\overline{B}\cdot C$
1	1	0	0	1	$\rightarrow A\cdot B\cdot\overline{C}$
1	1	1	1	0	

Solution: We find SOP for $\overline{Z}$ ...

$$\overline{Z} = A\cdot\overline{B}\cdot\overline{C} + A\cdot\overline{B}\cdot C + A\cdot B\cdot\overline{C}$$

$$Z = \overline{\overline{Z}} = \overline{A\cdot\overline{B}\cdot\overline{C} + A\cdot\overline{B}\cdot C + A\cdot B\cdot\overline{C}}$$
$$= \left(\overline{A\cdot\overline{B}\cdot\overline{C}}\right)\cdot\left(\overline{A\cdot\overline{B}\cdot C}\right)\cdot\left(\overline{A\cdot B\cdot\overline{C}}\right)$$
$$= \left(\overline{A}+B+C\right)\cdot\left(\overline{A}+B+\overline{C}\right)\cdot\left(\overline{A}+\overline{B}+C\right)$$

POS form

Example 8

Find Z in terms of A, B and C in product-of-sum (POS) form from the following truth table.

Solution: Alternatively, we look for **0** in the truth table…

A	B	C	Z
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

$(\overline{A}+B+C)$

$(\overline{A}+B+\overline{C})$

$(\overline{A}+\overline{B}+C)$

This term would have a value of 0 if and only if $\overline{A}=0, B=0, C=0$

This term would have a value of 0 if and only if $\overline{A}=0, B=0, \overline{C}=0$

This term would have a value of 0 if and only if $\overline{A}=0, \overline{B}=0, C=0$

$$Z=\left(\overline{A}+B+C\right)\cdot\left(\overline{A}+B+\overline{C}\right)\cdot\left(\overline{A}+\overline{B}+C\right)$$ ⇦ Compare this with that in Example 8.a

Karnaugh Maps

From the previous examples we can see that rules of Boolean algebra can be applied in order to simplify expressions. Apart from being laborious (and requiring us to remember all the laws) this method can lead to solutions which, though they appear minimal, are not. The **Karnaugh map** (or **K map**) provides a simple and straightforward method of minimizing Boolean expressions. With the K map Boolean expressions having up to four and even six variables can be simplified easily.

Maurice Karnaugh
American
1924–

The simplified logical expression is then used so that minimum hardware is utilized in the implementation of logical circuits.

A K map provides a pictorial method of grouping together expressions with common factors and therefore eliminating unwanted variables. The values inside the squares are copied from the output column of the truth table, therefore there is one square in the map for every row in the truth table. Around the edge of the K map are the values of the two input variable.

Simplification Process

1. Draw out the pattern of output 1's and 0's in a matrix of input values
2. Construct the K map and place 1s and 0s in the squares according to the truth table.
3. Group the isolated 1s which are not adjacent to any other 1s (single loops).
4. Group any pair which contains a 1 adjacent to only one other 1 (double loops).
5. Group any quad that contains one or more 1s that have not already been grouped, making sure to use the minimum number of groups.
6. Group any pairs necessary to include any 1s that have not yet been grouped, making sure to use the minimum number of groups.
7. Form the OR sum of all the terms generated by each group.

We illustrate the concept of K Maps in examples…

K Map for 2 Variables

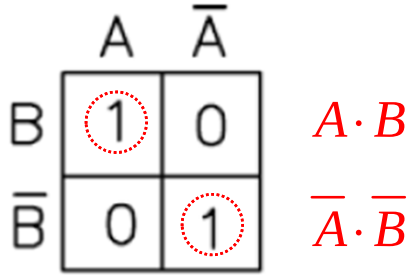
Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

- n = 0, $2^0 = 1$
- n = 1, $2^1 = 2$
- n = 2, $2^2 = 4$
- n = 3, $2^3 = 8$

Consider the following truth table.

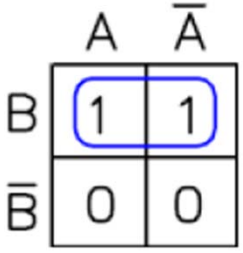
A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

K Map



The logical expression X is given by $X = A \cdot B + \overline{A} \cdot \overline{B}$

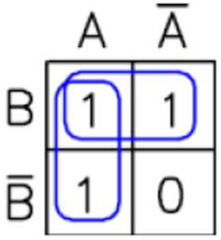
Consider $Y = A \cdot B + \overline{A} \cdot B$



$Y = B$

K Map

Consider $Y = A \cdot B + \overline{A} \cdot B + A \cdot \overline{B}$



$Y = A + B$

K Map

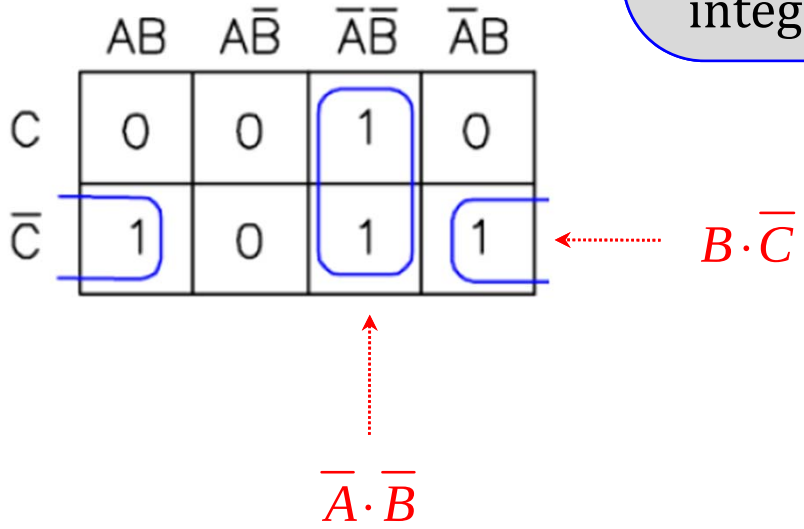
K Map for 3 Variables

Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

Consider the following truth table.

A	B	C	X
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

K Map



The corresponding Boolean expression using SOP is

$$X = \overline{A} \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + A \cdot B \cdot \overline{C}$$

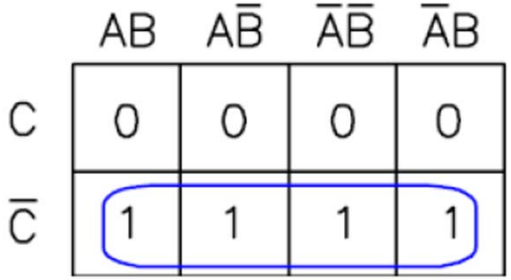
$$X = \overline{A} \cdot \overline{B} + B \cdot \overline{C}$$

More Examples for K Maps with 3 Variables…

Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

Consider

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

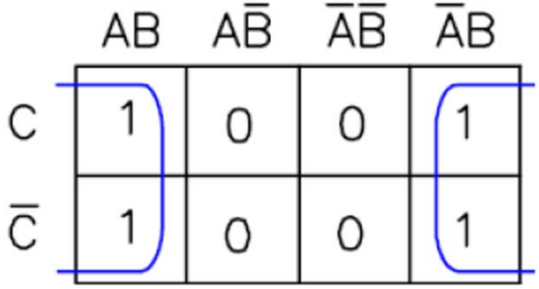


K Map

$$Y = \overline{C}$$

Consider

$$W = A \cdot B \cdot C + \overline{A} \cdot B \cdot C + A \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot \overline{C}$$



K Map

$$W = B$$

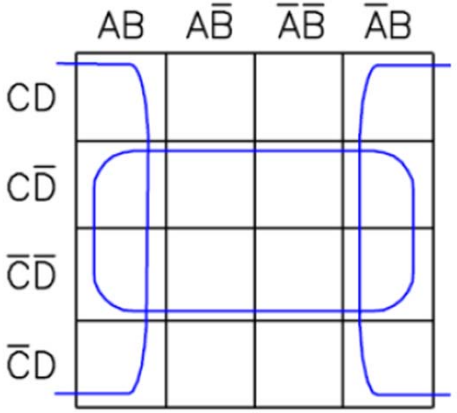
It is obvious that K maps is an excellent tool for simplifying Boolean expressions…

K Map for 4 Variables

Knowing how to generate Gray code should allow us to build larger maps. Actually, all we need to do is look at the left to right sequence across the top of the 3-variable map, follow a similar sequence for the other two variables and write it down on the left side of the 4-variable map.

K map of four variables A, B, C and D is shown in the following figure. As we have shown in the previous examples, we may easily prove that:

Combining eight adjacent squares in K map eliminates three variables from the resulting Boolean expression of the corresponding squares.



Example 9

Consider $X = \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot D + \overline{A} \cdot \overline{B} \cdot C \cdot D + \overline{A} \cdot B \cdot \overline{C} \cdot D + \overline{A} \cdot B \cdot C \cdot D$

	AB	$A\overline{B}$	$\overline{A}\overline{B}$	$\overline{A}B$
CD	0	0	1	1
$C\overline{D}$	0	0	0	0
$\overline{C}\overline{D}$	0	0	0	0
$\overline{C}D$	0	0	1	1

K Map

$$X = \overline{A} \cdot D$$

B is ***don't care*** and

C is also ***don't care***...

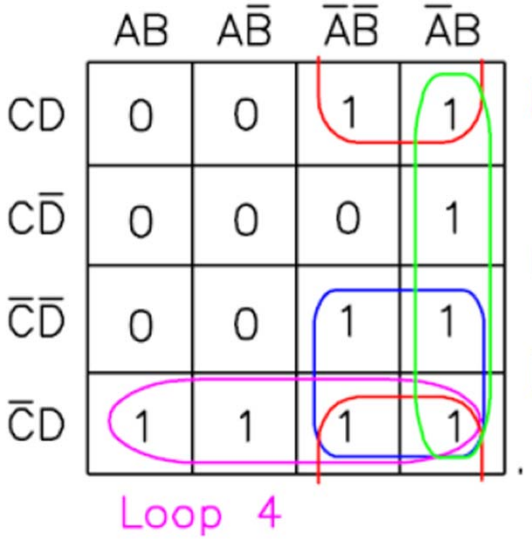
Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

Example 10

Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

Consider $Y = A\cdot B\cdot \overline{C}\cdot D + A\cdot \overline{B}\cdot \overline{C}\cdot D + \overline{A}\cdot \overline{B}\cdot \overline{C}\cdot D$
$+\overline{A}\cdot B\cdot \overline{C}\cdot D + \overline{A}\cdot \overline{B}\cdot \overline{C}\cdot \overline{D} + \overline{A}\cdot B\cdot \overline{C}\cdot \overline{D}$
$+\overline{A}\cdot B\cdot C\cdot \overline{D} + \overline{A}\cdot B\cdot C\cdot D + \overline{A}\cdot \overline{B}\cdot C\cdot D$

K Map



Loop 3 → *B* is ***don't care***, *C* is ***don't care***, $\overline{A}\cdot D$

Loop 1 → *C* is ***don't care***, *D* is ***don't care***, $\overline{A}\cdot B$

Loop 2 → *B* is ***don't care***, *D* is ***don't care***, $\overline{A}\cdot \overline{C}$

A is ***don't care***, *B* is ***don't care***, $\overline{C}\cdot D$

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

Example 11: K Map Simplification for POS

Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

From the table shown below, find Z in terms of A, B and C using Product-of-sums (POS) form. Use K map to simplify the resulting expression.

A	B	C	Z	$\overline{Z}$
0	0	0	1	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	0	1
1	1	1	0	1

Solution: Follow the usual K map simplification for $\overline{Z}$:

	AB	$A\overline{B}$	$\overline{A}\,\overline{B}$	$\overline{A}B$
C	1	0	1	0
$\overline{C}$	1	1	0	1

$$\overline{Z} = B\cdot\overline{C} + A\cdot B + A\cdot\overline{C} + \overline{A}\cdot\overline{B}\cdot C$$

Then, the POS simplification can be obtained by the following manipulations:

$$Z = \overline{\overline{Z}} = \overline{B\cdot\overline{C} + A\cdot B + A\cdot\overline{C} + \overline{A}\cdot\overline{B}\cdot C}$$

$$= \left(\overline{B\cdot\overline{C}}\right)\cdot\left(\overline{A\cdot B}\right)\cdot\left(\overline{A\cdot\overline{C}}\right)\cdot\left(\overline{\overline{A}\cdot\overline{B}\cdot C}\right) \qquad \textit{De Morgan's Law}$$

$$= \left(\overline{B}+C\right)\cdot\left(\overline{A}+\overline{B}\right)\cdot\left(\overline{A}+C\right)\cdot\left(A+B+\overline{C}\right) \qquad \textit{De Morgan's Law}$$

Summary: K Map Simplification Process

1. Draw out the pattern of output 1's and 0's in a matrix of input values
2. Construct the K map and place 1s and 0s in the squares according to the truth table.
3. Group the isolated 1s which are not adjacent to any other 1s (single loops).
4. Group any pair which contains a 1 adjacent to only one other 1 (double loops).
5. Group any quad that contains one or more 1s that have not already been grouped, making sure to use the minimum number of groups.
6. Group any pairs necessary to include any 1s that have not yet been grouped, making sure to use the minimum number of groups.
7. Form the OR sum of all the terms generated by each group.

Design Example

A majority detector has four input variables **A**, **B**, **C** and **D** and three output light indicators. **G**reen light will be on if majority of the input variables are equal to 1. **R**ed light will be on if majority of the input variables are equal to 0. **Y**ellow light will be on if there is a tie. Design an appropriate logic circuit for this application.

Design Procedure…

- Product or Design Specifications
- Identify necessary input and output variables
- Obtain a relationship between input and output variables
- Determine a logical expression characterizing the input-output relationship
- Implement the logical expression using logic gates

Truth Table

Solution: Construct the corresponding truth table…

Inputs				Outputs		
A	B	C	D	G	R	Y
0	0	0	0	0	1	0
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	1	0
0	1	0	1	0	0	1
0	1	1	0	0	0	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	0	1
1	0	1	0	0	0	1
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	1	0	0

K Maps and Logical Expressions

Group 2^n number of 1s which are adjacent to each other, with n being the largest possible integer.

K Map for G

	AB	$A\overline{B}$	$\overline{A}\,\overline{B}$	$\overline{A}B$
	1	1	0	1
$C\overline{D}$	1	0	0	0
$\overline{C}\,\overline{D}$	0	0	0	0
$\overline{C}D$	1	0	0	0

$$G = A \cdot B \cdot C + A \cdot C \cdot D + B \cdot C \cdot D + A \cdot B \cdot D$$

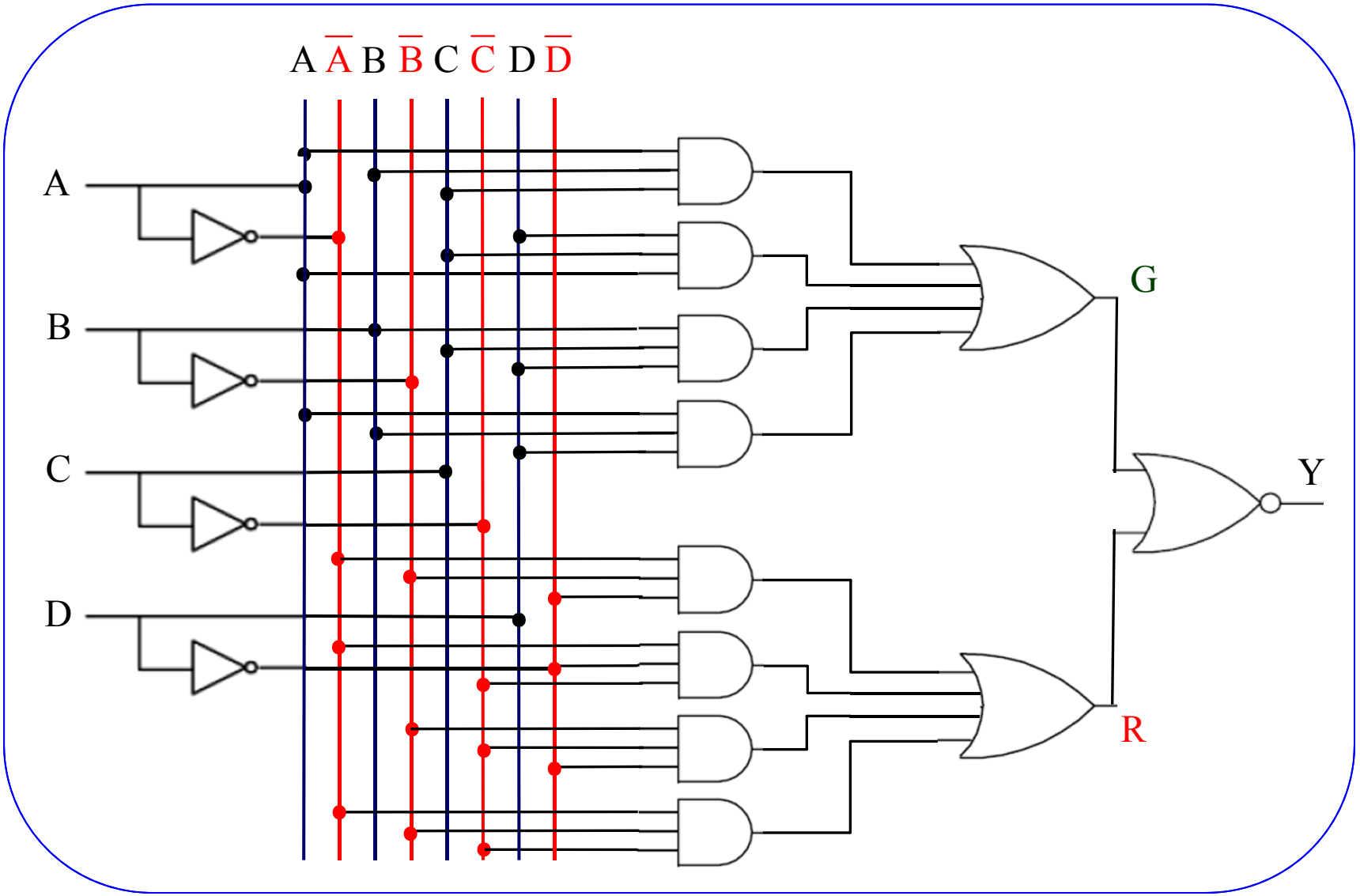
K Map for R

	AB	$A\overline{B}$	$\overline{A}\overline{B}$	$\overline{A}B$
CD	0	0	0	0
$C\overline{D}$	0	0	1	0
$\overline{C}\,\overline{D}$	0	1	1	1
$\overline{C}D$	0	0	1	0

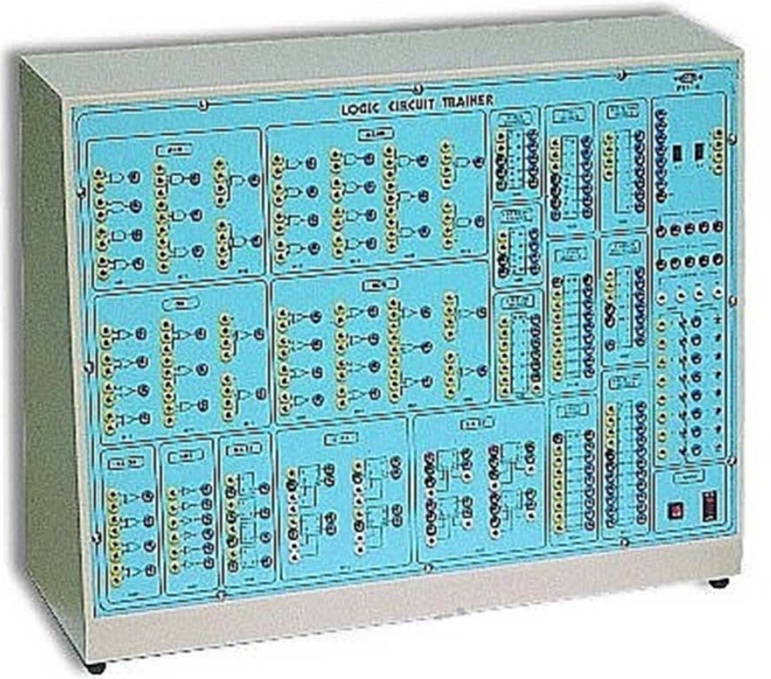
$$R = \overline{A} \cdot \overline{B} \cdot \overline{D} + \overline{A} \cdot \overline{C} \cdot \overline{D} + \overline{B} \cdot \overline{C} \cdot \overline{D} + \overline{A} \cdot \overline{B} \cdot \overline{C}$$

Exercise: Construct K map for Y and show that $Y = \overline{G} \cdot \overline{R} = \overline{G + R}$

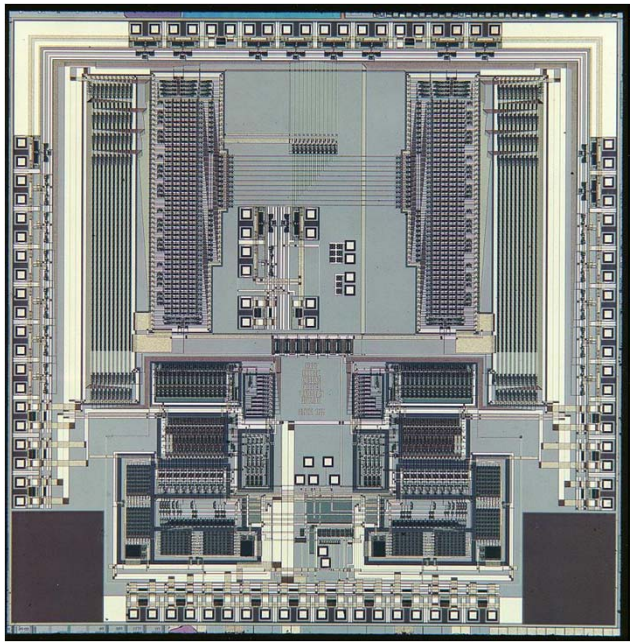
Logic Circuit Implementation



Actual Logic Circuit Implementation

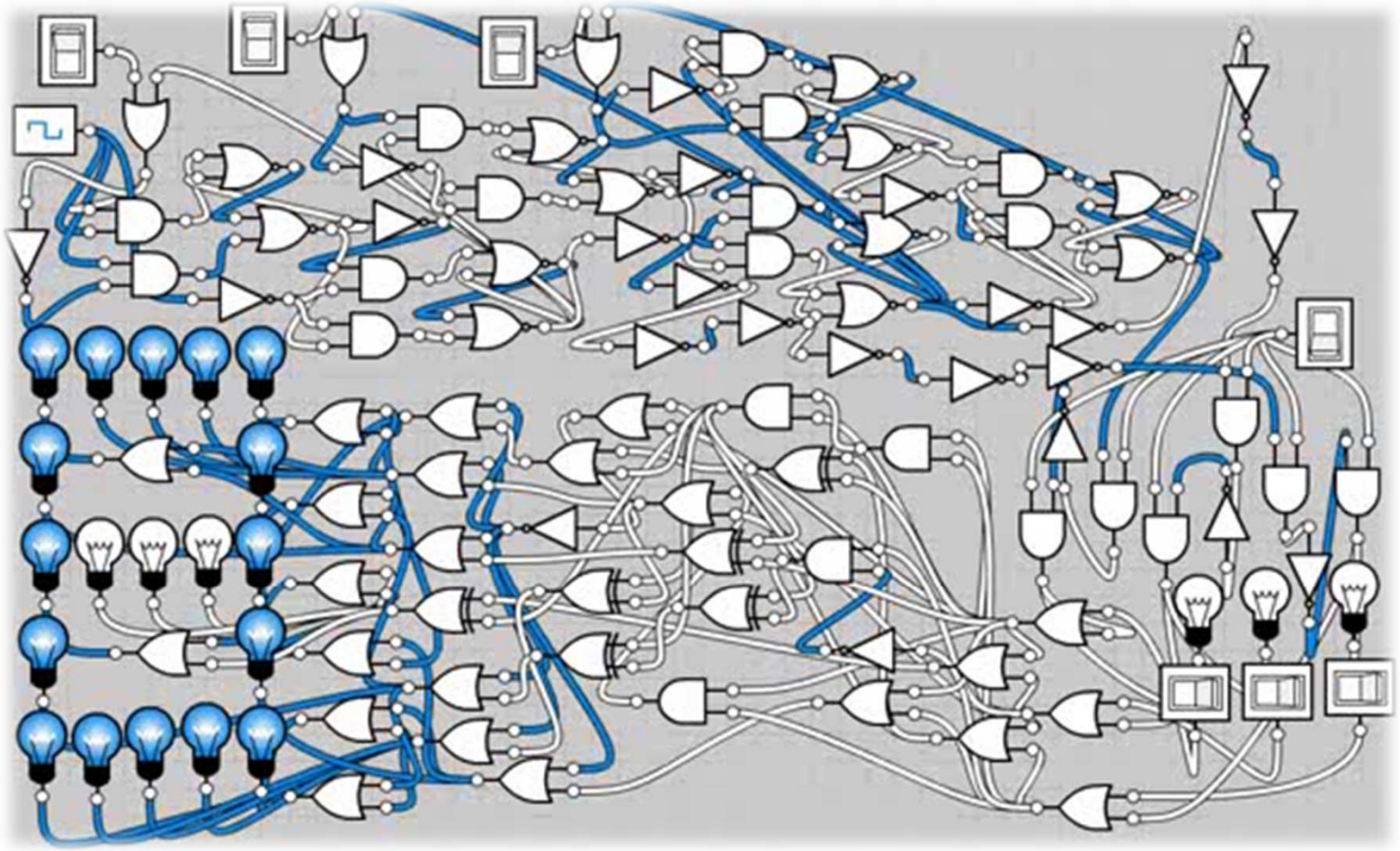


Logic Circuit Trainer



Integrated Circuit Chip

Video: 3-bit Binary Counter



Final Exam & Final Grade

Final Examination:

- There will be a final exam for this module.
- The final exam will have 4 questions (2 from Part 1 and 2 from Part 2).
- The final exam will have a duration of 2 hours.
- The final exam is of closed book.
- Students are allowed to bring an A4 sheet (double sided) of formulae with them into the examination venue.

Final Grade:

- Your Final Grade = 80% of Final Exam Marks + 20% Lab Marks

Acknowledgement…

… Special thanks to Prof Dipti Srinivasan of NUS ECE
for sharing parts of materials
covered in this second part of EG1108 …

… To UAV Research Group of NUS ECE
for materials on unmanned aerial systems …

… To YouTube for video clips on
3-bit Binary Counter, DC Power Supply, and
Motors and Generators …

That's all, folks!

Thank You!

Bugs Bunny
Hollywood Movie Star
1940–?