LECTURE NOTES

ON

INDUSTRIAL AUTOMATION AND CONTROL
(AEE511)

Prepared By:

Dr. V.Chandra Jagan Mohan
Dr.M.Pala Prasad Reddy

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal – 500043,Hyd

UNIT – I
INTRODUCTION TO INDUSTRIAL AUTOMATION AND CONTROL

Introduction to Industrial Automation and Control:

Industry:

In a general sense the term "Industry" is defined as follows.

Definition: *Systematic Economic Activity that could be related to Manufacture/Service/ Trade.*

Automation:

The word 'Automation' is derived from Greek words "Auto"(self) and "Matos"(moving). Automation therefore is the mechanism for systems that "move by itself". However, apart from this original sense of the word, automated systems also achieve significantly superior performance than what is possible with manual systems, in terms of power, precision and speed of operation.

Definition: *Automation is a set of technologies that results in operation of machines and systems without significant human intervention and achieves performance superior to manual operation.*

Control:

It is perhaps correct to expect that the learner for this course has already been exposed toa course on Control Systems, which is typically introduced in the final or pre-final year of an undergraduate course in Engineering in India. The word control is therefore expected to be familiar and defined as under.

Definition: *Control is a set of technologies that achieves desired patterns of variations of operational parameters and sequences for machines and systems by providing the input signals necessary.*

Industrial Automation also involves significant amount of hardware technologies, related to Instrumentation and Sensing, Actuation and Drives, Electronics for Signal Conditioning, Communication and Display, Embedded as well as Stand-alone Computing Systems etc.

As Industrial Automation systems grow more sophisticated in terms of the knowledge and algorithms they use, as they encompass larger areas of operation comprising several units or the whole of a factory, or even several of them, and as they integrate manufacturing with other areas of business, such as, sales and customer care, finance and the entire supply chain of the business, the usage of IT increases dramatically. However, the lower level Automation Systems that only deal with individual or , at best, a group of machines, make less use of IT and more of hardware, electronics and embedded computing.

Industrial information systems are generally reactive in the sense that they receive stimuli from their universe of discourse and in turn produce responses that stimulate its environment. Naturally, a crucial component of an industrial information system is its interface to the world.

Most of industrial information systems have to be real-time. By that we mean that the computation not only has to be correct, but also must be produced in time. An accurate result, which is not timely may be less preferable than a less accurate result produced in time. Therefore systems have to be designed with explicit considerations of meeting computing time deadlines.

Many industrial information systems are considered mission-critical, in the sense that the malfunctioning can bring about catastrophic consequences in terms of loss of human life or property. Therefore extraordinary care must be exercised during their design to make them flawless. In spite of that, elaborate mechanisms are often deployed to ensure that any unforeseen circumstances can also be handled in a predictable manner. Fault-tolerance to emergencies due to hardware and software faults must often be built in.

Role of automation in industry:

Manufacturing processes, basically, produce finished product from raw/unfinished material using energy, manpower and equipment and infrastructure. Since an industry is essentially a “systematic economic activity”, the fundamental

Objective of any industry is to make profit.

Similarly, systems such as Automated Guided Vehicles, Industrial Robots, Automated Crane and Conveyor Systems reduce material handling time. Automation also reduces cost of production significantly by efficient usage of energy,

manpower and material. The product quality that can be achieved with automated precision machines and processes cannot be achieved with manual operations. Moreover, since operation is automated, the same quality would be achieved for thousands of parts with little variation. Industrial Products go through their life cycles, which consist of various stages. At first, a product is conceived based on Market feedbacks, as well as Research and

Development Activities. Once conceived the product is designed. Prototype Manufacturing is generally needed to

prove the design. Once the design is proved, Production Planning and Installation must be carried out to ensure that the necessary resources and strategies for mass manufacturing are in place. This is followed by the actual manufacture and quality control activities through which the product is mass-produced. This is followed by a number of commercial activities through which the product is actually sold in the market. Automation also reduces the overall product life cycle i.e., the time required to complete (i) Product conception and design (ii) Process planning and installation (iii) Various stages of the product life cycle.

Types of production systems :

Major industrial processes can be categorized as follows based on their scale and scope of production.

Continuous flow process: Manufactured product is in continuous quantities i.e., the product is not a discrete object. Moreover, for such processes, the volume of production is generally very high, while the product variation is relatively low. Typical examples of such processes include Oil Refineries, Iron and Steel Plants, Cement and Chemical Plants. Mass Manufacturing of Discrete Products: Products are discrete objects and manufactured in large volumes. Product variation is very limited. Typical examples are Appliances, Automobiles etc. Batch Production: In a batch production process the product is either discrete or continuous. However, the variation in product types is larger than in continuous-flow processes. The same set of equipment is used to manufacture all the product types. However for each batch of a given product type a distinct set of operating parameters must be established. This set is often referred to as the "recipe" for the batch. Typical examples here would be Pharmaceuticals, Casting Foundries, Plastic moulding, Printing etc.

Job shop Production: Typically designed for manufacturing small quantities of discrete products, which are custom built, generally according to drawings supplied by customers. Any variation in the product can be made. Examples include Machine Shops, Prototyping facilities etc.

Types of Automation Systems:

Automation systems can be categorized based on the flexibility and level of integration in manufacturing process operations. Various automation systems can be classified as follows

Fixed Automation: It is used in high volume production with dedicated equipment, which has a fixed set of operation and designed to be efficient for this set. Continuous flow and Discrete Mass Production systems use this automation. e.g. Distillation Process, Conveyors, Paint Shops, Transfer lines etc. A process using mechanized machinery to perform fixed and repetitive operations in order to produce a high volume of similar parts.

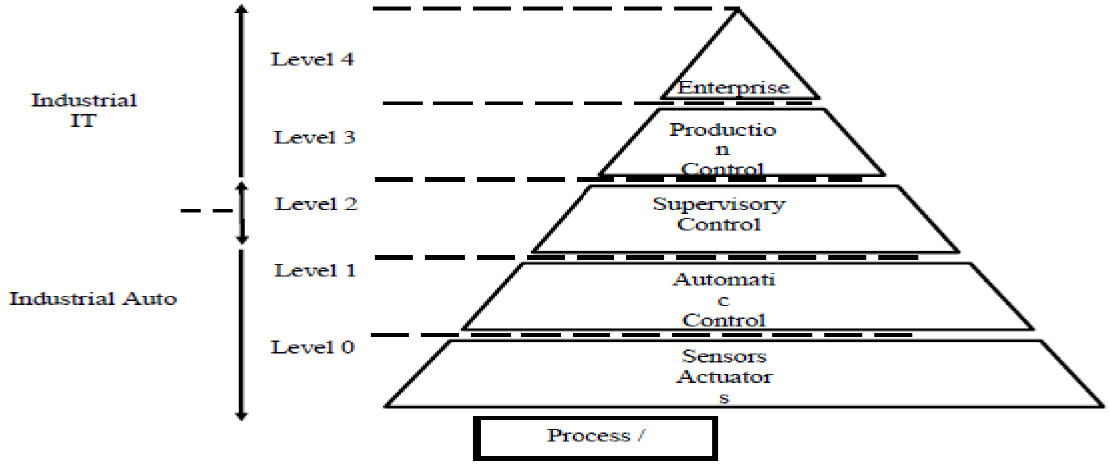
Programmable Automation: It is used for a changeable sequence of operation and configuration of the machines using electronic controls. However, non-trivial programming effort may be needed to reprogram the machine or sequence of operations. Investment on programmable equipment is less, as production process is not changed frequently. It is typically used in Batch process where job variety is low and product volume is medium to high, and sometimes in mass production also. e.g. in Steel Rolling Mills, Paper Mills etc.

Flexible Automation: It is used in Flexible Manufacturing Systems (FMS) which is invariably computer controlled. Human operators give high-level commands in the form of codes entered into computer identifying product and its location in the sequence and the lower level changes are done automatically. Each production machine receives settings/instructions from computer. These automatically loads/unloads required tools and carries out their processing instructions. After processing, products are automatically transferred to next machine. It is typically used in job shops and batch processes where product varieties are high and job volumes are medium to low. Such systems typically use Multipurpose CNC machines, Automated Guided Vehicles (AGV) etc.

Integrated Automation: It denotes complete automation of a manufacturing plant, with all processes functioning under computer control and under coordination through digital information processing. It includes technologies such as computer-aided design and manufacturing, computer-aided process planning, computer numerical control machine tools, flexible machining systems, automated storage and retrieval systems, automated material handling systems such as robots and automated cranes and conveyors, computerized scheduling and production control. It may also integrate a business system through a common database. In other words, it symbolizes full integration of process and management operations using information and communication technologies. Typical examples of such technologies are seen in Advanced Process Automation Systems and Computer Integrated Manufacturing (CIM)

The Architecture of Elements: The Automation Pyramid

Industrial automation systems are very complex having large number of devices with confluence of technologies working in synchronization. In order to know the performance of the system we need to understand the various parts of the system. Industrial automation systems are organized hierarchically as shown in the following figure.



Various components in an industrial automation system can be explained using the automation pyramid as shown above. Here, various layers represent the wideness (in the sense of no. of devices), and fastness of components on the time-scale.

Sensors and Actuators Layer: This layer is closest to the processes and machines, used to translate signals sothat signals can be derived from processes for analysis and decisions and hence control signals can be applied to the processes. This forms the base layer of the pyramid also called ‘level 0’ layer.

Automatic Control Layer: This layer consists of automatic control and monitoring systems, which drive the actuators using the process information given by sensors. This is called as ‘level 1’ layer.

Supervisory Control Layer: This layer drives the automatic control system by setting target/goal to the controller. Supervisory Control looks after the equipment, which may consist of several control loops. This is called as ‘level 2’ layer.

Production Control Layer: This solves the decision problems like production targets, resource allocation, task allocation to machines, maintenance management etc. This is called ‘level 3’ layer. ***Enterprise control layer:*** This deals less technical and more commercial activities like supply, demand, cash flow, product marketing etc. This is called as the ‘level 4’ layer.

The spatial scale increases as the level is increased e.g. at lowest level a sensor works in a single loop, but there exists many sensors in an automation system which will be visible as the level is increased. The lowest level is faster in the time scale and the higher levels are slower. The aggregation of information over some time interval is taken at higher levels.

All the above layers are connected by various types of communication systems. For example the sensors and actuators may be connected to the automatic controllers using a point-to-point digital communication, while the automatic controllers themselves may be connected with the supervisory and production control systems using computer networks. Some of these networks may be proprietary.

Measurement Systems Specifications:

1. Static Characteristics:

Static characteristics refer to the characteristics of the system when the input is either held constant or varying very slowly. The items that can be classified under the heading static characteristics are mainly:

Range (or span):

It defines the maximum and minimum values of the inputs or the outputs for which the instrument is recommended to use. For example, for a temperature measuring instrument the input range may be 100-500 ^{o}C and the output range may be 4-20 mA.

Sensitivity:

It can be defined as the ratio of the *incremental output* and the *incremental input*. While defining the sensitivity, we assume that the input-output characteristic of the instrument is approximately linear in that range. Thus if the sensitivity of a thermocouple is denoted as $10_0/VC\mu$, it indicates the sensitivity in the linear range of the thermocouple voltage vs. temperature characteristics. Similarly sensitivity of a spring balance can be expressed as 25 mm/kg (say), indicating additional load of 1 kg will cause additional displacement of the spring by 25mm.

Again sensitivity of an instrument may also vary with temperature or other external factors.This is known as *sensitivity drift*. Suppose the sensitivity of the spring balance mentioned above is 25 mm/kg at 20 ^{o}C and 27 mm/kg at 30^{o}C. Then the sensitivity drift/^{o}C is 0.2 (mm/kg)/^{o}C. In order to avoid such sensitivity drift, sophisticated instruments are either kept at controlled temperature, or suitable in-built temperature compensation schemes are provided inside the instrument.

Linearity:

Linearity is actually a measure of nonlinearity of the instrument. When we talk about sensitivity, we assume that the input/output characteristic of the instrument to be approximately linear. But in practice, it is normally nonlinear, as shown in Fig.1. The *linearity* is defined as the maximum deviation from the linear characteristics as a percentage of the full scale output. Thus,

$$Linearity = \frac{\Delta O}{O_{\max} - O_{\min}} \qquad (1)$$

where, $\Delta O = \max(\Delta O_1, \Delta O_2)$.

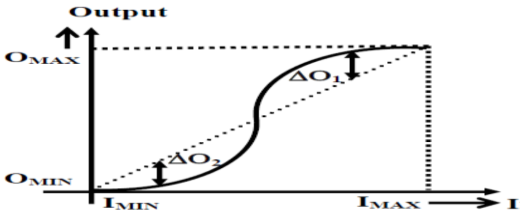


Fig. 1 Linearity

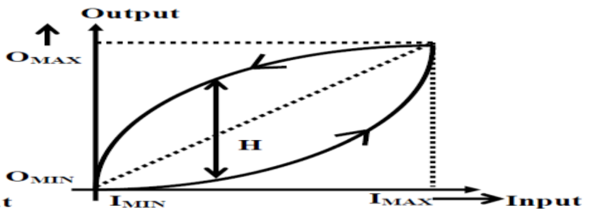


Fig. 2 Hysteresis

Hysteresis:

Hysteresis exists not only in magnetic circuits, but in instruments also. For example, the deflection of a diaphragm type pressure gage may be different for the same pressure, but one for increasing and other for decreasing, as shown in Fig.2. The *hysteresis* is expressed as the maximum hysteresis as a full scale reading, i.e., referring fig.2,

$$Hysteresis = \frac{H}{O_{\max} - O_{\min}} X100. \qquad (2)$$

Resolution:

In some instruments, the output increases in discrete steps, for continuous increase in the input, as shown in Fig.3. It may be because of the finite graduations in the meter scale; or the instrument has a digital display, as a result the output indication changes discretely. A $3\frac{1}{2}$-digit voltmeter, operating in 0-2V range, can have maximum reading of 1.999V, and it cannot measure any change in voltage below 0.001V. *Resolution* indicates the minimum change in input variable that is detectable. For example, an eight-bit A/D converter with $+5V$ input can measure the minimum voltage of $\frac{5}{2^8-1}$ or 19.6 *mv*. Referring to fig.3, *resolution* is also defined in terms of percentage as:

$$Resolution = \frac{\Delta I}{I_{max} - I_{min}} X100 \qquad (3)$$

The quotient between the measuring range and resolution is often expressed as *dynamic range* and is defined as:

$$\text{Dynamic range} = \frac{measurement\ range}{resolution} \qquad (4)$$

And is expressed in terms of dB. The dynamic range of an *n*-bit ADC, comes out to be approximately 6*n* dB.

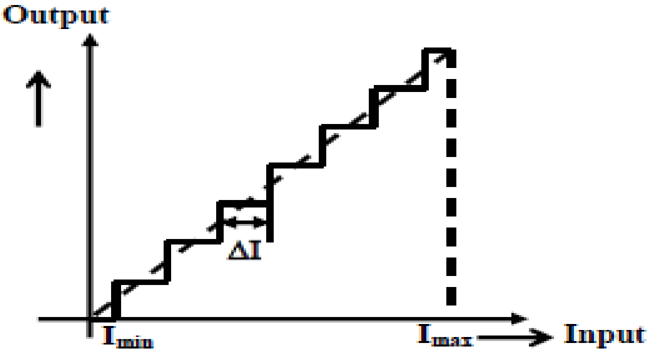


Fig. 3 Resolution

Accuracy:

Accuracy indicates the closeness of the measured value with the actual or true value, and is expressed in the form of the *maximum error (= measured value – true value)* as a percentage of full scale reading. Thus, if the accuracy of a temperature indicator, with a full scale range of 0-500 °C is specified as ±0.5%, it indicates that the measured value will always be within ±2.5 °C of the true value, if measured through a standard instrument during the process of calibration. But if it indicates a reading of 250 °C, the error will also be ±2.5 °C, i.e. ±1% of the reading. Thus it is always better to choose a scale of measurement where the input is near full-scale value. But the true value is always difficult to get. We use standard calibrated instruments in the laboratory for measuring true value if the variable.

Precision:

Precision indicates the repeatability or reproducibility of an instrument (but does not indicate accuracy). If an instrument is used to measure the same input, but at different instants, spread over the whole day, successive measurements may vary randomly. The random fluctuations of readings, (mostly with a Gaussian distribution) are often due to random variations of several other factors which have not been taken into account, while measuring the variable. A precision instrument indicates that the successive reading would be very close, or in other words, the standard deviation $_e\sigma$ of the set of measurements would be very small. Quantitatively, the precision can be expressed as:

$$Precision = \frac{measured\ range}{\sigma_e} \qquad (5)$$

The difference between precision and accuracy needs to be understood carefully. Precision means repetition of successive readings, but it does not guarantee accuracy; successive readings may be close to each other, but far from the true value. On the other hand, an accurate instrument has to be precise also, since successive readings must be close to the true value (that is unique).

2. Dynamic Characteristics

Dynamic characteristics refer to the performance of the instrument when the input variable is changing rapidly with time. For example, human eye cannot detect any event whose duration is more than one-tenth of a second; thus the dynamic performance of human eye cannot be said to be very satisfactory. The dynamic performance of an instrument is normally expressed by a differential equation relating the input and output quantities. It is always convenient to express the input-output dynamic characteristics in form of a linear differential equation. So, often a nonlinear mathematical model is linearised and expressed in the form:

$$a_n \frac{d^n x_0}{dt^n} + a_{n-1} \frac{d^{n-1} x_0}{dt^{n-1}} + \cdots + a_1 \frac{dx_0}{dt} + a_0 x_0 = b_m \frac{d^m x_i}{dt^m} + b_{m-1} \frac{d^{m-1} x_i}{dt^{m-1}} + \cdots + b_1 \frac{dx_i}{dt} + b_0 x_i \qquad (6)$$

where $_ix$ and $_0x$ are the input and the output variables respectively. The above expression can also be expressed in terms of a transfer function, as:

$$G(s) = \frac{x_0(s)}{x_i(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} \cdots + b_1 s + b_0}{a_n s^n + b_{n-1} s^{n-1} \cdots + a_1 s + a_0} \qquad (7)$$

Normally *m*<*n* an *n* is called the order of the system. Commonly available sensor characteristics can usually be approximated as either *zero-th order*, *first order* or *second order* dynamics. Here are few such examples:

Potentiometer

Displacement sensors using potentiometric principle (Fig.4) have no energy storing elements. The output voltage e_o can be related with the input displacement x_i by an algebraic equation:

$$e_o(t)x_t = Ex_i(t); \quad or, \quad \frac{e_o(s)}{x_i(s)} = \frac{E}{x_t} = constant \qquad (8)$$

Where x_t is the total length of the potentiometer and *E* is the excitation voltage.. So, it can be termed as a *zeroth order system*.

Thermocouple

A bare thermocouple (Fig.5) has a mass (*m*) of the junction. If it is immersed in a fluid at a temperature T_f, then its dynamic performance relating the output voltage e_o and the input temperature T_f, can be expressed by the transfer function:

$$\frac{e_o(s)}{T_f(s)} = \frac{K_v}{1+s\tau} \qquad (9)$$

where, K_v = steady state voltage sensitivity of the thermocouple in V/ °C.

τ = time constant of the thermocouple $= \frac{mC}{hA}$

m = mass of the junction

C = specific heat

h = heat transfer co-efficient

A = surface area of the hot junction.

Hence, the bare thermocouple is a first order sensor. But if the bare thermocouple is put inside a metallic protective well (as it is normally done for industrial thermocouples) the order of the system increases due to the additional energy storing element (thermal mass of the well) and it becomes a second order system.

Seismic Sensor

Seismic sensors (Fig.6.) are commonly used for vibration or acceleration measurement of foundations. The transfer function between the input displacement $_ix$ and output displacement $_ox$ can be expressed as:

$$\frac{x_o(s)}{x_i(s)} = \frac{Ms^2}{Ms^2+Bs+K} \qquad (10)$$

where: M = mass of the seismic body

B = damping constant

K= spring constant

From the above transfer function, it can be easily concluded that the seismic sensor is a *second order system*.

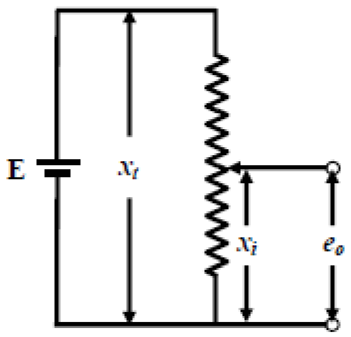


Fig. 4 Potentiometer

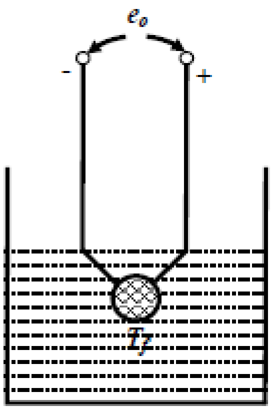


Fig. 5 Thermocouple

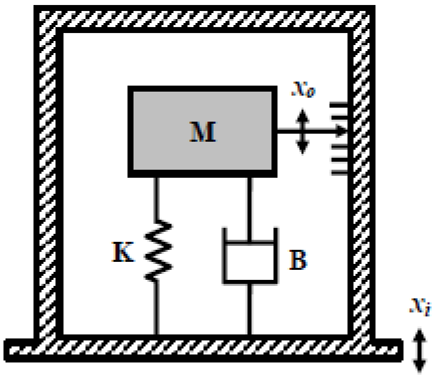


Fig. 6 Seismic sensor

Dynamic characteristics specifications are normally referred to the referred to the performance of the instrument with different test signals, e.g. impulse input, step input, ramp input and sinusoidal input. Few important specifications are:

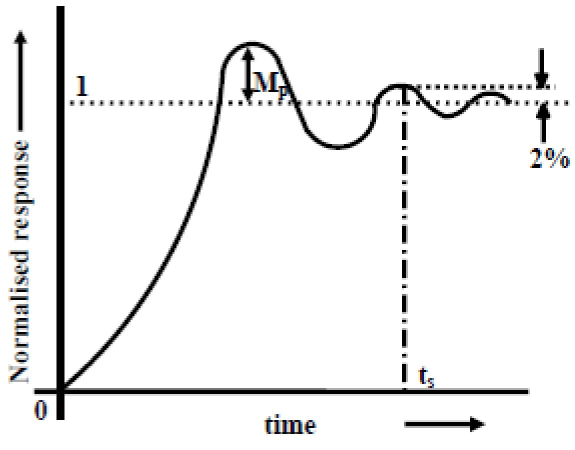


Fig. 7 Step response of a dynamic system

Step response performance

The normalized step response of a measurement system normally encountered is shown in Fig. 7. Two important parameters for classifying the dynamic response are:

Peak Overshoot *(M_p):* It is the maximum value minus the steady state value, normally expressed in terms of percentage.

Settling Time *(t_s):* It is the time taken to attain the response within ±2% of the steady state value.

Rise time *(t_r):* It is the time required for the response to rise from 10% to 90% of its final value.

Frequency Response Performance

The frequency response performance refers to the performance of the system subject to sinusoidal input of varying frequency.

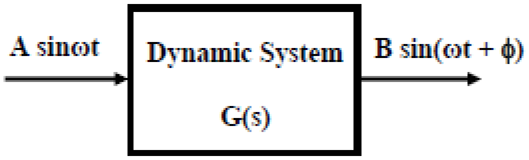


Fig. 8 Frequency Response

Suppose *G(s)* is the transfer function of the dynamic measurement system, represented by the general relation (7). If the input is a sinusoidal quantity of amplitude *A* and frequency ω, then in the steady state, the output will also be of same frequency, but of different amplitude *B*, and there would be a phase difference between the input and output. It can be shown that the amplitude ratio and the phase difference can be obtained as:

$$\frac{B}{A} = |G(j\omega)| \quad \text{and} \quad \phi = \angle G(j\omega) \qquad (11)$$

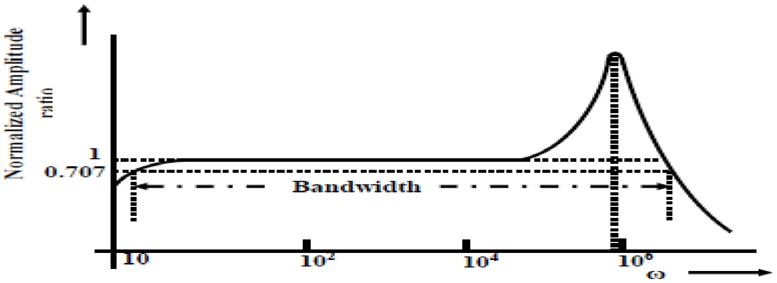


Fig. 9 Amplitude vs. frequency characteristics of a piezoelectric accelerometer

The plots showing variations of amplitude ratios and phase angle with frequency are called the magnitude and phase plots of the frequency response. Typical amplitude vs. frequency characteristics of a piezoelectric accelerometer is shown in Fig.9.

Bandwidth and Natural Frequency

From fig. 9, it is apparent that the amplitude is fairly constant over a range of frequencies. This range is called the *bandwidth* of the measuring system (to be precise, it is the frequency range in which the normalized amplitude ratio does not fall below 0.707, or -3 dB limit). The instrument is suitable for use in this range. The lower and upper limits are called the *lower* and *upper* cut off frequencies. The frequency at which the amplitude ratio attains a peak is called the (damped) *natural frequency* of the system. For further details, the reader is requested to refer any standard book on control systems.

3. Random Characteristics

If repeated readings of the same quantity of the measurand are taken by the same instrument, under same ambient conditions, they are bound to differ from each other. This is often due to some inherent sources of errors of the instrument that vary randomly and at any point of time it is very difficult to exactly say, what would be its value. For example, the characteristics of resistance and diode elements of an electronic circuit are random, due to two sources of noises: *thermal noise* and *flicker noise*. To characterize these behaviors, statistical terminologies are often used. Most common among them are *Mean* and *Standard deviation*. The mean of a set of readings is the most accurate estimation of the actual value, since, the positive and negative errors often cancelled out. On the other hand the standard deviation (σ) is a measure of the spread of the readings. If successive measurements of the same parameter under same ambient condition are taken and the mean and standard deviations are calculated, then assuming normal distribution of the randomness in measurements, we can say that 68% of the readings would fall within the range of mean±σ. Naturally, smaller the value ofσ, more would be the repeatability and higher would be the precision (refer equation (5)). This uncertainty limit is often extended to the 3σlimit, that means, that with 99% confidence, we can say that the any reading taken, would give a value within the range of mean±3σ. The interval of uncertainty is often called as the *confidence interval.*

Temperature Measurement:

The word *temper* was used in the seventeenth century to describe the quality of steel. It seems, after the invention of crude from of thermometer, the word *temperature* was coined to describe the degree of hotness or coolness of a material body. It was the beginning of seventeenth century when the thermometer – a temperature measuring instrument was first developed. Galileo Galilei is credited with the construction of first thermometer, although a Dutch scientist Drebbel also made similar instrument independently. The principle was simple. A bulb containing air with long vertical tube was inverted and dipped into a basin of water or coloured liquid. With the change in temperature of the bulb, the gas inside expanded or contracted, thus changing the level of the liquid column inside the vertical tube. A major drawback of the instrument was that it was sensitive not only to variation of temperature, but also to atmospheric pressure variation.

Successive developments of thermometers came out throughout seventeenth and eighteenth century. The liquid thermometer was developed during this time. The importance of two reference fixed temperatures was felt while graduating the temperature scales. Boiling point of water and melting point of ice provided two easily available references. But some other references were also tried. Fahrenheit developed a thermometer where, it seems, temperature of ice and salt mixture was taken as 0^{o} and temperature of human body as 96^{o}. These two formed the reference points, with which, the temperature of melting ice came as 32^{o} and that of boiling water as 212^{o}. In Celsius scale, the melting point of ice was chosen as 0^{o} and boiling point of water as 100^{o}. The concept of Kelvin scale came afterwards, where the absolute temperature of gas was taken as 0^{0} and freezing point of water as 273^{o}.

The purpose of early thermometers was to measure the variation of atmospheric or body temperatures. With the advancement of science and technology, now we require temperature measurement over a wide range and different atmospheric conditions, and that too with high accuracy and precision. To cater these varied requirements, temperature sensors based on different principles have been developed. They can be broadly classified in the following groups:

1. Liquid and gas thermometer
2. Bimetallic strip
3. Resistance thermometers (RTD and Thermistors)
4. Thermocouple
5. Junction semiconductor sensor
6. Radiation pyrometer

Within the limited scope of this course, we shall discuss few of the above mentioned temperature sensors, that are useful for measurement in industrial environment.

Resistance Thermometers

It is well known that resistance of metallic conductors increases with temperature, while that of semiconductors generally decreases with temperature. Resistance thermometers employing metallic conductors for temperature measurement are called *Resistance Temperature Detector (RTD)*, and those employing semiconductors are termed as *Thermistors*. RTDs are more rugged and have more or less linear characteristics over a wide temperature range. On the other hand Thermistors have high temperature sensitivity, but nonlinear characteristics.

Resistance Temperature Detector

The variation of resistance of metals with temperature is normally modeled in the form:

$$R_t = R_0[1 + \alpha(t - t_0) + \beta(t - t_0)^2 + ...] \qquad (1)$$

where R_t and R_0 are the resistance values at t° C and t_0°C respectively; α, β, etc. are constants that depends on the metal. For a small range of temperature, the expression can be approximated as:

$$R_t = R_0[1 + \alpha(t - t_0)] \qquad (2)$$

For Copper, $\alpha = 0.00427/^\circ C$.

Copper, Nickel and Platinum are mostly used as RTD materials. The range of temperature measurement is decided by the region, where the resistance-temperature characteristics are approximately linear. The resistance versus temperature characteristics of these materials is shown in fig.1, with t^o as 0^oC. Platinum has a linear range of operation upto 650^oC, while the useful range for Copper and Nickel are 120^oC and 300^oC respectively.

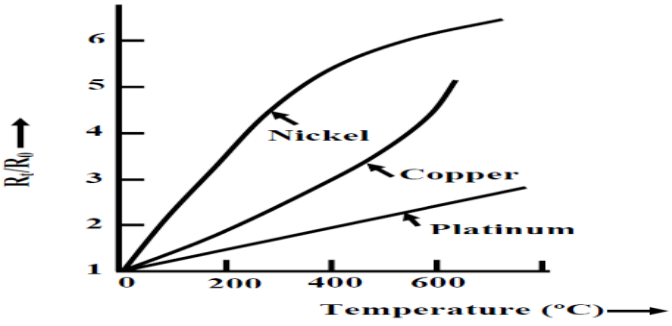


Fig. 1 Resistance-temperature characteristics of metals

Construction

For industrial use, bare metal wires cannot be used for temperature measurement. They must be protected from mechanical hazards such as material decomposition, tearing and other physical damages. The salient features of construction of an industrial RTD are as follows:

• The resistance wire is often put in a stainless steel well for protection against mechanical hazards. This is also
useful from the point of view of maintenance, since a defective sensor can be replaced by a good one while the
plant is in operation.
• Heat conducting but electrical insulating materials like mica is placed in between the well and the resistance
material.
• The resistance wire should be carefully wound over mica sheet so that no strain is developed due to length
expansion of the wire.

Fig. 2 shows the cut away view of an industrial RTD.

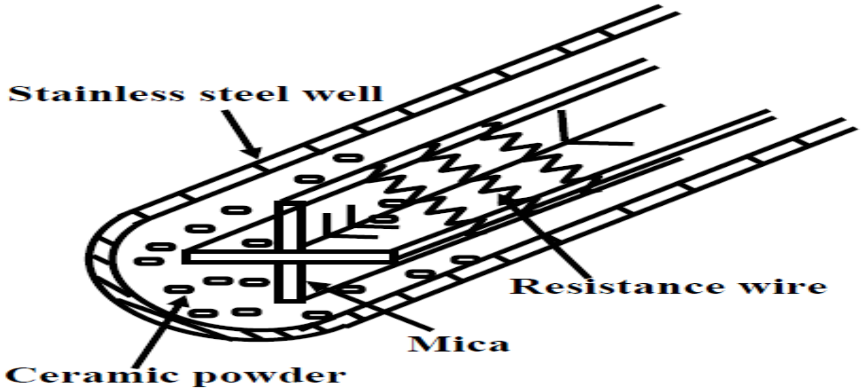
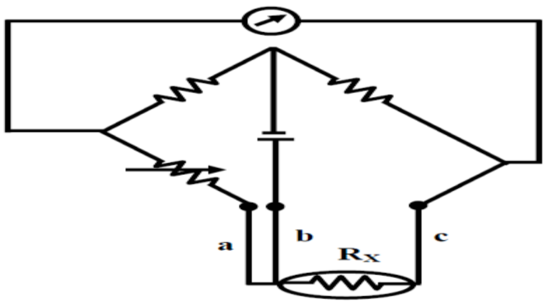


Fig. 2 Construction of an industrial RTD

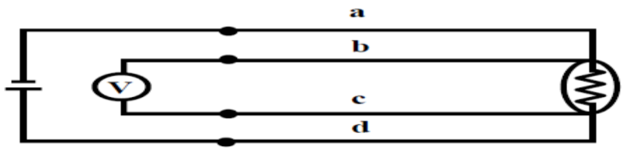
Signal conditioning

The resistance variation of the RTD can be measured by a bridge, or directly by volt-ampere method. But the major constraint is the contribution of the lead wires in the overall resistance measured.

Since the length of the lead wire may vary, this may give a false reading in the temperature to be measured. There must be some method for compensation so that the effect of lead wires is resistance measured is eliminated. This can be achieved by using either a *three wire RTD*, or a *four wire RTD*. Both the schemes of measurement are shown in fig. 3. In three wire method one additional dummy wire taken from the resistance element and connected in a bridge (fig. 3(a)) so that the two lead wires are connected to two adjacent arms of the bridge, thus canceling each other’s effect. In fig. 3(b) the four wire method of measurement is shown. It is similar to a four terminal resistance and two terminals are used for injecting current, while two others are for measuring voltage.



a. b. c: lead wires
Fig. 3(a) Three wire RTD



a. b. c: lead wires
Fig. 3(b) Four wire RTD

Thermistor

Thermistors are semiconductor type resistance thermometers. They have very high sensitivity but highly nonlinear characteristics. This can be understood from the fact that for a typical 2000 Ω the resistance change at 25^{o}C is

80Ω/^{o}C, whereas for a 2000 Ω platinum RTD the change in resistance at 25^{o}C is 7Ω/^{o}C. Thermistors can be of two types: (a) Negative temperature co-efficient (NTC) thermistors and (b) Positive temperature co-efficient (PTC) thermistors. Their resistance-temperature characteristics are shown in fig. 4(a) and 4(b) respectively.

The NTC thermistors, whose characteristics are shown in fig. 4(a) is more common. Essentially, they are made from oxides of iron, manganese, magnesium etc. Their characteristics can be expressed as:

$$R_T = R_0 e^{\beta(\frac{1}{T} - \frac{1}{T_0})} \qquad (3)$$

where,

R_T is the resistance at temperature T (K)
R_0 is the resistance at temperature T_0 (K)
T_0 is the reference temperature, normally 25°C
β is a constant, its value is decided by the characteristics of the material, the nominal value is taken as 4000.

From (3), the resistance temperature co-efficient can be obtained as:

$$\alpha_T = \frac{1}{R_T}\frac{dR_T}{dT} = -\frac{\beta}{T^2} \qquad (4)$$

It is clear from the above expression that the negative sign of α_T indicates the negative resistance-temperature characteristics of the NTC thermistor.

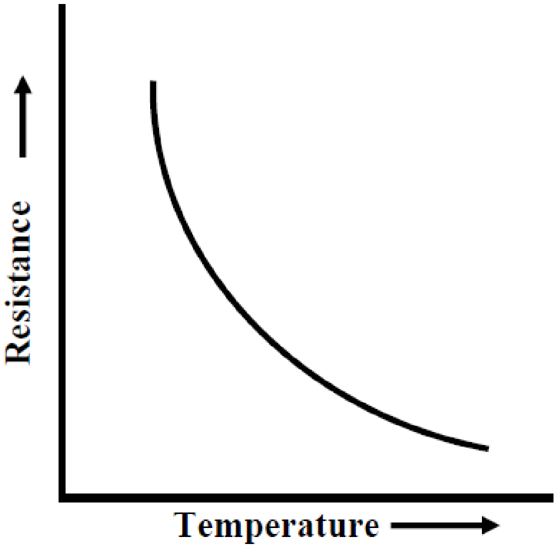


Fig. 4(a) Characteristics of a NTC

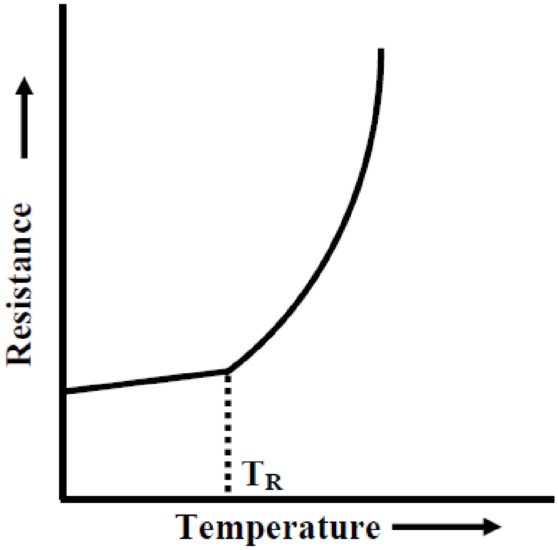


Fig. 4(b) Characteristics of a PTC

Useful range of themistors is normally -100 to +300^{o}C. A single thermistor is not suitable for the whole range of measurement. Moreover, existing thermistors are not interchangeable. There is a marked spread in nominal resistance and the temperature coefficient between two thermistors of same type. So, if a defective thermistor is to be replaced by a new thermistor similar type, a fresh calibration has to be carried out before use. Commercially available thermistors have nominal values of 1K, 2K, 10K, 20K, 100K etc. The nominal values indicate the resistance value at 25^{o}C. Thermistors are available in different forms: bead type, rod type disc type etc. The small size of the sensing element makes it suitable for measurement of temperature at a point. The time constant is also very small due to the small thermal mass involved. The nonlinear negative temperature characteristics also give rise to error due to *self-heating effect*. When current is flowing through the thermistor, the heat generated due to the -loss may increase the temperature of the resistance element, which may further decrease the resistance and increase the current further. This effect, if not tackled properly, may damage the thermistor permanently. Essentially, the current flowing should be restricted below the specified value to prevent this damage. Alternatively, the thermistor may be excited by a constant current source. RI$_2$ The nonlinear characteristics of thermistors often creates problem for temperature measurement, and it is often desired to linearise the thermistor characteristics. This can be done by adding one fixed resistance parallel to the thermistor.

The resistance temperature characteristics of the equivalent resistance would be more linear, but at the cost of sensitivity.

The Positive Temperature Coefficient (PTC) thermistor have limited use and they are particularly used for protection of motor and transformer widings. As shown in fig. 4(b), they have low and relatively constant resistance below a threshold temperature T_R, beyond which the resistance increases rapidly. The PTC thermistors are made from compound of barium, lead and strontium titanate.

3. Thermocouple

Thomas Johan Seeback discovered in 1821 that thermal energy can produce electric current. When two conductors made from dissimilar metals are connected forming two common junctions and the two junctions are exposed to two different temperatures, a net thermal emf is produced, the actual value being dependent on the materials used and the temperature difference between hot and cold junctions. The thermoelectric emf generated, in fact is due to the combination of two effects: *Peltier effect* and *Thomson effect*. A typical thermocouple junction is shown in fig. 5. The emf generated can be approximately expressed by the relationship:

$$e_0 = C_1(T_1 - T_2) + C_2(T_1^2 - T_2^2)\,\mu v \qquad (5)$$

where T_1 and T_2 are hot and cold junction temperatures in K. C_1 and C_2 are constants depending upon the materials. For Copper/ Constantan thermocouple, C_1=62.1 and C_2=0.045.

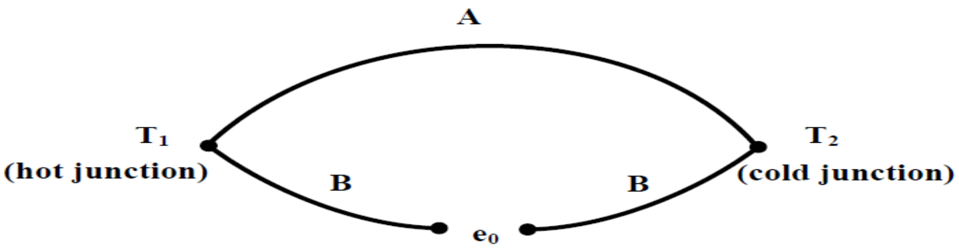


Fig. 5 A typical thermocouple

Pressure and Force Measurement

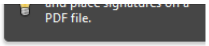
Pressure Measurement

Measurement of pressure inside a pipeline or a container in an industrial environment is a challenging task, keeping

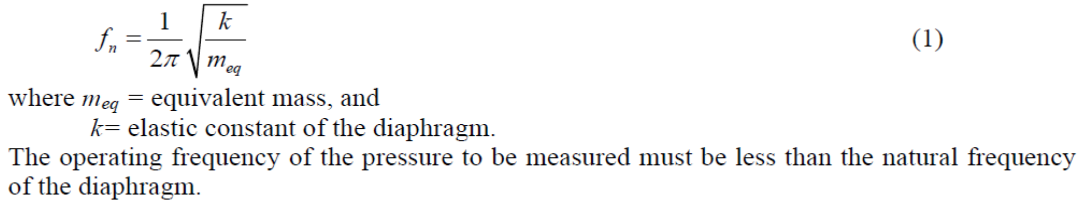
in mind that pressure may be very high, or very low (vacuum); the medium may be liquid, or gaseous. We will not discuss the vacuum pressure measuring techniques; rather try to concentrate on measurement techniques of pressure higher than the atmospheric. They are mainly carried out by using elastic elements: diaphragms, bellows and Bourdon tubes. These elastic elements change their shape with applied pressure and the change of shape can be measured using suitable deflection transducers. Their basic constructions and principle of operation are explained below.

Diaphragms

Diaphragms may be of three types: Thin plate, Membrane and Corrugated diaphragm. This classification is based on the applied pressure and the corresponding displacements. Thin plate (fig. 1(a)) is made by machining a solid block and making a circular cross sectional area with smaller thickness in the middle. It is used for measurement of relatively higher pressure. In a membrane the sensing section is glued in between two solid blocks as shown in fig. 1(b). The thickness is smaller; as a result, when pressure is applied on one side, the displacement is larger. The sensitivity can be further enhanced in a corrugated diaphragm (fig. 1(c)), and a large deflection can be obtained for a small change in pressure; however at the cost of linearity. The materials used are Bronze, Brass, and Stainless steel. In recent times, Silicon has been extensively used the diaphragm material in MEMS (Micro Electro Mechanical Systems) pressure sensor. Further, the natural frequency of a diaphragm can be expressed as:



PDF file.

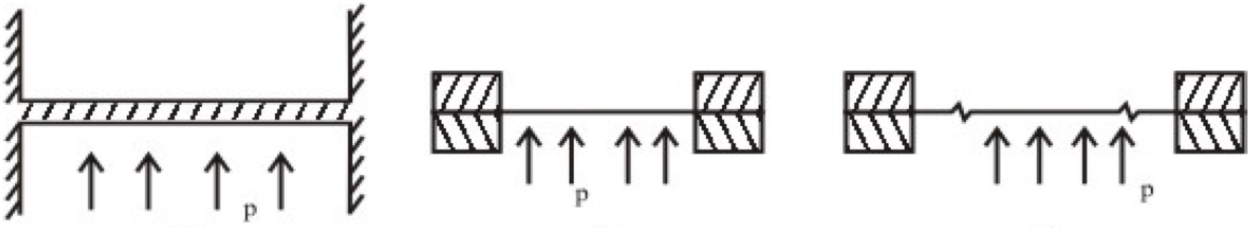


$$f_n = \frac{1}{2\pi}\sqrt{\frac{k}{m_{eq}}} \qquad (1)$$

where m_{eq} = equivalent mass, and

k= elastic constant of the diaphragm.

The operating frequency of the pressure to be measured must be less than the natural frequency of the diaphragm.



(a) (b) (c)

Fig. 1 (a) Thin plate, (b) Membrane and (c) Corrugated diaphragm.

When pressure is applied to a diaphragm, it deflates and the maximum deflection at the centre (y_0) can measured using a displacement transducer. For a Thin plate, the maximum deflection y_0 is small ($y_0 < 0.3t$) and referring fig. 2, a linear relationship between p and y_0 exists as:

$$y_0 = \frac{3}{16} p \frac{(1-\nu^2)}{Et^3} R^4 \qquad (2)$$

where, E= Modulus of elasticity of the diaphragm material, and

ν = Poisson's ratio.

However, the allowable pressure should be less than:

$$p_{\max} = 1.5\left(\frac{t}{R}\right)^2 \sigma_{\max} \qquad (3)$$

where, $\sigma_{\max}$ is the safe allowable stress of the material.

For a membrane, the deflection is larger, and the relationship between p and y_0 is nonlinear and can be expressed as (for $\nu = 0.3$):

$$p = 3.58 \frac{Et^3}{R^4} y_0^{\,3} \qquad (4)$$

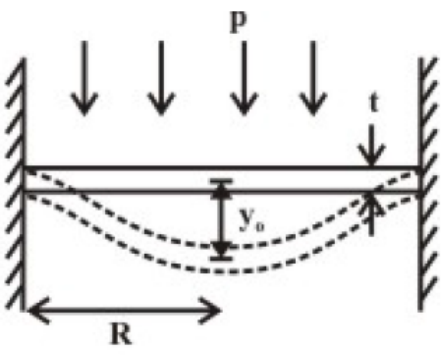


Fig. 2 Displacement of a diaphragm

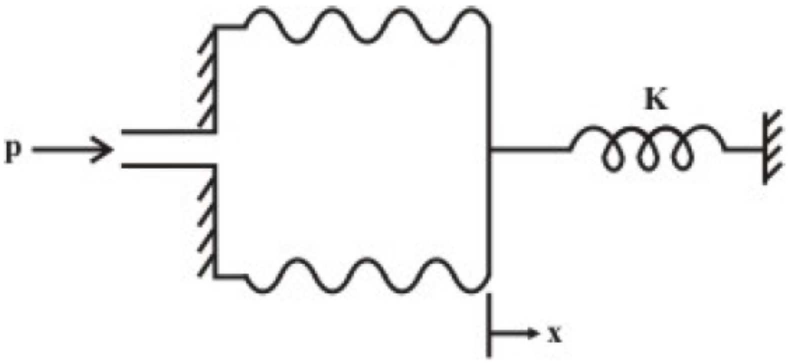


Fig. 3 Bellows

For a corrugated diaphragm, it is difficult to give any definite mathematical relationship between p and y_0; but the relationship is also highly nonlinear.

As the diaphragm deflates, strains of different magnitudes and signs are generated at different locations of the diaphragm. These strains can also be measured by effectively placing four strain gages on the diaphragm. The principle of strain gage will be discussed in the next section.

Bellows

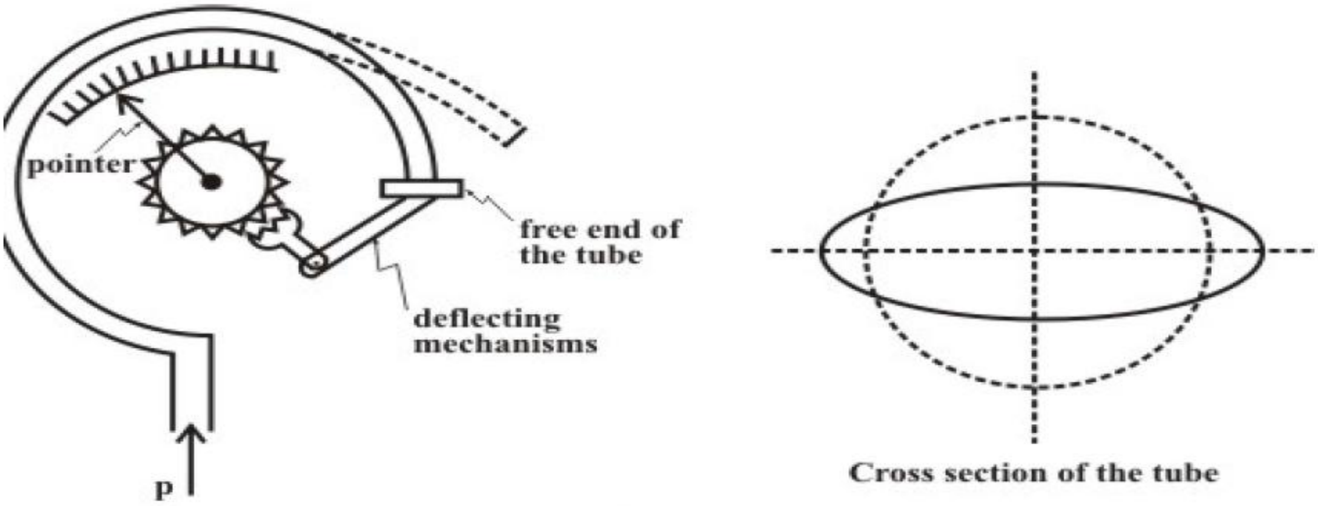
Bellows (fig. 3) are made with a number of convolutions from a soft material and one end of it is fixed, wherein air can go through a port. The other end of the bellows is free to move. The displacement of the free end increases with the number of convolutions used. Number of convolutions varies between 5 to 20. Often an external spring is used opposing the movement of the bellows; as a result a linear relationship can be obtained from the equation:

$$p\,A = k\,x \qquad (5)$$

where, *A* is the area of the bellows,
k is the spring constant and
x is the displacement of the bellows

Bourdon Tube

Bourdon tube pressure gages are extensively used for local indication. This type of pressure gages were first developed by E. Bourdon in 1849. Bourdon tube pressure gages can be used to measure over a wide range of pressure: form vacuum to pressure as high as few thousand psi. It is basically consisted of a C-shaped hollow tube, whose one end is fixed and connected to the pressure tapping, the other end free, as shown in fig. 4. The cross section of the tube is elliptical. When pressure is applied, the elliptical tube tries to acquire a circular cross section; as a result, stress is developed and the tube tries to straighten up. Thus the free end of the tube moves up, depending on magnitude of pressure. A deflecting and indicating mechanism is attached to the free end that rotates the pointer. The materials used are commonly Phosphor Bronze, Brass and Beryllium Copper. For a 2" overall diameter of the C-tube the useful travel of the free end is approximately"1/8. Though the C-type tubes are most common, other shapes of tubes, such as helical, twisted or spiral tubes are also in use.



Cross section of the tube

Fig. 4 Bourdon tube

Measurement of Force

The most popular method for measuring force is using strain gage. We measure the strain developed due to force using strain gages; and by multiplying the strain with the effective cross sectional area and Young's modulus of the material, we can obtain force. Load cells and Proving rings are two common methods for force measurement using strain gages. We will first discuss the principle of strain gage and then go for the force measuring techniques.

Strain Gage

Strain gage is one of the most popular types of transducer. It has got a wide range of applications. It can be used for measurement of force, torque, pressure, acceleration and many other parameters. The basic principle of operation of a strain gage is simple: when strain is applied to a thin metallic wire, its dimension changes, thus changing the resistance of the wire. Let us first investigate what are the factors, responsible for the change in resistance.

Gage Factor

Let us consider a long straight metallic wire of length *l* circular cross section with diameter *d* (fig. 5). When this wire is subjected to a force applied at the two ends, a strain will be generated and as a result, the dimension will change

(l changing to $l+\Delta l$, d changing to $d+\Delta d$ and A changing to $A+\Delta A$).For the time being, we are considering that all the changes are in positive direction. Now the resistance of the wire:

$$R = \frac{\rho l}{A}$$, where ρ is the resistivity.

From the above expression, the change in resistance due to strain:

$$\begin{aligned}\Delta R &= \left(\frac{\partial R}{\partial l}\right)\Delta l + \left(\frac{\partial R}{\partial A}\right)\Delta A + \left(\frac{\partial R}{\partial \rho}\right)\Delta \rho \\ &= \frac{\rho}{A}\Delta l - \frac{\rho}{A^2}\Delta A + \frac{l}{A}\Delta \rho \\ &= R\frac{\Delta l}{l} - R\frac{\Delta A}{A} + R\frac{\Delta \rho}{\rho}\end{aligned}$$

or,

$$\frac{\Delta R}{R} = \frac{\Delta l}{l} - \frac{\Delta A}{A} + \frac{\Delta \rho}{\rho} \qquad (6)$$

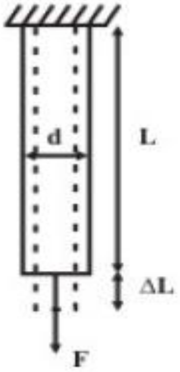


Fig. 5 Change of resistance with strain

Now, for a circular cross section, $A = \frac{\pi d^2}{4}$; from which, $\Delta A = \frac{\pi d}{2}\Delta d$. Alternatively,

$$\frac{\Delta A}{A} = 2\frac{\Delta d}{d}$$

Hence,

$$\frac{\Delta R}{R} = \frac{\Delta l}{l} - 2\frac{\Delta d}{d} + \frac{\Delta \rho}{\rho} \qquad (7)$$

Now, the *Poisson's Ratio* is defined as:

$$\upsilon = -\frac{lateral \quad strain}{longitudinal \quad strain} = -\frac{\Delta d/_{d}}{\Delta l/_{l}}$$

The Poisson's Ratio is the property of the material, and does not depend on the dimension. So, (6) can be rewritten as:

$$\frac{\Delta R}{R} = (1+2\upsilon)\frac{\Delta l}{l} + \frac{\Delta \rho}{\rho}$$

Hence,

$$\frac{\Delta R/_{R}}{\Delta l/_{l}} = 1 + 2\upsilon + \frac{\Delta \rho/_{\rho}}{\Delta l/_{l}}$$

The last term in the right hand side of the above expression, represents the change in resistivity of the material due to applied strain that occurs due to the *piezo-resistance property* of the material. In fact, all the elements in the right hand side of the above equation are independent of the geometry of the wire, subjected to strain, but rather depend on the material property of the wire. Due to this reason, a term *Gage Factor* is used to characterize the performance of a strain gage. The Gage Factor is defined as:

$$G := \frac{\Delta R/_{R}}{\Delta l/_{l}} = 1 + 2\upsilon + \frac{\Delta \rho/_{\rho}}{\Delta l/_{l}} \qquad (8)$$

For normal metals the Poisson's ratio υ varies in the range:

$$0.3 \le \upsilon \le 0.6 ,$$

while the piezo-resistance coefficient varies in the range:

$$0.2 \le \frac{\Delta \rho/_{\rho}}{\Delta l/_{l}} \le 0.6 .$$

Thus, the Gage Factor of metallic strain gages varies in the range 1.8 to 2.6. However, the semiconductor type strain gages have a very large Gage Factor, in the range of 100-150. This is attained due to dominant piezo-resistance property of semiconductors. The commercially available strain gages have certain fixed resistance values, such as, 120Ω, 350 Ω, 1000 Ω, etc. The manufacturer also specifies the Gage Factor and the maximum gage current to avoid self-heating (normally in the range 15 mA to 100 mA).

The choice of material for a metallic strain gage should depend on several factors. The material should have low temperature coefficient of resistance. It should also have low coefficient for thermal expansion. Judging from all these factors, only few alloys qualify for a commercial metallic strain gage. They are:

Advance (55% Cu, 45% Ni): Gage Factor between 2.0 to 2.2

Nichrome (80% Ni, 20% Co): Gage Factor between 2.2 to 2.5

Apart from these two, *Isoelastic* -another trademarked alloy with Gage Factor around 3.5 is also in use. Semiconductor type strain gages, though having large Gage Factor, find limited use, because of their high sensitivity and nonlinear characteristics.

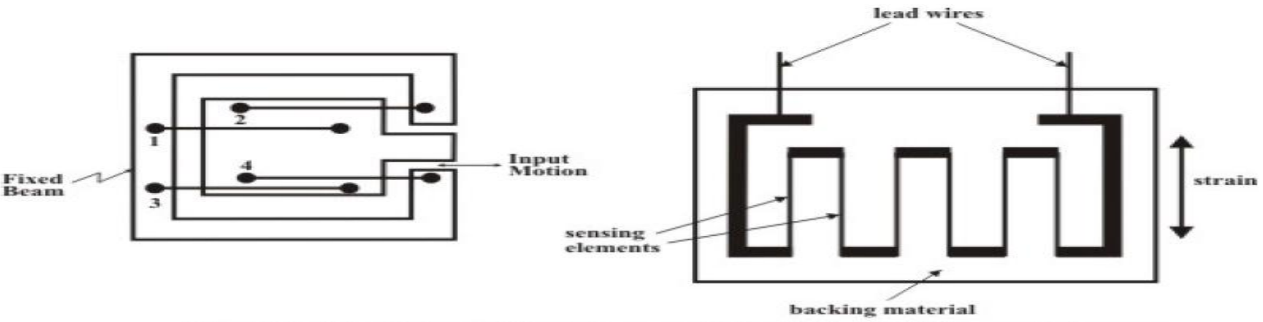


Fig. 6 (a) Unbonded metallic strain gage, (b) bonded metal foil type strain gage

Metallic Strain Gage

Most of the strain gages are metallic type. They can be of two types: *unbonded* and *bonded*. The unbonded strain gage is normally used for measuring strain (or displacement) between a fixed and a moving structure by fixing four metallic wires in such a way, so that two are in compression and two are in tension, as shown in fig. 6 (a). On the other hand, in the bonded strain gage, the element is fixed on a backing material, which is permanently fixed over a structure, whose strain has to be measured, with adhesive. Most commonly used bonded strain gages are *metal foil type*. The construction of such a strain gage is shown in fig. 6(b). The metal foil type strain gage is manufactured by photo-etching technique. Here the thin strips of the foil are the active elements of the strain gage, while the thick ones are for providing electrical connections. Because of large area of the thick portion, their resistance is small and they do not contribute to any change in resistance due to strain, but increase the heat dissipation area. Also it is easier to connect the lead wires with the strain gage. The strain gage in fig. 6(b) can measure strain in one direction only. But if we want to measure the strain in two or more directions at the same point, strain gage *rosette*, which is manufactured by stacking multiple strain gages in different directions, is used. Fig. 7 shows a three-element strain gage rosette stacked at $45^{0.}$

The *backing material*, over which the strain gage is fabricated and which is fixed with the strain measuring structure has to satisfy several important properties. Firstly, it should have high mechanical strength; it should also have high dielectric strength. But the most important it should have is that it should be non-hygroscopic, otherwise, absorption of moisture will cause bulging and generate local strain. The backing materials normally used are impregnated paper, fibre glass, etc. The *bonding material* used for fixing the strain gage permanently to the structure should also be non-hygroscopic. Epoxy and Cellulose are the bonding materials normally used.

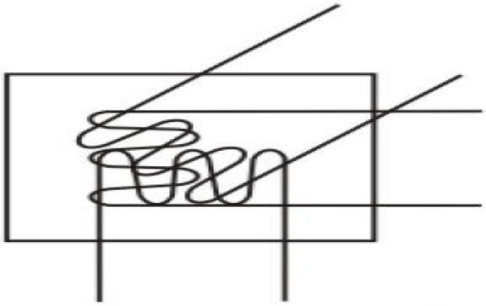


Fig. 7 Three-element strain gage rosette- 45° stacked.

Semiconductor type Strain Gage

Semiconductor type strain gage is made of a thin wire of silicon, typically 0.005 inch to 0.0005 inch, and length 0.05 inch to 0.5 inch. They can be of two types: *p*-type and *n*-type. In the former the resistance increases with positive strain, while, in the later the resistance decreases with temperature. The construction and the typical characteristics of a semiconductor strain gage are shown in fig.8.

MEMS pressure sensors is now a day’s becoming increasingly popular for measurement of pressure. It is made of a small silicon diagram with four piezo-resistive strain gages mounted on it. It has an in-built signal conditioning circuits and delivers measurable output voltage corresponding to the pressure applied. Low weight and small size of the sensor make it suitable for measurement of pressure in specific applications.

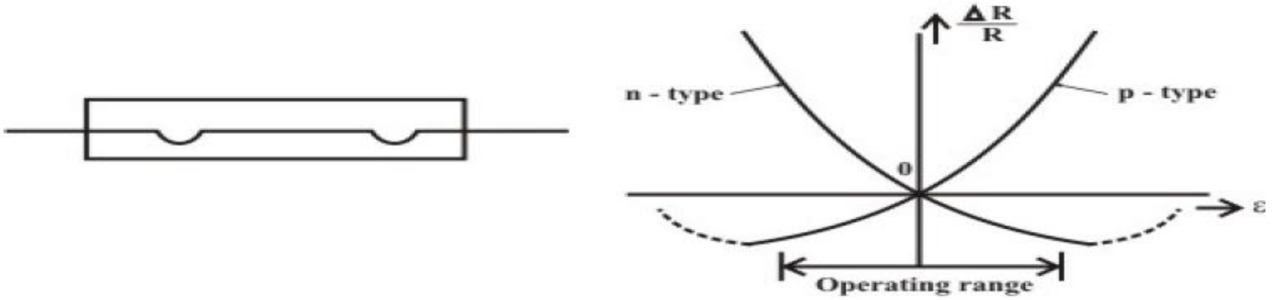


Fig. 8 (a) construction and (b) characteristics of a semiconductor strain gage

Strain Gage Bridge

Normal strain experienced by a strain gage is in the range of micro strain (typical value: 100 x 10^{-6}). As a result, the change in resistance associated with it is small (*R*Δ/Rε*G*=). So if a single strain gage is connected to a wheatstone bridge, with three fixed resistances, the bridge output voltage is going to be linear (recall, that we say the bridge output voltage would be linearly varying with *R*Δ/R, if *RR*Δ does not exceed 0.1). But still then, a single strain gage is normally never used in a wheatstone bridge. This is not because of improving linearity, but for obtaining *perfect temperature compensation*. Suppose one strain gage is connected to a bridge with three fixed arms. Due to temperature rise, the strain gage resistance will change, thus making the bridge unbalance, thus giving an erroneous signal, even if no strain is applied. If two identical strain gages are fixed to the same structure, one measuring compressional strain and the other tensile strain, and connected in the adjacent arms of the bridge, temperature compensation can be achieved. If the temperature increases, both the strain gage resistances will be affected in the same way, thus maintaining the bridge balance under no strain condition. One more advantage of using the push-pull configuration is increasing the sensitivity. In fact, all the four arms of the bridge can be formed by four active gages; this will improve the sensitivity further, while retaining the temperature compensation property. A typical strain gage bridge is shown in fig. 9. It can be shown that if nominal resistances of the strain gages are same and also equal gage factor *G*, then the unbalanced voltage is given be:

$$e_0 = \frac{EG}{4}(\varepsilon_1 + \varepsilon_3 - \varepsilon_2 - \varepsilon_4) \qquad (9)$$

where ε_1, ε_2, ε_3, ε_4 are the strains developed with appropriate signs.

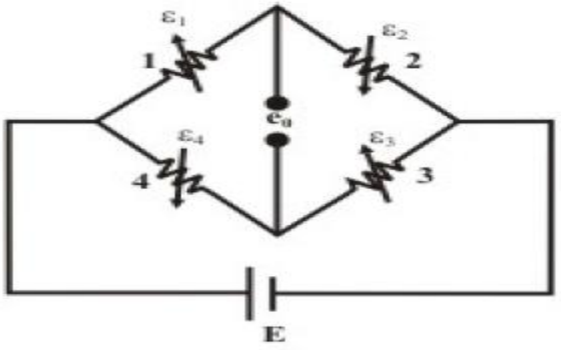
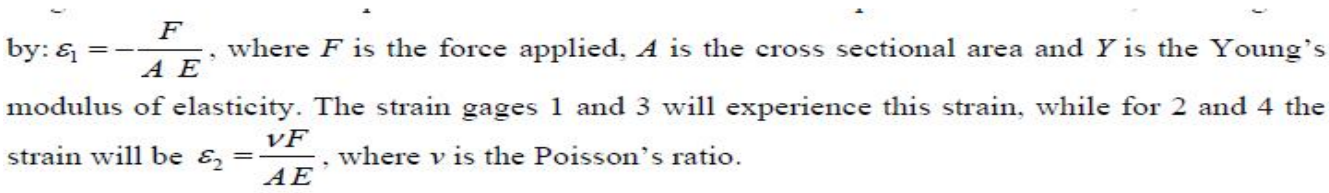


Fig. 9 Strain gage bridge with four active gages

Load Cell

Load cells are extensively used for measurement of force; weigh bridge is one of the most common applications of load cell. Here two strain gages are fixed so as to measure the longitudinal strain, while two other measuring the transverse strain, as shown in fig. 10. The strain gages, measuring the similar strain (say, tensile) are placed in the opposite arms, while the adjacent arms in the bridge should measure opposite strains (one tensile, the other compressional). If the strain gages are identical in characteristics, this will provide not only the perfect temperature coefficient, but also maximum obtainable sensitivity from the bridge. The longitudinal strain developed in the load cell would be compressional in nature, and is given



by: $\varepsilon_1 = -\dfrac{F}{A\ E}$, where F is the force applied, A is the cross sectional area and Y is the Young's modulus of elasticity. The strain gages 1 and 3 will experience this strain, while for 2 and 4 the strain will be $\varepsilon_2 = \dfrac{\nu F}{AE}$, where ν is the Poisson's ratio.

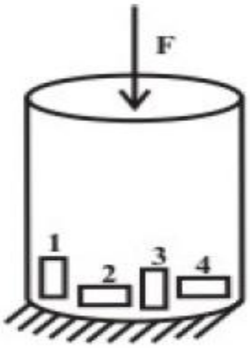


Fig. 10 Load cell with four strain gages

Proving Ring

Proving Rings can be used for measurement of both compressional and tensile forces. The advantage of a Proving Ring is that, because of its construction more strain can be developed compared to a load cell. The typical construction of a Proving Ring is shown in fig.11. It consists of a hollow cylindrical beam of radius *R*, thickness *t* and axial width *b*. The two ends of the ring are fixed with the structures between which force is measured. Four strain gages are mounted on the walls of the proving ring, two on the inner wall, and two on the outer wall. When force is applied as shown, gages 2 and 4 will experience strain –*ε* (compression), while gages 1 and 3 will experience strain + *ε* (tension). The magnitude of the strain is given by the expression:

$$\varepsilon = \frac{1.08FR}{Ebt^2} \qquad (10)$$

The four strain gages are connected in a bridge and the unbalanced voltage can easily be calibrated in terms of force to be measured.

Cantilever Beam

Cantilever beam can be used for measurement up to 10 kg of weight. One end of the cantilever is fixed, while the other end is free; load is applied at this end, as shown in fig. 12. The strain developed at the fixed end is given by the expression:

$$\varepsilon = \frac{6Fl}{Ebt^2} \qquad (11)$$

where,
l = length of the beam
t = thickness of the cantilever
b = width of the beam
E = Young’s modulus of the material

The strain developed can be measured by fixing strain gages at the fixed end: two on the top side of the beam, measuring tensile strain +*ε* and two on the bottom measuring compressional strain – *ε* (as shown in fig. 12) and using eqn. (9).

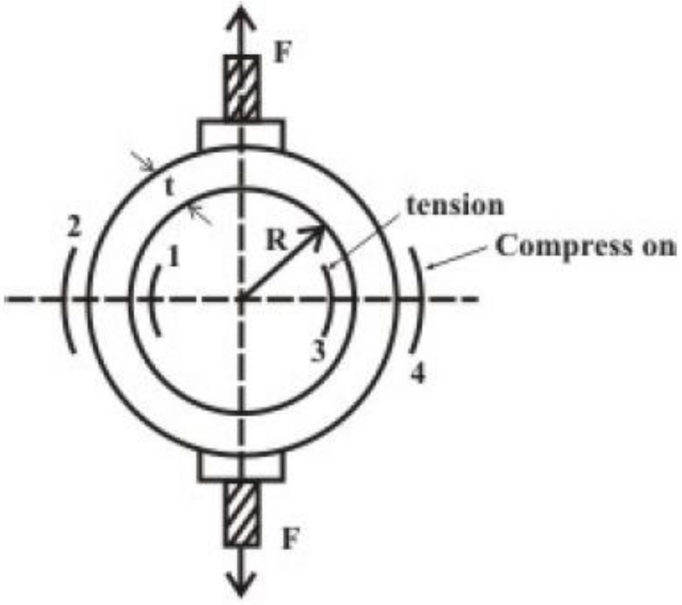


Fig. 11 Proving Ring

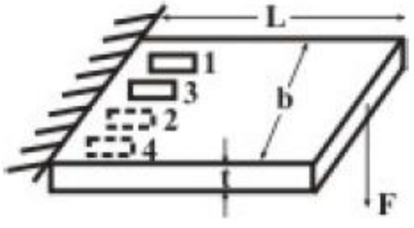


Fig. 12 Cantilever Beam

Displacement Measurement

Broadly speaking, displacement measurement can be of two types: contact and noncontact types. Besides the measurement principles can be classified into two categories: electrical sensing and optical sensing. In electrical sensing, passive electrical sensors are used variation of either inductance or capacitance with displacement is measured. On the other hand the optical method mainly works on the principle of intensity variation of light with distance. Interferometric technique is also used for measurement of very small displacement in order of nanometers. But this technique is more suitable for laboratory purpose, not very useful for industrial applications.

Potentiometer

Potentiometers are simplest type of displacement sensors. They can be used for linear as well as angular displacement measurement, as shown in Fig. 1. They are the resistive type of transducers and the output voltage is proportional to the displacement and is given by:

$$e_o = \frac{x_i}{x_t} E \quad ,$$

where x_i is the input displacement, x_t is the total displacement and E is the supply voltage.

The major problem with potentiometers is the contact problem resulting out of wear and tear between the moving and the fixed parts. As a result, though simple, application of potentiometers is limited.

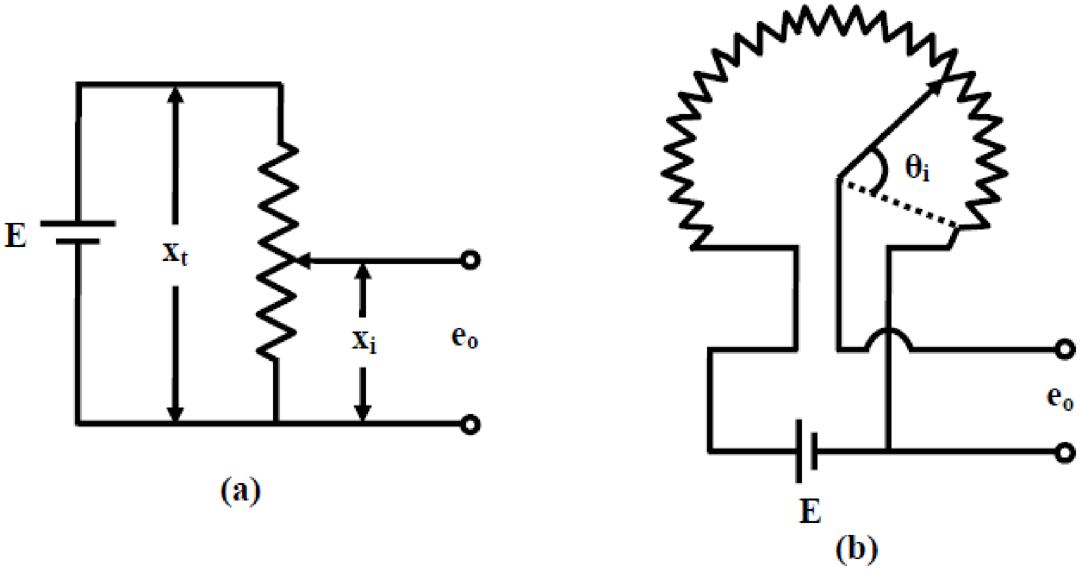


Fig. 1 Potentiometer
(a) Linear
(b) Rotary

Linear Variable Differential transformer (LVDT)

LVDT works on the principle of variation of mutual inductance. It is one of the most popular types of displacement sensor. It has good linearity over a wide range of displacement. Moreover the mass of the moving body is small, and the moving body does not make any contact with the static part, thus minimizing the frictional resistance. Commercial LVDTs are available with full scale displacement range of 0.25mm to ±±25mm. Due to the low inertia of the core, the LVDT has a good dynamic characteristics and can be used for time varying displacement measurement range.

The construction and principle of operation of LVDT can be explained with Fig. 2(a) and Fig. 2(b). It works on the principle of variation of the mutual inductance between two coils with displacement. It consists of a primary winding and two identical secondary windings of a transformer, wound over a tubular former, and a ferromagnetic core of annealed nickel-iron alloy moves through the former. The two secondary windings are connected in series opposition, so that the net output voltage is the difference between the two. The primary winding is excited by 1-10V r.m.s. A.C. voltage source, the frequency of excitation may be anywhere in the range of 50 Hz to 50 KHz. The output voltage is zero when the core is at central position (voltage induced in both the secondary windings are same, so the difference is zero), but increasing as the core moves away from the central position, in either direction. Thus, from the measurement of the output voltage only, one cannot predict, the direction of the core movement. A phase sensitive detector (PSD) is a useful circuit to make the measurement direction sensitive. It is connected at the output of the LVDT and compares the phase of the secondary output with the primary signal to judge the direction of movement. The output of the phase sensitive detector after low pass filtering becomes a d.c voltage for a steady deflection. The output voltage after PSD vs. displacement characteristics is shown in Fig. 2(c).

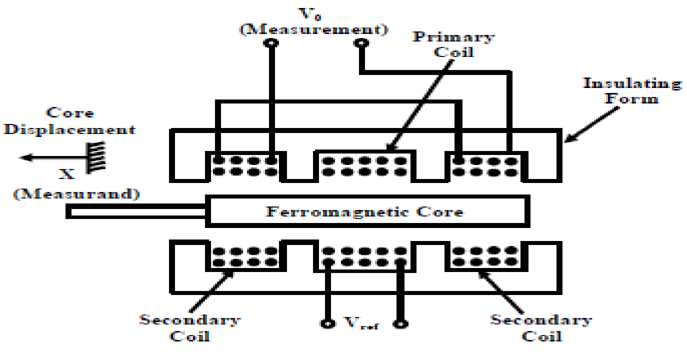


Fig. 2(a) Construction of LVDT.

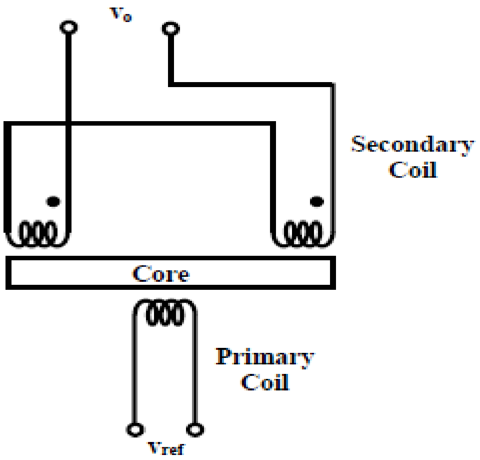


Fig. 2(b) Series opposition connection of secondary windings.

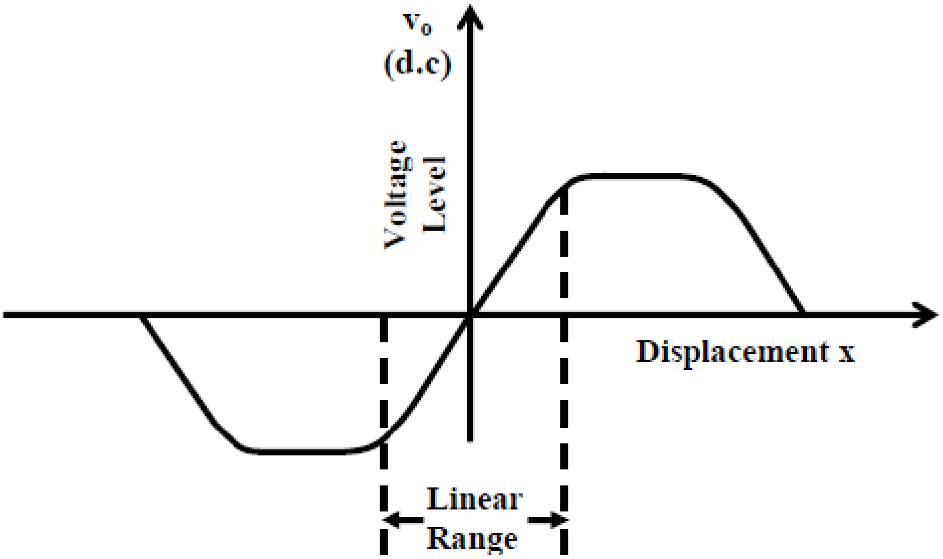


Fig. 2(c) Output voltage vs. displacement characteristics of LVDT after Phase sensitive detection.

Inductive type Sensors

LVDT works on the principle of variation of mutual inductance. There are inductive sensors for measurement of displacement those are based on the principle of variation of self inductance. These sensors can be used for proximity detection also. Such a typical scheme is shown in Fig. 3. In this case the inductance of a coil changes as a ferromagnetic object moves close to the magnetic former, thus change the reluctance of the magnetic path. The measuring circuit is usually an a.c. bridge.

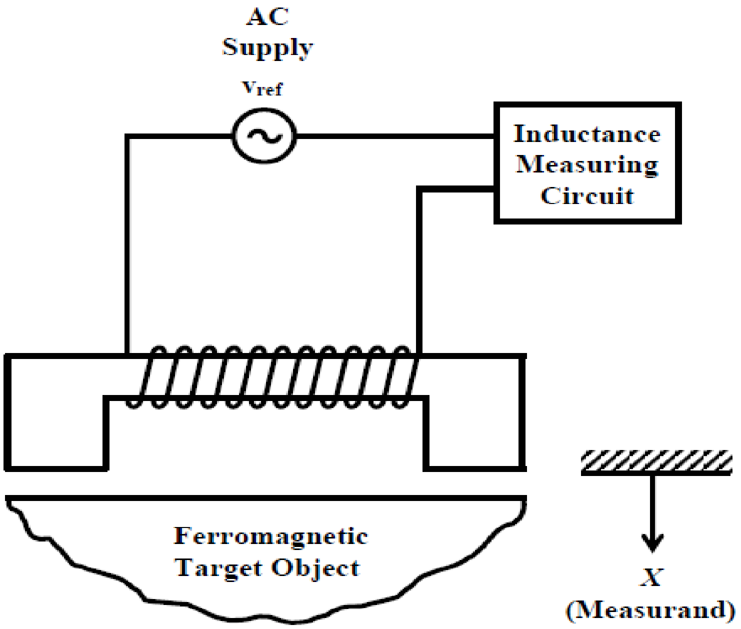


Fig. 3 Schematic diagram of a self inductance type proximity sensor.

Rotary Variable Differential Transformer (RVDT)

Its construction is similar to that of LVDT, except the core is designed in such a way that when it rotates the mutual inductance between the primary and each of the secondary coils changes linearly with the angular displacement. Schematic diagram of a typical RVDT is shown in Fig. 4.

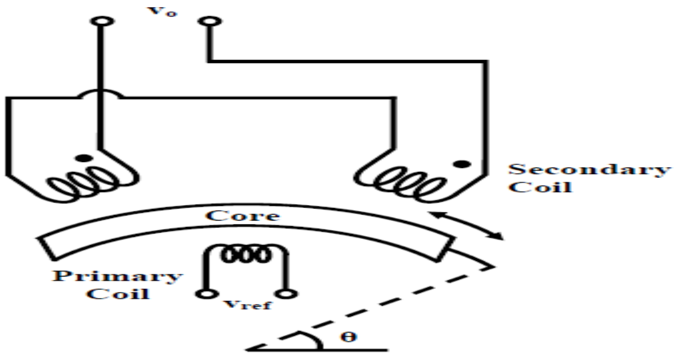


Fig. 4 Rotary Variable Differential Transformer (RVDT)

Resolver

Resolvers also work on the principle of mutual inductance variation and are widely used for measurement of rotary motion. The basic construction is shown in Fig. 5. A resolver consists of a rotor containing a primary coil and two stator windings (with equal number of turns) placed perpendicular to each other. The rotor is directly attached to the object whose rotation is being measured. If a.c. excitation of r.m.s voltage V_r is applied, then the induced voltages at two stator coils are given by:

$$v_{01} = KV_r \cos\theta$$

and $$v_{02} = KV_r \sin\theta \text{ ; where } K \text{ is a constant.}$$

By measuring these two voltages the angular position can be uniquely determined, provided . Phase sensitive detection is needed if we want to measure for angles in all the four quadrants. 0(090)θ≤≤

Synchros work widely as error detectors in position control systems. The principle of operation of synchros is similar to that of resolvers. However it will not be discussed in the present lesson.

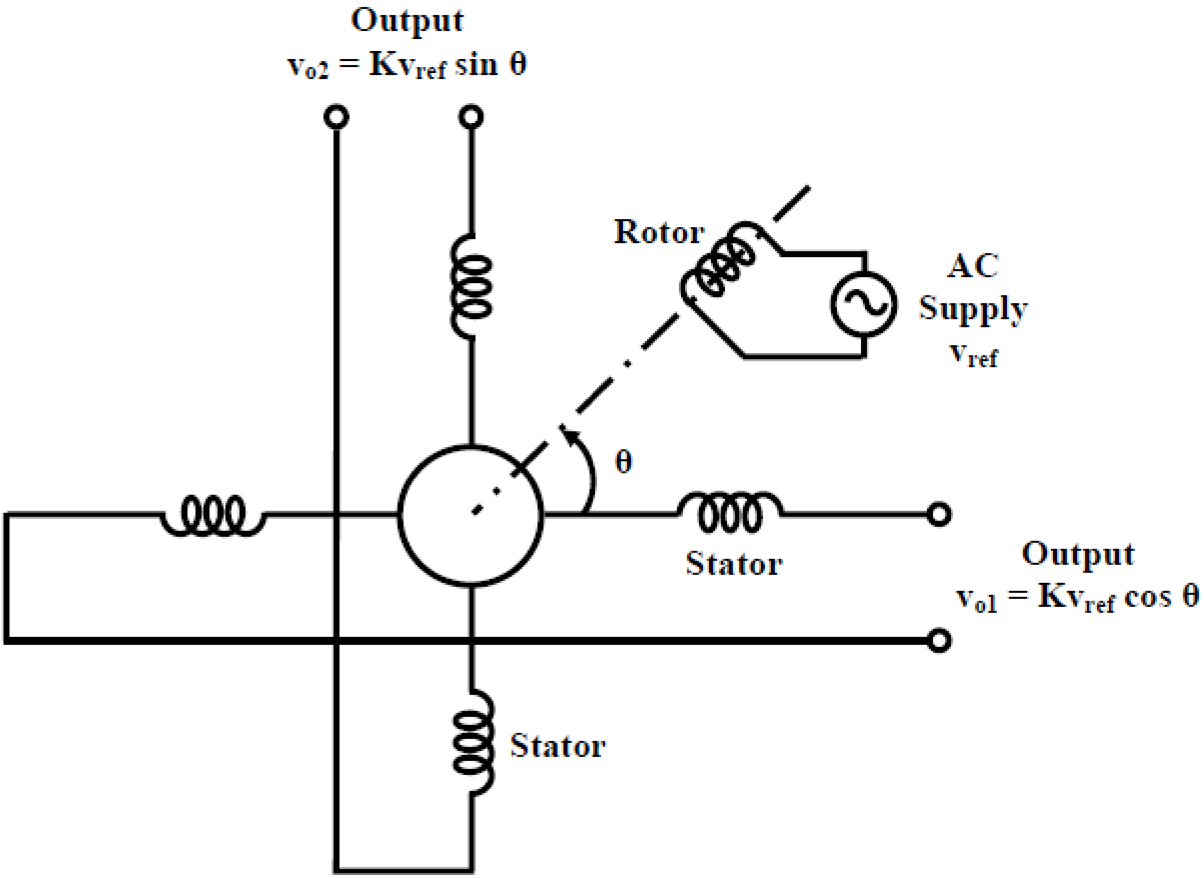

Output
$v_{o2} = Kv_{ref} \sin\theta$
Rotor
AC
Supply
v_{ref}
θ
Stator
Output
$v_{o1} = Kv_{ref} \cos\theta$
Stator

Fig. 5 Schematic diagram of the resolver.

Capacitance Sensors

The capacitance type sensor is a versatile one; it is available in different size and shape. It can also measure very small displacement in micrometer range. Often the whole sensor is fabricated in a silicon base and is integrated with the processing circuit to form a small chip. The basic principle of a capacitance sensor is well known. But to understand the various modes of operation, consider the capacitance formed by two parallel plates separated by a dielectric. The capacitance between the plates is given by:

$$C = \frac{\varepsilon_r \varepsilon_0 A}{d} \qquad (1)$$

where A=Area of the plates

d= separation between the plates

ε_r = relative permittivity of the dielectric

ε_0 = absolute permittivity in free space = $8.854 \times 10^{-12}\ F/m$.

capacitance sensor can be formed by either varying (i) the separation (d), or, (ii) the area (A), or (iii) the permittivity ($_r\varepsilon$). A displacement type sensor is normally based on the first two (variable distance and variable area) principles, while the variable permittivity principle is used for measurement of humidity, level, etc. Fig.6 Shows the basic constructions of variable gap and variable area types of capacitance sensors mentioned above. Fig. 6(a) shows a variable distance type sensor, where the gap between the fixed and moving plates changes. On the other hand, the area of overlap between the fixed plate and moving plate changes in Fig. 6(b), maintaining the gap constant. The variable area type sensor gives rise to linear variations of capacitance with the input variable, while a variable separation type sensor follows inverse relationship.

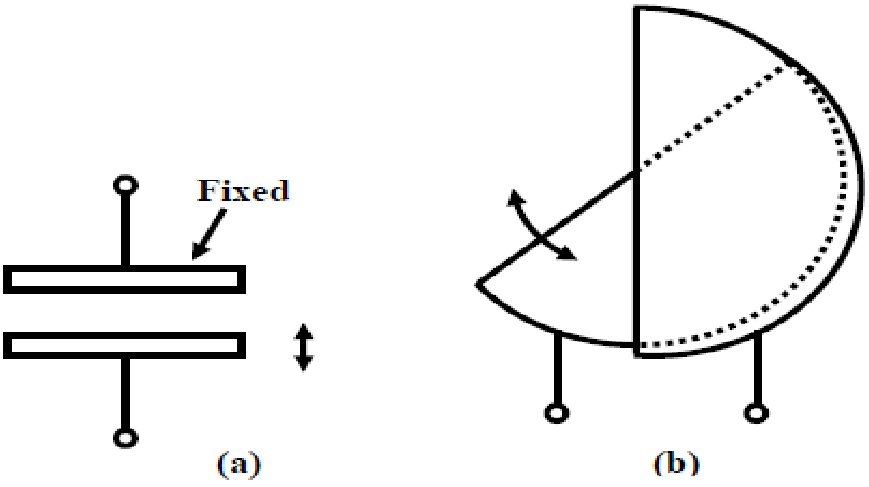


Fig.6 Capacitive type displacement Sensors:
a) variable separation type,
b) variable area type.

Capacitance sensors are also used for proximity detection. Such a typical scheme is shown in Fig. 4.

Capacitive proximity detectors are small in size, noncontact type and can detect presence of metallic or insulating objects in the range of approximately 0-5cm. For detection of insulating objects, the dielectric constant of the insulating object should be much larger than unity. Fig. 7 shows the construction of a proximity detector. Its measuring head consists of two electrodes, one circular (B) and the other an annular shaped one (A); separated by a small dielectrical spacing. When the target comes in the closed vicinity of the sensor head, the capacitance between the plates A and B would change, which can be measured by comparing with a fixed reference capacitor.

The measuring circuits for capacitance sensors are normally capacitive bridge type. But it should be noted that, the variation of capacitance in a capacitance type sensor is generally very small (few *pF* only, it can be even less than a *pF* in certain cases). These small changes in capacitor, in presence of large stray capacitance existing in different parts of the circuit are difficult. So the output voltage would generally be noisy, unless the sensor is designed and shielded carefully, the measuring circuit should also be capable of reducing the effects of stray fields.

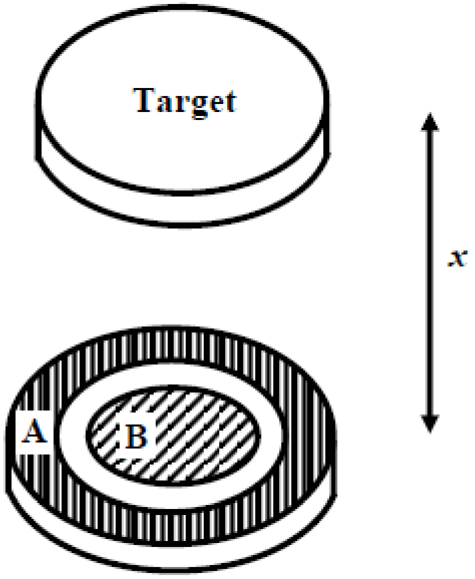


Fig. 7 Capacitance Proximity Detector.

Optical Sensors

Optical displacement sensors work on the basic principle that the intensity of light decreases with distance. So if the source and detector are fixed, the amount of light reflected from a moving surface will depend on the distance of the moving surface from the fixed ones. Measurement using this principle requires proper calibration since the amount of light received depends upon the reflectivity of the surface, intensity of the source etc. Yet it can provide a simple method for displacement measurement. Optical fibers are often used to transmit light to and from the measuring zone. Such a scheme with bundle fibers is shown in Fig. 8. It uses two bundle fibers, one for transmitting light from the source and the other to the detector. Light reflected on the receiving fiber bundle by the surface of the target object is carried to a photo detector. The light source could be Laser or LED; photodiodes or phototransistors are used for detection.

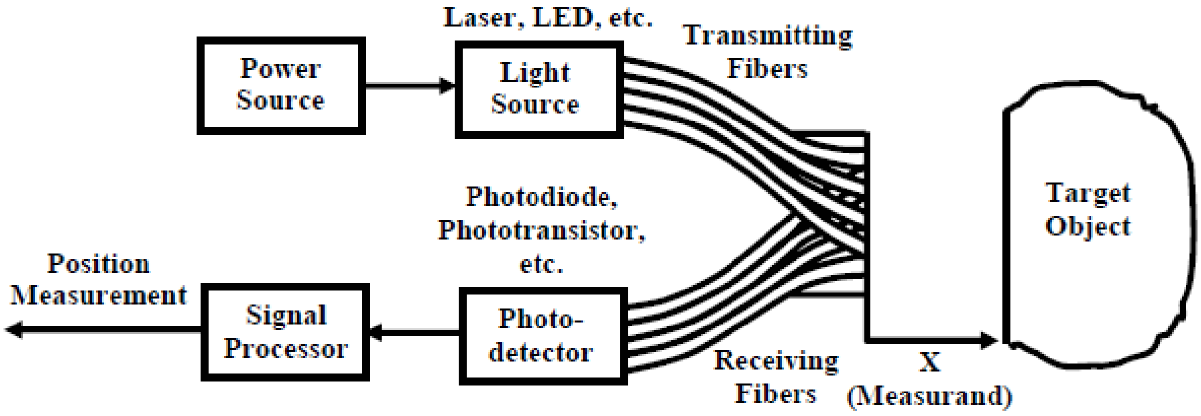


Fig. 8 A Fiber optic position sensor.

Speed Measurement

The simplest way for speed measurement of a rotating body is to mount a tachogenerator on the shaft and measure the voltage generated by it that is proportional to the speed. However this is a contact type measurement. There are other methods also for noncontact type measurements.

The first method is an optical method shown in Fig. 9. An opaque disc with perforations or transparent windows at regular interval is mounted on the shaft whose speed is to be measured. A LED source is aligned on one side of the disc in such a way that its light can pass through the transparent windows of the disc. As the disc rotates the light will alternately passed through the transparent windows and blocked by the opaque sections. A photodetector fixed on the other side of the disc detects the variation of light and the output of the detector after signal conditioning would be a square wave (as shown) whose frequency is decided by the speed and the number of holes (transparent windows) on the disc.

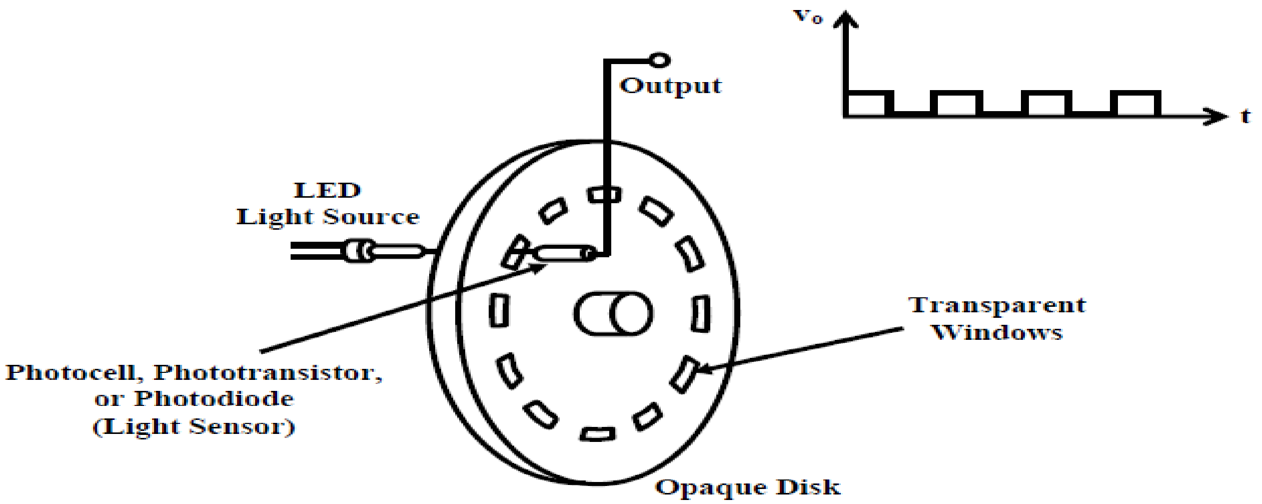


Fig. 9 Schematic arrangement of optical speed sensing arrangement.

Fig.10 shows another scheme for speed measurement. It is a variable reluctance type speed sensor. A wheel with projected teethes made of a ferromagnetic material is mounted on the shaft whose speed is to be measured. The static sensor consists of a permanent magnet and a search coil mounted on the same assembly and fixed at a closed distance from the wheel. The flux through the permanent magnet completes the path through the teeth of the wheel and cut the search coil. As the wheel rotates there would be change in flux cut and a voltage will be induced in the search coil. The variation of the flux can be expressed as:

$$\phi(t) = \phi_o + \phi_m \sin \omega t \qquad (2)$$

where ω is the angular speed of the wheel. Then the voltage induced in the coil is:

$$e = -N\frac{d\phi}{dt} = -N\omega\,\phi_m \cos \omega t \qquad (3)$$

where N is the number of turns in the search coil. So both the amplitude and frequency of the induced voltage is dependent on the speed of rotation. This voltage is fed to a comparator circuit that gives a square wave type voltage whose amplitude is constant, but frequency is proportional to the speed. A frequency counter is used to count the number of square pulses during a fixed interval and displays the speed.

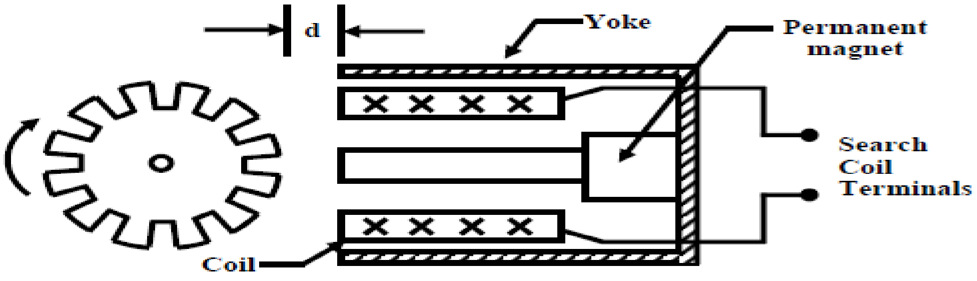


Fig. 10 Variable reluctance type speed sensor.

Signal Conditioning Circuits

The success of the design of any measurement system depends heavily on the design and performance of the signal conditioning circuits. Even a costly and accurate transducer may fail to deliver good performance if the signal conditioning circuit is not designed properly. The schematic arrangement and the selection of the passive and active elements in the circuit heavily influence the overall performance of the system. Often these are decided by the electrical output characteristics of the sensing element. Nowadays, many commercial sensors often have in-built signal conditioning circuit. This arrangement can overcome the problem of incompatibility between the sensing element and the signal conditioning circuit.

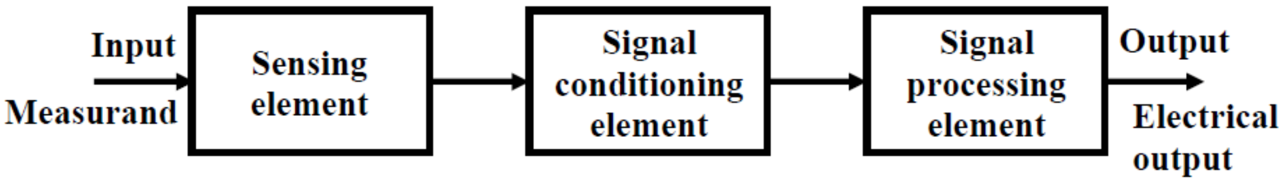


Fig. 1 Elements of a measuring system.

If one looks at the different cross section of sensing elements and their signal conditioning circuits, it can be observed that the majority of them use standard blocks like bridges (A.C. and D.C.), amplifiers, filters and phase sensitive detectors for signal conditioning. In this lesson, we would concentrate mostly on bridges and amplifiers and ponder about issues on the design issues.

Unbalanced D.C. Bridge

We are more familiar with balanced wheatstone bridge, compared to the unbalanced one; but the later one finds wider applications in the area of Instrumentation. To illustrate the properties of unbalanced d.c. bridge, let us consider the circuit shown in fig.2 .Here the variable resistance can be considered to be a sensor, whose resistance varies with the process parameter. The output voltage is , e_O which varies with the change of the resistance $R\Delta$/R=x. The arm ratio of the bridge is p and E is the excitation voltage.

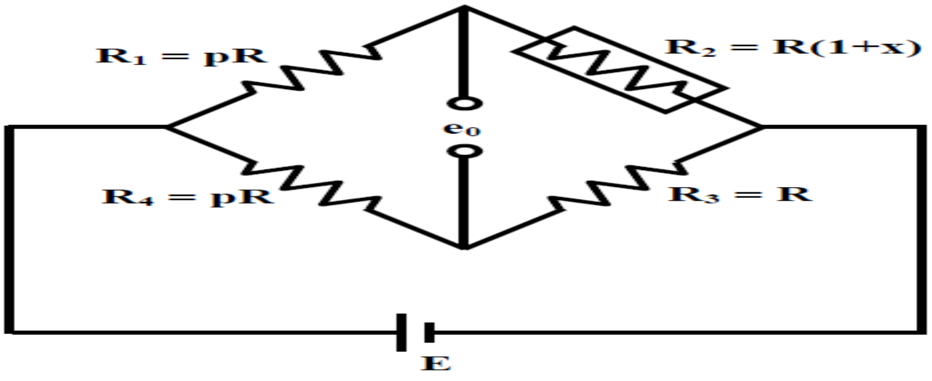


Fig. 2 Unbalanced D.C. bridge.

Then,

$$e_0 = \left[\frac{R(1+x)}{pR+R(1+x)} - \frac{R}{pR+R}\right]E$$

$$= \frac{px}{(p+1+x)(p+1)}E \qquad (1)$$

Push-pull Configuration

The characteristics of an unbalanced wheatstone bridge with single resistive element as one of the arms can greatly be improved with a *push-pull* arrangement of the bridge, comprising of two identical resistive elements in two adjacent arms: while the resistance of one sensor decreasing, the resistance of the other sensor is increasing by the same amount, as shown in fig.4. The unbalanced voltage can be obtained as:

$$e_0 = \left[\frac{R(1+x)}{R(1+x)+R(1-x)} - \frac{R}{2R}\right]E$$

$$= \left[\frac{1+x}{2} - \frac{1}{2}\right]E$$

$$= \frac{x}{2}E \qquad (4)$$

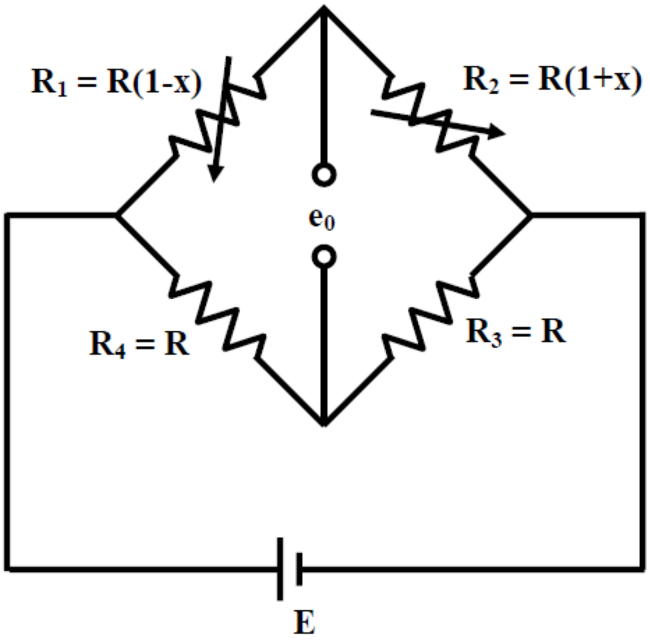


Fig. 4 Unbalanced D.C. bridge with push pull configuration of resistance sensors.

Looking at the above expression, one can immediately appreciate the advantage of using push-pull configuration. First of all, the nonlinearity in the bridge output can be eliminated completely. Secondly, the sensitivity is doubled compared to a single sensor element bridge.

The same concept can also be applied to A.C. bridges with inductive or capacitive sensors. These applications are elaborated below. **R$_2$ = R(1+x)R$_1$ = R(1-x)R$_4$ = R R$_3$ = R Ee$_0$Fig. 4 Unbalanced D.C. bridge with push pull configuration of resistance sensors.**

Unbalanced A.C. Bridge with Push-pull Configuration

Figures 5(a) and (b) shows the schematic arrangements of unbalanced A.C. bridge with inductive and capacitive sensors respectively with push-pull configuration. Here, the D.C. excitation is replaced by an A.C. source and two fixed resistances of same value are kept in the two adjacent arms and the inductive (or the capacitive) sensors are so designed that if the inductance (capacitance) increases by a particular amount, that of the other one would decrease by the same amount.

For fig. 5(a),

$$e_0 = \left[\frac{jwL(1+x)}{jwL(1+x) + jwL(1-x)} - \frac{R}{2R} \right] E ,$$

where w is the angular frequency of excitation, L is the nominal value of the inductance and $x = {}^{\Delta L}\!/_{L}$. Simplifying, we obtain,

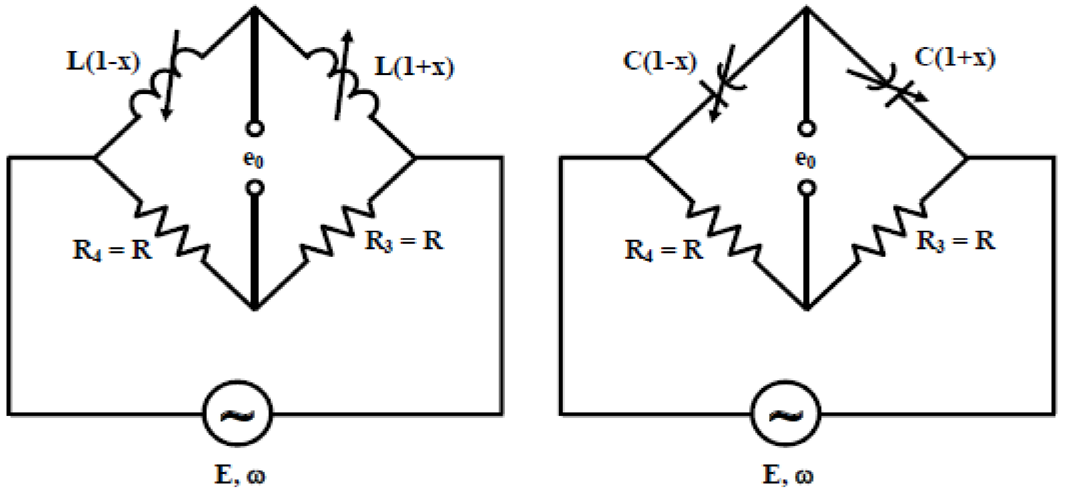


Fig. 5 Unbalanced A.C. bridge with push-pull configuration: (a) for inductive sensor, and (b) for capacitive sensor.

$$e_0 = \frac{x}{2} E , \qquad (5)$$

which again shows the linear characteristics of the bridge.

For the capacitance sensor with the arrangement shown in fig. 5(b), we have:

$$e_0 = \left[\frac{{}^{1}\!/_{jwC(1+x)}}{{}^{1}\!/_{jwC(1+x)} + {}^{1}\!/_{jwC(1-x)}} - \frac{R}{2R}\right]E$$

$$= \left[\frac{jwC(1-x)}{jwC(1+x)+jwC(1-x)} - \frac{R}{2R}\right]E$$

$$= -\frac{x}{2}E \qquad (6)$$

where $x = {}^{\Delta C}\!/_{C}$. As expected, we would also obtain here a complete linear characteristic, irrespective of whatever is the value of *x*. But here is a small difference between the performance of an inductive sensor bridge and that of a capacitance sensor bridge (equation (5) and (6)): a negative sign. This negative sign in an A.C. bridge indicates that the output voltage in fig. 4(b) will be 180^0 out of phase with the input voltage *E*. But this cannot be detected, if we use a simple A.C. voltmeter to measure the output voltage. In fact, if the value of *x* were negative, there would also be a phase reversal in the output voltage, which cannot be detected, unless a special measuring device for sensing the phase is used. This type of circuit is called a *Phase Sensitive Device* (PSD) and is often used in conjunction with inductive and capacitive sensors. The circuit of a PSD rectifies the small A.C. voltage into a D.C. one; the polarity of the D.C. output voltage is reversed, if there is a phase reversal

Capacitance Amplifier

Here we would present another type of circuit configuration, suitable for push-pull type capacitance sensor. The circuit can also be termed as a half bridge and a typical configuration has been shown in fig.6. Here two identical voltage sources are connected in series, with their common point grounded. This can be also achieved by using a center-tapped transformer. Two sensing capacitors C_1 and C_2 are connected as shown in the fig. 5 and the unbalanced current flows through an amplifier circuit with a feedback capacitor C_f. Now the current through the capacitors are:

$$I_1 = V.jwC_1 \quad and\ I_2 = -V.jwC_2$$

Hence the unbalanced current:

$$I = I_1 + I_2 = V.jw(C_1 - C_2)$$

And the voltage output of the amplifier:

$$V_0 = -\frac{I}{jwC_f} = -\frac{C_1 - C_2}{C_f}V \qquad (7)$$

As expected, a linear response can also be obtained by connecting a push-pull configuration of capacitance in fig.6. The gain can be adjusted by varying C_f. However, this is an ideal circuit, for a practical circuit, a high resistance has to be placed in parallel with C_f.

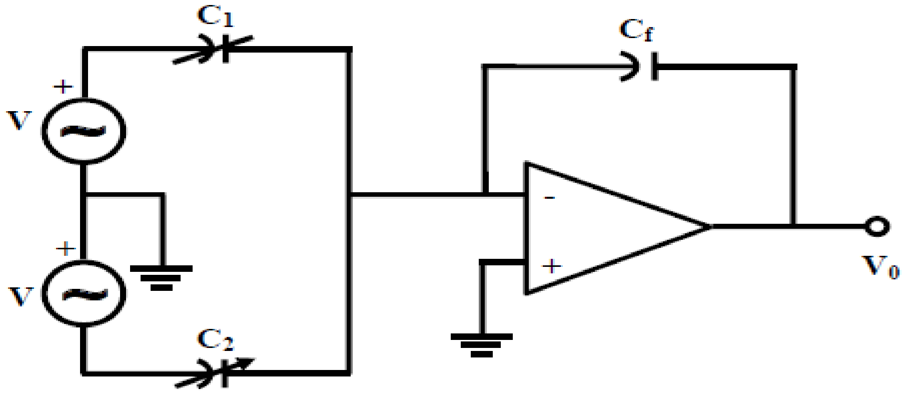


Fig. 6 A capacitance amplifier.

Amplifiers

An Amplifier is an integral part of any signal conditioning circuit. However, there are different configurations of amplifiers, and depending of the type of the requirement, one should select the proper configuration.

Inverting and Non-inverting Amplifiers

These two types are single ended amplifiers, with one terminal of the input is grounded. From the schematics of these two popular amplifiers, shown in fig.7, the voltage gain for the inverting amplifier is:

$$\frac{e_0}{e_i} = -\frac{R_2}{R_1}$$

while the voltage gain for the noninverting amplifier is:

$$\frac{e_0}{e_i} = 1 + \frac{R_2}{R_1}$$

Apparently, both the two amplifiers are capable of delivering any desired voltage gain, provided the phase inversion in the first case is not a problem. But looking carefully into the circuits, one can easily understand, that, the input impedance of the inverting amplifier is finite and is approximately R_1, while a noninverting amplifier has an infinite input impedance. Definitely, the second amplifier will perform better, if we want that, the amplifier should not load the sensor (or a bridge circuit).

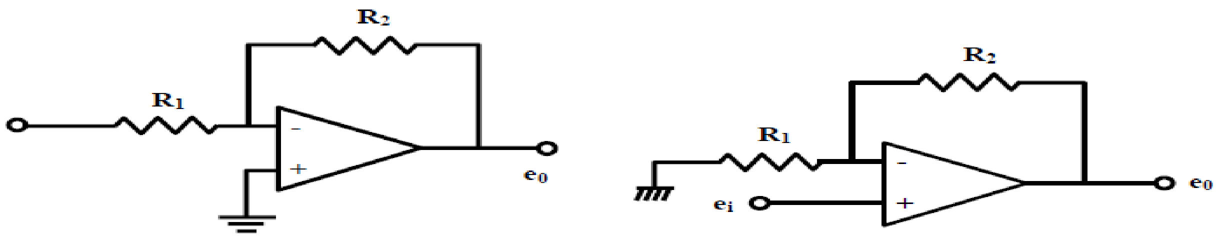


Fig. 7 (a) Inverting amplifier, (b) noninverting amplifier.

Differential Amplifier

Differential amplifiers are useful for the cases, where both the input terminals are floating. These amplifiers find wide applications in instrumentation. A typical differential amplifier with single op.amp. configuration is shown in fig.8. Here, by applying superposition theorem, one can easily obtain the contribution of each input and add them algebraically to obtain the output voltage as:

$$e_0 = \frac{R_4}{R_3 + R_4}(1 + \frac{R_2}{R_1}) e_2 - \frac{R_2}{R_1} e_1 \tag{8}$$

If we select

$$\frac{R_4}{R_3} = \frac{R_2}{R_1}, \tag{9}$$

then, the output voltage becomes:

$$e_0 = \frac{R_2}{R_1}(e_2 - e_1) \tag{10}$$

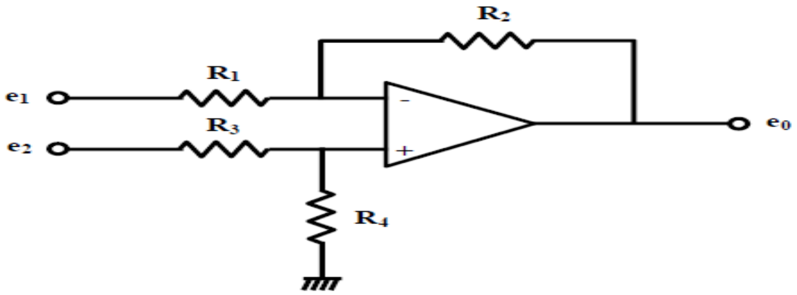


Fig. 8 Differential amplifier.

However, this type of differential amplifier with single op. amp. configuration also suffers from the limitation of finite input impedance. In fact, several criteria are used for judging the performance of an amplifier. These are mainly: (i) offset and drift, (ii) input impedance, (iii) gain and bandwidth, and (iv) common mode rejection ratio (CMRR).

The performance of an operational amplifier is judged by the gain- bandwidth product, which is fixed by the manufacturer’s specification. In the open loop, the gain is very high (around 10^5) but the bandwidth is very low. In the closed loop operation, the gain is low, but the achievable bandwidth is high. Normally, the gain of a single stage operational amplifier circuit is kept limited around 10, thus large bandwidth is achievable. For larger gains, several stages of amplifiers are connected in cascade.

CMRR is a very important parameter for instrumentation circuit applications and it is desirable to use amplifiers of high CMRR when connected to instrumentation circuits.

The CMRR is defined as

$$CMRR = 20\log_{10}\frac{A_d}{A_c} \qquad (11)$$

where, A_d is the differential mode gain A_c and is the common mode gain of the amplifier.

Instrumentation Amplifier

Often we need to amplify a small differential voltage few hundred times in instrumentation applications. A single stage differential amplifier, shown in fig.8 is not capable of performing this job efficiently, because of several reasons. First of all, the input impedance is finite; moreover, the achievable gain in this single stage amplifier is also limited due to gain bandwidth product limitation as well as limitations due to offset current of the op. amp. Naturally, we need to seek for an improved version of this amplifier.

A three op. amp. Instrumentation amplifier, shown in fig.10 is an ideal choice for achieving the objective. The major properties are (i) high differential gain (adjustable up to 1000) (ii) infinite input impedance, (iii) large CMRR (80 dB or more), and (iv) moderate bandwidth.

From fig. 10, it is apparent that, no current will be drawn by the input stage of the op. amps. (since inputs are fed to the non inverting input terminals). Thus the second property mentioned above is achieved. Looking at the input stage, the same current *I* will flow through the resistances R_1 and R_2. Using the properties of ideal op. amp., we can have:

$$I = \frac{e_1 - e_{i1}}{R_1} = \frac{e_{i1} - e_{i2}}{R_2} = \frac{e_{i2} - e_2}{R_1} \qquad (12)$$

from which, we obtain,

$$e_1 = e_{i1} + \frac{R_1}{R_2}(e_{i1} - e_{i2})$$

$$e_2 = e_{i2} - \frac{R_1}{R_2}(e_{i1} - e_{i2})$$

Therefore,

$$e_1 - e_2 = (1 + \frac{2R_1}{R_2})(e_{i1} - e_{i2})$$

The second stage of the instrumentation amplifier is a simple differential amplifier, and hence, using (10), the over all gain:

$$e_0 = \frac{R_4}{R_3}(e_2 - e_1) = \frac{R_4}{R_3}(1 + \frac{2R_1}{R_2})(e_{i2} - e_{i1}) \qquad (13)$$

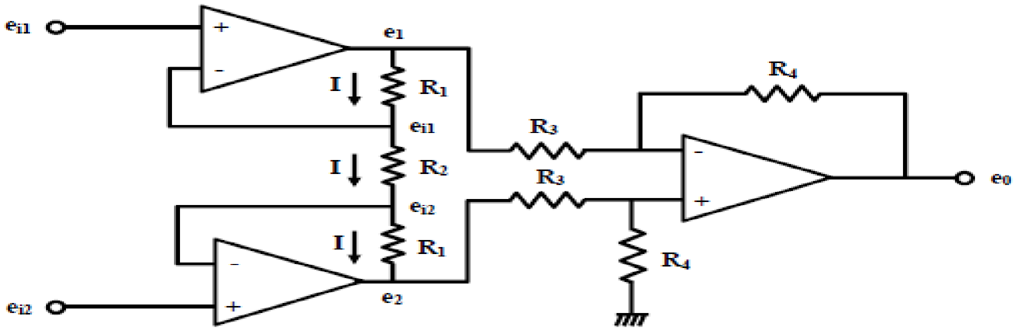


Fig. 10 Three op. amp. Instrumentation Amplifier.

Thus by varying R_2 very large gain can be achieved, but the relationship is inverse. Since three op. amps. are responsible for achieving this gain, the bandwidth does not suffer.

There are many commercially available single chip instrumentation amplifiers in the market. Their gains can be adjusted by connecting an external resistance, or by selecting the gains (50, 100 or 500) through jumper connections.

Errors and Calibration

Error Analysis

The term *error* in a measurement is defined as:

$$Error = Instrument\ reading - true\ reading. \qquad (1)$$

Error is often expressed in percentage as:

$$\%Error = \frac{Instrument\ reading - true\ reading}{true\ reading} X100 \qquad (2)$$

The errors in instrument readings may be classified in to three categories as:

1. *Gross errors*
2. *Systematic errors*
3. *Random Errors.*

Gross errors arise due to human mistakes, such as, reading of the instrument value before it reaches steady state, mistake of recording the measured data in calculating a derived measured, etc. Parallax error in reading on an analog scale is also is also a source of gross error. Careful reading and recording of the data can reduce the gross errors to a great extent.

Systematic errors are those that affect all the readings in a particular fashion. Zero error, and bias of an instrument are examples of systematic errors. On the other hand, there are few errors, the cause of which is not clearly known, and they affect the readings in a random way. This type of errors is known as *Random error*. There is an important difference between the systematic errors and random errors. In most of the case, the systematic errors can be corrected by calibration, whereas the random errors can never be corrected, the can only be reduced by averaging, or error limits can be estimated.

Systematic Errors

Systematic errors may arise due to different reasons. It may be due to the shortcomings of the instrument or the sensor. An instrument may have a zero error, or its output may be varying in a nonlinear fashion with the input, thus deviating from the ideal linear input/output relationship. The amplifier inside the instrument may have input offset voltage and current which will contribute to zero error. Different nonlinearities in the amplifier circuit will also cause error due to nonlinearity. Besides, the systematic error can also be due to improper design of the measuring scheme. It may arise due to the loading effect, improper selection of the sensor or the filter cut off frequency. Systematic errors can be due to environmental effect also. The sensor characteristics may change with temperature or other environmental conditions.

The major feature of systematic errors is that the sources of errors are recognisable and can be reduced to a great extent by carefully designing the measuring system and selecting its components. By placing the instrument in a controlled environment may also help in reduction of systematic errors. They can be further reduced by proper and regular calibration of the instrument.

Random Errors

It has been already mentioned that the causes of random errors are not exactly known, so they cannot be eliminated. They can only be reduced and the error ranges can be estimated by using some statistical operations. If we measure the same input variable a number of times, keeping all other factors affecting the measurement same, the same measured value would not be repeated, the consecutive reading would rather differ in a random way. But fortunately, the deviations of the readings normally follow a particular distribution (mostly normal distribution) and we may be able to reduce the error by taking a number of readings and averaging them out.

Few terms are often used to chararacterize the distribution of the measurement, namely,

$$\textit{Mean Value}\ \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (3)$$

where n is the total number of readings and x_i is the value of the individual readings. It can be shown that the mean value is the most probable value of a set of readings, and that is why it has a very important role in statistical error analysis. The *deviation* of the individual readings from the mean value can be obtained as :

$$\text{Deviation} \quad d_i = x_i - \bar{x} \qquad (4)$$

We now want to have an idea about the deviation, i.e., whether the individual readings are far away from the mean value or not. Unfortunately, the *mean of deviation* will not serve the purpose, since,

$$\text{Mean of deviation} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x}) = \bar{x} - \frac{1}{n}(n\bar{x}) = 0$$

So instead, *variance* or the *mean square deviation* is used as a measure of the deviation of the set of readings. It is defined as:

$$\text{Variance} \quad V = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sigma^2 \qquad (5)$$

The term σ is denoted as *standard deviation.* It is to be noted that in the above expression, the averaging is done over *n-1* readings, instead of *n* readings. The above definition can be justified, if one considers the fact that if it is averaged over *n*, the variance would become zero when *n*=1 and this may lead to some misinterpretation of the observed readings. On the other hand the above definition is more consistent, since the variance is undefined if the number of reading is *one*. However, for a large number of readings (*n>30*), one can safely approximate the variance as,

$$\text{Variance}\, V = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \sigma^2 \qquad (6)$$

The term *standard deviation* is often used as a measure of uncertainty in a set of measurements. Standard deviation is also used as a measure of quality of an instrument. It has been discussed in Lesson-3 that *precision*, a measure of reproducibility is expressed in terms of standard deviation.

Propagation of Error

Quite often, a variable is estimated from the measurement of two parameters. A typical example may be the estimation of power of a d.c circuit from the measurement of voltage and current in the circuit. The question is that how to estimate the uncertainty in the estimated variable, if the uncertainties in the measured parameters are known. The problem can be stated mathematically as,

$$\text{Let} \quad y = f(x_1, x_2, \ldots\ldots, x_n) \qquad (7)$$

If the uncertainty (or deviation) in x_i is known and is equal to Δx_i, $(i = 1, 2, ..n)$, what is the overall uncertainty in the term y?

Differentiating the above expression, and applying Taylor series expansion, we obtain,

$$\Delta y = \frac{\partial f}{\partial x_1}\Delta x_1 + \frac{\partial f}{\partial x_2}\Delta x_2 + \ldots\ldots + \frac{\partial f}{\partial x_n}\Delta x_n \qquad (8)$$

Since Δx_i can be either +ve or –ve in sign, the maximum possible error is when all the errors are positive and occurring simultaneously. The term *absolute error* is defined as,

$$\textit{Absolute error}: |\Delta y| = \frac{\partial f}{\partial x_1}|\Delta x_1| + \frac{\partial f}{\partial x_2}|\Delta x_2| + \ldots\ldots + \frac{\partial f}{\partial x_n}|\Delta x_n| \qquad (9)$$

But this is a very unlikely phenomenon. In practice, $x_1, x_2, \ldots\ldots, x_n$ are independent and all errors do not occur simultaneously. As a result, the above error estimation is very conservative. To alleviate this problem, the cumulative error in y is defined in terms of the standard deviation. Squaring equation (8), we obtain,

$$(\Delta y)^2 = \left(\frac{\partial f}{\partial x_1}\right)^2 (\Delta x_1)^2 + \left(\frac{\partial f}{\partial x_2}\right)^2 (\Delta x_2)^2 + \ldots\ldots + 2\frac{\partial f}{\partial x_1}\frac{\partial f}{\partial x_2}.(\Delta x_1 \Delta x_2) + \ldots\ldots. \qquad (10)$$

If the variations of $x_1, x_2, \ldots..$ are independent, positive value of one increment is equally likely to be associated with the negative value of another increment, so that the some of all the cross product terms can be taken as zero, in repeated observations. We have already defined *variance* V as the mean squared error. So, the mean of $(\Delta y)^2$ for a set of repeated observations, becomes the variance of y, or

$$V(y) = \left(\frac{\partial f}{\partial x_1}\right)^2 V(x_1) + \left(\frac{\partial f}{\partial x_2}\right)^2 V(x_2) + \ldots\ldots \qquad (11)$$

So the standard deviation of the variable y can be expressed as:

$$\sigma(y) = \left[\left(\frac{\partial f}{\partial x_1}\right)^2 \sigma^2(x_1) + \left(\frac{\partial f}{\partial x_2}\right)^2 \sigma^2(x_2) + \ldots\ldots\right]^{1/2} \qquad (12)$$

Limiting Error

Limiting error is an important parameter used for specifying the accuracy of an instrument. The limiting error (or guarantee error) is specified by the manufacturer to define the maximum limit of the error that may occur in the instrument. Suppose the accuracy of a 0-100V voltmeter is specified as 2% of the full scale range. This implies that the error is guaranteed to be within 2V for any reading. If the voltmeter reads 50V, then also the error is also within 2V. As a result, the accuracy for this reading will be(2/5)100 = 4%. If the overall performance of a measuring system is dependent on the accuracy of several independent parameters, then the limiting or guarantee error is decided by the absolute error as given in the expression in (9). For example, if we are measuring the value of an unknown resistance element using a wheatstone bridge whose known resistors have specified accuracies of 1%, 2% and 3% respectively, then,

Since $R_x = \frac{R_1 R_2}{R_3}$, we have,

$$\Delta R_x = \frac{R_2}{R_3}\Delta R_1 + \frac{R_1}{R_3}\Delta R_2 - \frac{R_1 R_2}{R_3^2}\Delta R_3$$

or,
$$\frac{\Delta R_x}{R_x} = \frac{\Delta R_1}{R_1} + \frac{\Delta R_2}{R_2} - \frac{\Delta R_3}{R_3}$$

Then following the logic given to establish (9), the absolute error is computed by taking the positive values only and the errors will add up; as a result the limiting error for the unknown resistor will be 6%.

Importance of the Arithmetic Mean

It has been a common practice to take a number of measurements and take the arithmetic mean to estimate the average value. But the question may be raised: *why mean?* The answer is: *The most probable value of a set of dispersed data is the arithmetic mean.* The statement can be substantiated from the following proof.

Let $x_1, x_2, x_3, ..., x_n$ be a set of n observed data. Let X be the central value (not yet specified).

So the deviations from the central value are $(x_1 - X), (x_2 - X), ...(x_n - X)$.

The sum of the square of the deviations is:

$$S_{sq} = (x_1 - X)^2 + (x_2 - X)^2 + ... + (x_n - X)^2$$

$$= x_1^2 + x_2^2 + ... + x_n^2 - 2X(x_1 + x_2 + ... + x_n) + nX^2$$

So the problem is to find X so that S_{sq} is minimum. So,

$$\frac{dS_{sq}}{dX} = -2(x_1 + x_2 + ... + x_n) + 2nX = 0$$

or,

$$X = \frac{1}{n}(x_1 + x_2 + ... + x_n) = \bar{x}$$

So the arithmetic mean is the central value in the least square sense. If we take another set of readings, we shall reach at a different mean value. But if we take a large number of readings, definitely we shall come very close to the actual value (or universal mean). So the question is, how to determine the deviations of the different set of mean values obtained from the actual value?

Standard deviation of the mean

Here we shall try to find out the standard deviation of the mean value obtained from the universal mean or actual value.

Consider a set of n number of readings, $x_1, x_2, x_{3,}....,x_n$. The mean value of this set expressed as:

$$\bar{x} = \frac{1}{n}(x_1 + x_2 + ... + x_n) = f(x_1 + x_2 + ... + x_n)$$

Using (11) for the above expression, we can write:

$$V(\bar{x}) = \left(\frac{\partial f}{\partial x_1}\right)^2 V(x_1) + \left(\frac{\partial f}{\partial x_2}\right)^2 V(x_2) + ... + \left(\frac{\partial f}{\partial x_n}\right)^2 V(x_n)$$

$$= \frac{1}{n^2}[V(x_1) + V(x_2)..... + V(x_n)]$$

Now the standard deviation for the readings $x_1, x_2, ..., x_n$ is defined as:

$$\sigma = \left[\frac{1}{n}[V(x_1) + V(x_2)..... + V(x_n)]\right]^{1/2}$$, where n is large.

Therefore,

$$V(\bar{x}) = \frac{1}{n^2}(n.\sigma^2) = \frac{\sigma^2}{n}$$

Hence, the standard deviation of the mean,

$$\sigma(\bar{x}) = \frac{\sigma}{\sqrt{n}} \qquad (13)$$

which indicates that the precision can be increased, (i.e. $\sigma(\bar{x})$ reduced) by taking more number of observations. But the improvement is slow due to the $\sqrt{n}$ factor.

Least square Curve Fitting

Often while performing experiments, we obtain a set of data relating the input and output variables (e.g. resistance vs. temperature characteristics of a resistive element) and we want to fit a smooth curve joining different experimental points. Mathematically, we want to fit a polynomial over the experimental data, such that the sum of the square of the deviations between the experimental points and the corresponding points of the polynomial is minimum. The technique is known as least square curve fitting. We shall explain the method for a straight line curve fitting. A typical case of least square straight line fitting for a set of dispersed data is shown in Fig. 1. We want to obtain the best fit straight line out of the dispersed data shown.

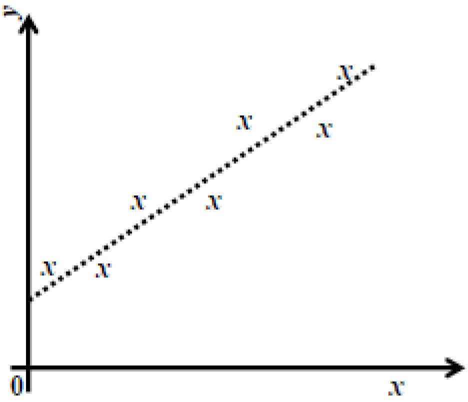
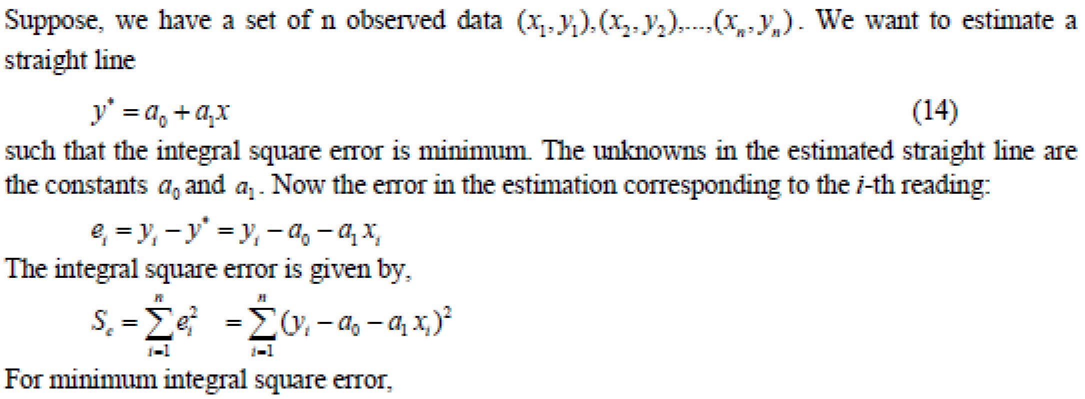


Fig. 1 Least square straight line fitting.



Suppose, we have a set of n observed data $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$. We want to estimate a straight line

$$y^* = a_0 + a_1 x \qquad (14)$$

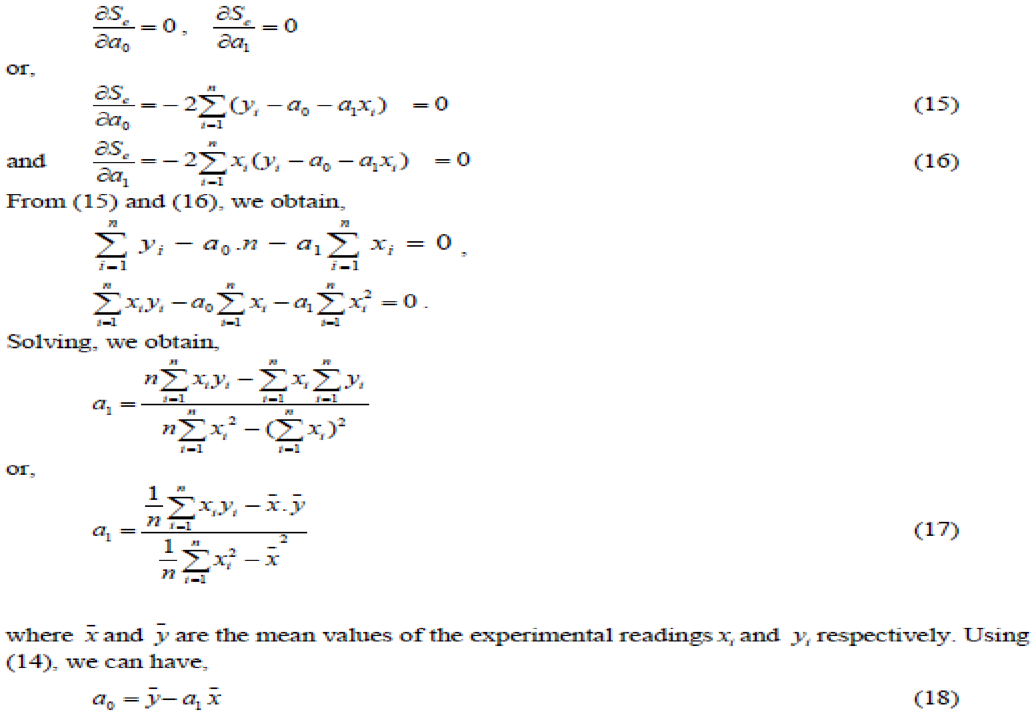
such that the integral square error is minimum. The unknowns in the estimated straight line are the constants a_0 and a_1. Now the error in the estimation corresponding to the *i*-th reading:

$$e_i = y_i - y^* = y_i - a_0 - a_1 x_i$$

The integral square error is given by,

$$S_e = \sum_{i=1}^{n} e_i^2 \quad = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2$$

For minimum integral square error,



$$\frac{\partial S_e}{\partial a_0} = 0\,, \quad \frac{\partial S_e}{\partial a_1} = 0$$

or,

$$\frac{\partial S_e}{\partial a_0} = -2\sum_{i=1}^{n}(y_i - a_0 - a_1 x_i) \quad = 0 \qquad (15)$$

and $$\frac{\partial S_e}{\partial a_1} = -2\sum_{i=1}^{n} x_i (y_i - a_0 - a_1 x_i) \quad = 0 \qquad (16)$$

From (15) and (16), we obtain,

$$\sum_{i=1}^{n} y_i - a_0 . n - a_1 \sum_{i=1}^{n} x_i = 0\,,$$

$$\sum_{i=1}^{n} x_i y_i - a_0 \sum_{i=1}^{n} x_i - a_1 \sum_{i=1}^{n} x_i^2 = 0\,.$$

Solving, we obtain,

$$a_1 = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}$$

or,

$$a_1 = \frac{\frac{1}{n}\sum_{i=1}^{n} x_i y_i - \bar{x}.\bar{y}}{\frac{1}{n}\sum_{i=1}^{n} x_i^2 - \bar{x}^2} \qquad (17)$$

where $\bar{x}$ and $\bar{y}$ are the mean values of the experimental readings x_i and y_i respectively. Using (14), we can have,

$$a_0 = \bar{y} - a_1 \bar{x} \qquad (18)$$

Calibration and error reduction

It has already been mentioned that the random errors cannot be eliminated. But by taking a number of readings under the same condition and taking the mean, we can considerably reduce the random errors. In fact, if the number of readings is very large, we can say that the mean value will approach the true value, and thus the error can be made almost zero. For finite number of readings, by using the statistical method of analysis, we can also estimate the range of the measurement error.

On the other hand, the systematic errors are well defined, the source of error can be identified easily and once identified, it is possible to eliminate the systematic error. But even for a simple instrument, the systematic errors arise due to a number of causes and it is a tedious process to identify and eliminate all the sources of errors. An attractive alternative is to calibrate the instrument for different known inputs *Calibration* is a process where a known input signal or a series of input signals are applied to the measuring system. By comparing the actual input value with the output indication of the system, the overall effect of the systematic errors can be observed. The errors at those calibrating points are then made zero by *trimming* few adjustable components, by using calibration charts or by using software corrections. Strictly speaking, calibration involves comparing the measured value with the *standard instruments* derived from comparison with the primary standards kept at Standard Laboratories. In an actual calibrating system for a pressure sensor (say), we not only require a standard pressure measuring device, but also a *test-bench*, where the desired pressure can be generated at different values. The calibration process of an acceleration measuring device is more difficult, since, the desired acceleration should be generated on a body, the measuring device has to be mounted on it and the actual value of the generated acceleration is measured in some indirect way. The calibration can be done for all the points, and then for actual measurement, the true value can be obtained from a *look-up table* prepared and stored before hand. This type of calibration, is often referred as *software calibration*. Alternatively, a more popular way is to calibrate the instrument at one, two or three points of measurement and trim the instrument through independent adjustments, so that, the error at those points would be zero. It is then expected that error for the whole range of measurement would remain within a small range. These types of calibration are known as single-point, two-point and three-point calibration. Typical input-output characteristics of a measuring device under these three calibrations are shown in fig.2. The single-point calibration is often referred as offset adjustment, where the output of the system is forced to be zero under zero input condition. For electronic instruments, often it is done automatically and is the process is known as *auto-zero* calibration. For most of the field instruments calibration is done at two points, one at zero input and the other at full scale input. Two independent adjustments, normally provided, are known as *zero* and *span* adjustments.

One important point needs to be mentioned at this juncture. The characteristics of an instrument change with time. So even it is calibrated once, the output may deviate from the calibrated points with time, temperature and other environmental conditions. So the calibration process has to be repeated at regular intervals if one wants that it should give accurate value of the measurand through out.

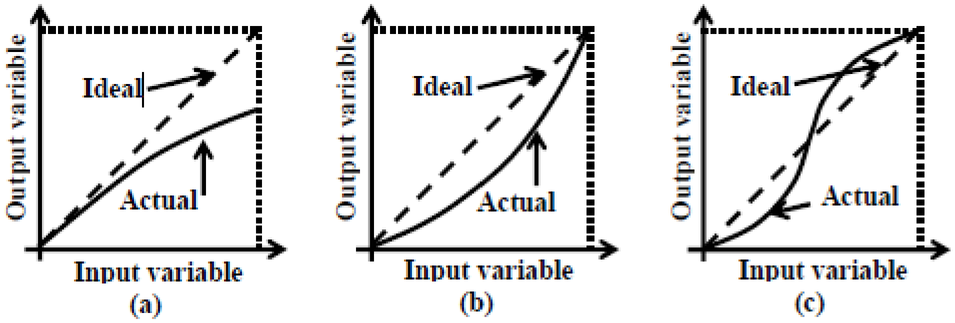


Fig. 2 (a) single point calibration, (b) two point calibration, (c) three point calibration

UNIT – II

PROCESS CONTROL

Introduction to Process Control

We often come across the term *process* indicating a set up or a plant that we want to control. Thus by a process we may mean a unit of chemical plant (say, a distillation column), or a manufacturing system (say, an assembly shop), or a food processing industry and so on. We may want to automate the process; we may also like to control certain parameters of the system output (say, level of a tank, pressure of steam etc.). Broadly speaking, there could be two types of control; we might want to carry out. The first one is called *sequential control*, where the control action is carried out in a sequence. A good example for this type of operation could be in an automated car manufacturing system, where the assembly of parts is carried out in a sequence (on a conveyor line). Here the control action is sequential in nature and works in a preprogrammed open loop fashion (implying that there is no feedback of the output signal to the controller). Programmable Logic Controller (PLC) is often used to carry out these operations.

But there are cases, where the control action needed is continuous in nature and precise control of the output variable is required. Take for example, the drum level control of a boiler. Here, the water level of the drum has to be maintained within a small band, in spite of variations of steam flow rate, steam pressure etc. This type of control is sometimes called *modulating control*, as the control variable is *modulated* to keep the process variable at a constant value. Feedback principle is used for these types of control.

Now onwards, we would concentrate on the control of these types of processes. But in order to design a controller effectively, we must have a thorough knowledge about the dynamics of the process. A mathematical model of the process dynamics often helps us to understand the process behaviour under different operational conditions

2. Characteristics of a Process

Different processes have different characteristics. But, broadly speaking, there are certain characteristics features those are more or less common to most of the processes. They are:

(i) *The mathematical model of the process is nonlinear in nature.*

(ii) *The process model contains the disturbance input*

(iii) *The process model contains the time delay term.*

In general a process may have several input variables and several output variables. But only one or two (at most few) of the input variables are used to control the process. These inputs, used for manipulating the process are called *manipulating variables*. The other inputs those are left uncontrolled are called *disturbances*.

Few outputs are measured and fed back for comparison with the desired set values. The controller operates based on the error values and gives the command for controlling the manipulating variables. The block diagram of such a closed loop process can be drawn as shown in Fig. 1.

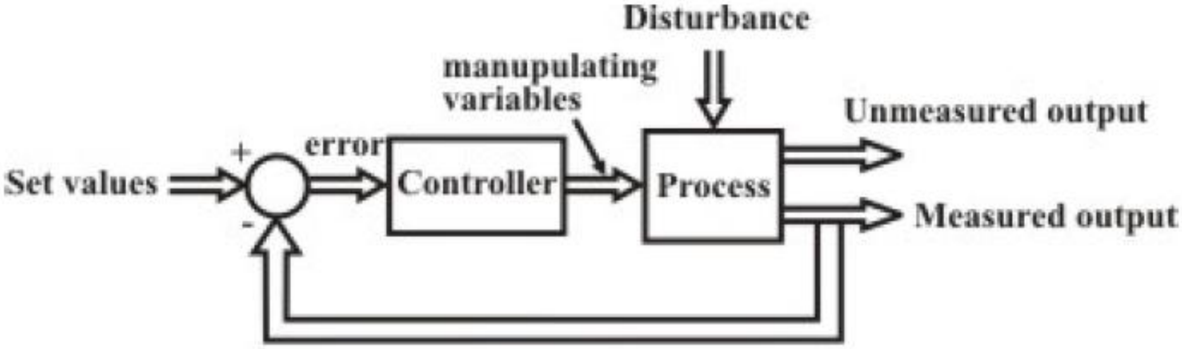


Fig. 1 General Description of a Closed loop process

In order to understand the behavour of a process, let us take up a simple open loop process as shown in Fig. 2. It is a tank containing certain liquid with an inflow line fitted with a valve V_1 and an outflow line fitted with another valve V_2. We want to maintain the level of the liquid in the tank; so the *measured output* variable is the liquid level *h*. It is evident from Fig.2 that there are two variables, which affect the measured output (henceforth we will call it only *output*) - the liquid level. These are the throttling of the valves V_1 and V_2. The valve V_1 is in the inlet line, and it is used to vary the inflow rate, depending on the level of the tank. So we can call the inflow rate as the manipulating variable. The outflow rate (or the throttling of the valve V_2) also affect the level of the tank, but that is decided by the demand, so not in our hand. We call it a *disturbance* (or sometimes as *load*).

The major feature of this process is that it has a single input (manipulating variable) and a single output (liquid level). So we call it a *Single-Input-Single-Output* (SISO) process. We would see afterwards that there are *Multiple-Input-Multiple-Output* (MIMO) processes also.

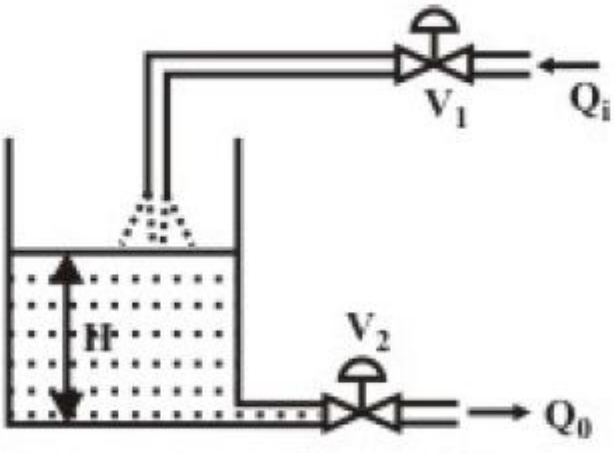


Fig. 2 Example of a Physical Process

3. Mathematical Modeling

In order to understand the behaviour of a process, a mathematical description of the dynamic behaviour of the process has to be developed. But unfortunately, the mathematical model of most of the physical processes is nonlinear in nature. On the other hand, most of the tools for analysis, simulation and design of the controllers, assumes, the process dynamics is linear in nature. In order to bridge this gap, the linearization of the nonlinear model is often needed. This linearization is with respect to a particular operating point of the system. In this section we will illustrate the nonlinear mathematical behaviour of a process and the linearization of the model. We will take up the specific example of a simple process described in Fig.2.

Let Q_i and Q_o are the inflow rate and outflow rate (in m^3/sec) of the tank, and H is the height of the liquid level at any time instant. We assume that the cross sectional area of the tank be A. In a steady state, both Q_i and Q_o are same, and the height H of the tank will be constant. But when they are unequal, we can write,

$$Q_i - Q_o = A\frac{dH}{dt} \qquad (1)$$

But the outflow rate Q_o is dependent on the height of the tank. Considering the Valve V_2 as an orifice, we can write, (please refer eqn.(4) in Lesson 7 for details)

$$Q_o = \frac{C_d A_2}{\sqrt{1-\beta^4}}\sqrt{\frac{2g}{\gamma}(P_1 - P_2)} \qquad (2)$$

We can also assume that the outlet pressure P_2=0 (atmospheric pressure) and

$$P_1 = \rho g H \qquad (3)$$

Considering that the opening of the orifice (valve V_2 position) remains same throughout the operation, equation (2) can be simplified as:

$$Q_o = C\sqrt{H} \qquad (4)$$

Where, C is a constant. So from equation (1) we can write that,

$$Q_i - C\sqrt{H} = A\frac{dH}{dt} \qquad (5)$$

The nonlinear nature of the process dynamics is evident from eqn.(5), due to the presence of the term $\sqrt{H}$.

$$Q_i - Q_o = A\frac{dH}{dt} \qquad (1)$$

But the outflow rate Q_o is dependent on the height of the tank. Considering the Valve V_2 as an orifice, we can write, (please refer eqn.(4) in Lesson 7 for details)

$$Q_o = \frac{C_d A_2}{\sqrt{1-\beta^4}}\sqrt{\frac{2g}{\gamma}(P_1 - P_2)} \qquad (2)$$

We can also assume that the outlet pressure P_2=0 (atmospheric pressure) and

$$P_1 = \rho g H \qquad (3)$$

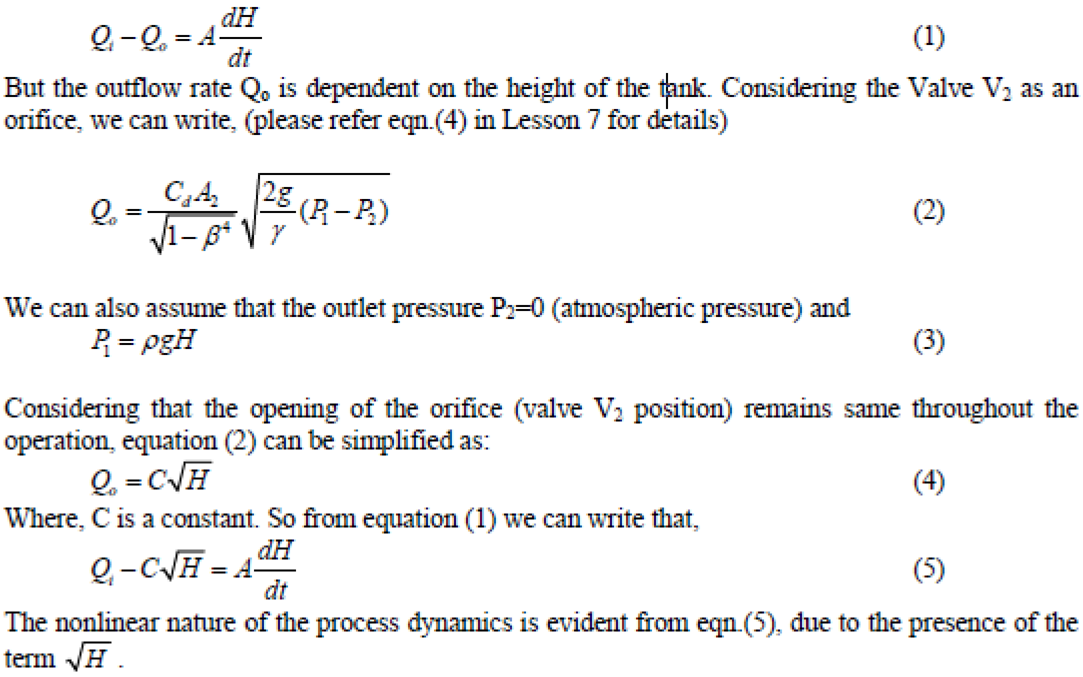
Considering that the opening of the orifice (valve V_2 position) remains same throughout the operation, equation (2) can be simplified as:

$$Q_o = C\sqrt{H} \qquad (4)$$

Where, C is a constant. So from equation (1) we can write that,

$$Q_i - C\sqrt{H} = A\frac{dH}{dt} \qquad (5)$$

The nonlinear nature of the process dynamics is evident from eqn.(5), due to the presence of the term $\sqrt{H}$.



$$Q_i - Q_o = A\frac{dH}{dt} \qquad (1)$$

But the outflow rate Q_o is dependent on the height of the tank. Considering the Valve V_2 as an orifice, we can write, (please refer eqn.(4) in Lesson 7 for details)

$$Q_o = \frac{C_d A_2}{\sqrt{1-\beta^4}}\sqrt{\frac{2g}{\gamma}(P_1 - P_2)} \qquad (2)$$

We can also assume that the outlet pressure P_2=0 (atmospheric pressure) and

$$P_1 = \rho g H \qquad (3)$$

Considering that the opening of the orifice (valve V_2 position) remains same throughout the operation, equation (2) can be simplified as:

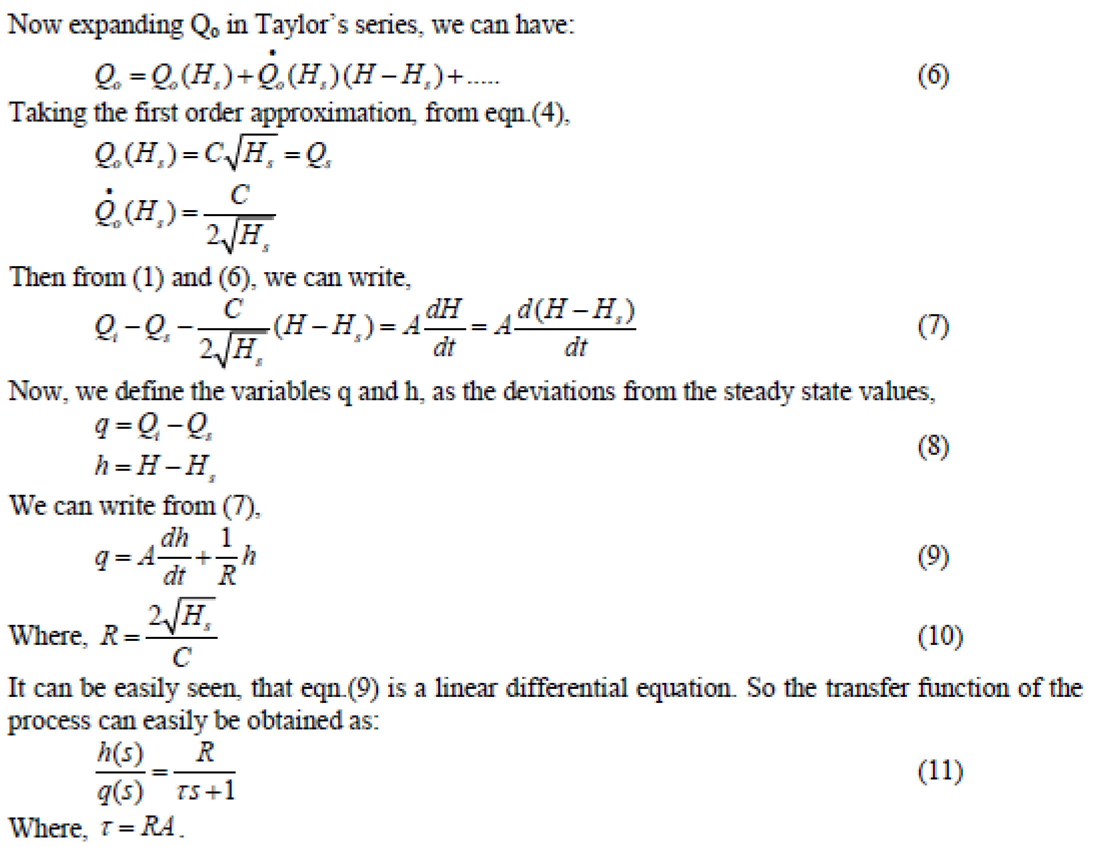
$$Q_o = C\sqrt{H} \qquad (4)$$

Where, C is a constant. So from equation (1) we can write that,

$$Q_i - C\sqrt{H} = A\frac{dH}{dt} \qquad (5)$$

The nonlinear nature of the process dynamics is evident from eqn.(5), due to the presence of the term $\sqrt{H}$.

In order to linearise the model and obtain a transfer function between the input and output, let us assume that initially Q_i =Q_o =Q_s; and the liquid level has attained a steady state value H_s. Now suppose the inflow rate has slightly changed, then how the height will change?



Now expanding Q_o in Taylor's series, we can have:

$$Q_o = Q_o(H_s) + \dot{Q}_o(H_s)(H - H_s) + ... \qquad (6)$$

Taking the first order approximation, from eqn.(4),

$$Q_o(H_s) = C\sqrt{H_s} = Q_s$$

$$\dot{Q}_o(H_s) = \frac{C}{2\sqrt{H_s}}$$

Then from (1) and (6), we can write,

$$Q_i - Q_s - \frac{C}{2\sqrt{H_s}}(H - H_s) = A\frac{dH}{dt} = A\frac{d(H - H_s)}{dt} \qquad (7)$$

Now, we define the variables q and h, as the deviations from the steady state values,

$$q = Q_i - Q_s$$
$$h = H - H_s \qquad (8)$$

We can write from (7),

$$q = A\frac{dh}{dt} + \frac{1}{R}h \qquad (9)$$

Where, $R = \dfrac{2\sqrt{H_s}}{C}$ (10)

It can be easily seen, that eqn.(9) is a linear differential equation. So the transfer function of the process can easily be obtained as:

$$\frac{h(s)}{q(s)} = \frac{R}{\tau s + 1} \qquad (11)$$

Where, $\tau = RA$.

It is to be noted that all the input and output variables in the transfer function model represent, the deviations from the steady state values. If the operating point (the steady state level H_s in the present case) changes, the parameters of the process (R andτ) will also change.

The importance of linearisation needs to be emphasized at this juncture. The mathematical models of most of the physical processes are nonlinear in nature; but most of the tools for design and analysis are for linear systems only. As a result, it is easier to design and evaluate the performance of a system if its mathematical model is available in linear form. Linearised model is an approximation of the actual model of the system, but it is preferred in order to have a physical insight of the system behaviour. It is to be kept in mind that this model is valid as long as the variation of the variables around the operating point is small. There are few systems whose dynamic behaviour is highly nonlinear and it is almost impossible to have a linear model of a system. For example, it is possible to develop the linearised transfer function model of an a.c. servomotor, but it is not possible for a step motor.

Referring to Fig. 2, if the valve V_1 is motorized and operated by electrical signal, we can also develop the model relating the electrical input signal and the output. Again, we have so far assumed that the opening of the valve V_2 to be constant, during the operation. But if we also consider its variation, that would also affect the dynamics of the tank model. So, the effect of disturbance can be incorporated in the overall plant model, as shown in Fig.3, by introducing a *disturbance transfer function D(s)*. *D(s)* can be easily by using the same methodology as described earlier in this section.

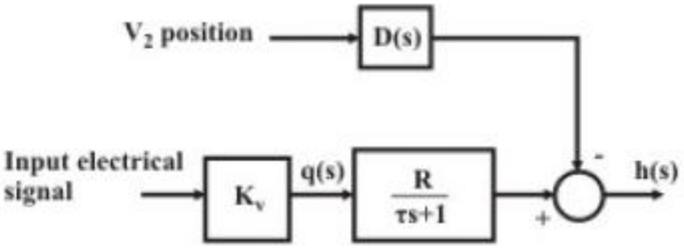


Fig. 3 Block diagram representation of the process shown in Fig. 2.

4. Higher Order System Model

We have considered a single tank and developed the linearised model of it. So it has a single time constant τ. But there are more complex processes. If there are two tanks coupled together, as shown in Fig.4, then we would have two time constants $_1\tau$and $_2\tau$. But it is evident that the dynamics of two tanks are coupled. Considering the change in the inflow *q(t)* as the input and the change in the level of the second tank h(t) as the output variable, with a little bit of calculation, it can be shown that the transfer function of this coupled tank system is

$$\frac{h_2(s)}{q(s)} = \frac{R_2}{\tau_1\tau_1 s^2 + (\tau_1 + \tau_2 + A_1 R_2)s + 1} \qquad (12)$$

The constants are similar to the earlier section with added suffixes corresponding to tank 1 and tank 2 respectively. In this case we have neglected the effects of the disturbances

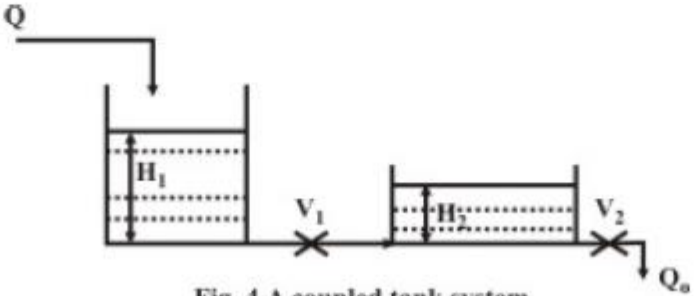


Fig. 4 A coupled tank system

5. Time delay

It has been mentioned earlier that one of the major characteristics of a process is the presence of time delay. This time delay term is often referred as “transportation lag”, since it is generated due to the delay in transportation of the output to the measuring point. The presence and effect of time delay can be easily explained with an example of a simple heat exchanger, as shown in Fig.

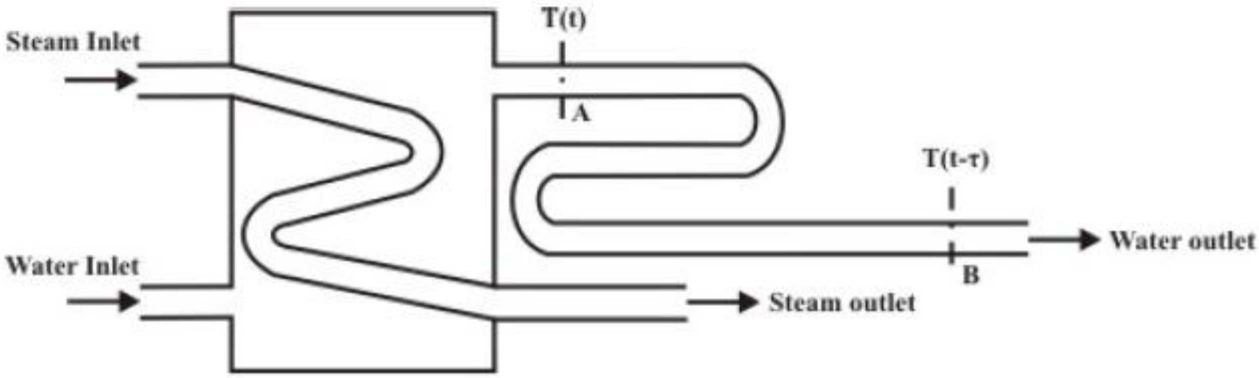


Fig. 5 Example of a time-delay system: a heat exchanger

In this case the transfer of heat takes place between the steam in the jacket and water in the tank. The measured output is the water temperature at the outlet T(t). For controlling this temperature, we may vary the steam flow rate at its inlet. So the manipulating variable is the steam flow rate. We can also identify a number of input variables those act as the disturbance, thus affecting the temperature at the water outlet; for example, inlet steam temperature, inlet water temperature and the water flow rate.

The temperature transducer should be placed at a location in the water outlet line just after the tank (location A in Fig. 5). But suppose, due to the space constraint, the transducer was placed at location B, at a distance L from the tank. In that case, there would be a delay sensing this temperature. If *T(t)* is the temperature measured at location A, then the temperature measured at location B would be ($_d$*Tt* τ−. The time delay term $_d$τ can be expressed in terms of the physical parameters as:

$$\tau_d = L/v \qquad (13)$$

where *L* is the distance of the pipeline between locations A and B;
and *v* is the velocity of water through the pipeline.

Noting from the Laplace Transformation table,

$$\mathcal{L}\ f(t-\tau_d) = e^{-s\tau_d}F(s) \qquad (14)$$

we can conclude that an additional term of $e^{-s\tau_d}$ would be introduced in the transfer function of the system due to the time delay factor. Thus the transfer function of an ordinary first order plant with time delay is

$$G(s) = \frac{Ke^{-s\tau_d}}{1+s\tau}$$

and its step response to a unit step input is as shown in Fig. 6.

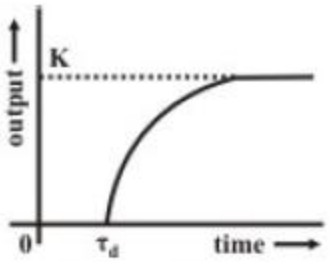


Fig. 6 Unit step response of a 1st order system with time delay

It can be seen that though the input has been at t = 0, the output remains zero till t = τ_d. This time delay present in the system may often be the main cause for instability of a closed loop system operation.

6. Multiple Input Multiple Output Systems

these cases, we had a single manipulating variable to control a single output variable. But in many cases, we have a number of inputs to control a number of outputs simultaneously, and the input-outputs are not decoupled. This will be evident if we consider a system, slightly modified system from that one shown in Fig. 4. In the modified system, we have added another inlet flow line in tank 2, as shown in Fig. 7.

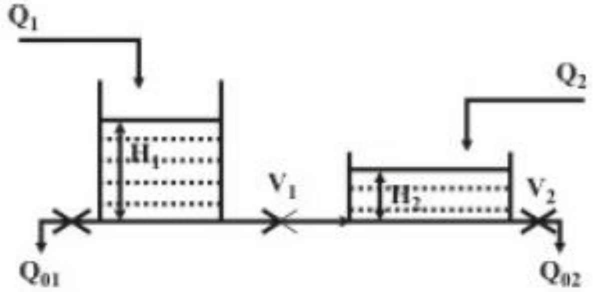


Fig. 7 A two-input two-output coupled tank system

If we consider the changes in inflow rates and are in inputs and the changes in the liquid levels of the two tanks $h_1 q_2 q_1$ and h_2 as the outputs, then the complete input-output behavior can be modeled using the transfer function matrix, as shown below:

$$\begin{bmatrix} h_1(s) \\ h_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} q_1(s) \\ q_2(s) \end{bmatrix} \tag{15}$$

We define $G(s)$ as the transfer function matrix and

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix}$$

In general, if there are *m* inputs and *p* outputs, then the order of the transfer function matrix is *p X m*. The MIMO system can also be further classified depending on the number of inputs and outputs. If the number of inputs is more than the number of outputs (*m*>*p*), then the system is called an *overactuated system*. If the number of inputs is less than the number of outputs (*m*<*p*), then the system is an *underactuated system*; while they are equal then the system is *square* (implying the *G(s)* is a square matrix).

A Multi-input-multi-output nonlinear system can be described in its state variable form as:

$$\dot{x} = f(x,u)$$
$$y = g(x,u)$$

where *x* is the state vector, *u* is the input vector and *y* is the output vector. *f* and *g* are nonlinear functions of *x* and *u*.

The above nonlinear system can also be linearised over its operating point and can be described in the state-space form as:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned} \qquad (16)$$

where *u* is the input vector of dimension *m*; *y* is the output vector of dimension *p* and *x* is an *n*-dimensional vector representing the states. The transfer G(s) function matrix can be obtained as:

$$G(s) = C\left(sI - A\right)^{-1} B + D \qquad (17)$$

For more details, please refer any book on Control Systems.

P-I-D Control

The basic control loop can be simplified for a single-input-single-output (SISO) system as in Fig.1. Here we are neglecting any disturbance present in the system.

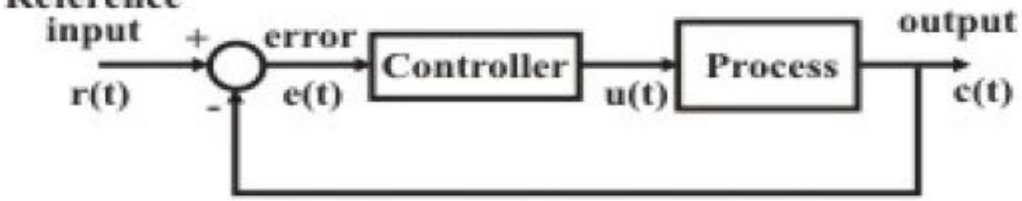


Fig. 1 A closed loop SISO system

The controller may have different structures. Different design methodologies are there for designing the controller in order to achieve desired performance level. But the most popular among them is Proportional-Integral-derivative (PID) type controller. In fact more than 95% of the industrial controllers are of PID type.

As is evident from its name, the output of the PID controller u(t) can be expressed in terms of the input e(t), as:

$$u(t) = K_p\left[e(t) + \tau_d \frac{de(t)}{dt} + \frac{1}{\tau_i}\int_0^t e(\tau)d\tau\right] \qquad (1)$$

and the transfer function of the controller is given by:

$$C(s) = K_p\left(1 + \tau_d s + \frac{1}{\tau_i s}\right) \qquad (2)$$

The terms of the controller are defined as:

K_p = Proportional gain

τ_d = Derivative time, and

τ_i = Integral time.

In the following sections we shall try to understand the effects of the individual components- proportional, derivative and integral on the closed loop response of this system. For the sake of simplicity, we consider the transfer function of the plant as a simple first order system without time delay as:

$$P(s) = \frac{K}{1 + \tau s} \qquad (3)$$

Proportional control

With the proportional control action only, the closed loop system looks like:

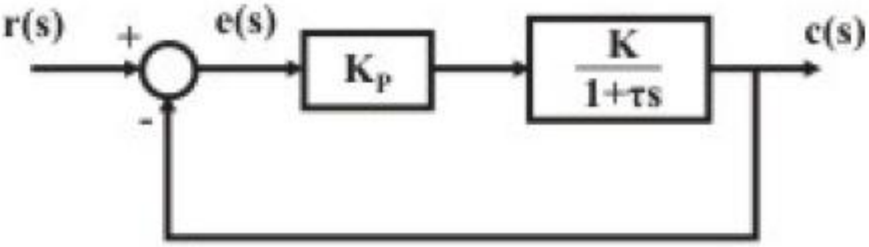


Fig. 2 Proportional Control

Now the closed loop transfer function can be expressed as:

$$\frac{c(s)}{r(s)} = \frac{\frac{KK_p}{1+\tau s}}{1+\frac{KK_p}{1+\tau s}} = \frac{KK_p}{1+KK_p+\tau s} = \frac{KK_p}{1+KK_p}\frac{1}{1+\tau^{'} s} \qquad (4)$$

where $\tau^{'} = \frac{\tau}{1+KK_p}$.

For a step input $r(s) = \frac{A}{s}$,

$$c(s) = \frac{KK_p}{1+KK_p}\frac{A}{s\left(1+\tau^{'} s\right)}$$

or, $$c(t) = \frac{AKK_p}{1+KK_p}\left(1-e^{-s\tau/\tau^{'}}\right) \qquad (5)$$

The system response is shown in Fig. 2.

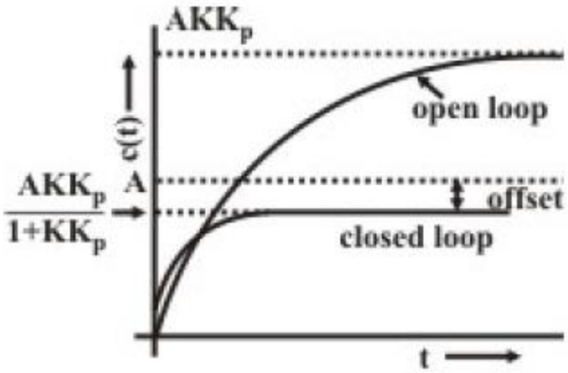


Fig. 3 Response with a proportional controller

From eqn. (5) and Fig. 2, it is apparent that:

1. The time response improves by a factor $\frac{1}{1+KK_p}$ (i.e. the time constant decreases).
2. There is a steady state offset between the desired response and the output response = $A\left(1-\frac{KK_p}{1+KK_p}\right) = \frac{A}{1+KK_p}$.

This offset can be reduced by increasing the proportional gain; but that may also cause increase oscillations for higher order systems.

The offset, often termed as "steady state error" can also be obtained from the error transfer function and the error function *e(t)* can be expressed in terms of the Laplace transformation form:

$$e(s) = \frac{1}{1+\frac{KK_p}{1+\tau s}}\frac{A}{s} = \frac{1+\tau s}{1+KK_p+\tau s}\frac{A}{s}$$

Using the final value theorem, the steady state error is given by:

$$e_{ss} = \underset{t\to\infty}{Lt}\; e(t) = \underset{s\to 0}{Lt}\; s\,e(s) = \underset{s\to 0}{Lt}\frac{1+\tau s}{1+KK_p+\tau s}\frac{A}{s} = \frac{A}{1+KK_p}$$

Often, the proportional gain term, K_p is expressed in terms of "Proportional Band". It is inversely proportional to the gain and expressed in percentage. For example, if the gain is 2, the proportional band is 50%. Strictly speaking, proportional band is defined as the %error to move the control valve from fully closed to fully opened condition. However, the meaning of this statement would be clear to the reader afterwards.

Integral Control

If we consider the integral action of the controller only, the closed loop system for the same process is represented by the block diagram as shown in Fig. 3.

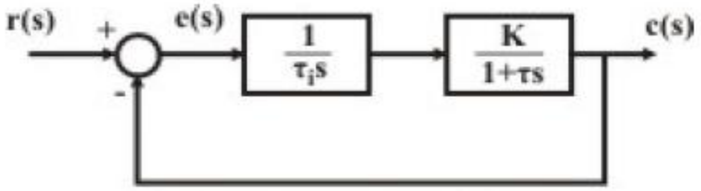


Fig. 4 Integral control action

Proceeding in the same way as in eqn. (4), in this case, we obtain,

$$\frac{c(s)}{r(s)} = \frac{\frac{K}{\tau_i s(1+\tau s)}}{1+\frac{K}{\tau_i s(1+\tau s)}} = \frac{K}{K+\tau_i s+\tau\tau_i s^2}$$

From the first observation, it can be seen that with integral controller, the order of the closed loop system increases by one. This increase in order may cause instability of the closed loop system, if the process is of higher order dynamics.

For a step input $r(s)=\frac{A}{s}$,

$$e(s)=\frac{1}{1+\frac{K}{\tau_i(1+\tau s)}}\frac{A}{s}=\frac{\tau_i s(1+\tau s)}{\tau_i s(1+\tau s)+K}\frac{A}{s}$$

$$e_{ss}=\underset{s\to 0}{Lt}\,s\,e(s)=0$$

So the major advantage of this integral control action is that the steady state error due to step input reduces to zero. But simultaneously, the system response is generally slow, oscillatory and unless properly designed, sometimes even unstable. The step response of this closed loop system with integral action is shown in Fig. 4.

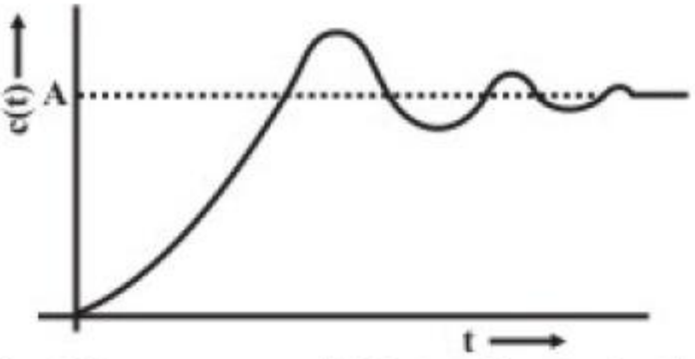


Fig. 5 Step response with integral control action

Proportional Plus Integral (P-I) Control

With P-I controller the block diagram of the closed loop system with the same process is given in Fig. 5.

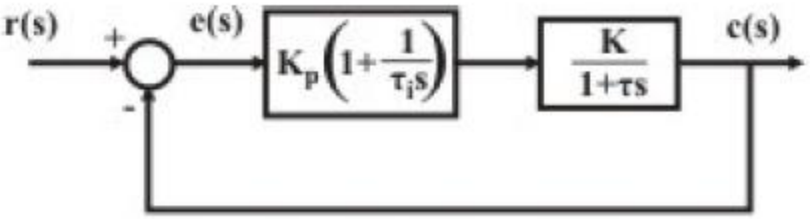


Fig. 6 Proportional plus Integral Control action

It is evident from the above discussions that the P-I action provides the dual advantages of fast response due to P-action and the zero steady state error due to I-action. The error transfer function of the above system can be expressed as:

$$\frac{e(s)}{r(s)} = \frac{1}{1+\frac{KK_p(1+\tau_i s)}{\tau_i s(1+\tau s)}} = \frac{\tau_i s(1+\tau s)}{s^2\tau\tau_i + (1+KK_p)\tau_i s + KK_p}$$

In the same way as in integral control, we can conclude that the steady state error would be zero for P-I action. Besides, the closed loop characteristics equation for P-I action is:

$$s^2\tau\tau_i + (1+KK_p)\tau_i s + KK_p = 0\,;$$

from which we can obtain, the damping constant as:

$$\xi = \left(\frac{1+KK_p}{2}\right)\sqrt{\frac{\tau_i}{KK_p\tau}}$$

whereas, for simple integral control the damping constant is:

$$\xi = \left(\frac{1}{2}\right)\sqrt{\frac{\tau_i}{K\tau}}$$

Comparing these two, one can easily observe that, by varying the term K_p, the damping constant can be increased. So we can conclude that by using P-I control, the steady state error can be brought down to zero, and simultaneously, the transient response can be improved. The output responses due to (i) P, (ii) I and (iii) P-I control for the same plant can be compared from the sketch shown in Fig. 6.

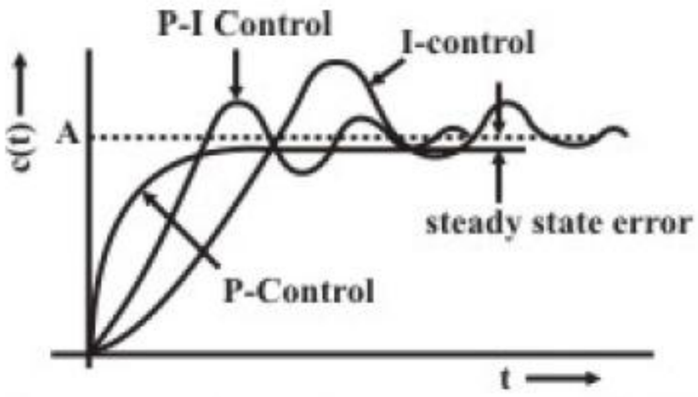


Fig. 7 Comparison among the transient responses with P, I and P-I control

Proportional Plus Derivative (P-D) Control

The transfer function of a P-D controller is given by:

$$C(s) = K_p(1+\tau_d s)$$

P-D control for the process transfer function $P(s) = \dfrac{K}{1+\tau s}$ apparently is not very useful, since it cannot reduce the steady state error to zero. But for higher order processes, it can be shown that the stability of the closed loop system can be improved using P-D controller. For this, let us take up the process transfer function as $P(s) = \dfrac{1}{Js^2}$. Looking at Fig.7, we can easily conclude that with proportional control, the closed loop transfer function is

$$\frac{c(s)}{r(s)} = \frac{\dfrac{K_p}{Js^2}}{1+\dfrac{K_p}{Js^2}} = \frac{K_p}{Js^2 + K_p}$$

and the characteristics equation is $Js^2 + K_p = 0$; giving oscillatory response. But with P-D controller, the closed loop transfer function is:

$$\frac{c(s)}{r(s)} = \frac{\dfrac{K_p(1+\tau_d s)}{Js^2}}{1+\dfrac{K_p(1+\tau_d s)}{Js^2}} = \frac{K_p(1+\tau_d s)}{Js^2 + K_p(1+\tau_d s)}$$

whose characteristics equation is $Js^2 + K_p\tau_d s + K_p = 0$; that will give a stable closed loop response.

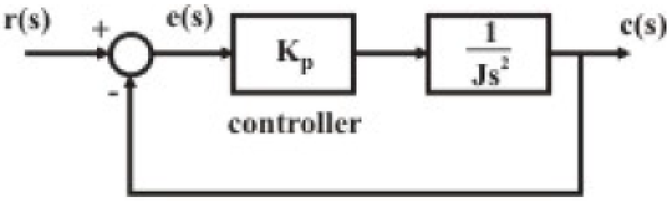


Fig. 8 Control action with a higher order process

The step responses of this process with P and P-D controllers are compared in Fig.8.

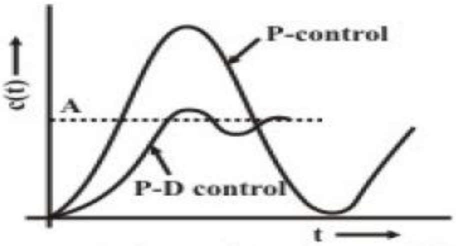


Fig. 9 Improvement of transient response with P-D control

Proportional-Integral-Derivative (PID) control

It is clear from above discussions that a suitable combination of proportional, integral and derivative actions can provide all the desired performances of a closed loop system. The transfer function of a P-I-D controller is given by:

$$C(s) = K_p\left(1 + \tau_d s + \frac{1}{\tau_i s}\right)$$

The order of the controller is low, but this controller has universal applicability; it can be used in any type of SISO system, e.g. linear, nonlinear, time delay etc. Many of the MIMO systems are first decoupled into several SISO loops and PID controllers are designed for each loop. PID controllers have also been found to be robust, and that is the reason, it finds wide acceptability for industrial processes. However, for proper use, a controller has to be tuned for a particular process; i.e. selection of P,I,D parameters are very important and process dependent. Unless the parameters are properly chosen, a controller may cause instability to the closed loop system. The method of tuning of P,I,D parameters would be taken up in the next lesson.

It is not always necessary that all the features of proportional, derivative and integral actions should be incorporated in the controller. In fact, in most of the cases, a simple P-I structure will suffice.

Controller Tuning

It is needless to say that the controller parameters influence heavily the performance of the closed loop system. Again, the choice of the value of the P, I and D parameters is very much process dependent. As a result, thorough knowledge about the plant dynamics is important for selection of these parameters. In most of the cases, it is difficult to obtain the exact mathematical model of the plant. So, we have to rely on the experimentation for finding out the optimum settings of the controller for a particular process. The process of experimentation for obtaining the optimum values of the controller parameters with respect to a particular process is known as controller tuning. It is needless to say, that controller tuning is very much process dependent and any improper selection of the controller settings may lead to instability, or deterioration of the performance of the closed loop system. In 1942 two practicing engineers, J.G. Ziegler and N.B. Nichols, after carrying out extensive experiments with different types of processes proposed certain tuning rules, there were readily accepted and till now are used as basic guidelines for tuning of PID controllers. Subsequently, G.H. Cohen and G.A.

Coon in 1953 proposed further modifications of the above techniques. Still then, the methods are commonly known as Ziegler-Nichols method. Substantial amount of research has been carried out on tuning of P-I-D controllers since last six decades. Several other methods have also been proposed. Most of them are model based, i.e. they assume that the mathematical model of the system is available to the designer. In fact, if the mathematical model of the system is available, many of them perform better than conventional Ziegler-Nichols method. But the strength of the ZN method is that it does not require a mathematical model, but controller parameters can simply be chosen by experimentation. We would be discussing the three experimental techniques those come under the commonly known Ziegler-Nichols method.

Now let us look back to whatever discussed in lessons 11 and 12. The closed loop system can be described as shown in Fig. 1.

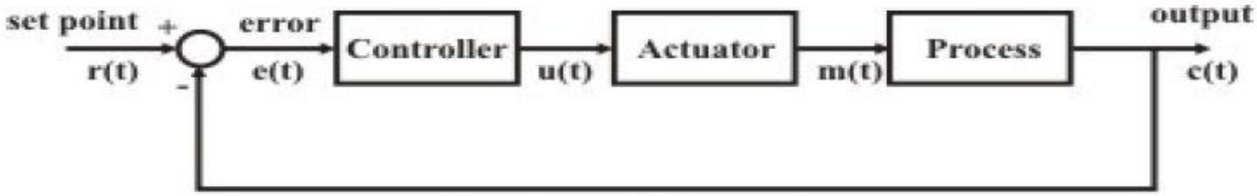


Fig. 1 Closed loop system.

The error signal is fed to the controller and the controller provides output u(t). Since the capacity of the controller to deliver output power is limited, an actuator is needed in between the controller and the process, which will actuate the control signal. It may be a valve positioner to open or close a valve; or a damper positioner to control the airflow through a damper. The controller considered here is a P-I-D controller whose input and output relationship is given by the equation:

$$u(t) = K_p \left[e(t) + \tau_d \frac{de(t)}{dt} + \frac{1}{\tau_i} \int_0^t e(\tau) d\tau \right]$$

Our objective is to find out the optimum settings of the P,I,D parameters, namely K_p , τ_d and τ_i through experimentation, which will provide satisfactory closed loop performance, of the particular process in terms of, say, stability, overshoot, setting time etc. Three methods of tuning are elaborated in the following sections.

Reaction Curve Technique

This is basically an open loop technique of tuning. Here the process is assumed to be a stable first order system with time delay. The closed loop system is broken as shown in Fig.2; a step input is applies at , output is measured at *b*. In fact, a bias input may be necessary so that the plant output initially becomes close to the nominal value. The step input is superimposed on this bias value. The input and the output response are plotted by suitable means as shown in Fig. 3. *m*

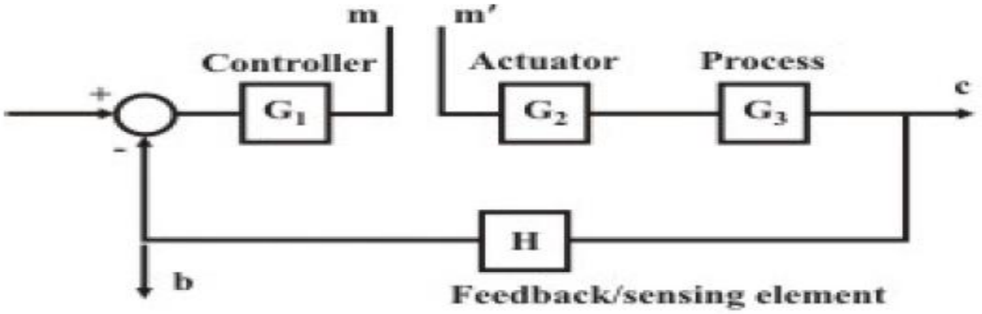


Fig. 2 Reaction curve technique for controller tuning.

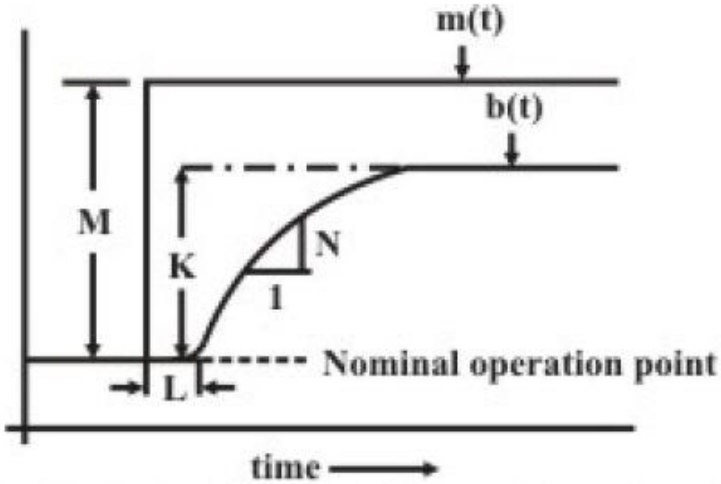


Fig. 3 Input and output plots under the condition shown in Fig. 1.

M,L and K are measured. Let us define the following terms corresponding to Fig. 2:

Slope = N,

Time Constant T=K/N

Lag Ratio R=L/T

Then, the recommended optimum settings, for P, P-I and P-I-D controller are as follows

Optimum settings

P-Control: $K_p = \frac{M}{NL}(1 + \frac{R}{3})$

P-I Control: $K_p = \frac{M}{NL}(\frac{9}{10} + \frac{R}{12})$; $\tau_i = L\left(\frac{30 + 3R}{9 + 20R}\right)$

P-I-D Control: $K_p = \frac{M}{NL}(\frac{4}{3} + \frac{R}{4})$; $\tau_i = L\left(\frac{32 + 6R}{13 + 8R}\right)$

$$\tau_d = L\left(\frac{4}{11 + 2R}\right)$$

Closed Loop Technique (Continuous Cycling method)

The major objection to the tuning methodology using reaction curve technique is that process has to be run in open loop that may not always be permissible. For tuning the controller when the process is in under closed loop operation, there are two methodologies. The first one, continuous cycling method is explained below.

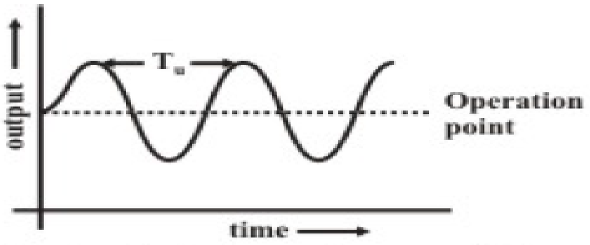
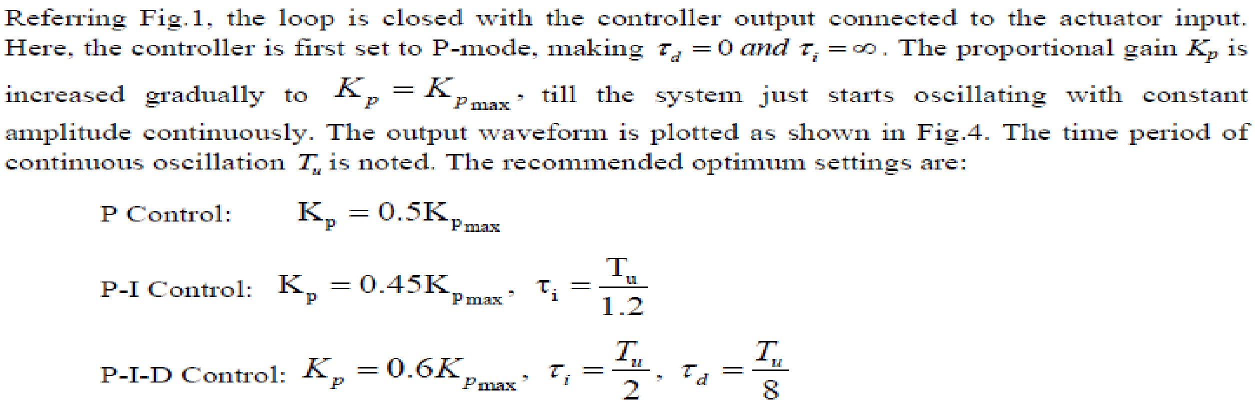


Fig. 4 Controller tuning continuous oscillation mode.



Referring Fig.1, the loop is closed with the controller output connected to the actuator input. Here, the controller is first set to P-mode, making $\tau_d = 0$ *and* $\tau_i = \infty$. The proportional gain K_p is increased gradually to $K_p = K_{P\max}$, till the system just starts oscillating with constant amplitude continuously. The output waveform is plotted as shown in Fig.4. The time period of continuous oscillation T_u is noted. The recommended optimum settings are:

P Control: $K_p = 0.5K_{P\max}$

P-I Control: $K_p = 0.45K_{P\max}$, $\tau_i = \frac{T_u}{1.2}$

P-I-D Control: $K_p = 0.6K_{P\max}$, $\tau_i = \frac{T_u}{2}$, $\tau_d = \frac{T_u}{8}$

Closed Loop Technique (Damped oscillation method)

In many cases, plants are not allowed to undergo through sustained oscillations, as is the case for tuning using continuous cycling method. Damped oscillation method is preferred for these cases. Here, initially the closed loop system is operated initially with low gain proportional control mode with $\tau_d = 0 \; and \; \tau_i = \infty$. The gain is increased slowly till a decay ratio (p_2/p_1) of 1/4th is obtained in the step response in the output, as shown in Fig. 5. Under this condition, the period of damped oscillation, T_d is also noted. Let K_d be the proportional gain setting for obtaining 1/4th decay ratio.

The optimum settings for a P-I-D controller are:

$$K_p = K_d; \tau_i = \frac{T_d}{6}; \tau_d = \frac{T_d}{1.5}$$

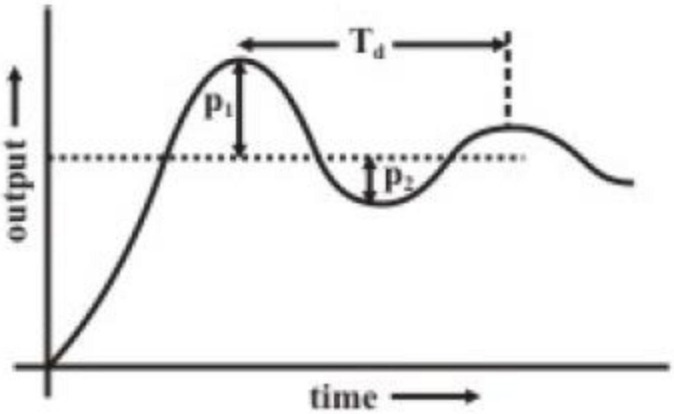


Fig. 5 Controller tuning using damped oscillation technique.

General comments about controller tuning

The different methodologies of controller tuning, known as Ziegler-Nichols method have been illustrated in the earlier sections. It is to be remembered that the recommended settings are empirical in nature, and obtained from extensive experimentation with number of different processes; there is no theoretical basis behind these selections. As a result, a better combination of the P, I, D values may always be found, that will give less oscillation and better settling time. But with no a-priori knowledge of the system, it is always advisable to perform the experimentation and select the controller settings, obtained from Ziegler-Nichols method. But there is always scope for improving the performance of the controller by fine-tuning. So, Ziegler-Nichols method provides initial settings that will give satisfactory, result, but it is always advisable to fine-tune the controller further for the particular process and better performance is expected to be achieved.

Nowadays digital computers are replacing the conventional analog controllers. P-I-D control actions are generated through digital computations. Digital outputs of the controllers are converted to analog signals before they are fed to the actuators. In many cases, commercial software are available for *Auto tuning* the process. Here the controller generates several commands those are fed to the plant. After observing the output responses, the controller parameters are selected, similar to the cases discussed above.

Integration windup and Bumpless transfer

Two major issues of concern with the close loop operation with P-I-D controllers are the *Integration Windup* and the requirement of providing *Bumpless Transfer*. These two issues are briefly elaborated below. The methodologies for providing Anti-integration Windup and Bumpless Transfer would be discussed in the next lesson.

Integration Windup

A significant problem with integral action is that when the error signal is large for a significant period of time. This can occur every time when there is large change in set point. If there is a sudden large change in set point, the error will be large and the integrator output in a P-I-D control will build up with time. As a result, the controller output may exceed the saturation limit of the actuator. This windup, unless prevented may cause continuous oscillation of the process that is not desirable.

Bumpless Transfer

When a controller is switched from manual mode to auto-mode, it is desired that the input of the process should not change suddenly. But since there is always a possibility that the decision of the manual mode of control and the auto mode of control be different, there may be a sudden change in the output of the controller, giving rise to a sudden jerk in the process operation. Special precautions are taken for *bumpless transfer* from manual to auto-mode.

Implementation of P-I-D Controllers

Looking back to the history of the PID controller, the PID controllers in the initial days were all pneumatic. In fact, all the experimentation by Ziegler and Nichols were carried out with pneumatic controllers. But pneumatic controllers were slow in nature. After the development of electronic devices and operational amplifiers, the electronic controllers started replacing the conventional pneumatic controllers. But with the advent of the microprocessors and microcontrollers, the focus of development is now towards the implementation with digital PID controllers. The major advantage of using digital PID controllers is that the controllers parameter can be programmed easily; as a result, they can be changed without changing any hardware. Moreover, the same digital computer can be used for a number of other applications besides generating the control action.

Bumpless Transfer

It is quite normal to set up some processes using manual control initially, and once the process is close to normal operating point, the control is transferred to automatic mode through auto/manual switch. In such cases, in order to avoid any jerk in the process the controller output immediately after the changeover should be identical to the output set in the manual mode. This can be achieved by forcing the integral output at the instant of transfer to balance the proportional and derivative outputs against the previous manual output; i.e.

Integral output = {(previous manual) – (proportional + derivative) output}.

Similarly, for automatic to manual transfer, initially the manual output is set equal to the controller output and the difference is gradually reduced by incrementing or decrementing the manual output to the final value of the manual signal and thus effecting a change over.

Another way to transfer from Auto to Manual mode in a bumpless manner, the set point may be made equal to the present value of the process variable and then slowly changing the set point to its desired value.

The above features can be easily be implemented if a digital computer is used as a controller. This provision eliminates the chance of the process receiving sudden jolt during transfer.

Prevention of Integration Windup

The effect of integration windup has been discussed in the last lesson. If there is a sudden large change in set point, the error between the set point and the process output will suddenly shoot up and the integrator output due to this error will build up with time. As a result, the controller output may exceed the saturation limit of the actuator. This windup, unless prevented may cause continuous oscillation of the process.

There exits several methods through which integration windup can be prevented. Before we go to the actual methods, let us consider the input-output characteristics of an actuator as shown in Fig. 1.

Its characteristics is similar to of an amplifier, where the output varies linearly with the input till the input is within a certain range; beyond that the output becomes constant either at the maximum or the minimum values of the output. The upper and lower limits of the output may correspond to the flow rates of a control valve when the valve is at fully open and fully closed position.

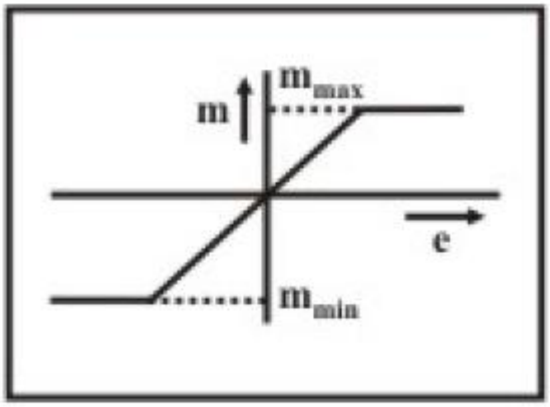


Fig. 1 Typical actuator characteristics

The first method uses a switch to break the integral action, whenever the actuator goes to saturation. This can be illustrated by Fig. 2. Consider schematic arrangement of a controller shown in the figure. When the switch is closed, transfer function of the controller can be obtained as:

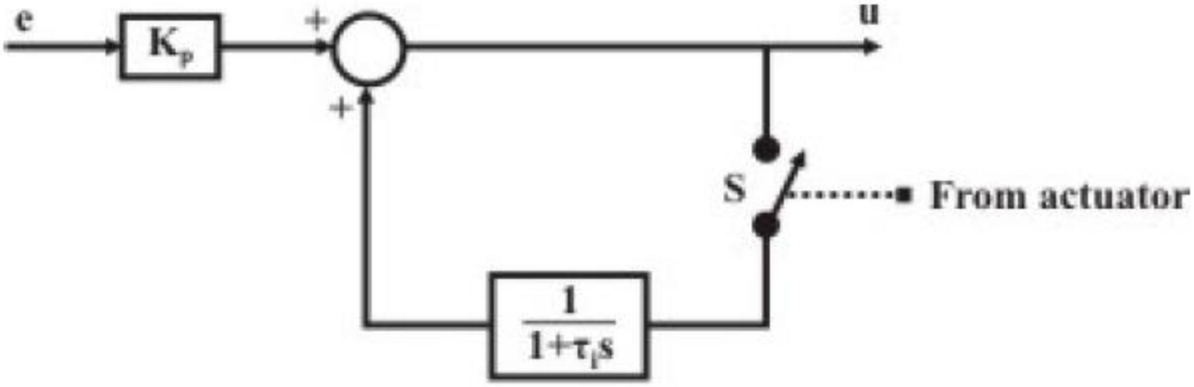


Fig. 2 Scheme for Anti integration windup

$$\frac{u(s)}{e(s)} = K_p\left(\frac{1}{1-\dfrac{1}{\tau_i s+1}}\right) = K_p\left(\frac{1+\tau_i s}{\tau_i s}\right) = K_p(1+\frac{1}{\tau_i s})$$

So when the switch is closed, the controller acts as a P-I controller. On the other hand, if the switch is open, it is a simple P- controller. The switch is activated by the position of the actuator. If the actuator is operating in the linear range, the switch is closed, and the controller is in P-I mode. But whenever the actuator is in the saturation mode, the switch is automatically opened; the controller becomes a P-controller. As a result, any windup due to the presence of integral mode is avoided.

Another technique for antiwindup action is illustrated in Fig.3. Here we assume that the slope of the actuator in the linear range is unity. As a result, when the actuator is operating in the linear range the error e_A is zero, and the controller acts as a PI controller. But when the actuator is in saturation mode, the error e_A is negative for a positive *e*. This will reduce the integral action in the overall control loop.

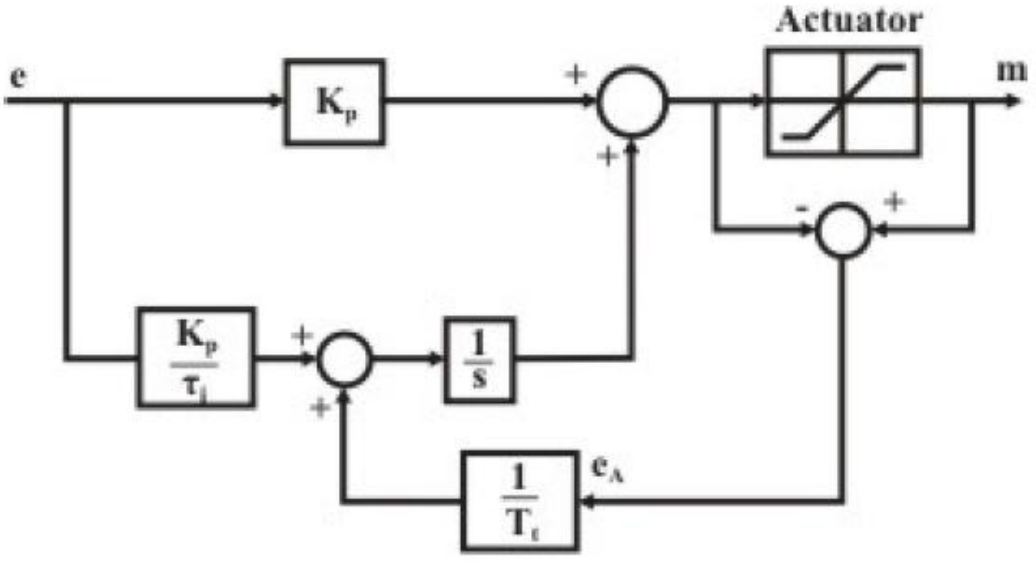
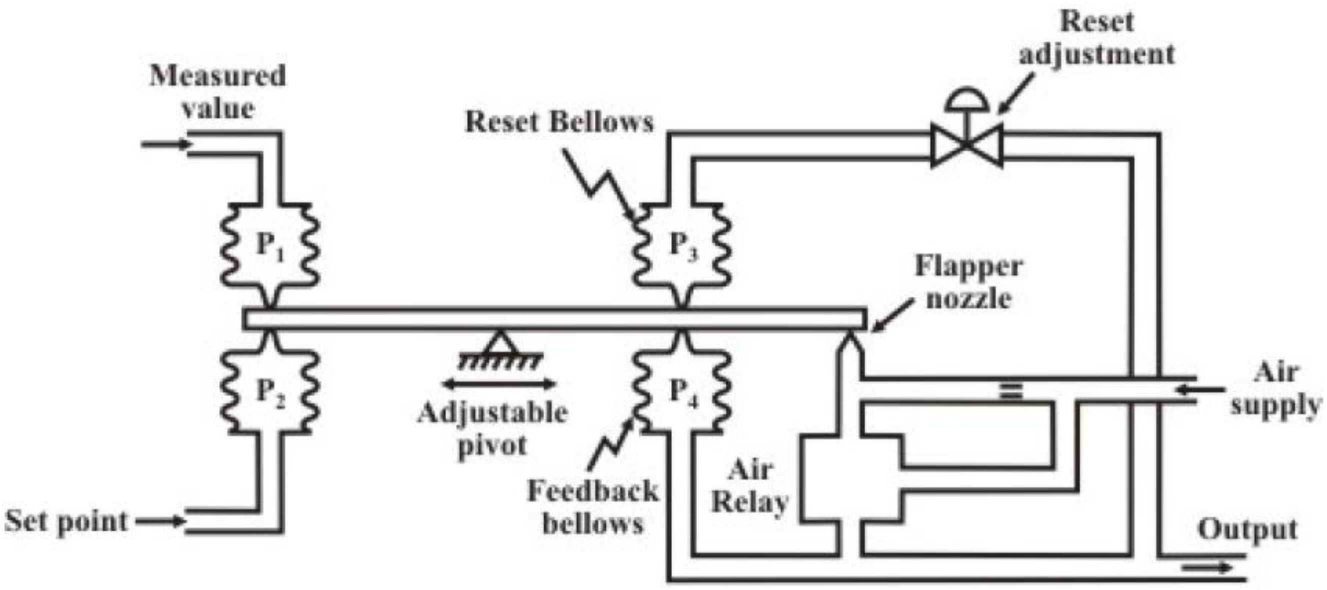


Fig. 3 Alternative arrangement for antiwindup action

Pneumatic Controller

It has been already mentioned that the early days PID controllers were all pneumatic type. The advantage of pneumatic controllers is its ruggedness, while its major limitation is its slow response. Besides it requires clean and constant pressure air supply. The major components of a pneumatic controller are bellows, flapper nozzle amplifier, air relay and restrictors (valves). The integral and derivative actions are generated by controlling the passage of air flow through restrictors to the bellows.



Here four bellows are connected to a force beam as shown. The measured process variable is converted to air pressure and connected to the bellows P_1. Similarly the air pressure corresponding to the set point signal is applied to the bellow P_2. The error corresponding to the measured value and the set point generates a force on the left hand side of the force beam. There is an adjustable pivot arrangement that sets the proportional gain of the amplifier. The right hand side of the force beam is connected to two bellows, P_3 and P_4 and a flapper nozzle amplifier. The output air pressure is dependent on the gap between the flapper and nozzle. An air relay enhances the air handling capacity. The output pressure is directly fed back to the feedback bellows P_4, and also to P_3 through a restrictor (valve). The opening of this restrictor decides the integral action to be applied. With a slight modification of this scheme, a pneumatic PID controller can also be implemented.

Electronic PID Controllers

Electronic PID controllers can be obtained using operational amplifiers and passive components like resistors and capacitors. A typical scheme is shown in Fig. 5. With little calculations, it can be shown that the circuit is capable of delivering the PID actions as:

$$e_0(t) = K_p\left[e(t) + \frac{1}{\tau_i}\int e(\tau)\ \ d\tau + \tau_d\frac{de(t)}{dt}\right] \qquad (1)$$

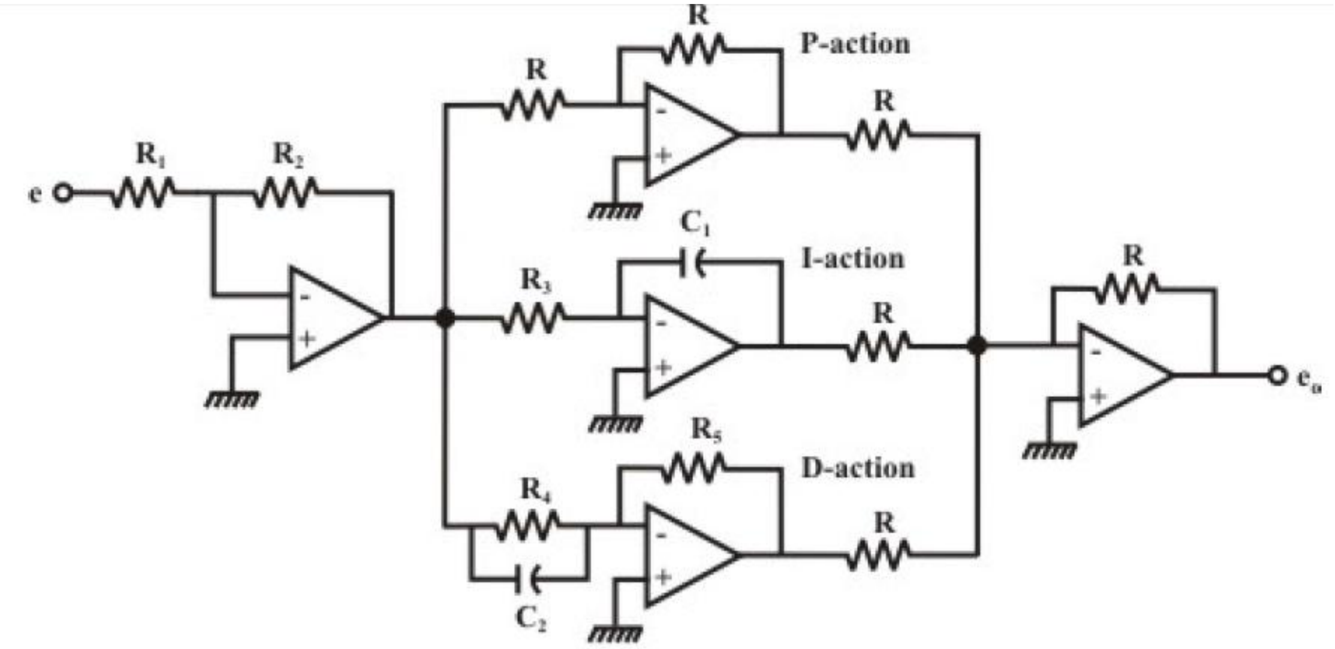


Fig. 5 Electronic PID controller

It is evident from Fig. 5, the proportional gain K_p is decided by the ratio $\frac{R_2}{R}$ of the first amplifier; the integral action is decided by R_3 and C_1 and the derivative action by R_5 and $C_{2.}$ The final output however comes out with a negative sign, compared to eqn. (1) (though the positive sign can also be obtained by using a noninverting amplifier at the input stage, instead of the inverting amplifier). The op. amps. Shown in the circuits are assumed to be ideal.

Digital P-I-D Control

In the digital control mode, the error signal is first sampled and the controller output is computed numerically through a digital processor.

Now Controller output for a continuous-type P-I-D controller:

$$u(t) = K_p \left[e(t) + \frac{1}{\tau_i} \int e(\tau)\, d\tau + \tau_d \frac{de(t)}{dt} \right] \qquad (1)$$

The above equation can be discretised at small sampling interval T_0 as shown in Fig. 6.

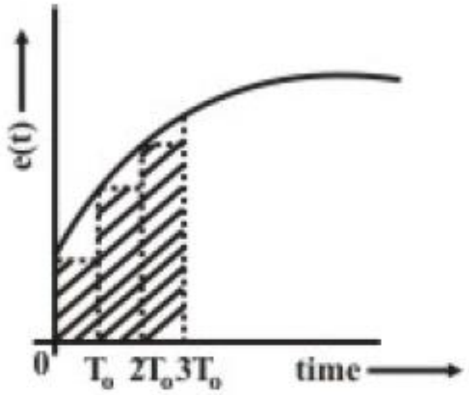


Fig. 6 Discretisation of the error signal

Taking the first order derivative,

$$\frac{de}{dt} \Rightarrow \frac{1}{T_0}\left[e(k)-e(k-1)\right]$$

and using rectangular integration, we can approximate as:

$$\int_0^t e(\tau)dt \Rightarrow T_0\sum_{i=0}^{k-1} e(i) \quad ; \quad t=kT_0$$

Now replacing the derivative and integral terms in eqn. (1), one can obtain,

$$u(k)=K_p\left[e(k)+\frac{T_0}{\tau_i}\sum_{i=0}^{k-1}e(i)+\frac{\tau_d}{T_0}\{e(k)-e(k-1)\}\right] \qquad (2)$$

The above algorithm is known as *Position algorithm*. But the major problem here is that the error values at all the time instants are to be stored (or at least the second term of the r.h.s of Eqn. (2) at each instant have to be stored). An alternative approach known as velocity algorithm can be obtained as follows.

From (2), one can write the error signal at the (k-1) th instant as:

$$u(k-1)=K_p\left[e(k-1)+\frac{T_0}{\tau_i}\sum_{i=0}^{k-2}e(i)+\frac{\tau_d}{T_0}\{e(k-1)-e(k-2)\}\right] \qquad (3)$$

Subtracting eqn. (3) from (2), we can have:

$$\Delta u(k)=u(k)-u(k-1)$$

$$=K_p\left[e(k)-e(k-1)+\frac{T_0}{\tau_i}e(k-1)+\frac{\tau_d}{T_0}\{e(k)-2e(k-1)+e(k-2)\}\right]$$

$$=q_0e(k)+q_1e(k-1)+q_2e(k-2) \qquad (4)$$

where,

$$q_0 = K_p(1+\frac{\tau_d}{T_0})$$

$$q_1 = -K_p(1+\frac{2\tau_d}{T_0}-\frac{T_0}{\tau_i})$$

$$q_2 = K_p\frac{\tau_d}{T_0}$$

The above algorithm is known as *Velocity algorithm*. The major advantage of this algorithm is that it is of recursive type. It calculates the incremental output at each sample instant. As a result, it requires only to store three previous values: e(k), e(k-1) and e(k-2). Besides it has got several other advantages also those are elaborated below:

Special Control Structures: Feedforward and Ratio Control

Feedforward Control

When the disturbance is measurable, feedforward control is an effective means for cancelling the effects of disturbance on the system output. This is advantageous, since in a simple feedback system, the corrective action starts after the effect of disturbance is reflected at the output. On the other hand, in feedforward control the change in disturbance signal is measured and the corrective action takes place immediately. As a result, the speed and performance of the overall system improves, if feedforward control, together with feedback action is employed.

In order to illustrate the effect of feedforward control, let us consider the heat exchange process shown in Fig.1. The cold water comes from a tank and flows to the heat exchanger. The flow rate of cold water can be considered as a disturbance. The change in input flow line may occur due to the change in water level in the tank. Suppose, the feedforward line is not connected, and the controller acts as a feedback control only. If the water inlet flow rate increases, the temperature of the outlet hot water flow will decrease. This will be sensed by the temperature sensor that will compare with the set point temperature and the temperature controller will send signal to open the control valve to allow more steam at the steam inlet. The whole operation is a time consuming and as a result the response of the controller due to the disturbance (inlet water flow rate) is normally slow. But if we measure the change in inlet flow rate by a flowmeter and feed this information to the controller, the controller can immediately take the correcting action anticipating the change in outlet temperature. This will improve the speed of response. Thus feedforward action, in addition to the feedback control improves the performance of the system, but provided, the disturbance is measurable.

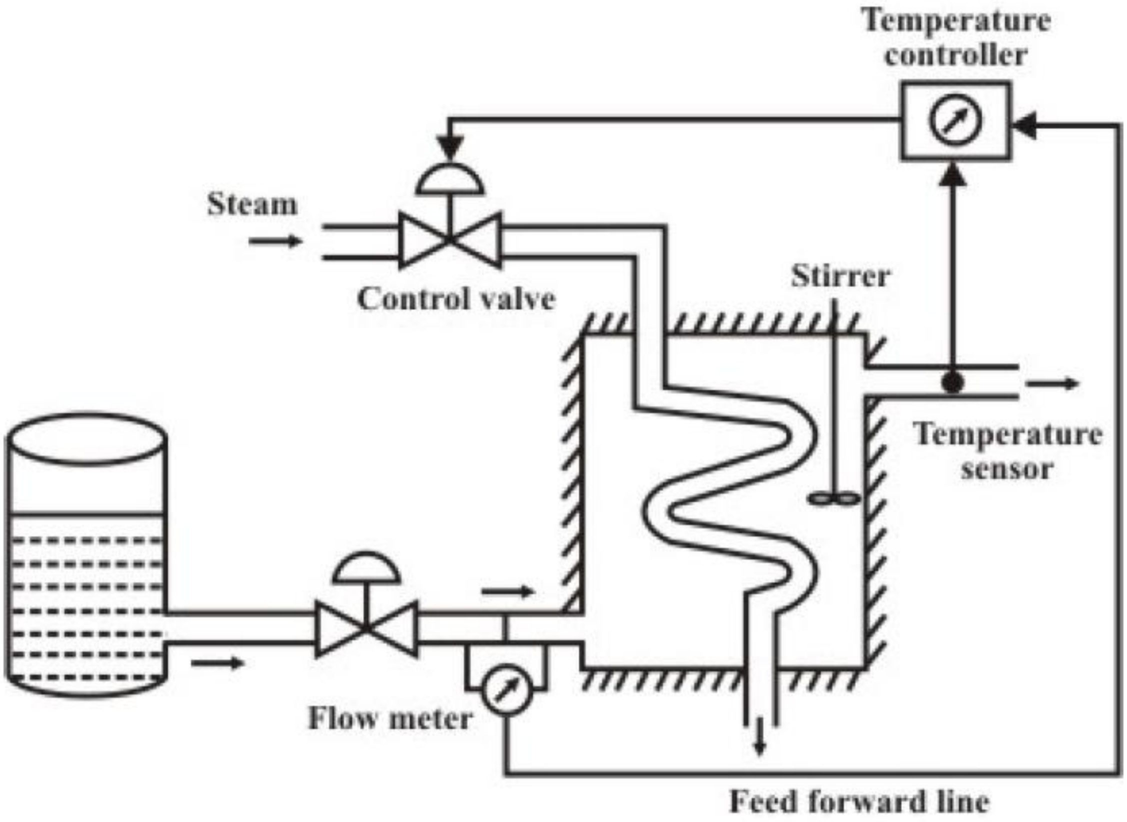


Fig. 1 Feed forward control action in a heat exchange.

Let us now draw the block diagram of the overall control operation of the system shown in Fig. 1. The block diagram representation is shown in Fig. 2.

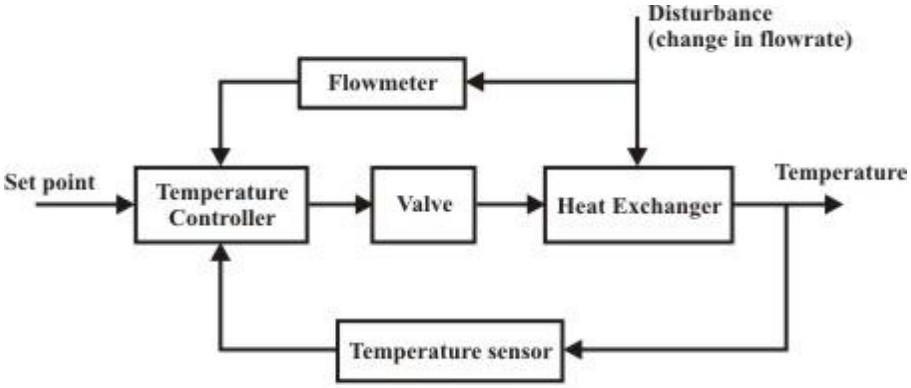


Fig. 2 Block diagram representation of the control action shown in Fig. 1

In general, the structure of the feedforward-feedback action in terms of the block diagram of transfer functions can be represented as shown in Fig. 3. Where,

$G(s)$ = Transfer function of the process (manipulating variable to output)

$G_n(s)$ = Disturbance transfer function (disturbance to output)

$G_c(s)$ = Transfer function of feedback controller

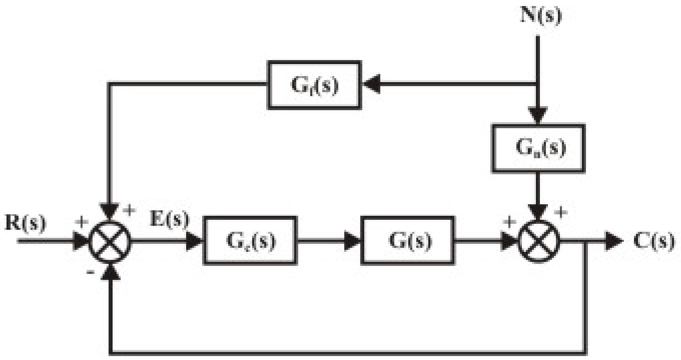


Fig. 3 Transfer function representation of the feedforward-feedback control action

$G_f(s)$ = Transfer function of the feedforward controller

So there are two controllers, one is the conventional feedback controller $G_c(s)$, while the other is the feedforward controller that is intended to nullify the effect of disturbance at the output. From Fig. 3, the overall output is:

$$\begin{aligned}\mathrm{C(s)} &= \mathrm{G_c}(s)\,\mathrm{G(s)}\,\mathrm{E(s)} + \mathrm{G_n}(s)\,\mathrm{N(s)}\\ &= \mathrm{G_c}(s)\,\mathrm{G(s)}\left[\mathrm{R(s)} - \mathrm{C(s)} + \mathrm{G_f N(s)}\right] + \mathrm{G_n(s)\,N(s)}\\ &= \mathrm{G_c}(s)\,\mathrm{G(s)}\left[\mathrm{R(s)} - \mathrm{C(s)}\right] + \left[\mathrm{G_c}(s)\,\mathrm{G(s)}\,\mathrm{G_f(s)} + \mathrm{G_n(s)}\right]\mathrm{N(s)}\end{aligned}$$

If we want to select the feedforard transfer function, such that, the effect of disturbance at the output is zero, then we require the co-efficient of N(s) in above equation to be set to zero. Thus,

$$\mathrm{G_c}(s)\,\mathrm{G(s)}\,\mathrm{G_f(s)} + \mathrm{G_n(s)} = 0$$

or,
$$\mathrm{G_f(s)} = -\frac{\mathrm{G_n(s)}}{\mathrm{G_c(s)\,G(s)}} \qquad (1)$$

It is to be noted that complete disturbance rejection can be obtained if the transfer functions $G_p(s)$ and $G_n(s)$ are known accurately, which is not possible in actual situation. As a result, the performance of the feedforward controller will deteriorate and complete disturbance rejection can not be achieved, through the effect can be reduced considerably. However the feedback controller would reduce the residual error due to imperfect feedforward control and at the output, the effect of imperfect cancellation may not be felt.

Ratio Control

Ratio control is a special type of feedforward control where the disturbance is measured and the ratio of the process output and the disturbance is held constant. It is mostly used to control the ratio of flow rates of two streams. Flow rates of both the stream are measured, but only one of them is controlled. There can be many examples of application of ratio control. Few examples are:

1. fuel-air ratio control in burners,
2. control of ratio of two reactants entering a reactor at a desired ratio,
3. maintaining the ratio of two blended streams constant in order to maintain the composition of the blend at the desired value.

There can be two schemes for achieving ratio control. The first scheme is shown in Fig. 5. In this configuration the ratio of flow rates of two streams is measured and compared with the desired ratio. The error is fed to the controller and the controller output is used to control the flow rate of stream B.

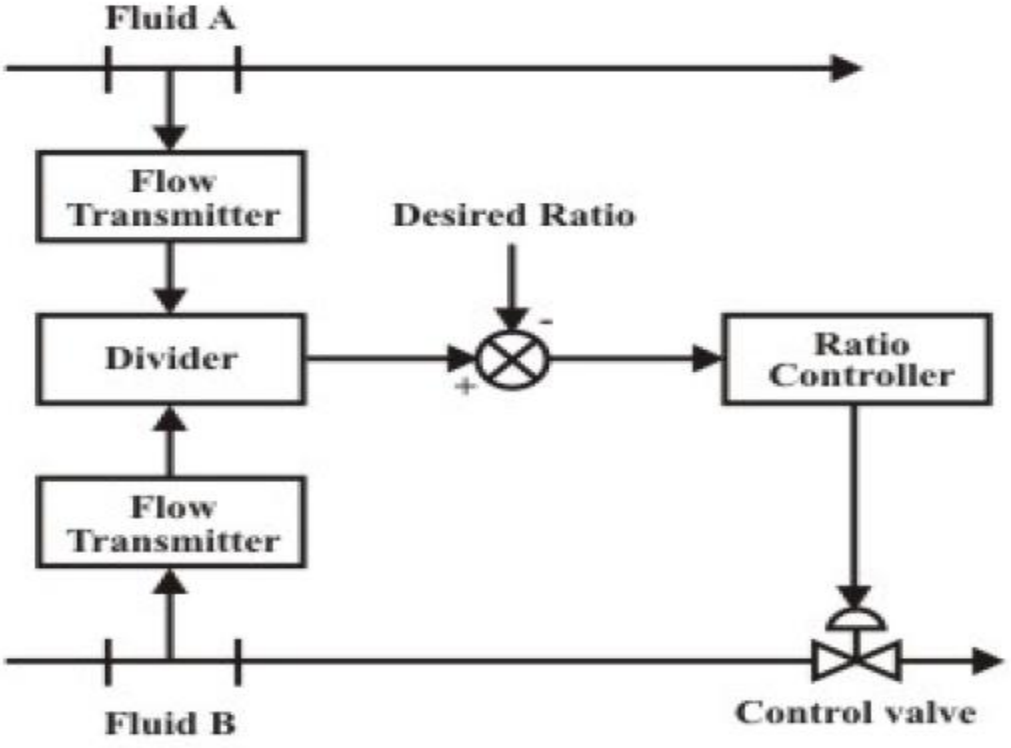


Fig. 5 A possible configuration for achieving ratio control

The second possible scheme for ratio control is shown in Fig. 6. Suppose the flow rate of fluid B has to be maintained at a constant fraction of flow rate of fluid A, irrespective of variation of flow rate of A (q_A). In this scheme the flow rate of fluid A is multiplied with the desired ratio (set externally) that gives the desired flow rate of fluid B. This is compared with the actual flow rate of fluid B and fed to the controller that operates the control valve.

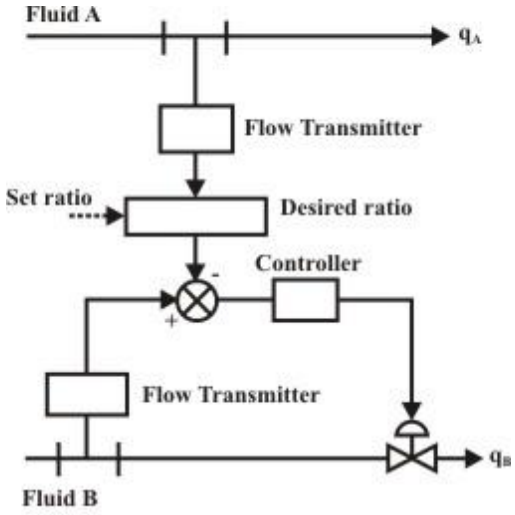


Fig. 6 Ratio control

Suppose that the above scheme (Fig. 6) is used for controlling the fuel-air ratio in a burner where airflow rate (fluid B) is controlled. But the desired ratio is also dependent on the temperature of the air. So an auxiliary measurement is needed to measure the temperature of air and set the desired ratio. Such a scheme is shown in Fig. 7.

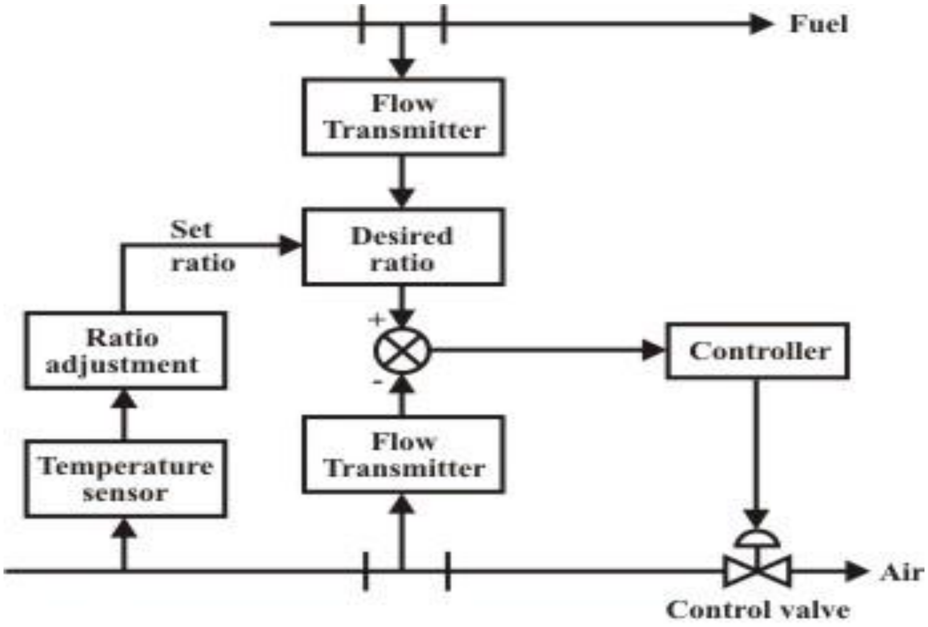
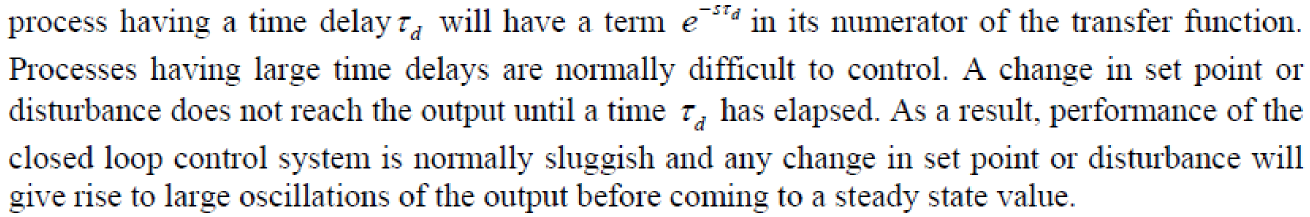


Fig. 7 Ratio control with ratio adjustment mechanism

The controllers in ratio control are usually P-I type. This is in order to achieve zero steady state error for maintaining the desired ratio. Derivative action is avoided because the flow is always noisy.

Special Control Structures: Predictive Control, Control of Systems with Inverse Response Predictive Control

It has already been mentioned earlier, how in a chemical process control transportation lag (time delay) comes into picture affecting the model and performance of the process.



process having a time delay τ_d will have a term $e^{-s\tau_d}$ in its numerator of the transfer function. Processes having large time delays are normally difficult to control. A change in set point or disturbance does not reach the output until a time τ_d has elapsed. As a result, performance of the closed loop control system is normally sluggish and any change in set point or disturbance will give rise to large oscillations of the output before coming to a steady state value.

In order to improve the closed loop performances of such time delay systems, O.J.M. Smith in 1957 first suggested a modification of conventional PID control schemes for processes having large time delay. The scheme for taking a predictive action in presence of transportation delay in the system is better known as *Smith Predictor*.

Let us consider that the transfer function of the process is given by:

$$G_p(s)=e^{-s\tau_d}G(s) \qquad (1)$$

where G(s) represents the system model without the delay. The basic scheme for Smith Predictor is shown in Fig. 1. Here G(s) is the conventional PID controller designed fro the process G(s). If the system model is exact, the output of the comparator-A would be zero and the outer loop can be ignored. The closed loop system can be simplified as the block diagram shown in Fig. 2. Since the time delay part is absent here, the controller will see the effect of control action much earlier and the sluggish response of the system will improve. The outer loop comes into play if the model of the process is not exact (normally that is expected). Fig. 1 can be redrawn as shown in Fig. 3 with the actual controller is shown inside the dashed line.

The performance of the Smith Predictor depends heavily on the actual knowledge of the plant model. Any change in the plant characteristics (particularly dead time) should be instantly taken care of, or otherwise, can cause deterioration of its performance. Besides, the hardware implementation of the controller using analog circuits is difficult. But the control scheme can be implemented with relative ease with a digital controller.

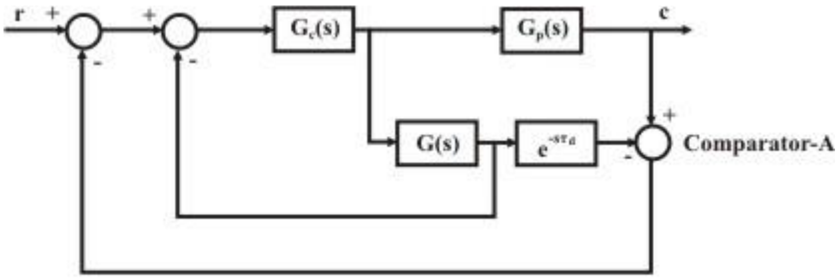


Fig. 1 Smith Predictor.

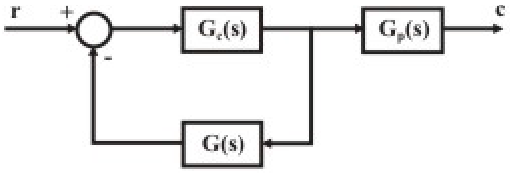


Fig. 2 Equivalent block diagram of Fig. 1 when the model is exact.

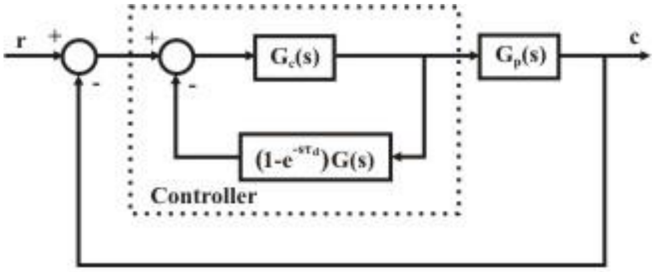


Fig. 3 Equivalent scheme of controller.

Application of Predictive Control in Gage Control of Steel Rolling Mills

In this section we shall give two examples of Predictive Control in control of gage thickness in steel rolling mills. *Automatic Gage Control* (AGC) is the most important control scheme in a rolling mill. Its main objective is to maintain the thickness of the sheet steel coming out of a rolling mill constant. The basic feedback scheme for AGC in a single stand rolling mill is shown in Fig. 4. The gage is controlled by varying the gap between the rollers in a stand. In fact, there are a number of stands in a rolling milling and the rolling is carried out in stages. Hydraulic actuators are normally used for roll gap adjustment. In this scheme, the gage of the strip is measured at the exit stand and compared with the reference gage command. The error is amplified to operate the servo valve of the hydraulic actuator. The basic control scheme shown here is P-type. Nucleonic detector is used for measurement of gage thickness at the exit stand. But because of the fact that the thickness measuring device is installed at a distance from the roll, it will introduce transportation lag in the closed loop system. There will be considerable time lag to detect the variation of sheet thickness at the roll stand and that will lead to oscillatory and unsatisfactory behaviour of closed loop gage control.

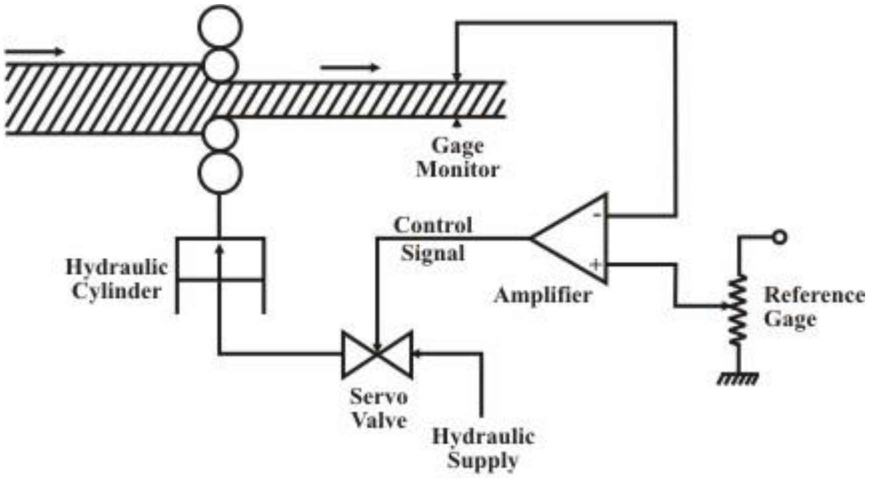


Fig. 4 Basic feedback loop for gage control.

In order to improve the performance of the closed loop scheme, Predictive Control mechanism are added with the basic feedback scheme. Here the actual roll gap (and subsequently the gage thickness) is estimated at the roll stand. This can be achieved by two possible methods:

- estimating the roll gap
- estimating the gap thickness from constant mass flow principle.

In both the cases, the gap is estimated through an approximate model of the rolling stand.

Smith Predictor by estimating the roll gap

In this method, the gage thickness is predicted by estimating the roll gap using the expression:

$$h = C_0 + \frac{P}{K_s} \qquad (2)$$

where h is the exit thickness of the rolled product, C_0 is the no-load roll gap, P is the roll force in the hydraulic actuator and K_s is the structural stiffness of the mill. The overall scheme can be represented as in Fig. 5.

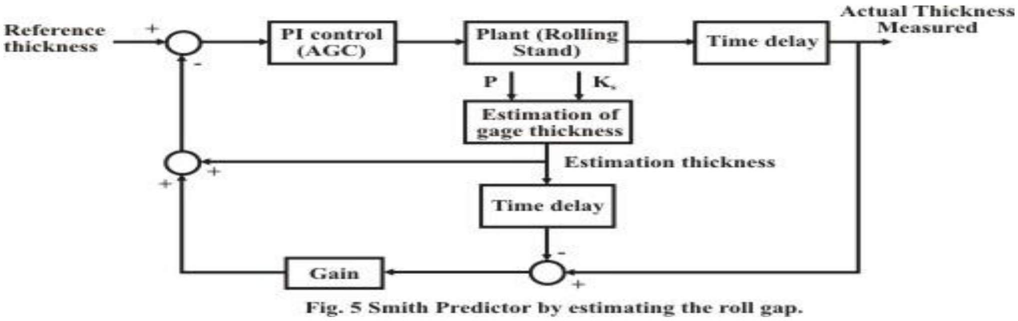


Fig. 5 Smith Predictor by estimating the roll gap.

Smith Predictor based on Constant Mass Flow principle

The thickness of the strip can also be predicted from constant mass flow principle. Consider a multi-stand mill. Let h_{i-1} and v_{i-1} be the thickness and the velocity of the strip at the inlet of the i-th stand (i.e. outlet at the (i-1)th stand). So h_i and v_i are the corresponding parameters at the outlet of the i-th stand. Then from the principle of conservation of mass, we can write,

Rate of mass flow in = Rate of mass flow out

Considering the width of the strip to be same before and after the stand, we can write:

$$h_{i-1}\, v_{i-1} = h_i\ \ v_i$$

or,

$$h_i = \frac{h_{i-1}\, v_{i-1}}{v_i} \qquad (3)$$

The velocities at the inlet and outlet of the stand can be measured by using a Laser Doppler velocity meter, or in a more conventional way by measuring the roll speeds at the two stands. However, some more correction factors are incorporated while predicting the exact gage thickness using constant mass flow principle. A scheme similar to Fig. 5 can be used here in the Smith Predictor scheme.

Systems with Inverse Response

We have seen different types of system response so far: first order system, second order system, system with time lag etc. Another type of system can also be classified with its typical step response pattern: system with inverse response. It is essentially a system whose transfer function is having a zero on the right half plane. This type of system is also called, *nonminimum phase system*.

Example of a system with inverse response

Before going into the details, let us consider a simple example of a system with inverse response. Consider the dynamic characteristics of a boiler drum in a water tube boiler of a steam power plant. High-pressure feedwater is pumped to the drum. Water from the drum circulates through the boiler tubes, gets heated and is converted to steam. This steam again comes back to the drum and subsequently is taken out through the steam flow line to the turbine. So the drum is filled up partially with water and partially with steam, both at high pressure. It is very important to control the water level of the drum at a desired level, by controlling the feedwater flow, with the varying demand of steam. The schematic arrangement can be shown as in Fig. 6.

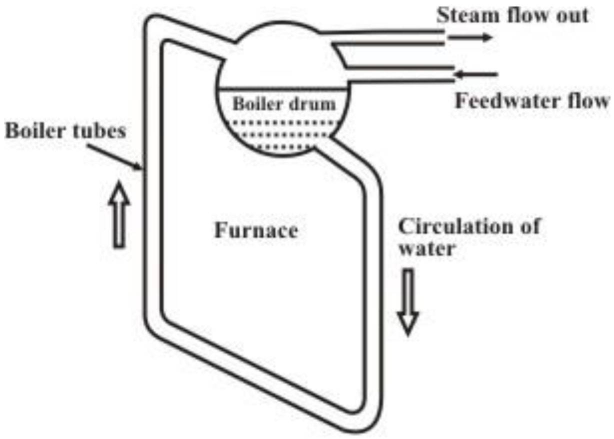


Fig. 6 A boiler drum.

The instantaneous level of water at the boiler drum is decided by the steam flow rate and the feedwater flow rate and it would reach a steady state when both are equal. Now suppose, the steam flow rate suddenly decreases, the feedwater flow rate remaining constant. At a first glance, it would appear that the drum level should rise. But actually the drum level will initially drop for some time before it rises and reaches a steady value. This is because of the fact that drop in steam flow rate will initially cause the rise in steam pressure in the drum. Due to the rise in pressure, the bubbles present inside the water in the drum will momentarily *shrink*. This will cause the temporary fall in the drum level. Similarly, for a sudden increase in steam flow rate, the drum level will momentarily *swell* before it drops down to a steady state vale. A typical response curve of the drum level due to the sudden fall in steam level is plotted in Fig. 7.

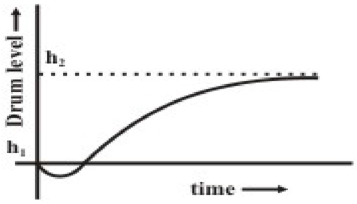


Fig. 7 Change in drum level due to sudden drop in steam flow rate.

Transfer function of a system with inverse response

The above response curve can be looked into as a combination of two first order process responses in opposition as shown in Fig. 8. The block diagram representation is shown in Fig. 9.

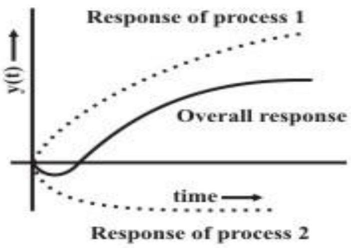


Fig. 8 Inverse response resulting from two first order opposing systems.

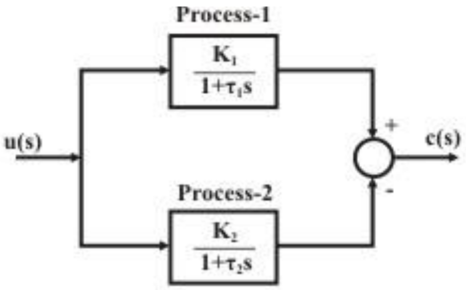


Fig. 9 Block diagram of two opposing first order systems.

The overall output response can be written as:

$$c(s)=\left[\frac{K_1}{1+\tau_1 s}-\frac{K_2}{1+\tau_2 s}\right]u(s)$$

or. $$c(s)=\frac{(K_1\tau_2-K_2\tau_1)s+(K_1-K_2)}{(1+\tau_1 s)(1+\tau_2 s)}u(s) \qquad (4)$$

Let us assume

$$\frac{\tau_1}{\tau_2}>\frac{K_1}{K_2}>1 \qquad (5)$$

Then for a step input, the response of process-2 will dominate during the initial phase, but ultimately, process- 1 will dominate in the steady state. It can also be seen that under these two conditions, the system has a zero on the right half of s-plane (it is evident from the fact that $K_1>K_2$ and $K_1\tau_2<K_2\tau_1$). The inverse response is a typical characteristics of a *nonminimum phase system*, i.e. a system with an unstable zero.

Control of a System with Inverse Response

It is evident that control of a system with inverse response is difficult as in the case of control of a system with large time lag. For a closed loop time lag system with a simple feedback controller, the controller does not see any effect of control action till a time τ_d has elapsed. On the other hand, for an inverse system, the controller will see an opposite effect to the expected one. So naturally, some special arrangement, similar to a Smith Predictor scheme is needed for control of inverse systems. Such an arrangement is shown in Fig. 10. This also requires an *apriori* knowledge of the system model.

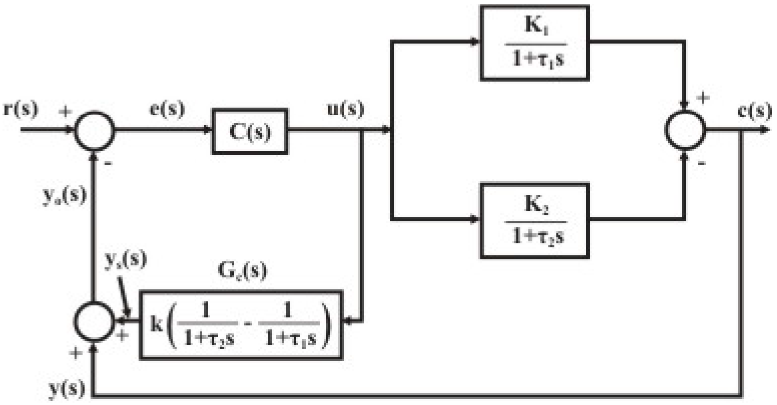


Fig. 10 A typical scheme for controlling a system with inverse response.

From Fig. 10, if we neglect the additional compensator loop with Gc(s) then it is evident that the feedback information received,

$$y(s) = C(s)\frac{(K_1\tau_2\text{-}K_2\tau_1)s+(K_1\text{-}K_2)}{(1+\tau_1 s)(1+\tau_2 s)}e(s) \tag{6}$$

that has a zero in the right half plane. To eliminate the effect of inverse response, one additional measurement signal must be added that excludes the information of inverse response. That can be achieved by the loop through the compensator Gc(s), that gives an additional output

$$\begin{aligned} y_s(s) &= C(s)G_c(s)e(s) \\ &= C(s)\,k\left(\frac{1}{1+\tau_2 s}-\frac{1}{1+\tau_1 s}\right)e(s) \end{aligned} \tag{7}$$

Combining (6) and (7), we can have,

$$\begin{aligned} y_0(s) &= y(s)+y_s(s) \\ &= C(s)\frac{(K_1\tau_2\text{-}K_2\tau_1)s+k(\tau_1\text{-}\tau_2)s+(K_1\text{-}K_2)}{(1+\tau_1 s)(1+\tau_2 s)}e(s) \end{aligned} \tag{8}$$

Now the loop transfer function, i.e. transfer function between y_o and e, will have a zero on the left half of s-plane, if the coefficient of s in the numerator of (8) is positive, i.e. using (6)

$$k > \frac{K_2\tau_1\text{-}K_1\tau_2}{\tau_1\text{-}\tau_2} \tag{9}$$

Thus it is evident that due the presence of the compensator, the inverse response behaviour (i.e. r.h.p zero) will not be felt by the controller. It can be also easily seen that the compensator in Fig. 10 has a similar configuration as in Fig. 6 for smith predictor. In other words, the compensator in Smith Predictor predicts the dead time behaviour of the process, while in the present case; the compensator $G_c(s)$ predicts the inverse behaviour of the process. The basic controller is normally chosen of P-I type.

UNIT – III

PROGRAMMABLE LOGIC CONTROL SYSTEMS

Introduction to Sequence/Logic Control and Programmable Logic Controllers

Many control applications do not involve analog process variables, that is, the ones which can assume a continuous range of values, but instead variables that are set valued, that is they only assume values belonging to a finite set. The simplest examples of such variables are binary variables, that can have either of two possible values, (such as 1 or 0, on or off, open or closed etc.). These control systems operate by turning on and off switches, motors, valves, and other devices in response to operating conditions and as a function of time. Such systems are referred to as sequence/logic control systems. For example, in the operation of transfer lines and automated assembly machines, sequence control is used to coordinate the various actions of the production system (e.g., transfer of parts, changing of the tool, feeding of the metal cutting tool, etc.).

Typically the control problem is to cause/ prevent occurrence of

♦ particular values of outputs process variables

♦ particular values of outputs obeying timing restrictions

♦ given sequences of discrete outputs

♦ given orders between various discrete outputs

Note that some of these can also be operated using analog control methods. However, in specific applications they may be viewed as discrete control or sensing devices for two reasons, namely,

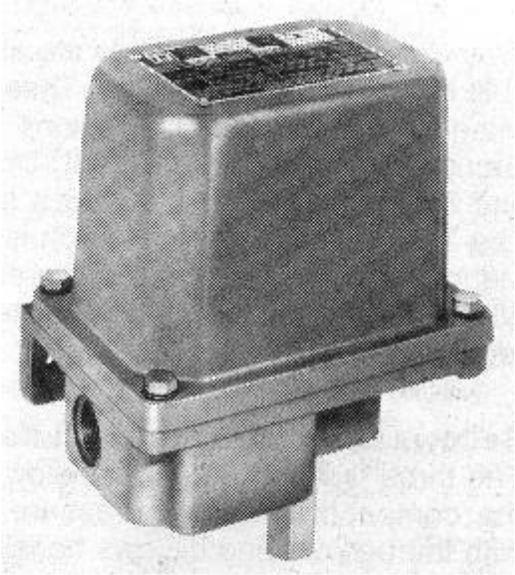
A. The inputs to these devices only belong to two specific sets. For example in the control of a reciprocating conveyor system, analog motor control is not applied. Simple on-off control is adequate. Therefore for this application, the motor-starter actuation system may be considered as discrete.

B. Often the control problem considered is supervisory in nature, where the problem is provide different types of supervisory commands to automatic control systems, which in turn carry out analog control tasks, such that over all system operating modes can be maintained and coordinated to achieve system objectives.

Industrial Example of Discrete Sensors and Actuators

There are many industrial sensors which provide discrete outputs which may be interpreted as the presence/absence of an object in close proximity, passing of parts on a conveyor, For example, tables x.y and a.b below show a set of typical sensors which provide a discrete set of output corresponding to process variables. **Type**	**Signal**	**Remark**
Switch	**Binary Command**	**External** **Input Device**
Limit switch	**Position**	**Feedback Sensor**

		Device
Thumbwheel switch	**Set valued Command**	**External Input Device**
Thermostat	**Temperature Level**	**Feedback Sensor Device**
Photo cell	**Position of objects**	**Feedback Sensor Device**
Proximity detector	**Position of objects**	**Feedback Sensor Device**
Push button	**Command (unlatched)**	**External Input Device**



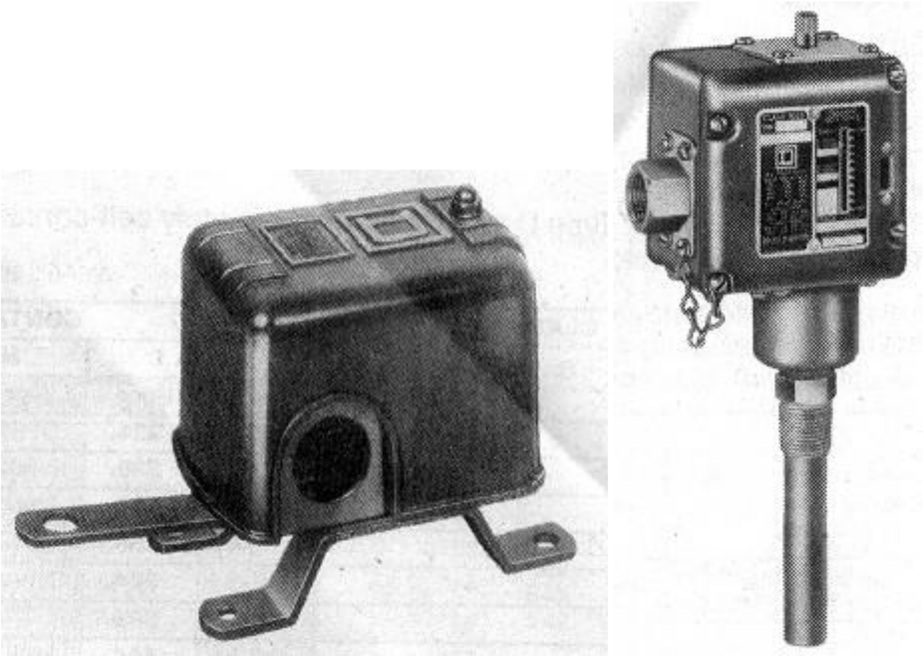


Fig. 18.1 Example Industrial Discrete Input and Sensing Devices: Type	Output Quantity	Energy Source
Relay, Contactor	**voltage**	electrical
Motor Starter	**motion**	electrical
Lamp	**indication**	electrical
Solenoid	**motion**	electrical
On-off Flow Control valve	**Flow**	pneumatic, hydraulic
Directional Valves	**Hydraulic Pressure**	pneumatic, hydraulic

Industrial Example

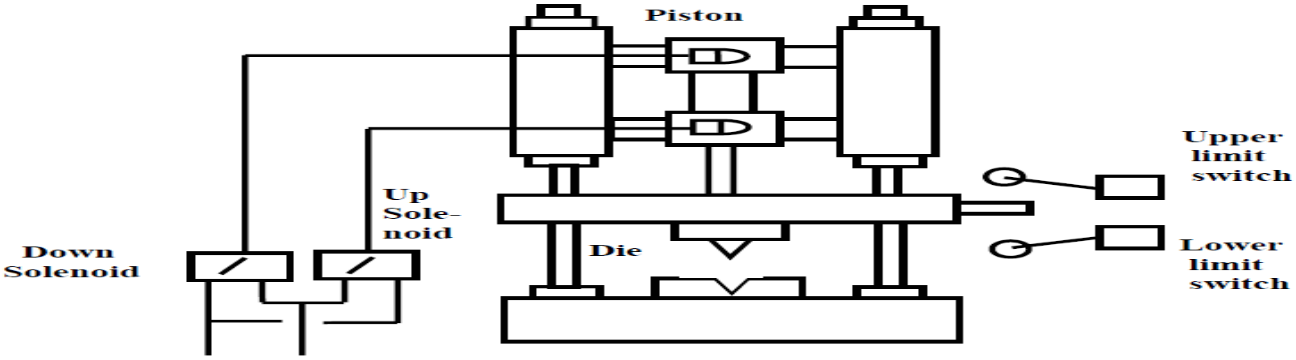


Fig. 18.2 An Industrial Logic Control Example

The die stamping process is shown in figure below. This process consists of a metal stamping die fixed to the end of a piston. The piston is extended to stamp a work piece and retracted to allow the work piece to be removed. The process has 2 actuators: an up solenoid and a down solenoid, which respectively control the hydraulics for the extension and retraction of the stamping piston and die. The process also has 2 sensors: an upper limit switch that indicates when the piston is fully retracted and a lower limit switch that indicates when the piston is fully extended. Lastly, the process has a master switch which is used to start the process and to shut it down. The control computer for the process has 3 inputs (2 from the limit sensors and 1 from the master switch) and controls 2 outputs (1 to each actuator solenoid).

The desired control algorithm for the process is simply as follows. When the master switch is turned on the die-stamping piston is to reciprocate between the extended and retracted positions, stamping parts that have been placed in the machine. When the master switch is switched off, the piston is to return to a shutdown configuration with the actuators off and the piston fully retracted

Comparing Logic and Sequence Control with Analog Control

The salient points of difference between Analog Control and Logic/Sequence control are presented in the table below.

Issue	**Logic/Sequence Control**	**Analog Control**
Model	Logical State-Transition	Numerical Different ial/Difference Eqn
	Simple Model/Easy to build	Complex Model/Hard to build
	Infrequent	Liable to change
Signal **Temporal Property**	Signal range/status	Signal value
	(Timed) sequence	(Timed)Function/Trajectory
Control Redesign/Tuning	On-off/logical	linear/non linear analog
	Supervisory	automatic
	Open/Closed Loop	Open/Closed Loop
	Infrequent	Tuning needed

Programmable Logic Controllers (PLC)

A modern controller device used extensively for sequence control today in transfer lines, robotics, process control, and many other automated systems is the **Programmable Logic Controller** (PLC). In essence, a PLC is a special purpose industrial microprocessor based real-time computing system, which performs the following functions in the context of industrial operations

- Monitor Input/Sensors
- Execute logic, sequencing, timing, counting functions for Control/Diagnostics
- Drives Actuators/Indicators
- Communicates with other computers

Some of the following are advantages of PLCs due to standardized hardware technology, modular design of the PLCs, communication capabilities and improved development program development environment:

- Easy to use to simple modular assembly and connection;
- Modular expansion capacity of the input, outputs and memory;
- Simple programming environments and the use of standardized task libraries and debugging aids;
- Communication capability with other programmable controllers and computers.

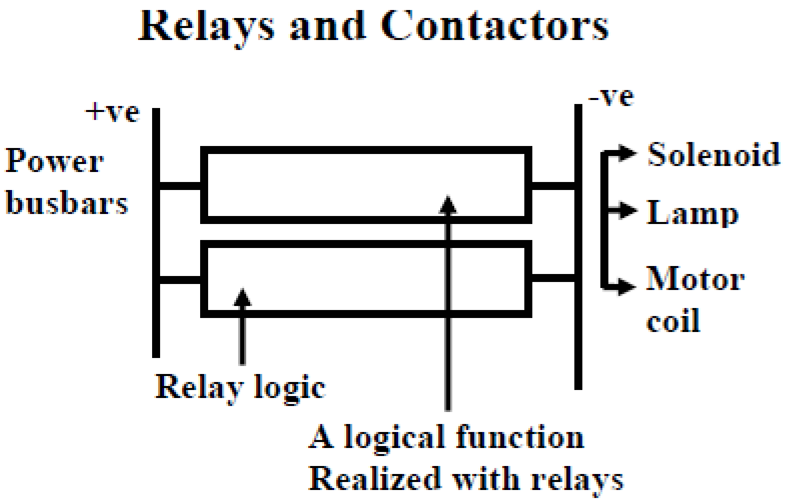
Evolution of the PLC

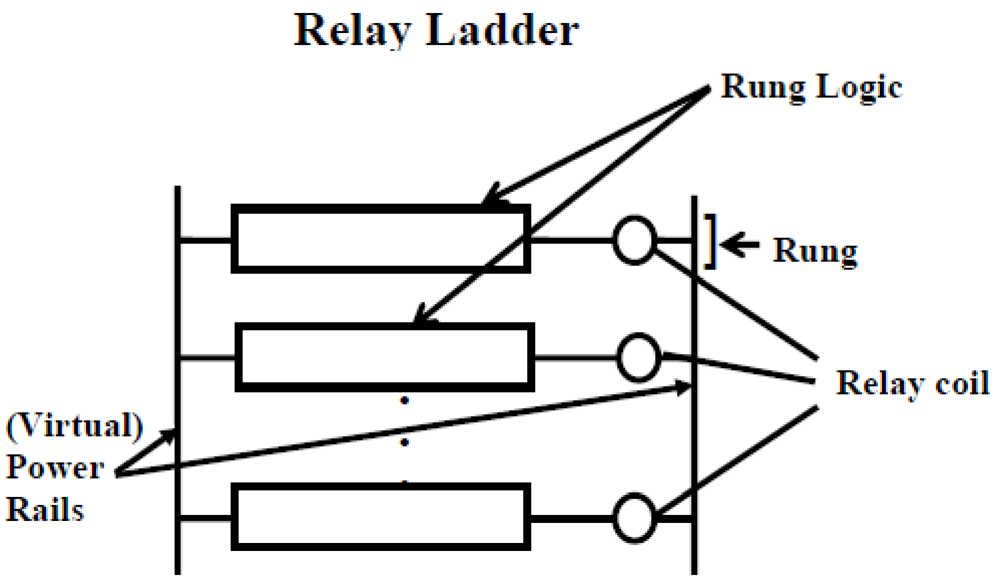
Before the advent of microprocessors, industrial logic and sequence control used to be performed using elaborate control panels containing electromechanical or solid-state relays, contactors and switches, indicator lamps, mechanical or electronic timers and counters etc., all hardwired by complex and elaborate wiring. In fact, for many applications such control panels are used even today. However, the development of microprocessors in the early 1980's quickly led to the development of the PLCs, which had significant advantages over conventional control panels. Some of these are:

- Programming the PLC is easier than wiring physical components; the only wiring required is that of connecting the I/O terminals.
- The PLC can be reprogrammed using user-friendly programming devices. Controls must be physically rewired.
- PLCs take up much less space.
- Installation and maintenance of PLCs is easier, and with present day solid-state technology, reliability is grater.
- The PLC can be connected to a distributed plant automation system, supervised and monitored.
- Beyond a certain size and complexity of the process, a PLC-based system compare favorably with control panels.
- Ability of PLCs to accept digital data in serial, parallel and network modes imply a drastic reduction in plant sensor and actuator wirings, since single cable runs to remote terminal I/O units can be made. Wiring only need to be made locally from that point.

- Special diagnostic and maintenance modes for quick troubleshooting and servicing, without disrupting plant operations.

However, since it evolved out of relay control panels the PLCs adopted legacy concepts, which were applicable to such panels. To facilitate maintenance and modification of the physically wired control logic, the control panel was systematically organized so that each control formed a rung much like a rung on a ladder. The development of PLCs retained the ladder logic concept where control circuits are defined like rungs on a ladder where each rung begins with one or more inputs and each rung usually ends with only one output. A typical PLC ladder structure is





Application Areas

Programmable Logic Controllers are suitable for a variety of automation tasks. They provide a simple and economic solution to many automation tasks such as

- Logic/Sequence control
- PID control and computing
- Coordination and communication
- Plant start-up, shut-down

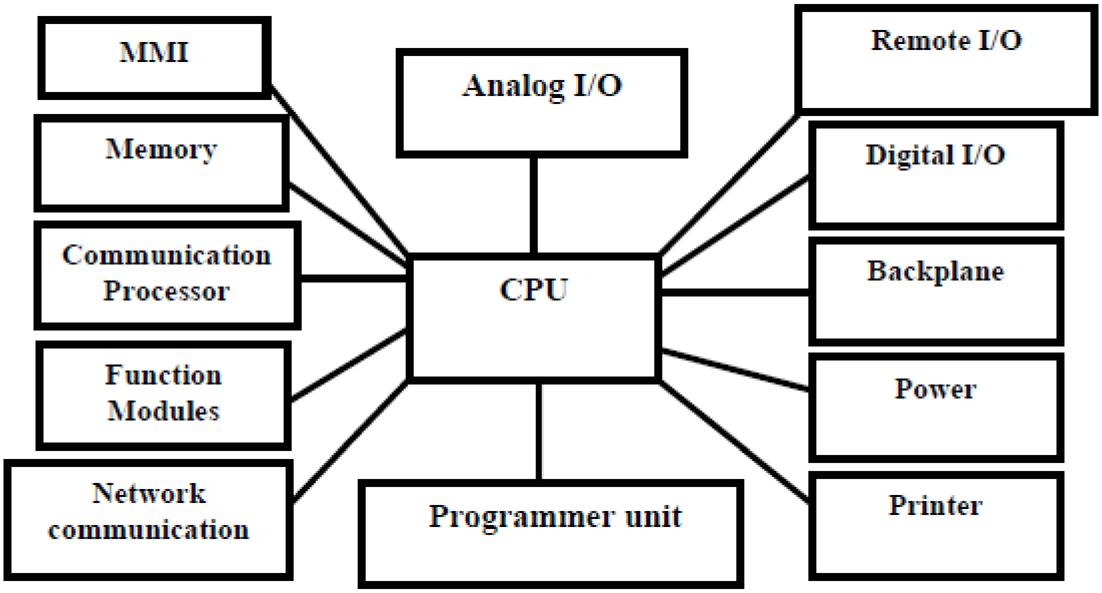
Any manufacturing application that involves controlling repetitive, discrete operations is a potential candidate for PLC usage, e.g. machine tools, automatic assembly equipment, molding and extrusion machinery, textile machinery and automatic test equipment. Some typical industrial areas that widely deploy PLC controls are named in Table x.y. The list is only illustrative and by no means exhaustive.

Chemical/ Petrochemical	**Metals**	**Manufacturing/Machining**
Batch process	Blast Furnace	Material Conveyors, Cranes
Pipeline Control	Continuous Casting	Assembly
Weighing, Mixing	Rolling Mills	Milling, Grinding, Boring

Finished Product Handling
Water/ Waste Treatment
Soaking Pit
Steel Melting Shop
Plating, Welding, Painting
Molding/ casting/forming

Architecture of PLCs

The PLC is essentially a microprocessor-based real-time computing system that often has to handle significant I/O and Communication activities, bit oriented computing, as well as normal floating point arithmetic. A typical set of components that make a PLC System is shown in Fig.



Central controller

The central controller (CC) contains the modules necessary for the main computing operation of the Programmable controller (PC). The central controller can be equipped with the following:

- ♦ Memory modules with RAM or EPROM (in the memory sub modules) for the program (main memory);
- ♦ Interface modules for programmers, expansion units, standard peripherals etc;
- ♦ Communications processors for operator communication and visualization, communication with other systems and configuring of local area networks.

A bus connects the CPUs with the other modules.

Central Processing units

The CPUs are generally microprogrammed processors sometimes capable of handling multiple data width of either 8, 16 or 24 bits. In addition some times additional circuitry, such as for bit processing is provided, since much of the computing involves logical operations involving digital inputs and auxiliary quantities. Memory with battery backup is also provided for the following:

♦ Flags (internal relays), timers and counters;

♦ Operating system data

♦ Process image for the signal states of binary inputs and outputs.

The user program is stored in memory modules. During each program scan, the processor reads the statement in the program memory, executes the corresponding operations. The bit processor, if it exists, executes binary operations. Often multiple central controllers can be configured in hot standby mode, such that if one processor fails the other can immediately pick up the computing tasks without any failure in plant operations.

Communications processors

Communications processors autonomously handle data communication with the following:

♦ Standard peripherals such as printers, keyboards and CRTs,

♦ Supervisory Computer Systems,

♦ Other Programmable controllers,

The data required for each communications processors is stored in a RAM or EPROM sub module so that they do not load the processor memories. A local area network can also be configured using communications processors. This enables the connection of various PLCs over a wide distance in various configurations. The network protocols are often proprietary. However, over the last decade, interoperable network protocol standards are also supported in modern PLCs.

Program and Data memory

The program and data needed for execution are stored in RAM or EPROM sub modules. These sub modules are plugged into the processors. Additional RAM memory modules can also be connected.

Expansion units

Modules for the input and output of signals are plugged into expansion units. The latter are connected to the central controller via interface modules. Expansion units can be connected in two configurations.

A. Centralized configuration

The expansion units (EU) are located in the same cabinet as the central controllers or in an adjacent cabinet in the centralized configuration, several expansion units can be connected to one central controller. The length of the cable from the central controller to the most distant expansion unit is often limited based on data transfer speeds.

B. Distributed configuration The expansion units can be located at a distance of up to 1000 m from the central controller. In the distributed configuration, up to 16 expansion units can be connected to one central controller. Four additional expansion units can be connected in the centralized configuration to each distributed expansion unit and to the central controller.

Input/Output Units

A host of input and output modules are connected to the PLC bus to exchange data with the processor unit. These can be broadly categorized into Digital Input Modules, Digital Output Modules, Analog Input Modules, Analog Output Modules and Special Purpose Modules.

Digital Input Modules

The digital inputs modules convert the external binary signals from the process to the internal digital signal level of programmable controllers.

Digital Output Modules

The digital output modules convert the internal signal levels of the programmable controllers into the binary signal levels required externally by the process.

Analog Input Modules

The analog input modules convert the analog signals from the process into digital values which are then processed by the programmable controller.

Analog Output Modules

The analog output modules convert digital values from the programmable controller into the analog signals required by the process.

Special Purpose Modules

These may include special units for:

- High speed counting
- High accuracy positioning
- On-line self-optimizing control
- Multi axis synchronisation, interpolation

These modules contain additional processors, and are used to relieve the main CPU from the high computational loads involved in the corresponding tasks.

Programmers

External programming units can be used to download programs into the program memory of the CPU. The external field programmers provide several software features that facilitate program entry in graphical form. The programmers also provide comprehensive aids for debugging and execution monitoring support logic and sequence control systems. Printer can be connected to the programmers for the purpose of documenting the program. In some cases, special programming packages that run on Personal Computers, can also be used as programming units. There are two ways of entering the program:

A. Direct program entry to the program memory (RAM) plugged into the central controller. For this purpose, the programmer is connected to the processor or to the programmer interface modules.

B. Programming the EPROM sub modules in the programmer without connecting it to the PC (off-line). The memory sub modules are then plugged into the central controller.

Other Miscellaneous Units

Other units such as Power Supply Units, Bus Units etc. can also be connected to the PLC system.

The Software Environment and Programming of PLCs

Structure of a PLC Program

There are several options in programming a PLC, as discussed earlier. In all the options the common control of them is that PLC programs are structured in their composition. i.e. they consist of individual, separately defined programs sections which are executed in sequence. These programs sections are called ‘blocks”. Each program section contains statements. The blocks are supposed to be functionally independent. Assigning a particular (technical) function to a specific block, which has clearly defined and simple interfaces with other blocks, yields a clear program structure. The testing of such programs in sections is substantially simplified.

Various types of blocks are available according to the function of the program section.

In general the major part of the program is contained in blocks that contain the program logic graphically represented. For improved modularity, these blocks can be called in a sequence or in nested configurations.

Special Function Blocks, which are similar to application library modules, are used to realize either frequently reoccurring or extremely complex functions. The function block can be “parameterized”.

The interface to the operating system of the PLC, which are similar to the system calls in application programming for Personal Computers, are defined in special blocks. They are only called upon by the system program for particular modes of execution and in the case of the faults.

Function blocks are also used where the realization of the logic control STEP 5 statements can’t be carried out graphically. Similarly, individual steps of a control sequence can be programmed into such a block and reused at various points in a program or by various programs. PLC manufacturers offer standard functions blocks for complex functions, already tested and documented. With adequate expertise the user can produce his own function blocks. Some very common function blocks (analog input put, interface function blocks for communication processors and others) may be integrated as standard function blocks and supported by the operating system of the PLC.

Users can also define separate data blocks for special purposes, such as monitoring, trending etc., and perform read/write on such areas. Such facilities of structured programming result in programs, which are easier to read, write, debug and maintain.

Program Execution

There are different ways and means of executing a user program. Normally a cyclic execution program is preferred and this cyclic operators are given due priorities. Program processing in a PLC happens cyclically with the following execution:

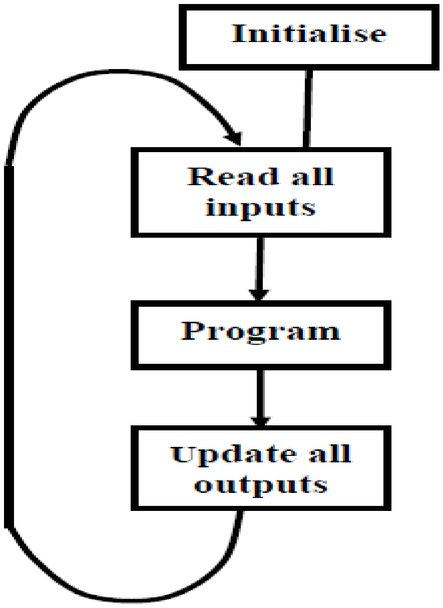
1. After the PLC is initialised, the **processor** reads the individual inputs. This status of the input is stored in the process- image input table (**PII**).

2. This processor processes the program stored in the program memory. This consists of a list of logic functions and instructions, which are successively processed, so that the required input information will already be accessed before the read in PII and the matching results are written into a process-image output table (**PIQ**). Also other storage areas for counters, timers and memory bits will be accessed during program processing by the processor if necessary.

3. In the third step after the processing of the user program, the status from the PIQ will transfer to the outputs and then be switched on and/or off. Afterwards it begins the execution of the next cycle from step 1.

The same cyclic process also acts upon an RLL program.

The time required by the microprocessor to complete one cycle is known as the **scan time**. After all rungs have been tested, the PLC then starts over again with the first rung. Of course the scan time for a particular processor is a function of the processor speed, the number of rungs, and the complexity of each rung.



Interrupt Driven and Clock Driven Execution Modes

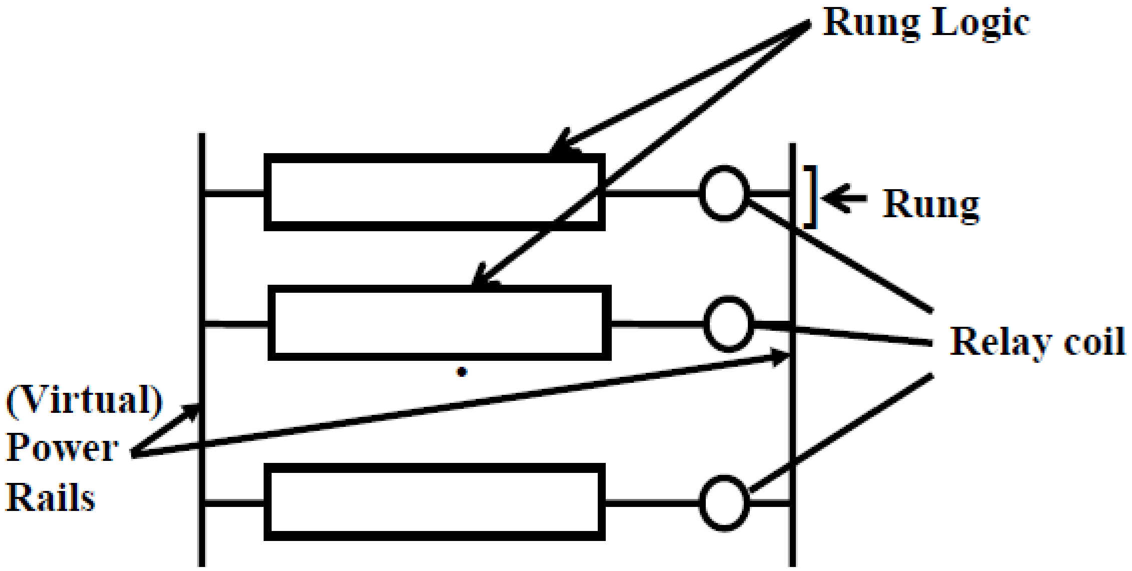
A cyclically executing program can however be interrupted by a suitably defined signal resulting in an interrupt driven mode of program execution (when fast reaction time is required). If the interrupting signal occurs at fixed intervals we can also realized time synchronous execution (i.e. with closed loop control function). The cyclic execution, synchronized by a real time clock is the most common program structure for a PLC.

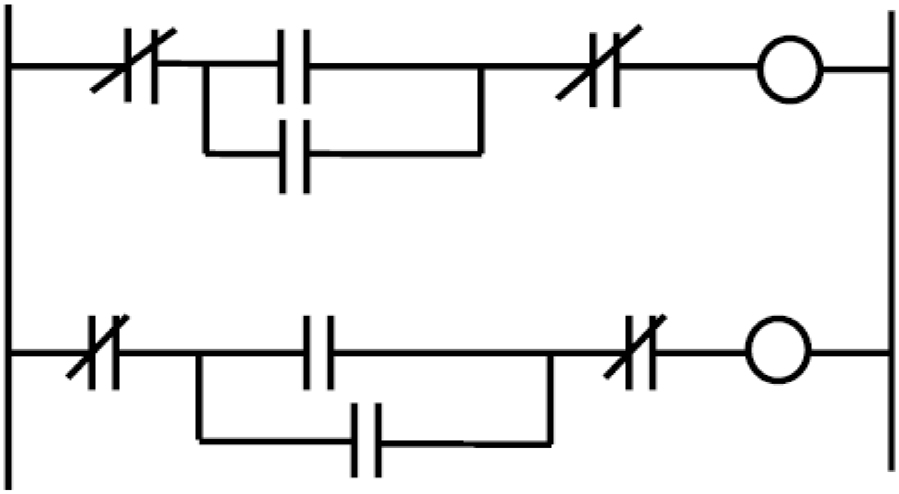
Similarly, programmers can also define error-handling routines in their programs. Specific and defined error procedures are then invoked if the PLC operating system encounters fault of given types during execution.

Programming Languages

PLC programs can be constructed using various methods of representation. Some of the common ones are, described below.

The Relay Ladder Logic (RLL) Diagram A Relay Ladder Logic (RLL) diagram, also referred to as a Ladder diagram is a visual and logical method of displaying the control logic which, based on the inputs determine the outputs of the program. The ladder is made up of a series of "rungs" of logical expressions expressed graphically as series and parallel circuits of relay logic elements such as contacts, timers etc. Each rung consist of a set of inputs on the left end of the rung and a single output at the right end of each rung.

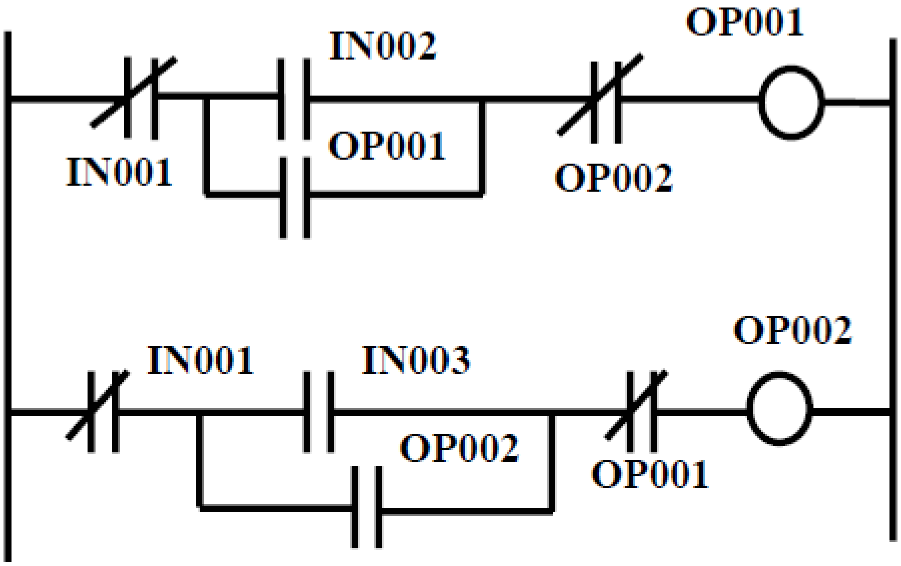




RLL Programming Paradigms: Merits and Demerits

For the programs of small PLC systems, RLL programming technique has been regarded as the best choice because a programmer can understand the relations of the contacts and coils intuitively. Additionally, a maintenance engineer can easily monitor the operation of the RLL program on its graphical representation because most PLC manufacturers provide an animated display that clearly identifies the states of the contacts and coils. Although RLL is still an important language of IEC 1131-3, as the memory size of today's PLC systems increases, a large-sized RLL program brings some significant problems because RLL is not particularly suitable for the well-structured programming: It is difficult to structure an RLL program hierarchically.

Example: Forward Reverse Control



This example explains the control process of moving a motor either in the forward direction or in the reverse direction. The direction of the motor depends on the polarity of the supply. So in order to control the motor, either in the forward direction or in the reverse direction, we have to provide the supply with the corresponding polarity. The Fig 19.2 depicts the procedure to achieve this using Relay Ladder Logic. Here, the Ladder consists of two rungs corresponding to forward and reverse motions.

The rung corresponding to forward motion consists of

1. A normally closed stop push-button (IN001),
2. A normally opened forward run push-button (IN002) in parallel with a normally opened auxillary contact(OP001),
3. A normally closed auxillary contact(OP002) and
4. The contacter for coil(OP001).

Similarly, the rung corresponding to reverse motion consists of

1. A normally closed stop push-button (IN001),
2. A normally opened forward run push-button (IN003) in parallel with a normally opened auxillary contact(OP002),
3. A normally closed auxillary contact(OP001) and
4. The contacter for coil(OP002).

Operation: The push-buttons(PB) represented by IN--- are real input push-buttons, which are to be manually operated. The auxillary contacts are operated through program. Initially the machine is at standstill, no voltage supply is present in the coils, and the PBs are as shown in the fig. The stop PB is intially closed, the motor will not move until the forward run PB/reverse run PB is closed. Suppose we want to run the motor in the forward direction from standstill, the outputs of the coils contacters have logic ‘0’ and hence both the auxillary contacts are turned on.

Once the coil contacter gives the logic ‘1’, the following consequences takes place simultaneously

A. The auxillary contact OP001 in the second rung becomes opened,which stops the voltage for reverse motion of the motor. At this stage, the second rung is not turned on even the reverse run PB is pressed by mistake.

B. The auxillary contact OP001 is the first rung is on, which provides the path for the positive voltage until the stop PB is pressed. Here the auxillary contact OP001 acts as a ‘latch’, which facilitates even to remove the PB IN002 once the coil OP001 is on.

If we want to rotate the motor in the reverse direction, the stop PB is to be pressed sothat no voltage in the coil is present, then we can turn on the PB corresponding to reverse run. This is a simple example of ‘interlocking’, where each rung locks the operation of the other rung.

There are several other programming paradigms for PLCs. Two of them are mentioned here for briefly.

The Function Chart (IEC)

Depicts the logic control task symbols in terms of functional blocks connected symbolically in a graphic format.

The Statement List (STL)

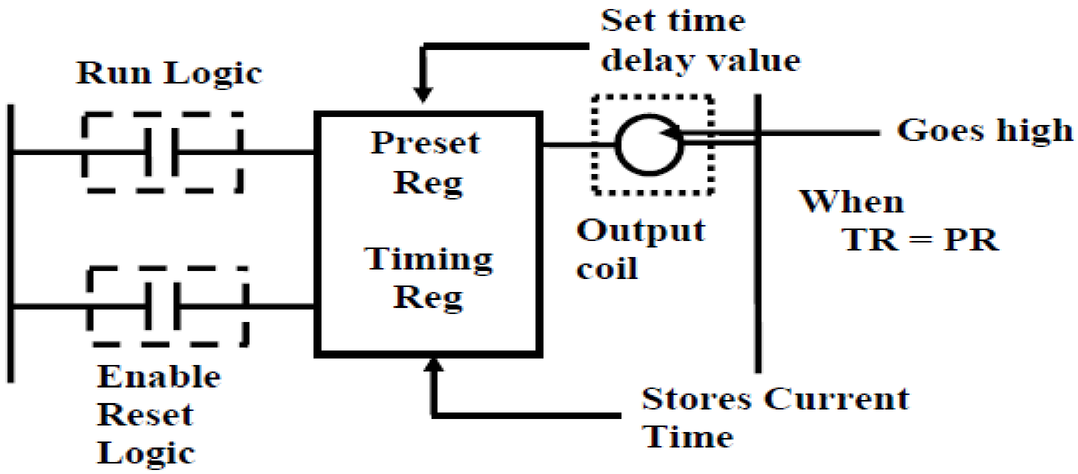
Is made up of series of assembly language like statements each one of which represents a logic control statement executable by the processor of the programmable controller. The statement list is the most unrestricted of all the methods of representation. Individual statements are made up of mnemonics, which represent the function to be executed. This method of representation is favoured by those who have already had experience in programming microprocessors.

Internal Variable Operands or Flags

In addition to the inputs and outputs, which correspond to physical signals in the controlled systems internal variables are required to save the intermediate computational values of the program. These are referred to as Flags, or the Auxiliary Contacts in Relay Ladder Logic parlance. The number of such variables admissible in a program may be limited. Such auxiliary contacts correspond to output values and are assumed to be activated by the corresponding output values. They may be either of an NO or an NC type. Therefore, an NO auxiliary contact would be closed if the corresponding output is active i.e. has value “1”.

Timer

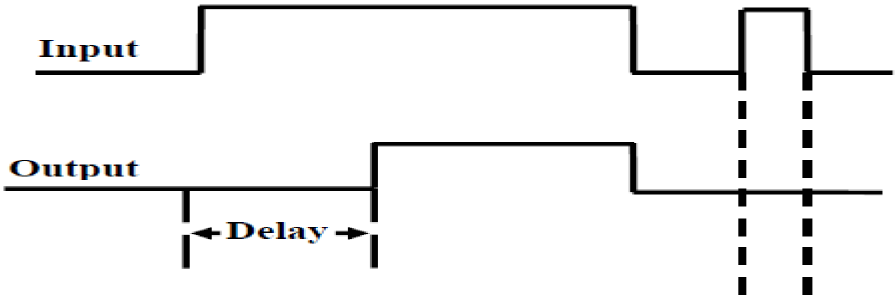
These are special operands of a PLC, which represent a time delay relay in a relay logic system. The time functions are a fixed component of the central processing unit. The number of these varies from manufacturer to manufacturer and from product to product. It is possible to achieve time delays in the range of few milliseconds to few hours.

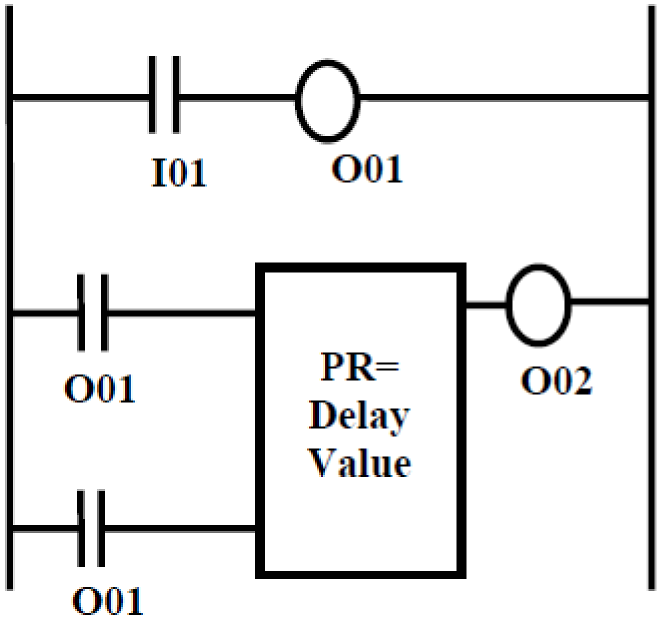


Timers have a preset register value, which represent the maximum count it can hold and can be set using software/program. The figure shown below has a ‘enable reset logic’ and ‘run logic’ in connection with the timer. The counter doesnot work and the register consists of ‘zero’ until the enable reset logic is ‘on’. Once the ‘enable reset logic’ is ‘on’, the counter starts counting when the ‘run logic’ is ‘on’. The output is ‘on’ only when the counter reaches the maximum count.

Various kinds of timers are explained as follows

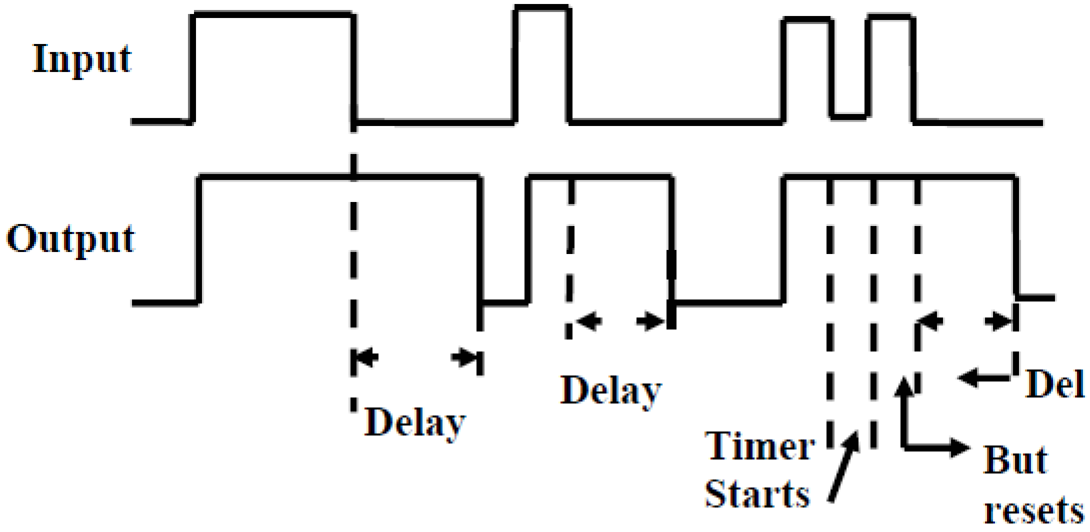
On delay timer: The input and output signals of the on delay timer are as shown in the Fig. 19.6. When the input signal becomes on, the output signal becomes on with certain delay. But when the input signal becomes off, the output signal also becomes off at the same instant. If the input becomes on and off with the time which less than the delay time, there is no change in the output and remains in the ‘off’ condition even the input is turned on and off i.e., output is not observed until the input pulse width is greater than the delay time.



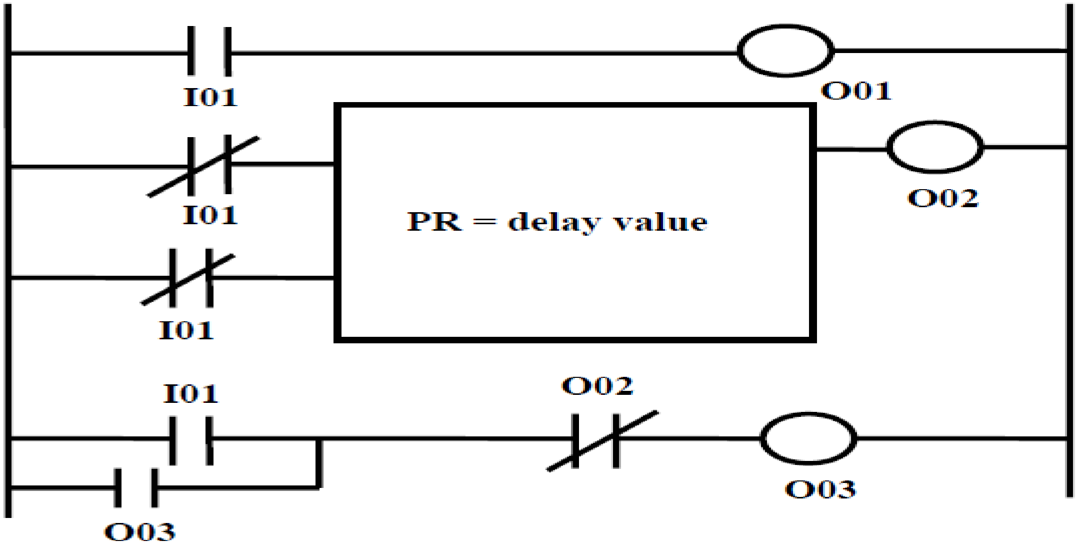


Realization of on-delay timer: The realization of on-delay timer using the basic timer shown in the previous fig is explained here. The realization is as shown in the Fig. 19.6, which shows a real input switch(IN001), coil1(OP002), two normally opened auxillary contacts(OP002), coil2(OP002). When the real input switch is 'on' the coil(OP002) is 'on' and hence both the auxillary switches are 'on'. Now the counter value starts increasing and the output of the timer is 'on' only after it reaches the maximum preset count. The behaviour of this timer is shown in figure, which shows the on-delay timer. The value in the counter is 'reset' when the input switch(IN001) is off as the 'enable reset logic' is 'off'. This is a non-retentive timer.

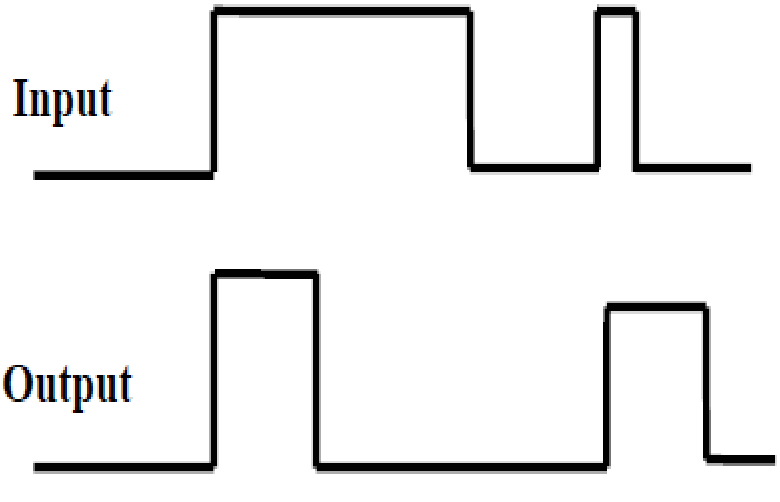
Off delay timer: The input and output signals of the off delay timer are as shown in the Fig. 19.7. When the input signal becomes on, the output signal becomes on at the same time. But when the input signal becomes off, the output signal becomes 'off' with certain delay. If the input becomes on and off with the time which less than the delay time, there is no change in the output and remains in the 'on' condition even the ipnut is turned on and off i.e., the delay in the output is not observed until the input pulse width is greater than the delay time.

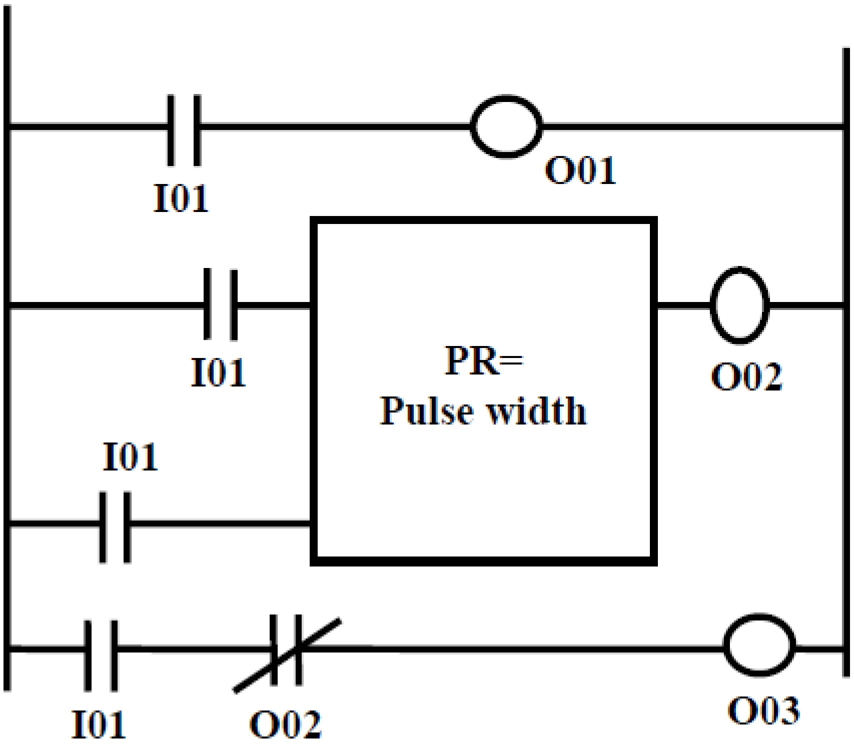


Realization of off-delay timer: The realization of on-delay timer using the basic timer shown in the previous fig is explained here. The realization is as shown in the Fig. 19.8, which shows a real input switch(IN001), coil1(OP002), two normally closed input contacts(IN001), output contacts (OP002,OP003). When the real input switch is 'on', the coil(OP002) is 'on' and both the auxillary input switches are 'off'. Now the output contact(OP002) becomes 'off' which in turn makes the auxillary contact(OP002) in the third rung to become 'on' and hence the output contact(OP003) is 'on'. When the real input switch is 'off', the counter value starts increasing and the output of the contact becomes 'on' after the timer reaches the maximum preset count. At this time the auxillary contact in the third rung becomes 'off' and so is the output contact(OP003). The input and output signals are as shown in the figure, which explain the off-delay timer.

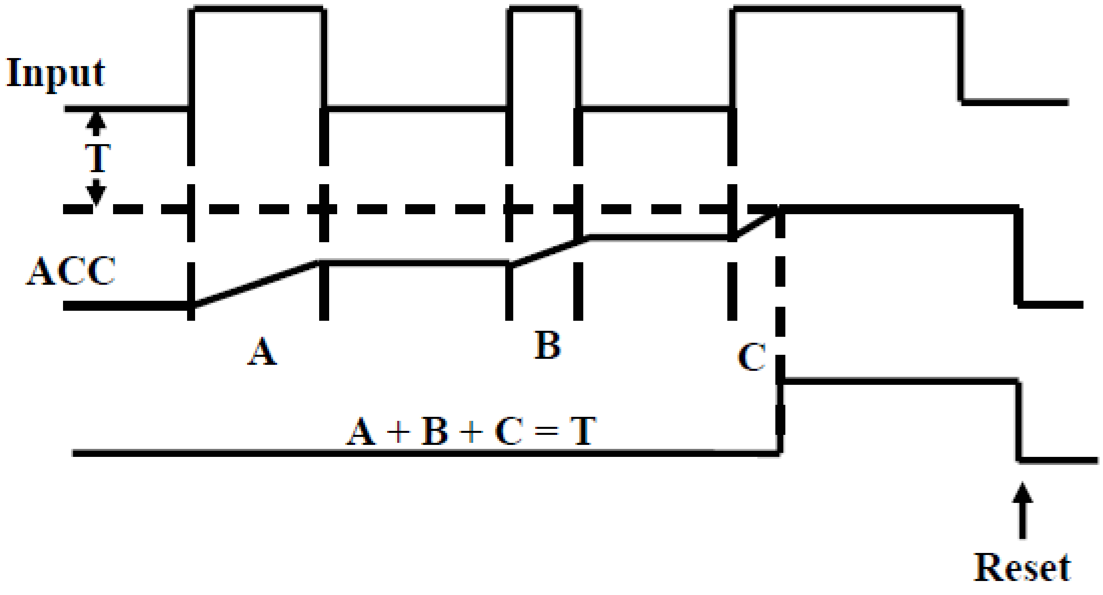


Fixed pulse width timer: The input and output signals of the fixed pulse width timer are as shown in the Fig . When the input signal becomes on, the output signal becomes on at the same time and remains on for a fixed time then becomes ‘off’. The output pulse width is independent of input pulse width.

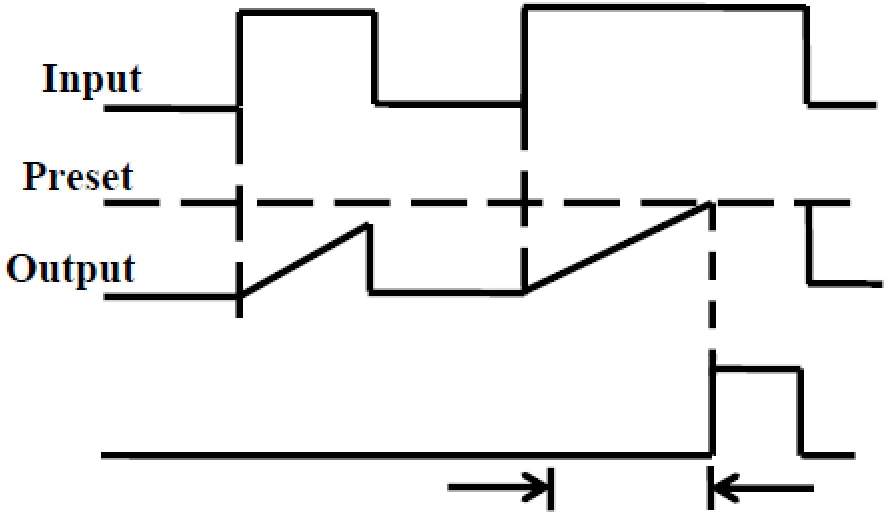




Retentive timer: The input and output signals of the retentive timer are as shown in the fig. This is also implemented internally in a register as in the previous case. When the input is 'on' , the internal counter starts counting until the input is 'off' and at this time, the counter holds the value till next input pulse is applied and then starts counting starting with the value existing in the register. Hence it is named as 'retentive' timer. The output is 'on' only when the counter reaches its 'terminal count'.

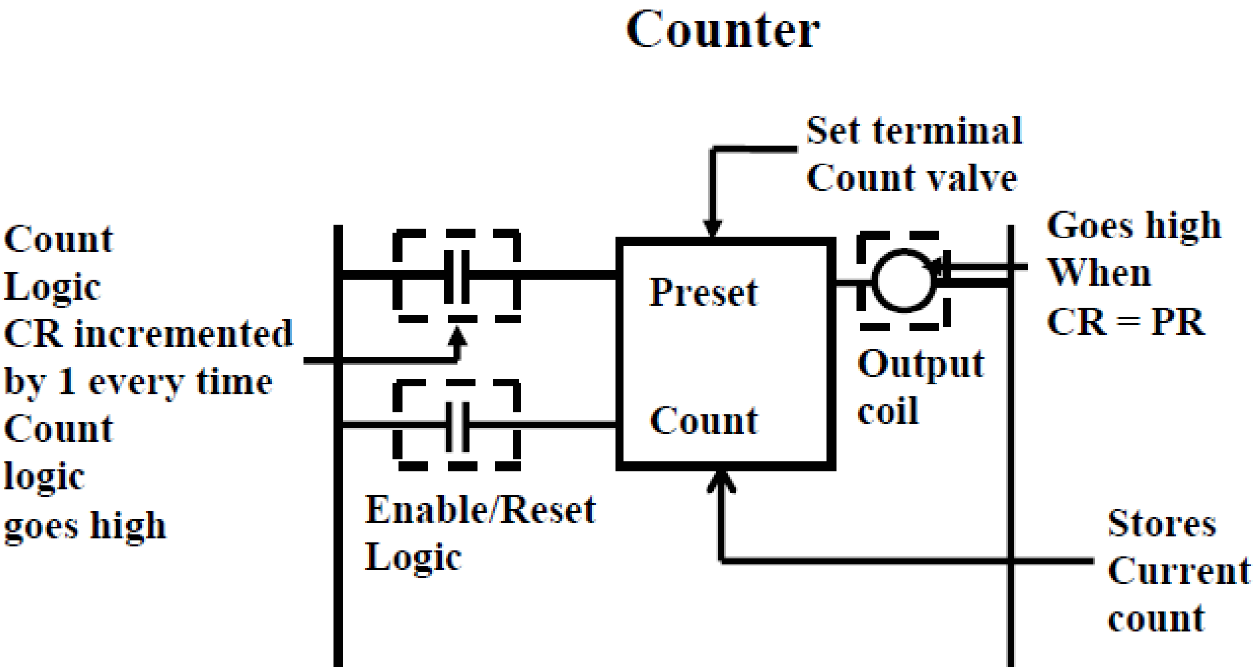


Non-retentive timer: The input and output signals of the non-retentive timer are as shown in the Fig. . This is implemented internally in a register. When the input is ‘on’, the internal counter starts counting until the input is ‘off’ and at this time the value in the counter is reset to zero. Hence it is named as non-retentive timer. The output is ‘on’ only when the counter reaches its ‘terminal count’.



Counter

The counting functions (C) operate as hardware counters, but are a fixed component of the central processing unit. The number of these varies for each of the programmable controllers. It is possible to count up as well as to count down. The counting range is from 0 to 999. The count is either dual or BCD coded for further processing.



User defined Data

If the memory capacity of the flag area is not sufficient to memorize the signal status and data, the operand area “data” (D) is applied. In general, in the flag area, primarily binary conditions apply, whereas in the data area digital values prevail and are committed to memory. The data is organized into data blocks (DB). 256 data words with 16 bit each can be addressed to each data block. The data is stored in the user memory sub module. The available capacity within the module has to be shared with the user program**.**

Addressing

The designation of a certain input or output within the program is referred to as addressing. Different PLC manufacturers adopt different conventions for specifying the address of a specific input or output signal. A typical addressing scheme adopted in PLCs manufacturers by Siemens is illustrated in the sequel.

The inputs and outputs of the PLCs are mostly defined in groups of eight on digital input and/or digital output devices. This eight unit is called a **byte**. Every such group receives a number as a **byte address.** Each in/output byte is divided into 8 indiindividual **bits**, through which it can respond with. These bits are numbered from bit 0 to bit 7. Thus one receives a **bit address**. For example, in the address I0.4, **I** denotes that the address type is specified as Input, **0** is the byte address and **4** the bit address. Similarly in the address Q5.7, **Q** denotes that the address type is specified as Output, **5** is the byte address and **7** is the bit address.

Operation Set

The operation set of PLC programming languages can be divided into four major function groups:

Binary or Logic functions

Program control functions and

Other statements

Binary function combines primarily binary signal status with logic operations. The logic functions (binary logic operation) are the AND and the OR functions, according to the series and the parallel circuit arrangement on the ladder diagram. The result of the logic operation together with the memory function is then assigned to the appropriate operand. The majority of operands are inputs (I), output (O) and flags (F). With the result of logic operations of binary logic, timers can be enabled and started and counters can be initiated, incremented to count up or decremented to count down. Because the results of the time and count functions can be combined with logic functions, the associated operations are considered to be part of the group of binary functions.

Arithmetic functions are primarily used to perform arithmetic on numerical values. These can be combined with logic operations, such that the numeric operations can be enabled or disabled in a given scan based on logic conditions, much like “if then else” programming constructs. Similarly, logic conditions can be derived from numeric variables using operations like comparison.

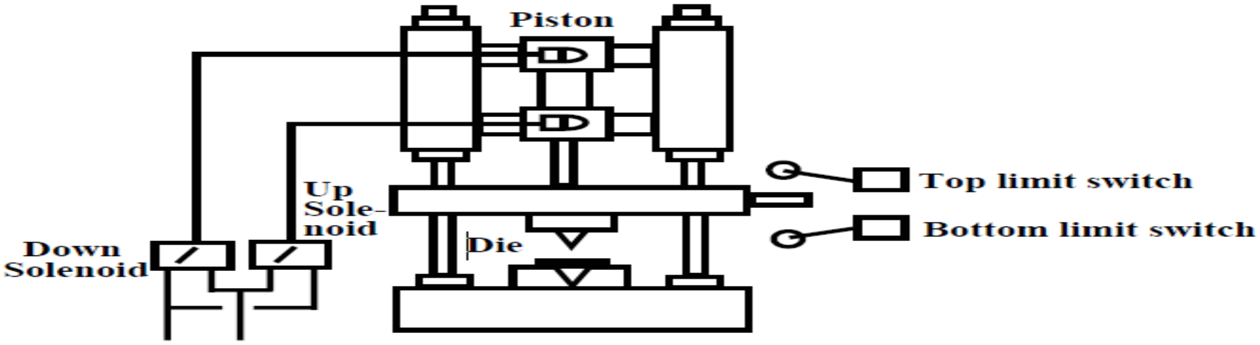
Logic operations are established in the register of the processor (in the “accumulators”). The registers are loaded with a loading operation (they are supplied with a value). The result of the logic operation is then transferred back to the operand area via a transfer operation. Digital functions are:

Program Control operations include, Function block calls and Jump functions.

Formal Modelling of Sequence Control Specifications and Structured RLL Programming

Industrial Logic Control Example Revisited

For convenience of reference the description and a pictorial representation of the process is reproduced below.



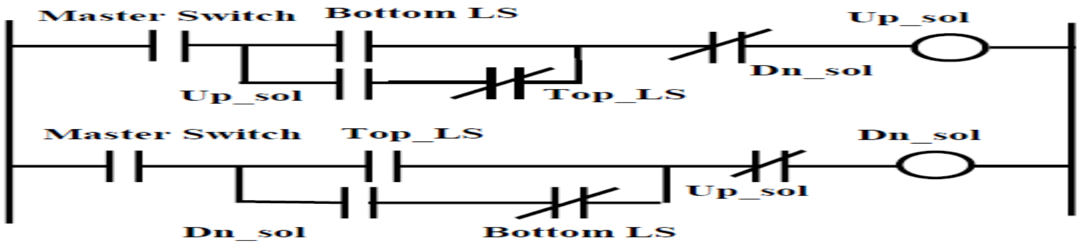
Linguistic description of the industrial stamping process

This process consists of a metal stamping die fixed to the end of a piston. The piston is extended to stamp a work piece and retracted to allow the work piece to be removed. The process has 2 actuators: an up solenoid and a down solenoid, which respectively control the electro-hydraulic direction control valves for the extension and retraction of the stamping piston and die. The process also has 2 sensors: an upper limit switch that indicates when the piston is fully retracted and a lower limit switch that indicates when the piston is fully extended. Lastly, the process has a master switch which is used to start the process and to shut it down.

The control computer for the process has 3 inputs (2 from the limit sensors and 1 from the master switch) and controls 2 outputs (1 to each actuator solenoid).

The desired control algorithm for the process is simply as follows. When the master switch is turned on, the die-stamping piston is to reciprocate between the extended and retracted positions, stamping parts that have been placed in the extended piston machine. When the master switch is switched off, the piston is to return to a shutdown configuration with the actuators off and the piston fully retracted.

At first, let us consider an Relay Ladder Logic program that has been written directly from the linguistic description and assess it for suitability of operations.The first version of sequence control program for the industrial stamping process



A hastily constructed RLL program for the above process may look like the one given in Figure . The above program logic indicates that the Up solenoid output becomes activated when the Master Switch is on and the bottom Limit Switch is on. Also there is interlock provided, so that when the Down solenoid is on, the Up solenoid cannot be on. Further, once the Up solenoid is on, the output is latched by an auxiliary contact, so that it remains on till the bottom LS is made on, when it turns off. A similar logic has been implemented for the activation of the Down solenoid.

However, on closer examination, several problems may be discovered with the above program. Some of these are discussed below.

- For example, there is no provision for a Master stop switch to stop the press from stopping in an emergency, except by turning the Master Switch off. This would indeed stop the process, however, if the press stops midway, both bottom and top limit switches would be off. Now the process would not start, even if the Master switch is turned on again. Therefore, either a manual jogging control needs to be provided, so that the operator can return the piston to the up position by manually operating the hydraulics, or special auto mode logic should be designed to perform this.

- As a second example, note that this process does not have a part detect sensor. This implies that the moment the Master switch is on the press would start going up and down at its own travel speed, regardless of whether a part has been placed for pressing or not. Apart from wastage of energy, this could be safety hazard for an operator who has to place the part on the machine between the interval of a cycle of operation.

The above discussion clearly indicates the need for a systematic approach towards the development of RLL programs for industrial logic control problems. This is all the more true since industrial process control is critical application domain where control errors can lead to loss of production or operator safety. Therefore, in this chapter, we discuss a systematic approach towards the design of RLL programs.

Steps in Sequence Control Design

The approach to Sequence control Design presented below is derived from the basic principles of modern software engineering practices. An interested student is referred to standard software engineering references for a more detailed discussion on these.

A brief description of the steps follows:

The broad initial steps are

- Requirement Analysis
- For Modelling of the process
- Design
- Design verification and validation

A. Requirements Analysis

This is a very important initial step. Errors in this step may be discovered very late during commissioning of the control system and can result in loss of significant amount of man-hours. Note that, since the control engineer is not necessarily an expert in plant operations. Therefore it is quite natural that he may not understand requirements and characteristics of process operations fully, without conscious and significant effort.

Therefore, this phase is to be carried out in close consultation with the plant engineers of the user organization. It basically involves studying the system behavioral aspects of the system to:

i) identify the feedback inputs from sensors and the external operator inputs from Man Machine Interface (MMI),

ii) identify the controller outputs to actuators and the outputs to the indicators/ MMI,

iii) study and document the sequence of actions and events under the various operational ‘modes’. This should not only indicate the ‘normal’ modes, but also ‘emergency’, ‘start-up’, ‘shut-down’ and other special modes operation that may not be occurring very frequently but important all the same.

iv) study the effects of possible failures in the process as well as sensors and identify possible means of recovery. Although a failure may be very rare in occurrence, unless they are considered, some of them may have devastating consequences when they occur. Failures in a process can occur in the sensors, the actuators, or some time the process equipment itself.

v) examine need for manual override control, additional sensors, indicators, and alarms for maintenance, operational efficiency or safety.

Initially, often the above information may be collected in the form of informal linguistic descriptions by direct discussion with plant personnel. Further, it must be remembered that software development is often an iterative process and therefore, the above analysis steps may have to be repeated a number of times for an actual design exercise.

Requirements Analysis for the Stamping Process

As given above the first step in requirement analysis is to identify the Process Control Inputs that need to be sensed and the Process Control outputs that need to be actuated.

Process Control Inputs

- *Part sensor*: A position switch that detects when a part has been placed. In cases where proper positioning of the part can take time. One may also consider using manual switch to be operated by the operator once he is satisfied that the part is properly placed and ready to be stamped. Here an automated part detect sensor has been assumed.

- *Auto PB* : A push button that indicates that machine is ready to stamp parts one after the other in the ‘automatic mode’

- *Stop PB*: A push button that the operator can use to stop the machine any time during the time that the piston is moving down. This is needed to avoid stamping a part if any last second error is discovered by the operator regarding, say, the placement of the part.

• *Reset PB*: In case the piston has been stopped due to some error condition, it is desired that the operator explicitly presses this push button to indicate that the error has been taken care of, and the machine is ready to return to stamping in the auto mode.

• *Bottom LS*: This sensor indicates when the piston has reached the bottom position.

• *Top LS*: This sensor indicates when the piston has reached the top position.

Process Control Outputs

• *Up Solenoid*: Control output that drives the Up solenoid of the electro-hydraulic direction control valve which in turn drives the piston up.

• *Down Solenoid*: Control output that drives the Up solenoid of the electro-hydraulic direction control valve which in turn drives the piston up.

• *Auto Mode Indicator*: An indicator lamp that indicates that the machine is in 'Auto' mode.

• *Part Hold*: A gripping actuator that holds the part firmly to avoid movements during stamping

Sequence of Events and Actions

A. The "Auto" PB turns the Auto Indicator Lamp on

B. When a part is detected, the press ram advances down to the bottom limit switch

C. The press then retracts up to top limit switch and stops

D. A "Stop" PB, if pressed, stops the press only when it is going down

E. If the "Stop" PB is pressed, the "Reset" PB must be pressed before the "Auto" PB can be pressed

F. After retracting, the press waits till the part is removed and the next part is detected

Effects of failures

Among possible failures for this process are drops in hydraulic pump pressure, failures in top and bottom limit switches etc. The exact nature of the failures and its impact need to be understood for the application context. While this should be done, it requires domain knowledge for the control engineer. This is therefore not attempted here for reasons of conciseness. However, the learner is encouraged to augment the control logic with additional logic to detect such failures rapidly and initiate activities for fault tolerance.

Formal process modelling

Once the requirements have been ascertained, formal process modelling can be undertaken. In this step the informal linguistic descriptions have to be rigorously checked for ambiguity, inconsistency or incompleteness. This is best achieved by converting linguistic descriptions into formal process models. Initially one may use intermediate forms like list of operations, flowchart etc. Eventually and before developing the control programs, these are to be converted into mathematically unambiguous and consistent description using a formal modeling framework such as a Finite State Machine (FSM). It is the experience of practical engineers that modelling paradigms that can be represented pictorially are particularly suited to human beings.

For formal modelling, a process often can be viewed as a Discrete Event System (DES). Many formalisms for creating timed or untimed models of DESs exist (e.g. Petri Nets). A detailed description of these is beyond the scope of this lesson. An interested reader is referred to literature on real-time systems for a more detailed discussion on these. In this lesson, it is shown how the process dynamics can be modeled as a Finite State Machine. The following facts which are very important to modelling are mentioned.

A. An FSM is a simple formalism for DES in which, at any time, the system exists in any one discrete-state of a finite set of such states.

B. A state is basically an assignment of values to the set of variables of the system. For a discrete event system, the process variables are assumed to take only a finite set of values. For example, the limit switches can only take two values each, namely, either ON or OFF.

C. Further, the set of the process variables have to be chosen in such a manner that, the future behaviour of the process would be determined solely based on the values of the chosen set of variables at the present time. For example, for the stamping press example, the set of process variables would include the values for the Top and Bottom limit switches. However, based only on these the behaviour of the process cannot be determined. This is because from these it cannot be determined whether the piston is moving up or moving down. Therefore one would have to add the state of the motion as a state variable. The set of values that this variable can take are: ‘going up’, ‘going down’ and ‘stationary’. In this case, one would also have to add another variable, namely the value of the part detect sensor output to be able to distinguish between the behavioral difference between the case when it is ON and when it is OFF, when the piston is at the top position.

D. Some of the state variables may be measured physically with sensors. Others may not be.

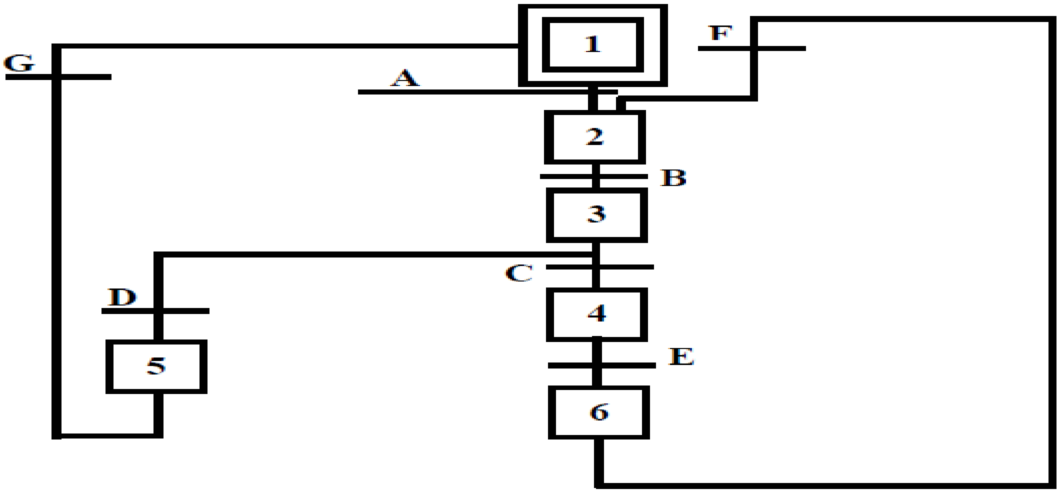
E. The choice of state variables can be subjective and different designers might pick others. The choice also depends on the nature of control actions that one would like to take. Thus, the choice of states is specific to the machine and its operation.

F. During its life cycle, the process moves from state to state over time. Thus it spends most of its time in the states. Occasionally however, it makes a transition from one state to another. The occurrence of a transition depends entirely on the occurrence of discrete events. Such events are names given to conditions involving states, some of which may change due to external factors, such as operator inputs, or due to internal factors, such passage of time. On occurrence of such an event, mechanisms causing state transitions are triggered. State transitions or events are generally considered instantaneous and thus, the system spends time only in the various states. State variables are modified by the occurrence of transitions. In fact, it is this change in the values of the state variables, which is taken to be a transition from one state to another.

G. All possible combinations of state variables may not be valid state assignments for a system. In other words, the system can have only some of all the possible combinations of state variables. These are said to be the combinations that are 'reachable' by the system.

H. One of the states is generally taken to be an 'initial state'. The system, when it starts its life cycle, that is, at the time from which its behaviour is described by the State Diagram, is supposed to be at the initial state.

I. At each state, a set of outputs are exercised. This is described by an output table, where the values for each output variable at each of the states is shown.



Design of RLL Program

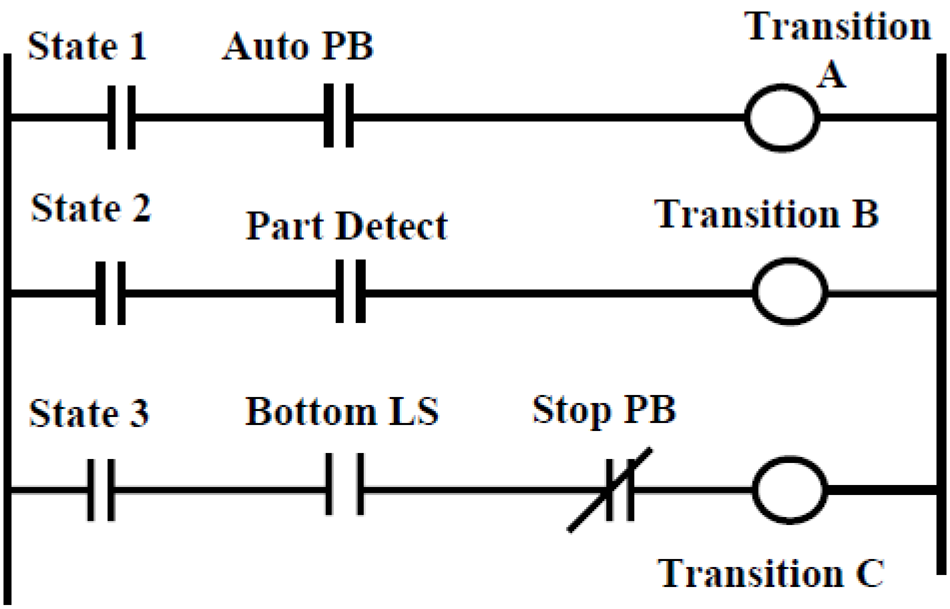
Based on the formal model, the sequence control program can be developed systematically. In fact, one of the main advantages of formal process modelling is that the process of development of the control program becomes mechanical. Thus it can be done quickly and with a much reduced chance of error. In the case of FSM models one has to write the RLL program such that over the scan cycles it executes the state machine itself. The outputs of the state machine go to the process, and since the state machine is nothing but a behavioral model for the process, the process also executes the transitions of the machine, as desired. The realization of a state machine by an RLL program involves computation of the process transitions, in terms of the inputs and the internal state variables of the program followed by computation of the new states and finally, the outputs corresponding to the states. This method is demonstrated here for RLL programming using the above example of the industrial stamping process.

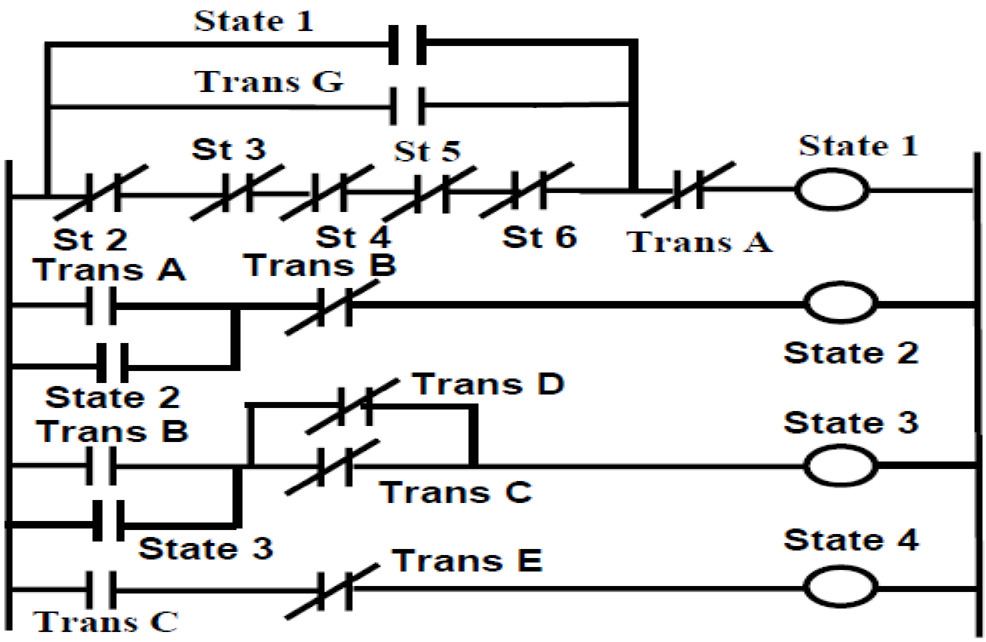
The ladder logic begins with a section to initialize the states and transitions to a single value, corresponding to the initial state. Some PLCs programming languages provide special instructions for such initialization. In this case, however, it is assumed that all auxiliary variables representing the states are set to zero initially. Logic is provided such that in the first scan the auxiliary state variable corresponding to the initial state would be set to 1.

The next section of the ladder logic considers the transitions and then checks for transition conditions. Each transition condition contains an auxiliary NO contact corresponding to the source state from which it is defined. For example, note that the logic for the rung corresponding to transition A contains a contact corresponding to state 1, which is the source state for transition A. Further, it contains the other logical terms corresponding to the input state variables as well as timer outputs, if applicable. In the case of transition A, the external condition is simply the pressing of the Auto PB. Note that, in any scan cycle, at most one transition can be enabled.

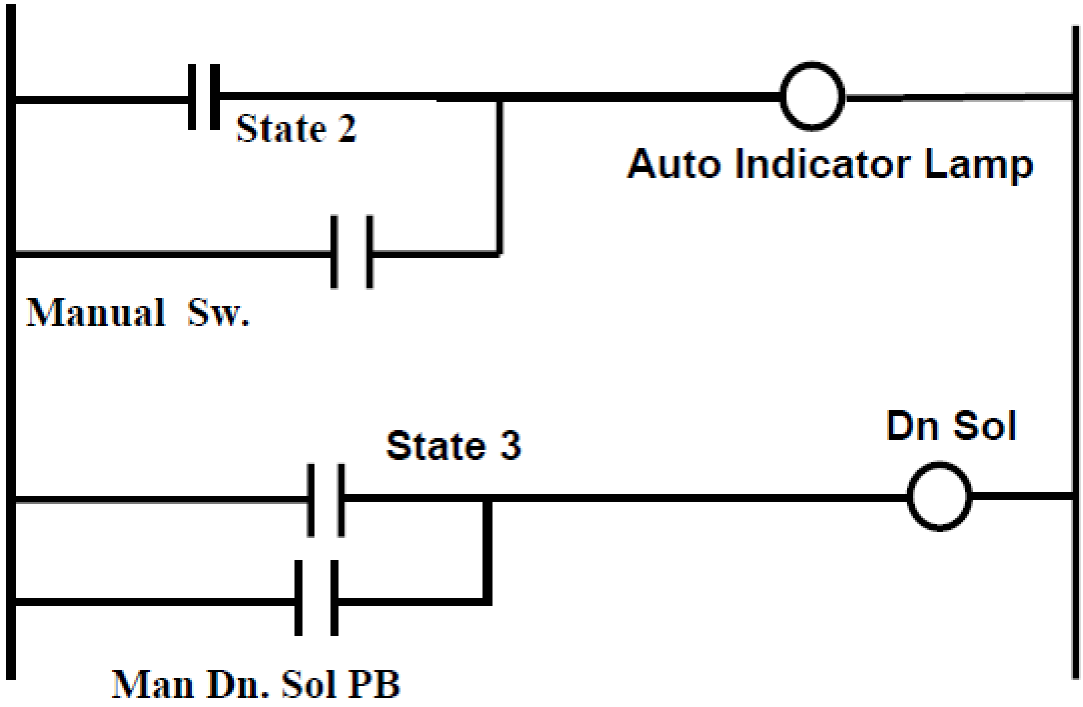
The next block of rungs constitutes the state logic. If the transition logic for any transition is satisfied, the following state logic which is the destination state for the enabled transition is to be turned on and the state logic which is the source state for the enabled transition is to be turned off. Therefore each of the rungs corresponding to a state contains one auxiliary NO contact corresponding to the transition for which that state is a destination state. Similarly, each of the rungs corresponding to a state contains one auxiliary NC contact corresponding to the transition for which that state is a source state in series with the above NO contact. If there is more than one transition for which the state is a destination, all the auxiliary NO contacts corresponding to these transitions should be put in parallel. Similarly, if there is more than one transition, for which the present state is a source, all the auxiliary NC contacts corresponding to these transitions are to be put in series. This occurs in the same scan cycle in which the transition logic is turned on, since the state logic rungs follow those corresponding to transition logic. Note that at the end of every scan cycle, there is only one state logic that is enabled.

Now that the state logic has changed, in the next scan cycle the transition that was enabled, turns off and the system stays in that state, till the next transition logic gets enabled. So that the state logic remains turned on, even if a transition for which this state is a source, turns off, an NO auxiliary contact corresponding to the state that latches the state logic is to be provided in parallel with the parallel block of all the auxiliary NO contacts for each transition for which the present state is a destination.





This is followed by ladder logic to turn on outputs as requires by the steps. This section of ladder logic corresponds to the actions for each step. The rung for each output therefore contains one NO auxiliary contact corresponding to the state in which it is enabled. If an output is enabled at more than one state, the auxiliary NO contacts corresponding to those states would be connected in parallel. Similarly, if Manual switch or PB contacts are required, they also have to be put in parallel with the contacts for the states.



Programming of PLCs: Sequential Function Charts

There are also other languages to program a PLC in, than the RLL. Most of the significant manufacturers support about 3 to 5 programming languages. Some of these languages, such as the RLL, have been in use for a long time. While most manufacturers used similar languages, these were not standardised in terms of syntactic features. Thus programs developed for one would not run in another without considerable modifications, often mostly syntactic. In the last few years there has been effort to standardise the PLC programming languages by the International Electrotechnical Commission (IEC). One of the languages, namely the Sequential Function Chart, which offers significant advantages towards development of complex structured PLC programs for concurrent industrial processes, is studied in detail. Know other languages are introduced in brief.

IEC 1131-3: The International Programmable Controller Language Standard

IEC 1131 is an international standard for PLCs formulated by the International Electrotechnical Commission (IEC). As regards PLC programming, it specifies the syntax, semantics and graphics symbols for the following PLC programming languages:

- Ladder diagram (LD)
- Sequential Function Charts (SFC)
- Function Block Diagram (FBD)
- Structured Text (ST)
- Instruction List (IL)

IEC 1131 was developed to address the industry demands for greater interoperability and standardisation among PLC hardware and software products and was completed in 1993. A component of the IEC 1131, the IEC 1131-3 define the standards for data types and programming. The goal for developing the standard was to propose a programming paradigm that would contain features to suit a large variety of control applications, which would eliminate proprietary barriers for the customer and their associated training costs. The language specification takes into account modern software engineering principles for developing clean, readable and modular code. One of the benefits of the standard is that it allows multiple languages to be used simultaneously, thus enabling the program developer to use the language.

Major Features of IEC 1131-3

The following are some of the major features of the standard.

1. ***Multiple Language Support*:** One of the main features of the standard is that it allows multiple languages to be used simultaneously, thus enabling the program developer to use the language best suited to each control task.

2. ***Code Reusability:*** The control algorithm can include reusable entities referred to as "program organization units (POUs)" which include Functions, Function Blocks, and Programs. These POUs are reusable within a program and can be stored in user-declared libraries for import into other control programs.

3. ***Library Support:*** The IEC-1131 Standard includes a library of pre-programmed functions and function blocks. An IEC compliant controller supports these as a "firmware" library, that is, the library is pre-coded in executable form into a prom or flash ram on the device. Additionally, manufacturers can supply libraries of their own functions. Users can also develop their own libraries, which can include calls to the IEC standard library and any applicable manufacturers' libraries.

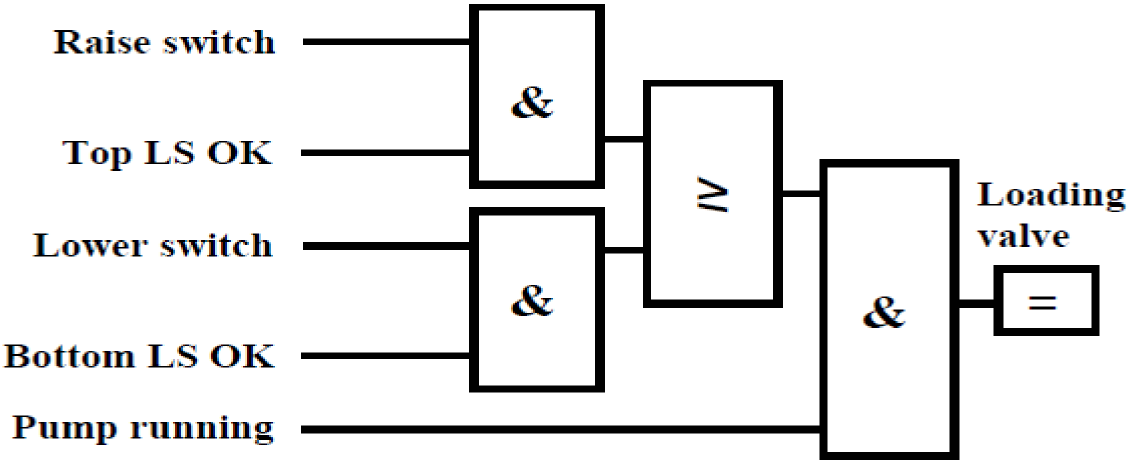
4. ***Execution Models:*** The general construct of a control algorithm includes the use of "tasks", each of which can have one or more Program POUs. A task is an independently schedulable software entity and can be assigned a cyclic rate of execution, can be event driven, or be triggered by specific system functions, such as startup.

IEC 1131-3 Programming Languages

IEC 1131-3 defines two graphical programming languages (Ladder Diagram and Function Block Diagram), two textual languages (Instruction List and Structured Text), and a fifth language (Sequential Function Chart) that is a tool to define the program architecture and execution semantics. The set of languages include assembly-like low-level language like the Instruction List, as well as Structured Text having features similar to those of a high level programming language. Using these, different computational tasks of a control algorithm can be programmed in different languages, then linked into a single executable file.

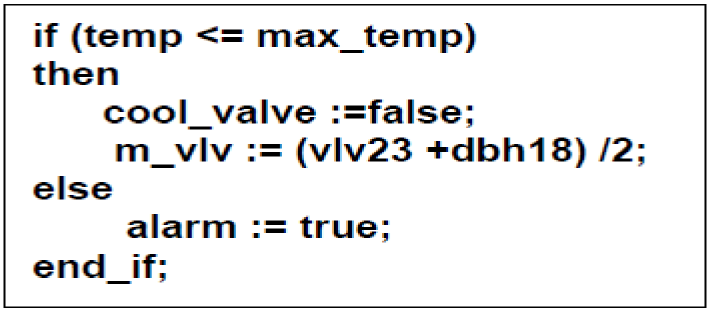
Function Block Diagram (FBD)

The function block diagram is a key product of the standard IEC 1131-3. FBD is a graphical language that lets users easily describe complex procedures by simply wiring together function blocks, much like drawing a circuit diagram with the help of a graphical editor. Function blocks are basically algorithms that can retain their internal state and compute their outputs using the persistent internal state and the input arguments. Thus, while a static mathematical function will always return the same output given the same input (e.g. sine, cosine), a function block can return a different value given the same input, depending on its internal state (e.g. filters, PID control). This graphical language clearly indicates the information or data flow among the different computational blocks and how the over all computation is decomposed among smaller blocks, each computing a well defined operation. It also provides good program documentation. The IEC 1131 standard includes a wide range of standard function blocks for performing a variety of operations, and both users and vendors can create their own. A typical simple function block is shown below in Fig.



Structured Text (ST)

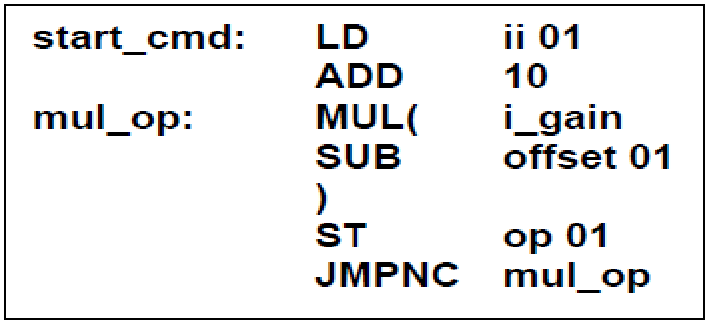
Structured Text (ST) is a high-level structured programming language designed for expressing algorithms with complex statements not suitable for description in a graphical format. ST supports a set of data types to accommodate analog and digital values, times, dates, and other data. It has operators to allow logical branching (IF), multiple branching (CASE), and looping (FOR, WHILE…DO and REPEAT…UNTIL). Typically, a programmer would create his own algorithms as Functions or Function Blocks in Structured Text and use them as callable procedures in any program. A typical simple program segment written in Structured Text is shown below in Fig.



```
if (temp <= max_temp)
then
      cool_valve :=false;
       m_vlv := (vlv23 +dbh18) /2;
else
        alarm := true;
end_if;
```

Instruction List (IL)

A low-level assembly-like language, IL is useful for relatively simple applications, and works on simple digital data types such as boolean, integer. It is tedious and error prone to write large programs in such low level languages. However, because complete control of the implementation, including elementary arithmetic and logical operations, rests with the programmer, it is used for optimizing small parts of a program in terms of execution times and memory. A typical simple program segment written in Instruction List is shown below in Fig.



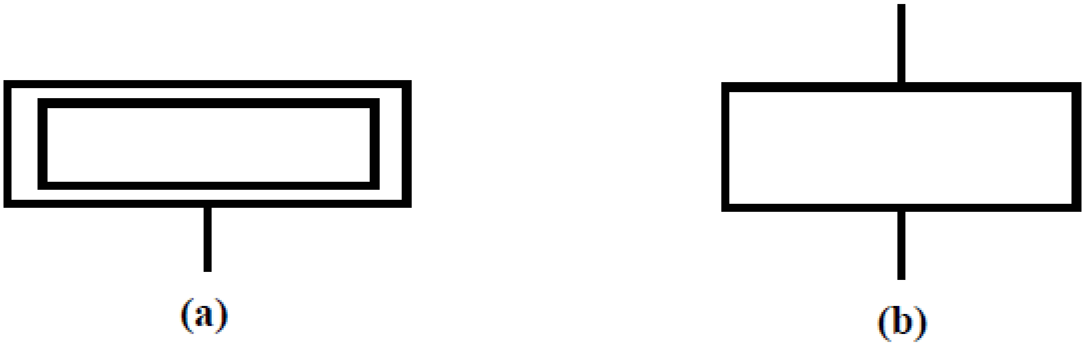
```
start_cmd:    LD        ii 01
              ADD       10
mul_op:       MUL(      i_gain
              SUB       offset 01
              )
              ST        op 01
              JMPNC     mul_op
```

Sequential Function Chart (SFC)

SFC is a graphical method, which represents the functions of a sequential automated system as a sequence of steps and transitions. SFC may also be viewed as an organizational language for structuring a program into well-defined steps, which are similar conceptually to states, and conditioned transitions between steps to form a sequential control algorithm. While an SFC defines the architecture of the software modules and how they are to be executed, the other four languages are used to code the action logic that exercises the outputs, within the modules to be executed within each step. Similar modules are used for computation of the logical enabling conditions for each transition. Each step of SFC comprises actions that are executed, depending on whether the step is active or inactive. A step is active when the flow of control passes from one step to the next through a conditional transition that is enabled when the corresponding transition logic evaluates to true. If the transition condition is true, control passes from the current step, which becomes inactive, to the next step, which then becomes active. Each control function can, therefore, be represented by a group of steps and transitions in the form of a graph with steps labeling the nodes and transitions labeling the edges. This graph is called a Sequential Function Chart (SFC).

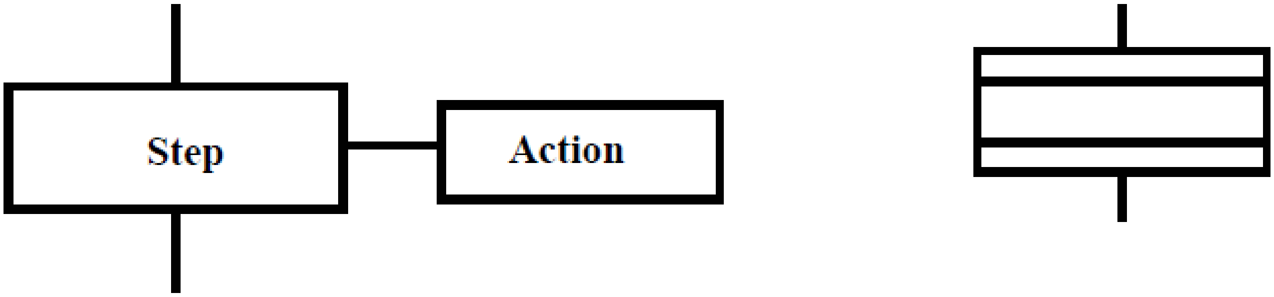
Steps

Each step is a control program module which may be programmed in RLL or any other language. Two types of steps may be used in a sequential function chart: initial and regular. They They are represented graphically as shown below in Fig.



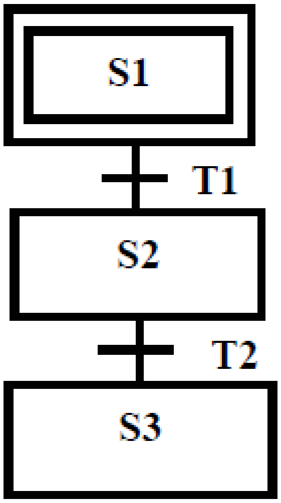
The ***initial step*** is executed the first time the SFC block is executed or as a result of a reset operation performed by a special function named SFC_RESET. There can be one and only one initial step in an SFC. The initial step cannot appear within a simultaneous branch construct, (which is described later in this section) but it may appear anywhere else. A regular step is executed if the transitional logic preceding the step makes the step active. There can be one or many regular steps in an SFC network, one or more of which may be active at a time. Only the active steps are evaluated during a scan.

Each step may have action logic consisting, say, of zero or more rungs programmed in Relay Ladder Diagram (RLD) logic language. *Action Logic* is the logic associated with a step, i.e., the logic, programmed by RLL or any other logic, which is executed when the step is active. When a step becomes inactive, its state is initialised to its default state. A collection of steps may be labeled together as a macro-step.



Transitions

Each transition is a program module like a step that finally evaluates a transition variable. Once a transition variable evaluates to true the step(s) following it are activated and those preceding it are deactivated. Only transitions following active states are considered active and evaluated during a scan. Transitions can also be a simpler entity such as a variable value whose value may be set by simple digital input. Transition logic can be programmed in any language. If programmed in RLL, each transition must contain a rung that ends with an output coil to set its transition variable.



Transitions connect steps in an SFC

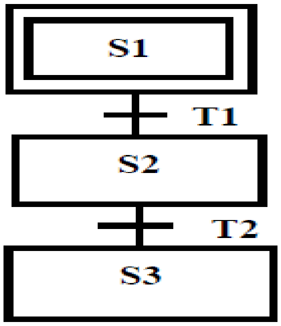
The SFC in Fig.shows how the transitions connect steps in an SFC. Initially, step S1 is active. Thus transition T1 is also active. When the transition variable T1 becomes true, immediately, S1 becomes inactive, S2 becomes active while T1 becomes inactive and T2 becomes active.

Basic Control Structures

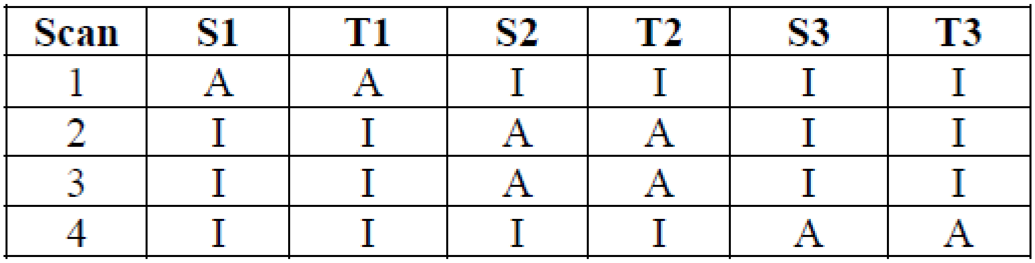
Six basic control structures used in a sequential function chart are discussed below. .

Simple Sequence

In a simple sequence, control passes from step S2 to step S3 **only if** step S2 is active and transition T2 evaluates true.



(a)



Scan	S1	T1	S2	T2	S3	T3
1	A	A	I	I	I	I
2	I	I	A	A	I	I
3	I	I	A	A	I	I
4	I	I	I	I	A	A

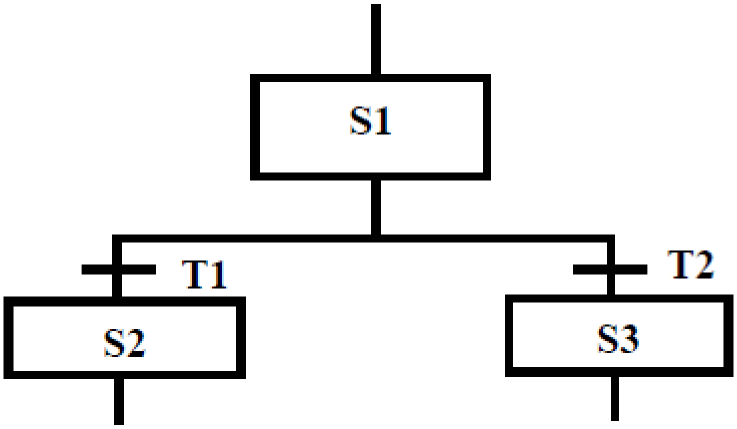
(b)

A simple sequence in an SFC (a) and its execution over scans (b)

The table in Fig. (b) indicates the status (A : active; I:inactive) of te steps and transitions over scan cycles.

Divergence of a Selective Sequence

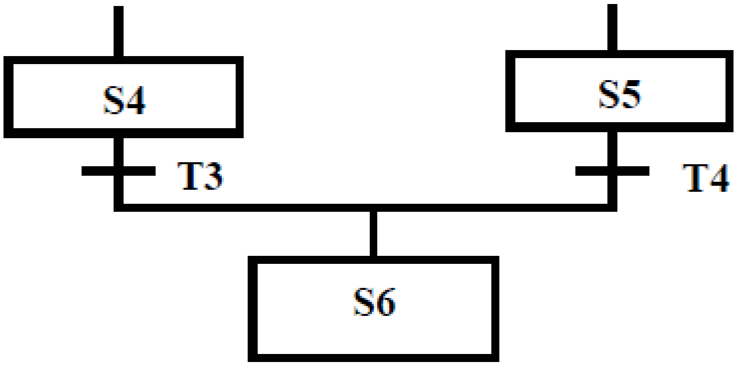
Divergence of a Selective Sequence means that after a step there is a choice of two or more transitions which can evaluate to be true. However, at a time, only one of which can be true, and therefore, the sequence of steps following that transition only are activated. For example, in the divergent selective sequence shown in Fig. 21.8, control passes from step S1 to step S2 if step S1 is active and transition T1 evaluates true. Control passes from step S1 to step S3 if step S1 is active, *transition T1 is not true, and* transition T2 is true (left-to-right priority of transition). Thus, at a time, exactly one branch of the selective sequence is selected. A left-to-right priority is used to determine the action branch if more than one transition evaluates true at the same time. A selective divergence must be preceded by one step. The first element after a selective divergence must be a transition. In terms of familiar programming constructs, this is similar to an IF…THEN…ELSEIF…ELSEIF….ELSE…construct.



Divergece of a selective sequence

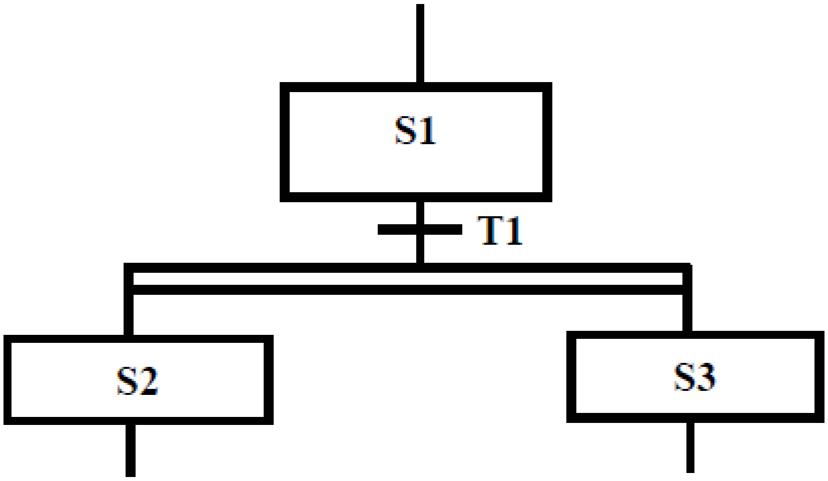
Convergence of a Selective Sequence

A convergence of the divergent selective sequences must follow. Here transitions from each branch of the selective sequence converges eventually to one step. As shown in Fig. in a convergent selective sequence, control passes from step S4 to step S6 if step S4 is active and transition T3 evaluates true. Similarly, control passes from step S5 to step S6 if step S5 is active and transition T4 is true.



Divergence of a Simultaneous Sequence

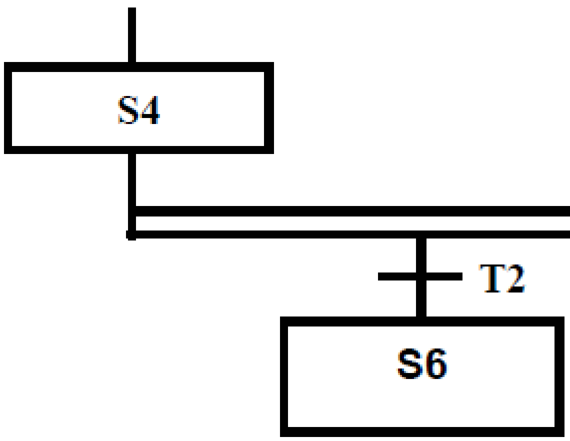
In contrast with a selective sequence, in a simultaneous divergent branch more than one step can become active. Thus, in this case, we have two or more branches to be active simultaneously. In the divergent simultaneous sequence shown in Fig. control transfers from step S1 to step S2 and step S3 if step S1 is active *and* transition T1 evaluates true. Both steps S2 and S3 will become active. Note that the action logic of one step will be executed before the action logic of the other. The order of which step is executed first is undefined in the standard, and depends on a particular implementation. Note that the same thing happens for the state logic too, since computation is necessarily sequential in a single processor system. However the sequence becomes noticeable for action logic, since the output is observed in the field. A simultaneous divergent branch must be preceded by one transition. A step must be the first element after a simultaneous branch.



Divergence of a simultaneous sequence

Convergence of a Simultaneous Sequence

A simultaneous convergent branch concludes a simultaneous sequence. It can only be preceded by step elements and not transitions as in the case of a selective sequence. It must be followed by one transition. In the convergent simultaneous sequence shown in Fig. control passes from step S4 and step S5 to step S6 if steps S5 and S6 are both active and transition T2 evaluates true.

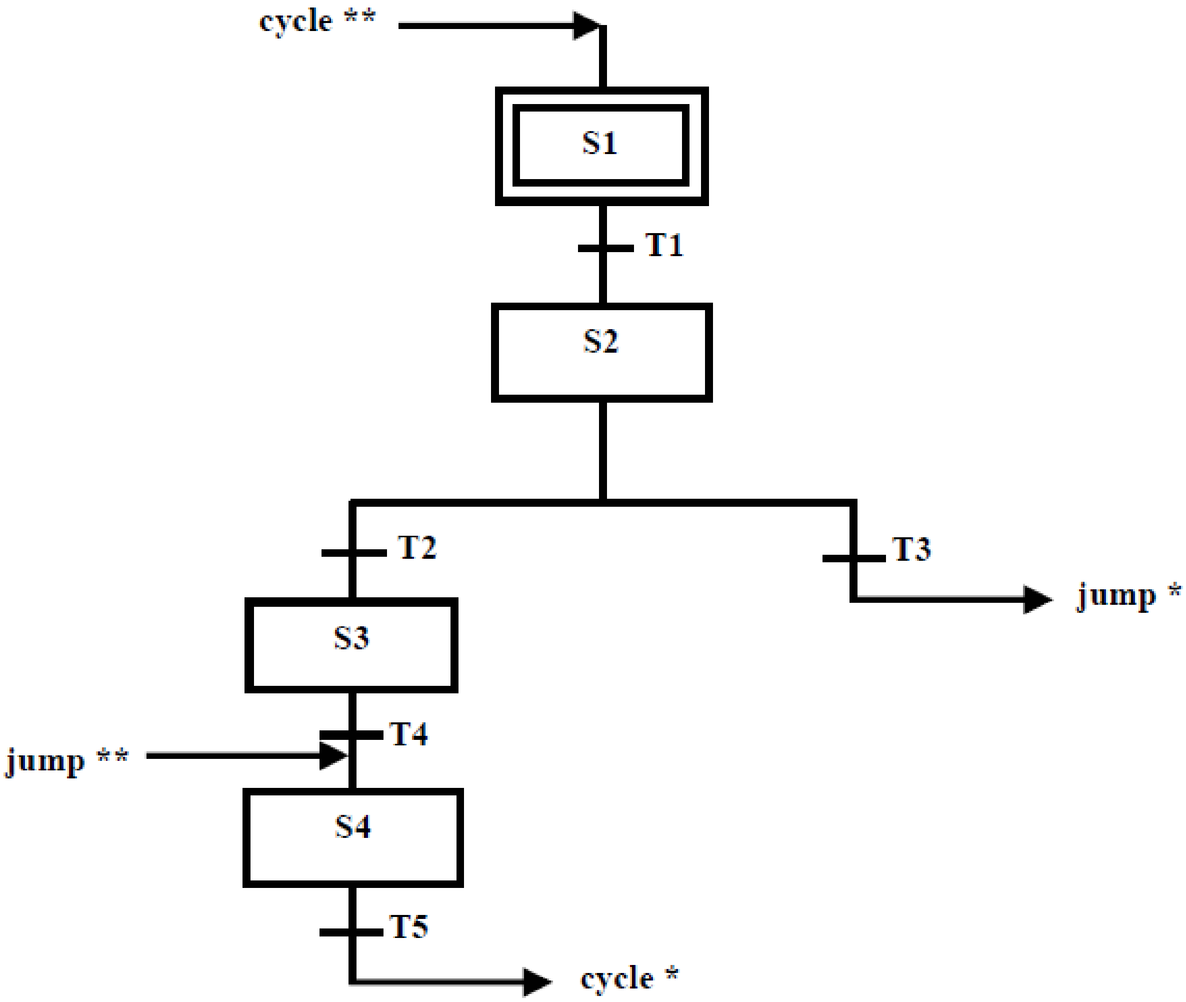


Convergence of a simultaneous sequence

The transition logic for T2 is only executed when all of the steps at the end of the simultaneous sequence are active.

Source and Destination Connectors

Source and destination connectors are used to create forward and backward jumps in an SFC. The keywords *jump* and *cycle* denote connectors in the SFC shown below. Backward jumps are called *cycles*. In the forward *jump* sequence shown in Fig. 21.12, control passes from step S2 to step S4 if step S2 is active, transition T2 evaluates to be false, and transition T3 evaluates to be true. In the backward jump (or cycle) sequence, control passes from step S4 to step S1 if step S4 is active and transition T5 evaluates true. Source and destination connectors cannot occur before a transition. Source connectors must occur immediately after the transition and indicate that the trasfer of control takes place once the transition fires. Destination connectors must occur immediately before a step indicating that the transfer of control after the transition before the corresponding source connector, this step becomes active.



Source and Destination Connectors

Many PLCs also allow SFCs to entered be as graphic diagrams. Small segments of ladder logic can then be entered for each transition and action. Each segment of ladder logic is kept in a separate program. The architecture of such programs is discussed next.

Control Program Architecture with SFCs

A typical architecture of a control program with SFCs is shown in Fig. Here the main program block is organised as an SFC. Each step and transition in the SFC of the main program block is coded as a module. These modules may be coded using any of the languages under the 1131-3 standard. These may be SFCs themselves. In general these modules may be organised in terms of a Preprocessing and a Post Processing block, in addition to the main sequential processing block.

Preprocessing

This section is processed at the start of every scan. Normally, RLD preprocessing logic is used to process, at the start of the scan cycle, events which may affect the sequential processing section of the program. These events may include:

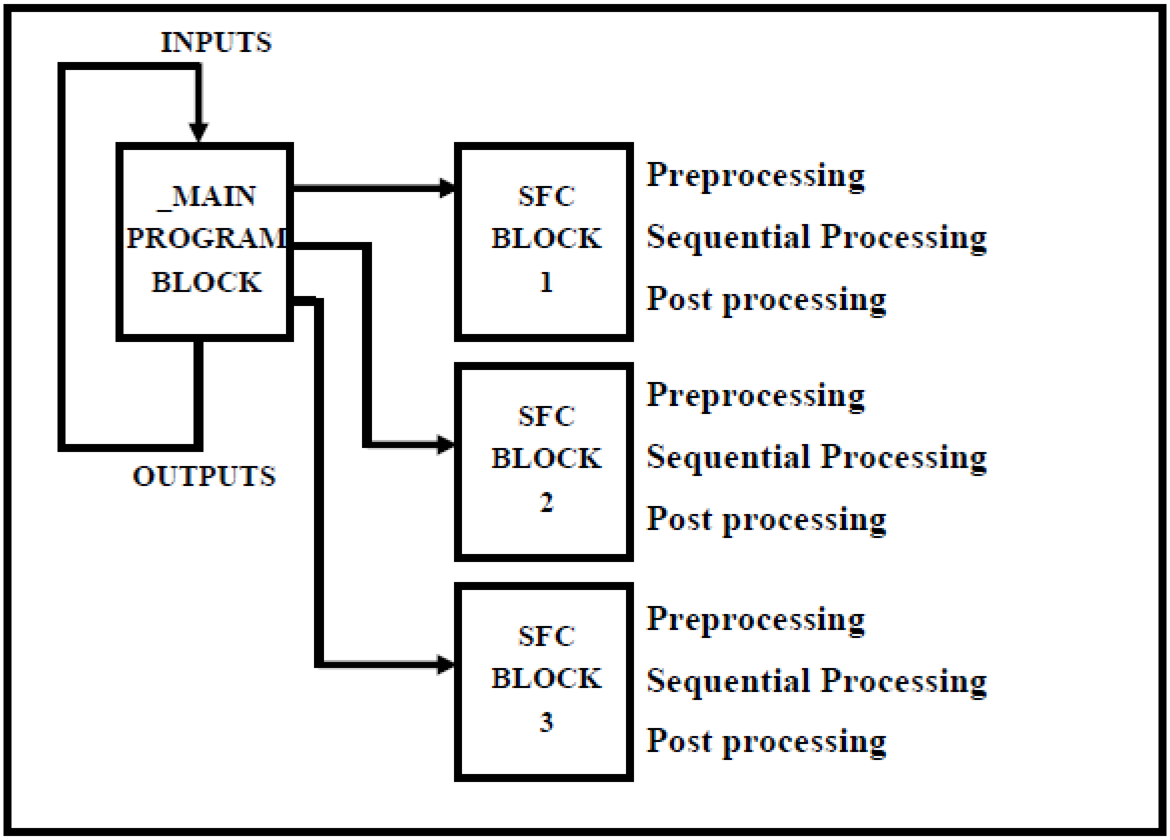
- Initialization;
- Operator commands;
- Resetting the SFC to the initial state.

Sequential Processing

This portion of the PLC scan consists of evolving the SFC to its next state and processing the action logic of any steps that become active. Only the logic associated with active steps and transitions is scanned by the PLC, leading to a significant reduction of scan time.

Post processing

This section is processed every scan after the SFC is complete. It may contain Relay Ladder Diagram (RLD) logic to process safety interlocks, etc.



Architecture of Control Software organized with SFCs

The PLC Hardware Environment

PLC systems are available in many hardware configurations, even from a single vendor, to cater to a variety of customer requirements and affordability. However, there are some common components present in each of these. These components are:

A. Power Supply - This module can be built into the PLC processor module or be an external unit. Common voltage levels required by the PLC are 5Vdc, 24Vdc, 220Vac. The voltage lends are stabilized and often the PS monitors its own health.

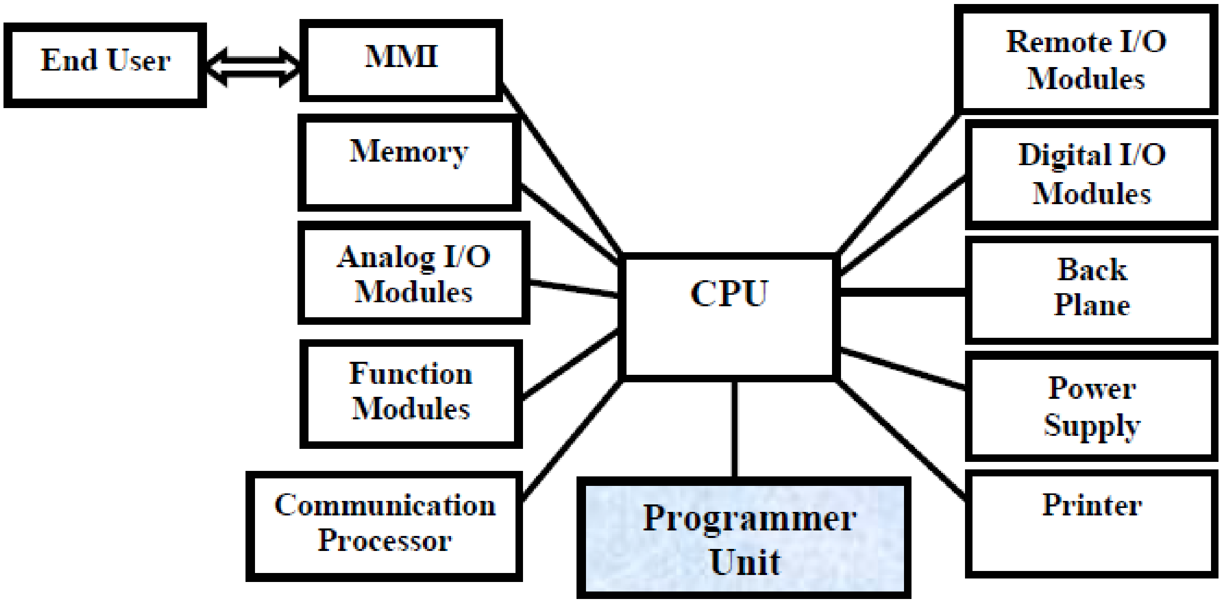
B. Processor - This is the main computing module where ladder logic ands other application programs are stored and processed.

C. Input/Output - A number of input/output modules must be provided so that the PLC can monitor the process and initiate control actions as specified in the application control programs. Depending on the size of the PLC systems the input-output subsystem can either span across several cards or even be integrated on the processor module. Some of there input-output

Input/output cards generates/accept TTL level, clean signals. Output ‘modules’ provide necessary power to the signals. Input ‘modules’ converts voltage levels, cleans up RF noise and isolates it from common mode voltages. I/O modules may also prevent over voltages to reach the CPU or low level TTL.

D. Indicator lights - These indicate the status of the PLC including power on, program running, and a fault. These are essential when diagnosing problems.

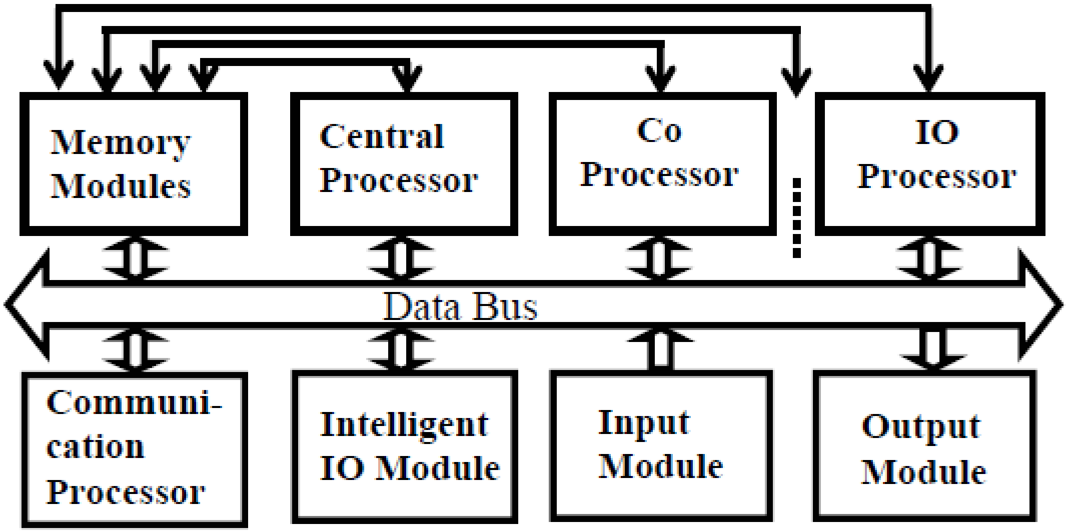
E. Rack, Slot, Backplane – These physically house and connect the electronic components of a PLC.



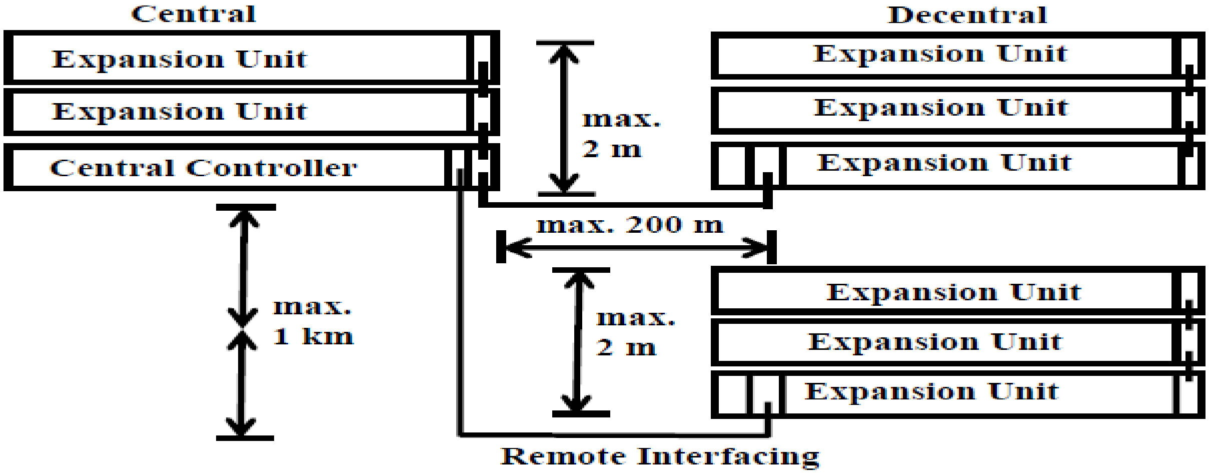
Typical Subsystems for a PLC system

The configuration of the PLC refers to the physical organization of the components. Typical configurations are listed below from largest to smallest.

A. Rack - A rack is often large and can hold multiple cards. These cards, which realize the CPU, power, communication, i/o and special function modules, are connected by a bus, often called a backplane. When necessary, multiple racks can be connected together by bus extenders. Each channel in a card can be addressed by a rack – slot – channel addressing scheme, which varies from vendor to vendor. These tend to be of highest cost, but also the most flexible and easy to maintain. The functional architecture of such a rack mounted PLC system is shown in Fig. The figure shows the various types of functional subsystems, which may or may not be on the same board, connected through a backplane. However this does not reflect the physical organization the various modules that make a PLC system. This is shown, distributed over a number of racks along with bus extension system shown in Fig. The figure shows that while direct connection may be possible for extension over small distances of a few meters. For extension over longer distances special bus extension units are needed to provide the necessary drives for reliable signal transmission over a distance.



Functional hardware organization of a PLC System



Physical layout and Bus extension System for PLCs

B. Mini - These are similar in function to PLC racks, but about half the size. Photograph of one such PLC system is shown in Fig. These generally are situated completely at one place and do not use an extended bus. Floor mounted or wall mounted.

A mini PLC System

C. Compact - A compact, all-in-one unit (about the size of a shoebox) that has limited expansion capabilities. Lower cost, and compactness make these ideal for small applications. Usually wall mounts.

D. Micro - These units can be as small as a deck of cards suitable for wall mounted or table top. They tend to have fixed quantities of I/O and limited abilities, but costs will be the lowest. Used for simple embedded applications. Often not suitable for industrial applications.

E. Software - A software based PLC requires a general purpose computer, like a PC, with an interface card. The software, utilizes the operating system resources of the computer to realize control, logic and i/o functions. An advantage of such a configuration is that it allows the PLC to be connected to sensors, other similar PLCs or to other computers across a general purpose network, such as the ethernet. The PLC can also function concurrently with other PC-based applications like a visualization software.

Processor Module

A wide range of processor modules, scalable in terms of performance and capacity, are available to meet the different needs of users. Processors manage the whole PLC station consisting of discrete input/output modules, analog modules and application-specific function modules (counting, axis control, stepper control, communication, etc.) located on one or more racks connected to the backplane. In terms of hardware, besides a CPU and possible co-processor, each processor module typically includes:

- a protected internal RAM memory which can take the application program and can be extended by memory extension cards (RAM or Flash EPROM)
- a realtime clock
- ports for connecting several devices simultaneously for purposes such as programming, human-machine interface etc.
- communication cards for various industrial communication standards such as, Modbus+ or Fieldbus, as well as serial links and Ethernet links
- Display block with LEDs, RESET button, used to activate a cold restart of the PLC system.

Typical specifications for a high end and a low end PLC processor module for a rack-based PLC system are given below.

Features	Low end	High end
No. of racks	6	24
No. of module slots	21	87
In-rack discrete I/O	512	2048
In-rack analog I/O	24	256
Application specific function modules	8	64
Process control loops	-	60
Process control channels	-	20
Network connection: TCP/IP, Modbus +, Ethernet	1	4
Fieldbus connection	0	2
Internal memory (16-bit words)	32K	176K
Memory extension (16-bit words)	64K	512K

Processor modules contain function block libraries, which can be configured to work with other modules, to realize various automation related functionality, such as,

- Counting up to 10 – 100 KHz

- PID Control with algorithms realized in different forms

- Controlled positioning for manufacturing by CNC machines with stepper / servo drives, and features such as rapid traverse / creep speed for high accuracy positioning of point to point axes, interpolation and multi axis synchronization for contouring axes

- Input/output: These may be categorized as digital / analog depending on the nature of the signal or as local/remote/networked, depending on the interface through which it is acquired. These are described in detail below

Input Module

Input modules convert process level signals from sensors (e.g. voltage face Contacts, 0-24v Dc, 4 – 20mA), to processor level digital signals such as 5V or 3.3 V. They also accept direct inputs from thermocouples and RTDs in the analog case, and limit switches or encoders in the digital case. Naturally, therefore these modules include circuitry for galvanic isolation, such as those using optocouplers.

Galvanic isolation-

Analog input modules

Analog input modules convert analog process level signals to digital values, which are then processed by the digital electronic hardware of the programmable controller. A set of typical parameters that define an analog input module are shown in Table . The analog modules sense 8/16 analog signals in the range ± 5 V, ± 10 V or 0 to 10 V. Each channel can either be single-ended, or differential. For single ended channels only one wire is connected to a channel terminal. The analog voltage on each channel terminal that is sensed is referred to a common ground.

In the case of differential channels, each channel terminal involves two wires and the voltage between the pair of wires is sensed. Thus both the wires can be at different voltages and only their difference is sensed and converted to digital. Differential channels are more accurate but consume more electronic resources of the module for their processing. Often these modules also house channels that output analog/digital signals, as well as excitation circuitry for sensors such as RTDs.

An analog module typically contains:

- Analog to digital (A/D) converters
- Analog multiplexers and simultaneous sample-hold (S/H)
- Analog Signal termination
- PLC bus ports

Analog IO Module

Module Parameter	Type/Number/Typical Value
Number of input	8/16 voltage/current/Pt 100/ RTD
Galvanic isolation	Yes /No
Input ranges	±50 mV to ±10 V; ±20 mA; Pt 100
Input impedance for various ranges (ohm)	±50mV: > 10 M ; ±10 V: > 50k; ±20 mA : 25; Pt 100 : > 10 M
Types of sensor connections	2-wire connection; 4-wire connection for Pt 100

Data format Conversion principle Conversion time	11 bits plus sign or 12 bit 2’s complement Integrating /successive approximation In ms (integrating) , µs (successive approx.)

Digital Input Modules

The digital inputs modules convert the external binary signals from the process to the internal digital signal level of programmable controllers. Digital input channel processing involves isolation and signal conditioning before inputting to a comparator for conversion to a 0 or a 1. The typical parameters that define a digital input module are shown in tabular form along with typical values in Table.

Module Parameter	**Typical Values**
Number of input	16/32
Galvanic isolation	yes
Nominal input voltage	+ 24 V DC
Input voltage range - “0” signal - “1” signal	-33…+7V +13…+33V
Input current	Typically in mA
Delay	Typically in µs
Maximum cable length	Typically within 1000m

Output Modules

Outputs to actuators allow a PLC to cause something to happen in a process. Common actuators include:

1. Solenoid Valves - logical outputs that can switch a hydraulic or pneumatic flow.

2. Lights - logical outputs that can often be powered directly from PLC boards.

3. Motor Starters - motors often draw a large amount of current when started, so they require motor starters, which are basically large relays.

4. Servo Motors - a continuous output from the PLC can command a variable speed or position to a servo motor drive system.

The outputs from these modules may be used to drive such actuators. Consequently, they include circuitry for current / power drive using solid-state electronics such as transistors for DC outputs or triacs for AC outputs. Continuous outputs require output cards with D/A converters. Sometimes they also provide potential free relay contacts (NO/NC), which may be used to drive higher power actuators using a separate power source. Since these modules straddle across the processor and the output power circuit, these must provide isolation.

However, most often, output modules act as modulators of the actuator power, which is actually applied to the equipment, machine or plant. External power supplies are connected to the output card and the card will switch the power on or off for each output. Typical output voltages are 120V ac, 24V dc, 12-48V ac/dc, 5V dc (TTL) or 230V ac. These cards typically have 8 to 16 outputs of the same type and can be purchased with different current ratings. A common choice when purchasing output cards is relays, transistors or triacs. Relays are the most flexible output devices. They are capable of switching both AC and DC outputs. But, they are slower (about 10ms switching is typical), they are bulkier, they cost more, and they wear out after a large number of cycles. Relays can switch high DC and AC voltage levels while maintaining isolation. Transistors are limited to DC outputs, and triacs are limited to AC outputs. Transistor and triac outputs are called switched outputs. In this case, a voltage is supplied to the PLC card, and the card switches it to different outputs using solid-state circuitry (transistors, triacs, etc.). Triacs are well suited to AC devices requiring less than 1A. Transistor outputs use NPN or PNP transistors up to 1A typically. Their response time is well under 1ms.

Analog Output Module

Analog output modules convert digital values from the PLC processor module into an analog signal required by the process. These modules therefore require a D/A converter for providing analog outputs. However, typically, servo-amplifiers for power amplification, required for driving high current loads directly, are not integrated on-board. Front connectors are used for terminating the signal cables. Modules and front connectors may be inserted and removed under power. The output signals can be disabled by means of an enable input. The last value then remains latched. Typical parameters that define an analog output module are shown in Table along with typical values.

Digital Output Module

Digital output modules convert internal signal levels of the programmable controllers into the binary signal levels required externally by the process. Output can be DC or AC. Up to 16 outputs can be connected in parallel. Indication for short-circuits, fuse blowing etc. are often provided.

Number of outputs	8 voltage and current output
Galvanic isolation	yes
Output ranges (rated values)	**± 10 V; 0…20 mA**
Load resistance - for voltage outputs min. - for current outputs max.	3.3 k 300
Digital representation of the signal	11 bits plus sign
Conversion time	In µs
Short-circuit protection	yes
Short-circuit current approx.	25 mA (for a voltage output)

Open-circuit voltage approx.	18 V (for a current output)
Linearity in the rated range	±0.25% + 2 LSB
Cable length max.	200 m

The typical parameters that define a digital output module are shown in Table along with typical values

Module Parameter	**Typical Value**
1.Number of outputs	**16/32**
2.Galvanic isolation	yes
Rated value of Supply voltage Permissible range	+ 24 V DC 20-30 V
Max. output current for “1” signal	0.5 A
Short-circuit protection	Yes
Max. switching frequency for resistive loads, lamps, inductive loads, respectively, in Hz.	100/11/2 Hz (at 0.3 A)
“0” signal level max “1” signal level max.	+3V $V_{pp}^{-}1.5$ V
Max. cable length (unshielded)	400 m

Function Modules

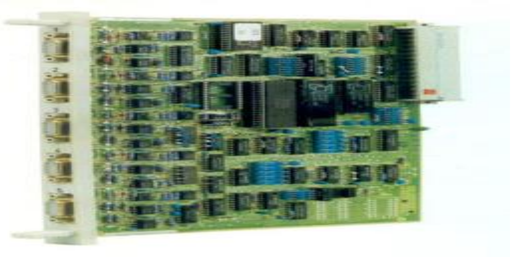
For high speed i/o tasks such as one required to measure speed by counting pulses from shaft angle encoders, or for precision position control applications, independent i/o modules that execute tasks independently of the central processor are required to meet the timing requirements of the i/o. The signal preprocessing “intelligent” I/O- modules make it possible to count fast impulse trains, to acquire and process travel increments, speed and time measure etc., i.e. they take on the critical timing control tasks which normally can’t be carried out fast enough by the central processor with its programmable logic control, as well as its primary logic control functions. These modules not only relieve the central processor of additional tasks, they also provide fast and specialized solutions to some common control problems. The processing of the signals is carried out primarily by the appropriate I/O- modules, which frequently operate with their own processor. Below we discuss two such modules, which are used with PLCs to handle specific high performance automation functions.

A. **A Count Module** is employed where pulses at high frequency have to be counted, i.e. when machines run fast. It can also be applied to output fast pulse trains or realize accurate timing signals.

B. **A Loop Controller Module** is primarily used where high speed closed loop control is required, such as with controlled drives. The preprogrammed, parameterized functions available with the module (e.g. for ramp-function generation, speed regulation,signal limit monitoring) can be easily parameterized via graphical interfaces by a programmer.

Count Module

A count module senses fast pulses, from sources such as shaft angle encoders, through several input ports. Counting frequency can be as high as 2 MHz and a typically, a counter of length 16 bit or more can count up and down. Counter modules can often also be applied for time and frequency measurement and as a frequency divider.



A high speed counter module

Typical counter module hardware contains, among possible other things, an interface to the processor through the system bus, a counter electronics block, a quartz controlled frequency generator and a frequency divider. For example, it may contain, say, 5 counters with, say, 16 bits, each of which are cascadable. In this way, up to 80 bit can be counted in various codes. Thus, decimals up to about 10^{24} can be counted. Each port input can be switched on to the counter at random. It is possible to place a frequency divider from 1 to 16, between the port input and the counter. The frequency of an internal frequency generator can be directed either straight to a counter or via the frequency divider to a port input. On reaching the terminal count, the counter outputs a level or edge signal.

For each counter there are a number of different operating modes, which can be set by a user program. With a comparator and an alarm register, a number of count values can be compared and under defined conditions configured to turn on a process alarm.

A counter can be programmed in many ways, such as:

- Count mode binary or BCD coded
- Count once or cyclically
- Count on rising or falling edge
- Count up or down
- Counting of internal clock or external pulses

Loop Controller Module

A loop controller module is suitable for solving fast control loop problems. A typical module can process several control loops with sampling times varying between a few milliseconds to several seconds. The process output values are measured via analog input ports and are compared with the set point values. The power circuits of the actuator units are driven through analog output ports. Such a module contains a microprocessor, which controls the sensing and processing of the process output and set point values and computes the control law and outputs the manipulated variables. The operating configurations of the loop controller module are assigned with a programmer and committed to memory located on the module. The central controller provides set point values, parameters and control commands and reads the output values.

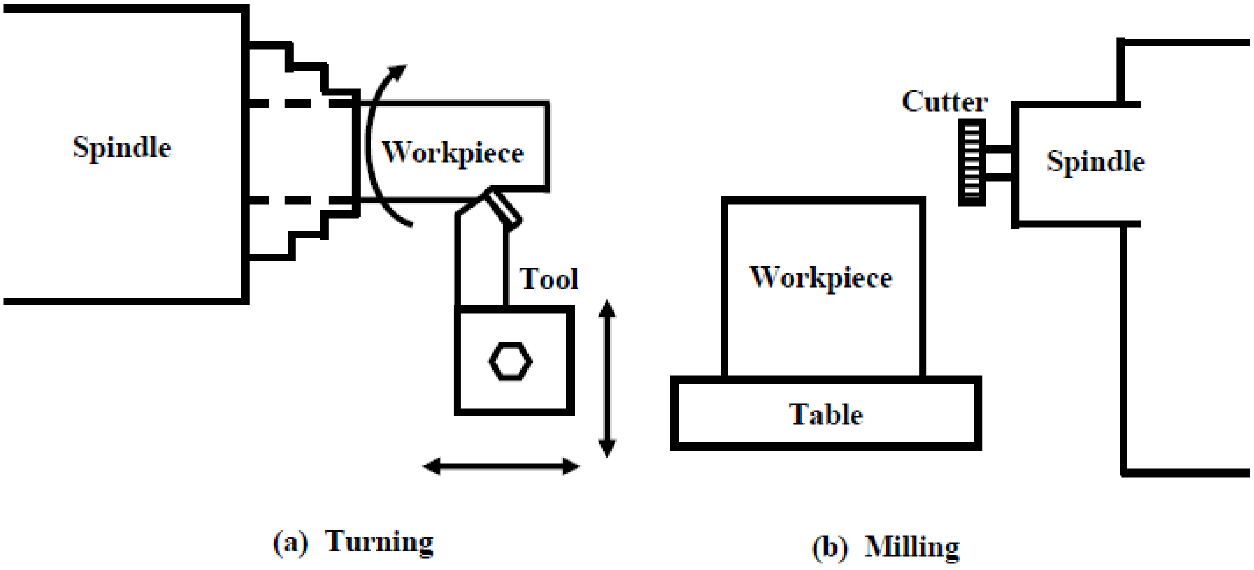
The application software for the module can be structured in terms of standard close loop functions (e.g. ramp function generator, speed controller, etc.). These standard functions can be interconnected to a closed loop structures with the aid of a programmer interface and are the resulting control code automatically compiled and downloaded to memory on the module. The microprocessor executes the standard functions in accordance with the designed closed loop structure for an application such as a motor drive or a standard cascade process control loop. A drive loop controller would comprise all necessary functions for the control of a drive, while a cascade loop controller would consist of a cascade of two control loops. The outer loop controller can, for an instant, be used for closed loop position control, while the inner loop for control of rotational speed control. Each loop controller can be equipped with P, D, PI, PD or PID algorithms. As additional functions there may exist limit monitoring indicators limit monitoring indicators, which for example, can monitor the actual armature current value, for thermal supervision of the drive.

UNIT - IV

CNC MACHINES AND ACTUATORS

Introduction to Computer Numerically Controlled (CNC) Machines

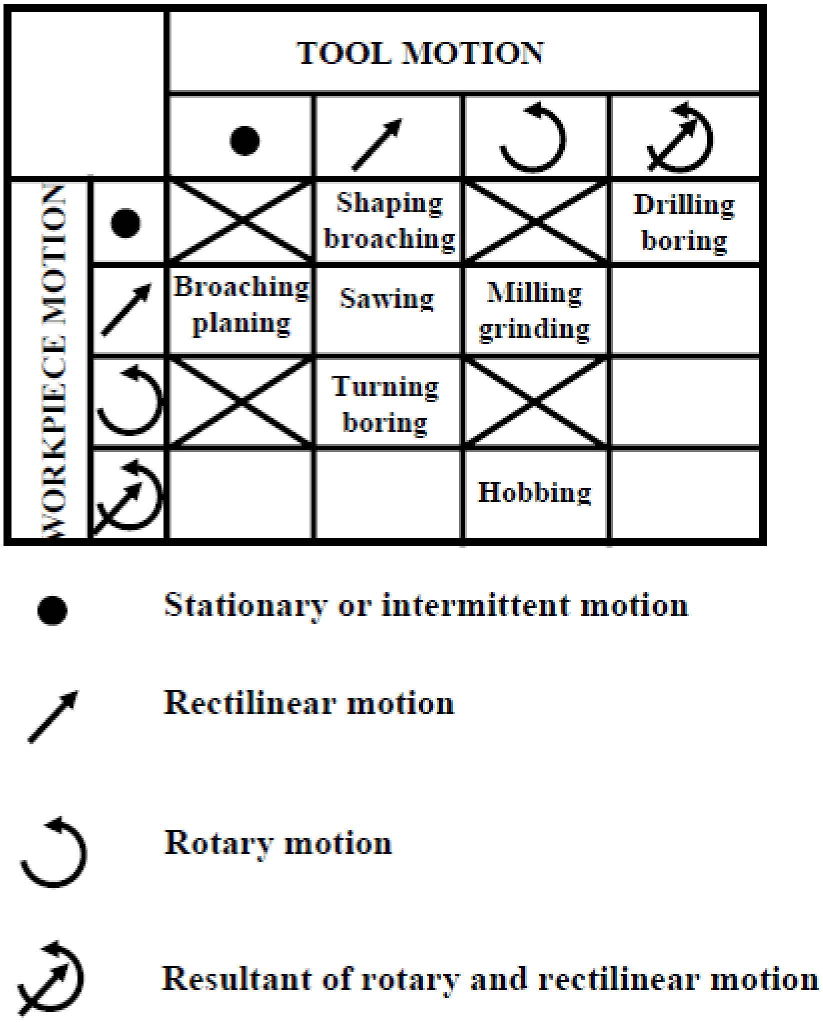
Introduction



Drive in a metal cutting

Introductory Concepts of Machining

Machining is basically removal of material, most often metal, from the workpiece, using one or more cutting tools to achieve the desired dimensions. There are different machining processes, such as, turning, milling, boring etc. In all these cases metal is removed by a shearing process, which occurs due to the relative motion between the workpiece and the tool. Generally, one of the two rotates at designated and generally high speed, causing the shearing of material (known as chips), from the workpiece. The other moves relatively slowly to effect removal of metal throughout the workpiece. For example, as seen above in a turning operation of lathes, the "job" or the workpiece rotates in a chuck, while the tool moves in two dimensions translationally. On the other hand, in milling, it is the cutter which rotates on a spindle, while the workpiece, which is fastened to a table, moves in X-Y dimensions. While, a precise and high speed rotational motion is needed for good finish of the machined surface, for dimensional accuracy, precise position and velocity control of the table drive are essential.

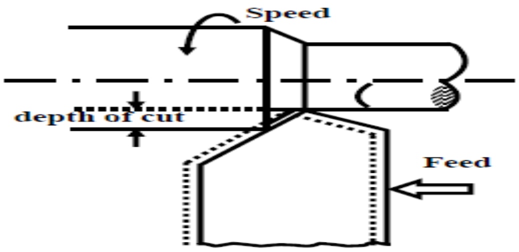


		TOOL MOTION			
		●	↗	↺	↗↺
WORKPIECE MOTION	●	✕	Shaping broaching	✕	Drilling boring
	↗	Broaching planing	Sawing	Milling grinding	
	↺	✕	Turning boring	✕	
	↗↺			Hobbing	

●	Stationary or intermittent motion
↗	Rectilinear motion
↺	Rotary motion
↗↺	Resultant of rotary and rectilinear motion

Nature of motion of the Job and the Tool for various Metal Cutting Processes

For all metal-cutting processes, the cutting speed, feed, and depth of cut are important parameters. The figure below shows the important geometry for the turning process. The cutting speed, which is a measure of the part cut surface speed relative to the tool. Speed is a velocity unit for the translational motion, which is may be stated in or meters/min. The depth of cut, DOC is the depth that the tool is plunged into the surface. Feed defines the relative lateral movement between the cutting tool and the workpiece. Thus, together with depth of cut, feed decides the cross section of the material removed for every rotation of the job or the tool, as the case may be. Feed is the amount of material removed for each revolution or per pass of the tool over the workpiece and is measured in units of length/revolution, length/pass or other appropriate unit for the particular process.



Speed, feed and depth of cut for turning operation

What is Computer Numerical Control?

Modern precision manufacturing demands extreme dimensional accuracy and surface finish. Such performance is very difficult to achieve manually, if not impossible, even with expert operators. In cases where it is possible, it takes much higher time due to the need for frequent dimensional measurement to prevent overcutting. It is thus obvious that automated motion control would replace manual “handwheel” control in modern manufacturing. Development of computer numerically controlled (CNC) machines has also made possible the automation of the machining processes with flexibility to handle production of small to medium batch of parts.

In the 1940s when the U.S. Air Force perceived the need to manufacture complex parts for high-speed aircraft. This led to the development of computer-based automatic machine tool controls also known as the Numerical Control (NC) systems. Commercial production of NC machine tools started around the fifties and sixties around the world. Note that at this time the microprocessor has not yet been invented.

Initially, the CNC technology was applied on lathes, milling machines, etc. which could perform a single type of metal cutting operation. Later, attempt was made to handle a variety of workpieces that may require several different types machining operations and to finish them in a single set-up. Thus CNC machining Centres capable of performing multiple operations were developed. To start with, CNC machining centres were developed for machining prismatic components combining operations like milling, drilling, boring and tapping. Gradually machines for manufacturing cylindrical components, called turning centers were developed.

Numerical Control

Automatically controlling a machine tool based on a set of pre-programmed machining and movement instructions is known as numerical control, or NC.

In a typical NC system the motion and machining instructions and the related numerical data, together called a *part program*, used to be written on a punched tape. The part program is arranged in the form of blocks of information, each related to a particular operation in a sequence of operations needed for producing a mechanical component. The punched tape used to be read one block at a time. Each block contained, in a particular syntax, information needed for processing a particular machining instruction such as, the segment length, its cutting speed, feed, etc. These pieces of information were related to the final dimensions of the workpiece (length, width, and radii of circles) and the contour forms (linear, circular, or other) as per the drawing. Based on these dimensions, motion commands were given separately for each axis of motion. Other instructions and related machining parameters, such as cutting speed, feed rate, as well as auxiliary functions related to coolant flow, spindle speed, part clamping, are also provided in part programs depending on manufacturing specifications such as tolerance and surface finish. Punched tapes are mostly obsolete now, being replaced by magnetic disks and optical disks.

Computer Numerically Controlled (CNC) machine tools, the modern versions of NC machines have an embedded system involving several microprocessors and related electronics as the Machine Control Unit (MCU). Initially, these were developed in the seventies in the US and Japan. However, they became much more popular in Japan than in the US. In CNC systems multiple microprocessors and programmable logic controllers work in parallel for simultaneous servo position and velocity control of several axes of a machine for contour cutting as well as monitoring of the cutting process and the machine tool. Thus, milling and boring machines can be fused into versatile machining centers. Similarly, turning centers can realize a fusion of various types of lathes. Over a period of time, several additional features were introduced, leading to increased machine utilisation and reduced operator intervention. Some of these are:

(a) Tool/work monitoring: For enhanced quality, avoidance of breakdowns.

(b) Automated tool magazine and palette management: For increased versatility and reduced operator intervention over long hours of operation

(c) Direct numerical control (DNC): Uses a computer interface to upload and download part programs in to the machine automatically.

Advantages of a CNC Machine

CNC machines offer the following advantages in manufacturing.

• *Higher flexibility*: This is essentially because of programmability, programmed control and facilities for multiple operations in one machining centre,

• *Increased productivity*: Due to low cycle time achieved through higher material removal rates and low set up times achieved by faster tool positioning, changing, automated material handling etc.

• *Improved quality*: Due to accurate part dimensions and excellent surface finish that can be achieved due to precision motion control and improved thermal control by automatic control of coolant flow.

- *Reduced scrap rate*: Use of Part programs that are developed using optimization procedures
- *Reliable and Safe operation*: Advanced engineering practices for design and manufacturing, automated monitoring, improved maintenance and low human interaction
- *Smaller footprint*: Due to the fact that several machines are fused into one.

On the other hand, the main disadvantages of NC systems are

- Relatively higher cost compared to manual versions
- More complicated maintenance due to the complex nature of the technologies
- Need for skilled part programmers.

The above disadvantages indicate that CNC machines can be gainfully deployed only when the required product quality and average volume of production demand it.

Classification of NC Systems

CNC machine tool systems can be classified in various ways such as :

1. Point-to-point or contouring : depending on whether the machine cuts metal while the workpiece moves relative to the tool

2. Incremental or absolute : depending on the type of coordinate system adopted to parameterise the motion commands

3. Open-loop

Contouring systems

In contouring systems, the tool is cutting while the axes of motion are moving, such as in a milling machine. All axes of motion might move simultaneously, each at a different velocity. When a nonlinear path is required, the axial velocity changes, even within the segment. For example, cutting a circular contour requires sinusoidal rates of change in both axes. The motion controller is therefore required to synchronize the axes of motion to generate a predetermined path, generally a line or a circular arc. A contouring system needs capability of controlling its drive motors independently at various speeds as the tool moves towards the specified position. This involves simultaneous motion control of two or more axes, which requires separate position and velocity loops. It also requires an interpolator program that generates the position and velocity setpoints for the two drive axes, continuously along the contour.

In modern machines there is capability for programming machine axes, either as point-to-point or as continuous (that is contouring)

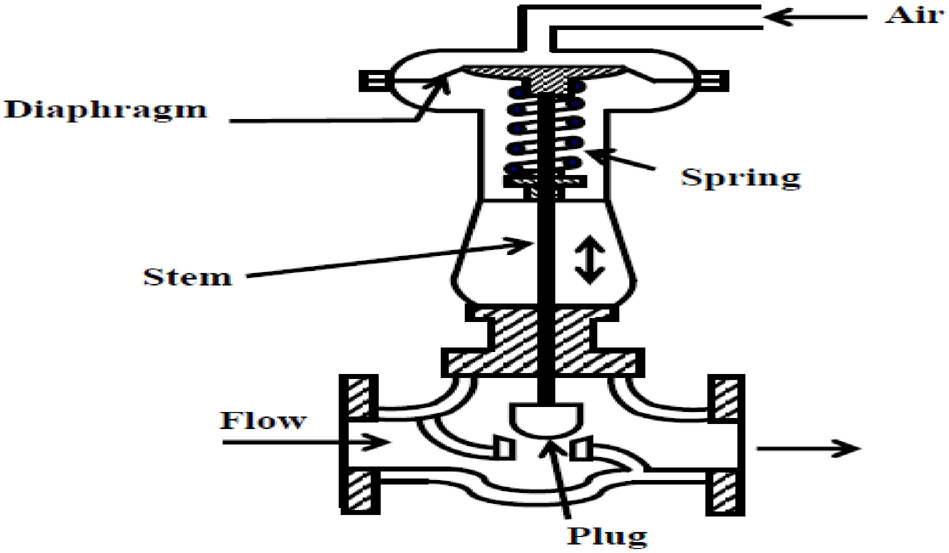
Before the next type of classification is introduced, it is necessary to present the basic coordinate system conventions in a machine tool.

Control Valves

The control action in any control loop system, is executed by the final control element. The most common type of final control element used in chemical and other process control is the control valve. A control valve is normally driven by a diaphragm type pneumatic actuator that throttles the flow of the manipulating variable for obtaining the desired control action. A control valve essentially consists of a plug and a stem. The stem can be raised or lowered by air pressure and the plug changes the effective area of an orifice in the flow path.

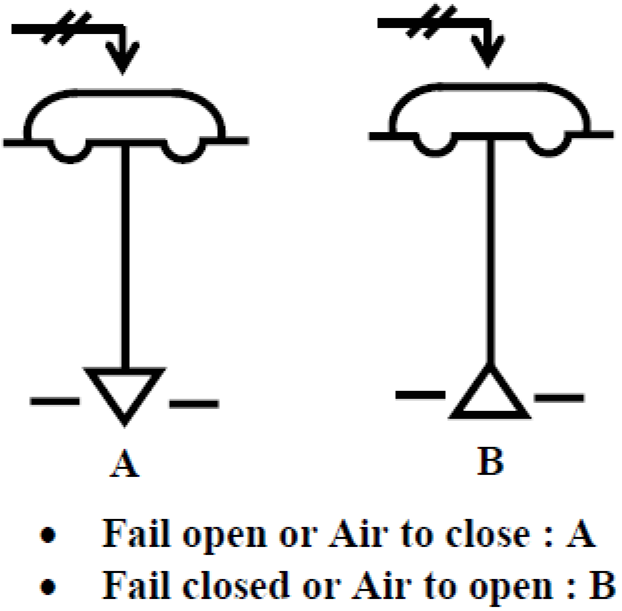
Classifications

Control valves are available in different types and shapes. They can be classified in different ways; based on: (a) action, (b) number of plugs, and (c) flow characteristics.



Control valve

(a) Action: Control valves operated through pneumatic actuators can be either (i) air to open, or (ii) air to close. They are designed such that if the air supply fails, the control valve will be either fully open, or fully closed, depending upon the safety requirement of the process. For example, if the valve is used to control steam or fuel flow, the valve should be shut off completely in case of air failure. On the other hand, if the valve is handling cooling water to a reactor, the flow should be maximum in case of emergency. The schematic arrangements of these two actions are shown in Fig. Valve A are air to close type, indicating, if the air fails, the valve will be fully open. Opposite is the case for valve B.

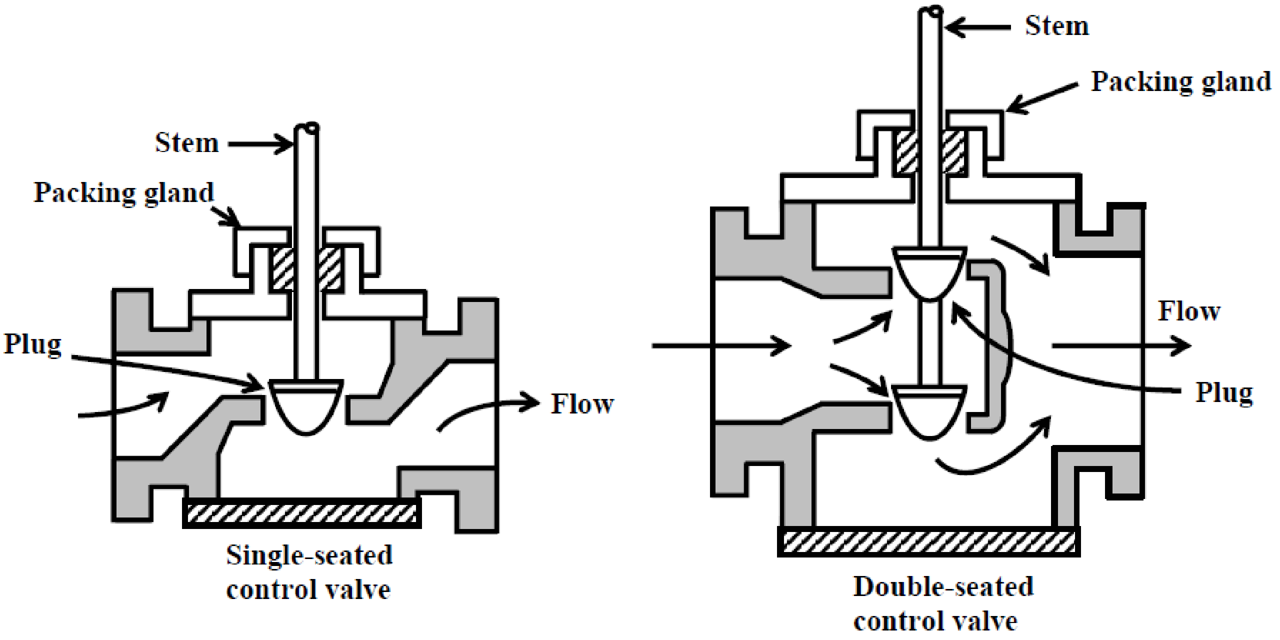


- Fail open or Air to close : A
- Fail closed or Air to open : B

Air to open and Air to close valves

(b) Number of plugs: Control valves can also be characterized in terms of the number of plugs present, as *single-seated valve* and *double-seated valve*. The difference in construction between a single seated and double-seated valve are illustrated in Fig. only one plug is present in the control valve, so it is single seated valve. The advantage of this type of valve is that, it can be fully closed and flow variation from 0 to 100% can be achieved. But looking at its construction, due to the pressure drop across the orifice a large upward force is present in the orifice area, and as a result, the force required to move the valve against this upward thrust is also large. Thus this type of valves is more suitable for small flow rates. On the other hand, there are two plugs in a double-seated valve; flow moves upward in one orifice area, and downward in the other orifice. The resultant upward or downward thrust is almost zero. As a result, the force required to move a double-seated valve is comparatively much less.

But the double-seated valve suffers from one disadvantage. The flow cannot be shut off completely, because of the differential temperature expansion of the stem and the valve seat. If one plug is tightly closed, there is usually a small gap between the other plug and its seat. Thus, single-seated valves are recommended for when the valves are required to be shut off completely. But there are many processes, where the valve used is not expected to operate near shut off position. For this condition, double-seated valves are recommended.



Single-seated control valve

Double-seated control valve

Single-seated and double-seated valves

(c) Flow Characteristics: It describes how the flow rate changes with the movement or lift of the stem. The shape of the plug primarily decides the flow characteristics. However, the design of the shape of a control valve and its shape requires further discussions. The flow characteristic of a valve is normally defined in terms of (a) inherent characteristics and (b) effective characteristics. An inherent characteristic is the ideal flow characteristics of a control valve and is decided by the shape and size of the plug. On the other hand, when the valve is connected to a pipeline, its overall performance is decided by its effective characteristic.

Ideal Characteristics

The control valve acts like an orifice and the position of the plug decides the area of opening of the orifice. Recall that the flow rate through an orifice can be expressed in terms of the upstream and downstream static pressure heads as:

$$q = K_1 a\sqrt{2g(h_1 - h_2)} \qquad (1)$$

where q = flow rate in m^3/sec.

K_1 = flow coefficient

a = area of the control valve opening in m^2

h_1 = upstream static head of the fluid in m

h_2 = downstream static head of the fluid in m

g = acceleration due to gravity in m/sec^2.

Now the area of the control valve opening (a) is again dependent on the stem position, or the lift. So if the upstream and downstream static pressure heads are somehow maintained constant, then the flow rate is a function of the lift (z), i.e.

$$q = f(z) \qquad (2)$$

The shape of the plug decides, how the flow rate changes with the stem movement, or lift; and the characteristics of q vs. z is known as the inherent characteristics of the valve.

Let us define

$$m = \frac{q}{q_{\max}} \qquad \text{and} \qquad x = \frac{z}{z_{\max}}$$

where, $q_{\max}$ is the maximum flow rate, when the valve is fully open and $z_{\max}$ is the corresponding maximum lift. So eqn. (2) can be rewritten in terms of m and x as:

$$m = f(x) \qquad (3)$$

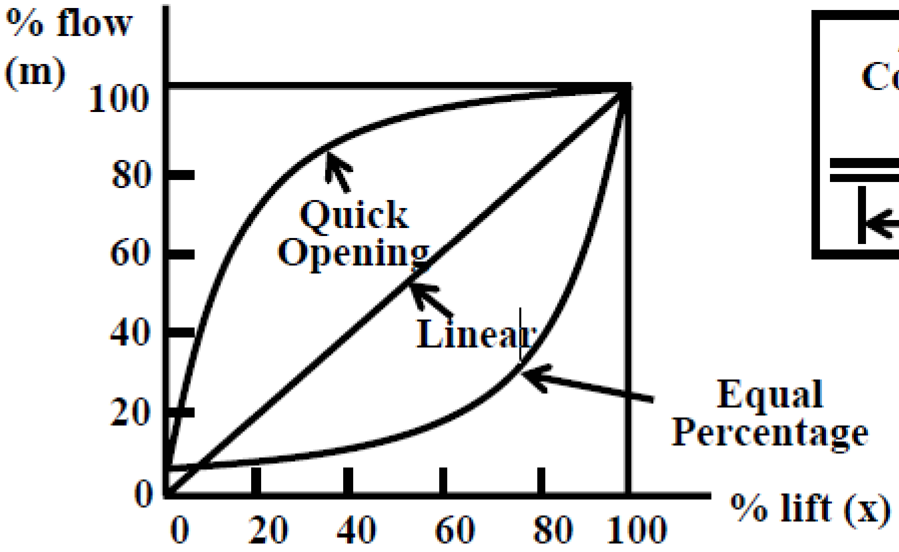
and the valve sensitivity is defined as dm/dx, or the slope of the curve m vs. x. In this way, the control valves can be classified in terms of their m vs. x characteristics, and three types of control valves are normally in use. They are:

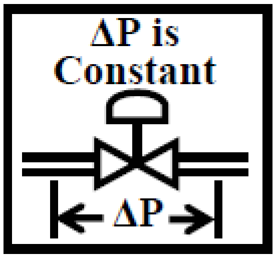
(a) Quick opening

(b) Linear

(c) Equal Percentage.

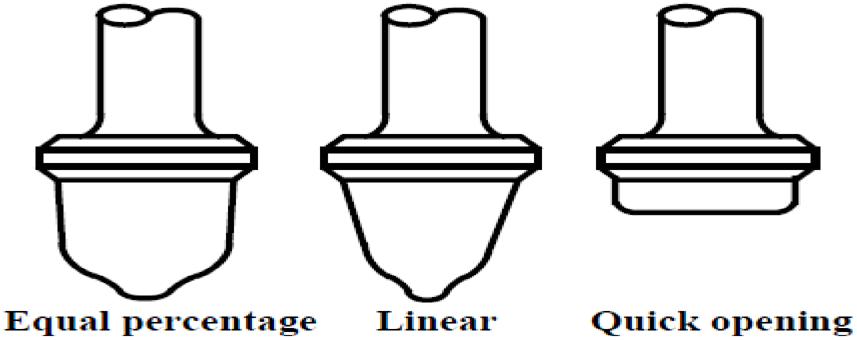
The characteristics of these control valves are shown in Fig. It has to be kept in mind that all the characteristics are to be determined after maintaining constant pressure difference across the valve as shown in Fig.





Flow characteristics of control valves

Different flow characteristics can be obtained by properly shaping the plugs. Typical shapes of the three types of valves are shown in Fig.



Valve plug shapes for the three common flow characteristics.

For a linear valve,*dm/dx*= 1, as evident from Fig.5 and the flow characteristics is linear throughout the operating range. On the other hand, for an equal percentage valve, the flow characteristics is mathematically expressed as:

$$\frac{dm}{dx} = \beta m \qquad (4)$$

where β is a constant.

The above expression indicates, that the slop of the flow characteristics is proportional to the present flow rate, justifying the term equal percentage. This flow characteristics is linear on a semilog graph paper. The minimum flow rate m_0 (flow rate at x=0) is never zero for an equal percentage valve and m can be expressed as:

$$m = m_0\, e^{\beta x} \qquad (5)$$

Rangeability of a control valve is defined as the ratio of the maximum controllable flow and the minimum controllable flow. Thus:

Rangeability = maximumcontrollableflow/minimum controllable flow

Rangeability of a control valve is normally in between 20 and 70.

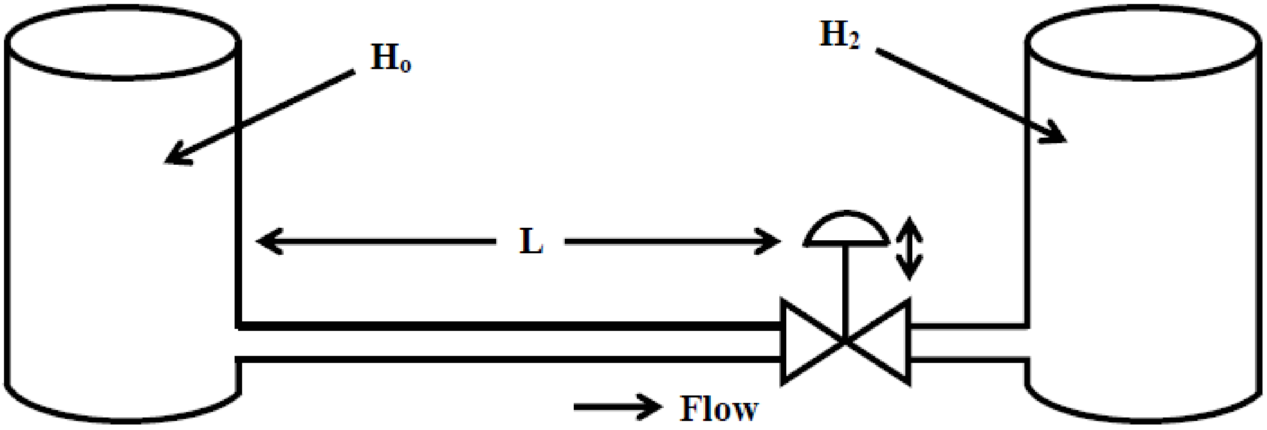
Effective Characteristics

So far we have discussed about the ideal characteristics of a control valve. It is decided by the shape of the plug, and the pressure drop across the valve is assumed to be held constant. But in practice, the control valve is installed in conjunction with other equipment, such as heat exchanger, pipeline, orifice, pump etc. The elements will have their own flow vs. pressure characteristics and cause additional frictional loss in the system and the effective characteristics of the valve will be different from the ideal characteristics. In order to explain the deviation, let us consider a control valve connected with a pipeline of length L in between two tanks, as shown in Fig. We consider the tanks are large enough so that the heads of the two tanks H_0 and H_2 can be assumed to be constant. We also assume that the ideal characteristic of the control valve is linear. From eqn. (1), we can write for a linear valve:

$$K_1 a = Kz$$

where K is a constant and z is the stem position or lift.

Now the pipeline will experience some head loss that is again dependent on the velocity of the fluid.



Effect of friction loss in pipeline for a control valve

The head loss Δh_L will affect the overall flow rate q and eqn.(1) can be rewritten as:

$$q = \left[K\sqrt{2g(H_0 - H_2 - \Delta h_L} \right] z \qquad (6)$$

The head loss (in m) can be calculated from the relationship:

$$\Delta h_L = F\frac{L}{D}\frac{v^2}{2g} \qquad (7)$$

where F = Friction coefficient
L = Length of the pipeline in m
D = inside diameter of the pipeline in m
v = velocity of the flow in m.

Further, the velocity of the fluid can be related to the fluid flow q (in m^3/sec) as:

$$v = \frac{q}{\frac{\pi}{4}D^2} \qquad (8)$$

Combining (7) and (8), we can write:

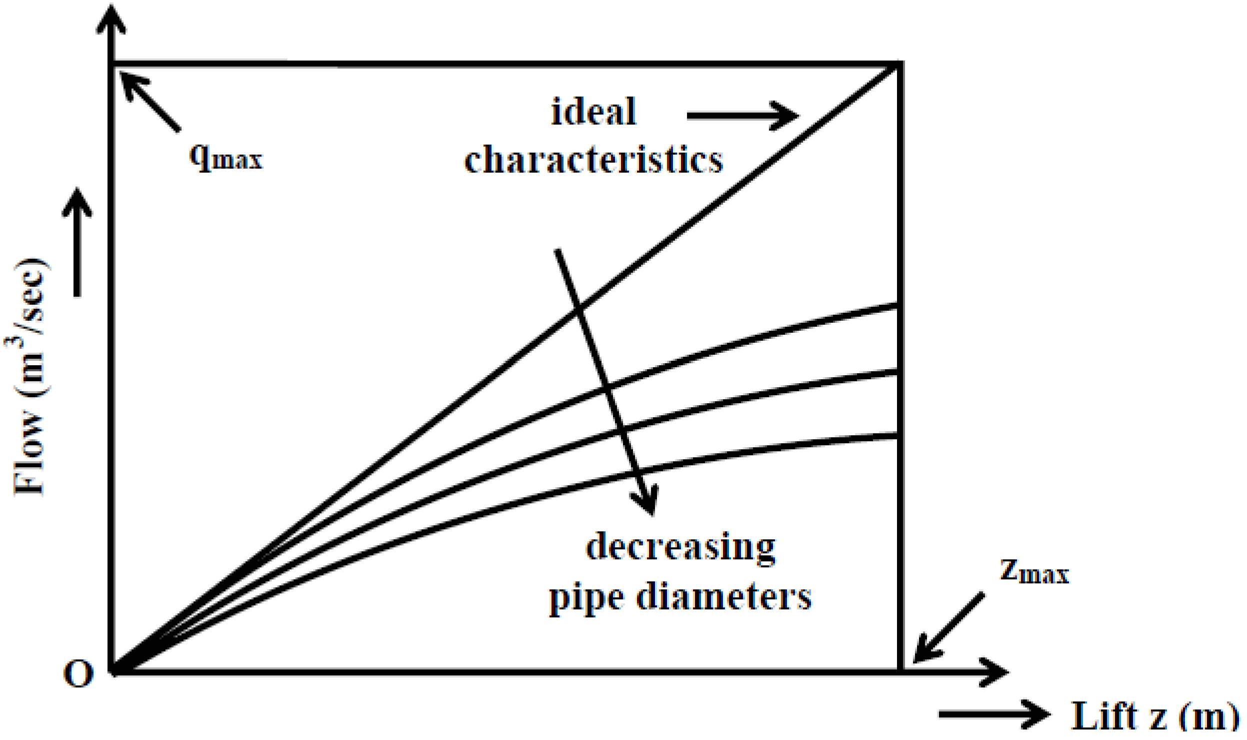
$$\Delta h_L = \frac{8}{\pi^2}\frac{FL}{gD^5}q^2 \qquad (9)$$

Substituting (9) in (6) and further simplifying, one can obtain:

$$q = \left[K\sqrt{\frac{2g(H_0 - H_2)}{1+\alpha z^2}} \right] z \qquad (10)$$

where $\alpha = \dfrac{16FLK^2}{\pi^2 D^5}$

From (10), it can be concluded that *q* is no longer linearly proportional to stem lift *z*, though the ideal characteristics of the valve is linear. This nonlinearity of the characteristics is dependent on the diameter of the pipeline *D*; i.e. smaller the pipe diameter, larger is the value of α and more is the nonlinearity. The nonlinearity of the effective valve characteristics can be plotted as shown in Fig.

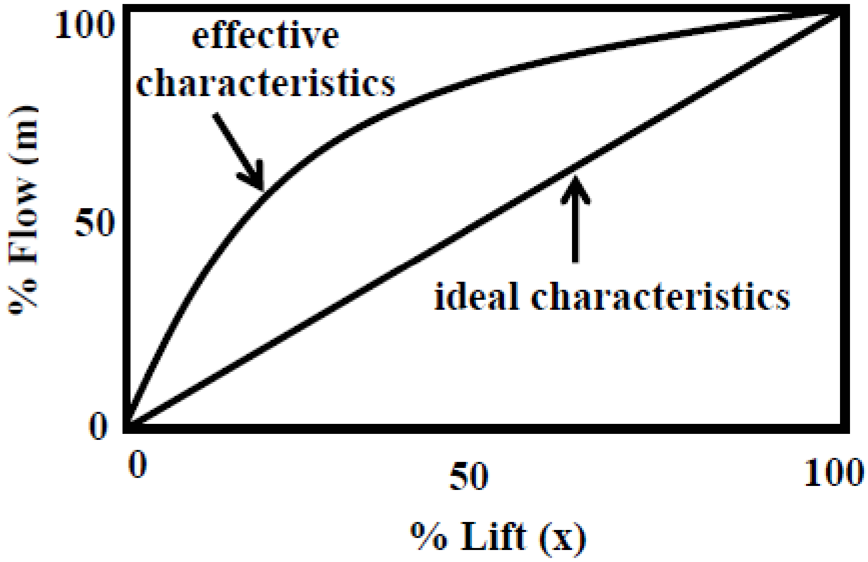


Effect of pipeline diameter on the effective flow characteristics of the control valve

The nonlinearity introduced in the effective characteristics can be reduced by mainly (i) increasing the line diameter, thus reducing the head loss, (ii) increasing the pressure of the source H_0, (iii) decreasing the pressure at the termination H_2.

The effective characteristics of the control valve shown in Fig. are in terms of absolute flow rate.

If we want to express the effective characteristics in terms of $m\,(=\dfrac{q}{q_{\max}})$ in eqn. (3) deviation from the ideal characteristics will also be observed. Linear valve characteristics will deviate upwards, as shown in Fig. An equal percentage valve characteristic will also shift upward from its ideal characteristic; thus giving a better linear response in the actual case.



Comparison of ideal and effective characteristics for a linear valve

Thus linear valves are recommended when pressure drop across the control valve is expected to be fairly constant. On the other hand, equal percentage valves are recommended when the pressure drop across the control valve would not be constant due to the presence of series resistance in the line. As the line loss increases, the effective characteristics of the equal percentage valve will move closer to the linear relationship in *m* vs. *x* characteristics.

Hydraulic Actuation Systems - I: Principle and Components

Hydraulic Actuators, as used in industrial process control, employ hydraulic pressure to drive an output member. These are used where high speed and large forces are required. The fluid used in hydraulic actuator is highly incompressible so that pressure applied can be transmitted instantaneously to the member attached to it.

It was not, however, until the 17th century that the branch of hydraulics with which we are to be concerned first came into use. Based upon a principle discovered by the French scientist Pascal, it relates to the use of confined fluids in transmitting power, multiplying force and modifying motions.

Then, in the early stages of the industrial revolution, a British mechanic named Joseph Bramah utilized Pascal's discovery in developing a hydraulic press. Bramah decided that, if a small force on a small area would create a proportionally larger force on a larger area, the only limit to the force a machine can exert is the area to which the pressure is applied.

Principle Used in Hydraulic Actuator System

Pascal's Law

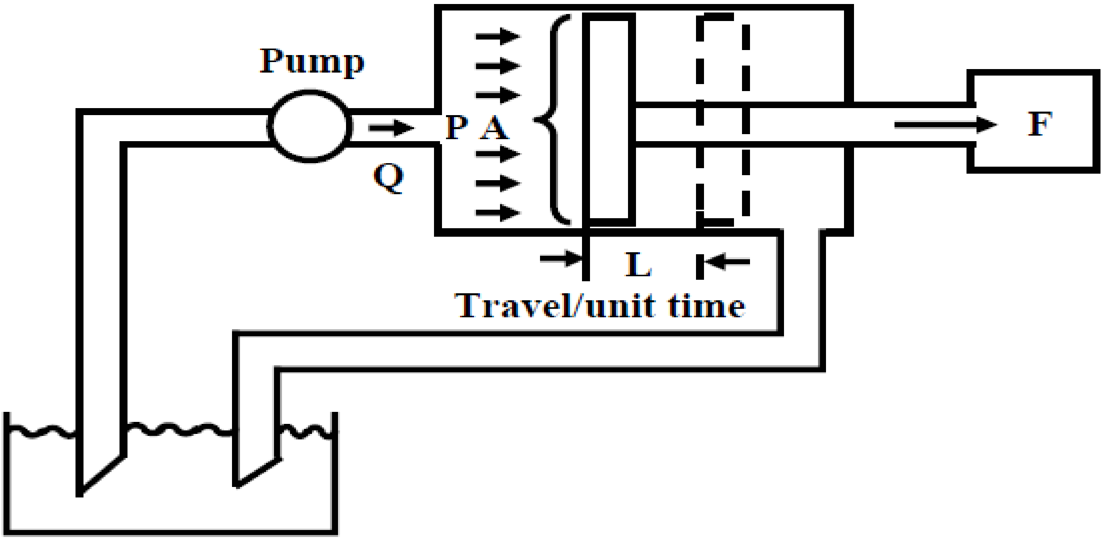
Pressure applied to a confined fluid at any point is transmitted undiminished and equally throughout the fluid in all directions and acts upon every part of the confining vessel at right angles to its interior surfaces.

Amplification of Force

Since pressure P applied on an area A gives rise to a force F, given as,

$F = P \times A$

Thus, if a force is applied over a small area to cause a pressure P in a confined fluid, the force generated on a larger area can be made many times larger than the applied force that created the pressure. This principle is used in various hydraulic devices to such hydraulic press to generate very high forces.



Major hydraulic and mechanical variables

Advantages of Hydraulic Actuation Systems

Hydraulics refers to the means and mechanisms of transmitting power through liquids. The original power source for the hydraulic system is a prime mover such as an electric motor or an engine which drives the pump. However, the mechanical equipment cannot be coupled directly to the prime mover because the required control over the motion, necessary for industrial operations cannot be achieved. In terms of these Hydraulic Actuation Systems offer unique advantages, as given below.

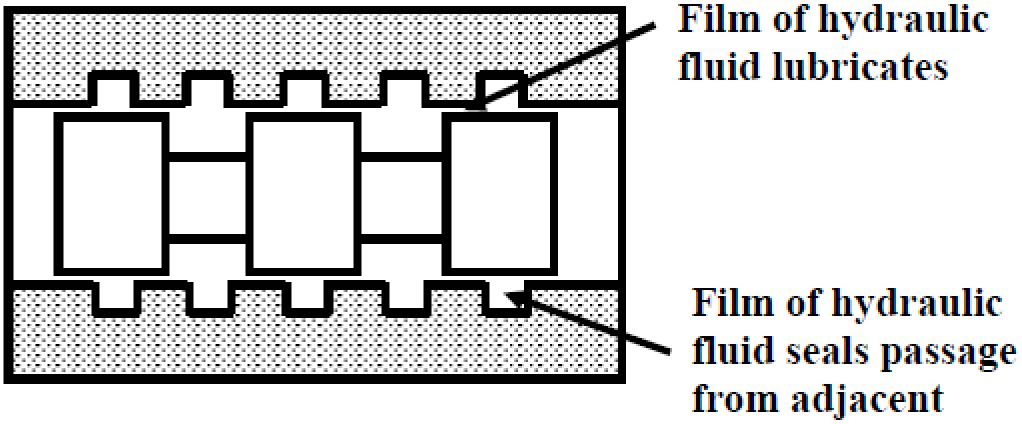
Variable Speed and Direction: Most large electric motors run at adjustable, but constant speeds. It is also the case for engines. The actuator (linear or rotary) of a hydraulic system, however, can be driven at speeds that vary by large amounts and fast, by varying the pump delivery or using a flow control valve. In addition, a hydraulic actuator can be reversed instantly while in full motion without damage. This is not possible for most other prime movers.

Power-to-weight ratio**:** Hydraulic components, because of their high speed and pressure capabilities, can provide high power output with vary small weight and size, say, in comparison to electric system components. Note that in electric components, the size of equipment is mostly limited by the magnetic saturation limit of the iron. It is one of the reasons that hydraulic equipment finds wide usage in aircrafts, where dead-weight must be reduced to a minimum.

Stall Condition and Overload Protection**:** A hydraulic actuator can be stalled without damage when overloaded, and will start up immediately when the load is reduced. The pressure relief valve in a hydraulic system protects it from overload damage. During stall, or when the load pressure exceeds the valve setting, pump delivery is directed to tank with definite limits to torque or force output. The only loss encountered is in terms of pump energy. On the contrary, stalling an electric motor is likely to cause damage. Likewise, engines cannot be stalled without the necessity for restarting.

Components of Hydraulic Actuation Systems Hydraulic Fluid

Hydraulic fluid must be essentially non-compressible to be able to transmit power instantaneously from one part of the system to another. At the same time, it should lubricate the moving parts to reduce friction loss and cool the components so that the heat generated does not lead to fire hazards. It also helps in removing the contaminants to filter. The most common liquid used in hydraulic systems is petroleum oil because it is only very slightly compressible. The other desirable property of oil is its lubricating ability. Finally, often, the fluid also acts as a seal against leakage inside a hydraulic component. The degree of closeness of the mechanical fit and the oil viscosity determines leakage rate. Figure below shows the role played by hydraulic fluid films in lubrication and sealing.


Film of hydraulic fluid lubricates
Film of hydraulic fluid seals passage from adjacent

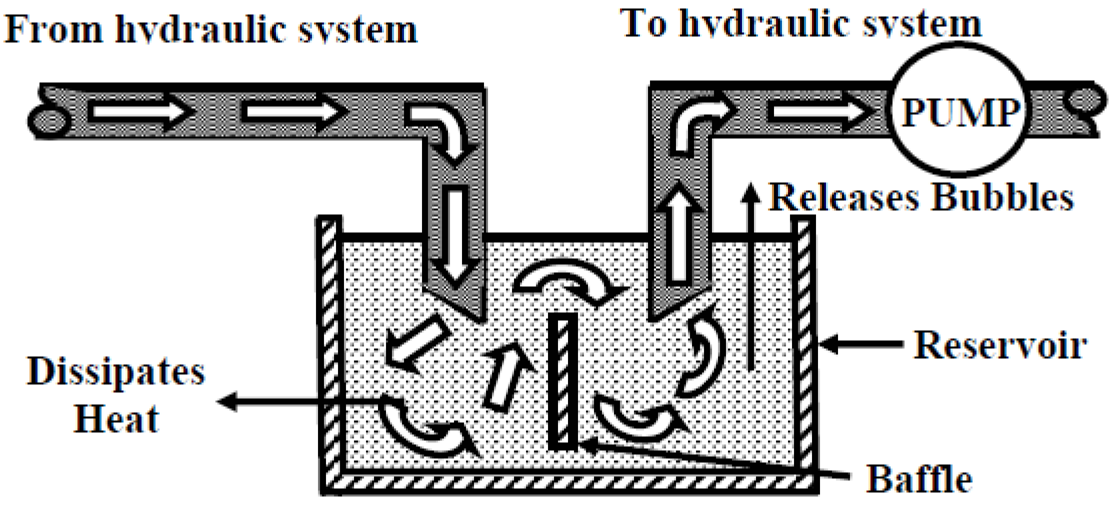
Lubrication and Sealing by Hydraulic Fluid

The Fluid Delivery Subsystem

It consists of the components that hold and carry the fluid from the pump to the actuator. It is made up of the following components.

Reservoir

It holds the hydraulic fluid to be circulated and allows air entrapped in the fluid to escape. This is an important feature as the bulk modulus of the oil, which determines the stiffness of hydraulic system, deteriorates considerably in the presence of entrapped air bubbles. It also helps in dissipating heat.



The functions of the reservoir

Filter

The hydraulic fluid is kept clean in the system with the help of filters and strainers. It removes minute particles from the fluid, which can cause blocking of the orifices of servo-valves or cause jamming of spools.

Line

Pipe, tubes and hoses, along with the fittings or connectors, constitute the conducting lines that carry hydraulic fluid between components. Lines are one of the disadvantages of hydraulic system that we need to pay in return of higher power to weight ratio. Lines convey the fluid and also dissipate heat. In contrast, for Pneumatic Systems, no return path for the fluid, which is air, is needed, since it can be directly released into the atmosphere. There are various kinds of lines in a hydraulic system. The working lines carry the fluid that delivers the main pump power to the load. The pilot lines carry fluid that transmit controlling pressures to various directional and relief valves for remote operation or safety. Lastly there are drain lines that carry the fluid that inevitably leaks out, to the tank.

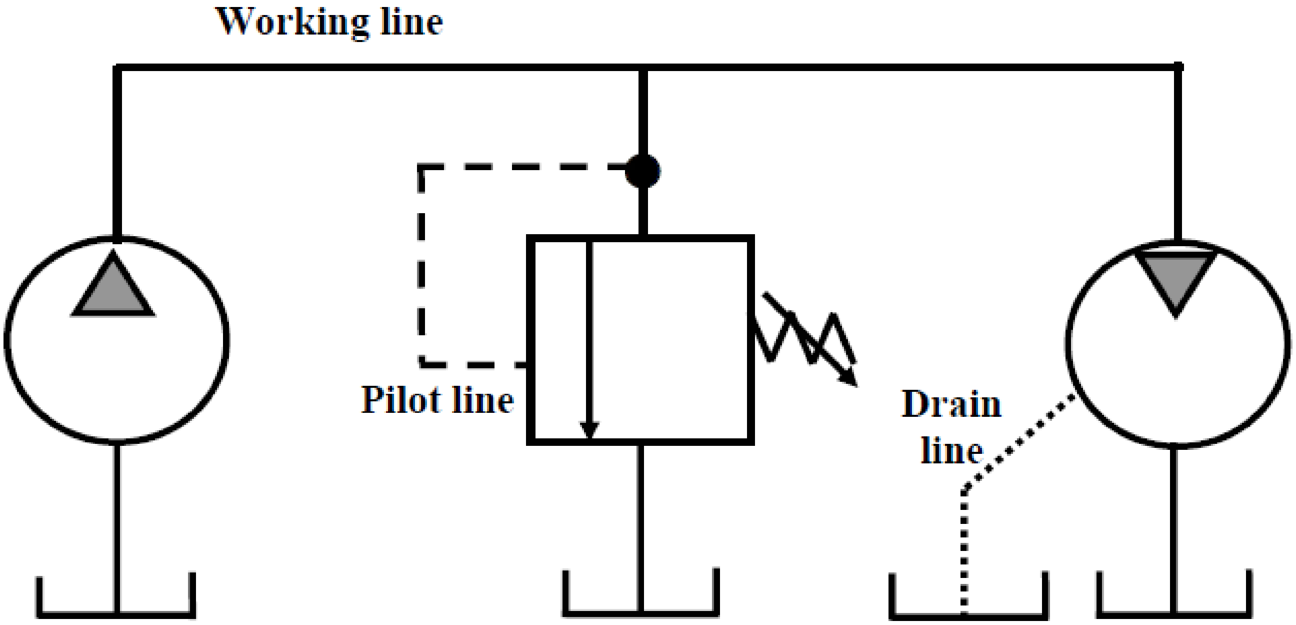
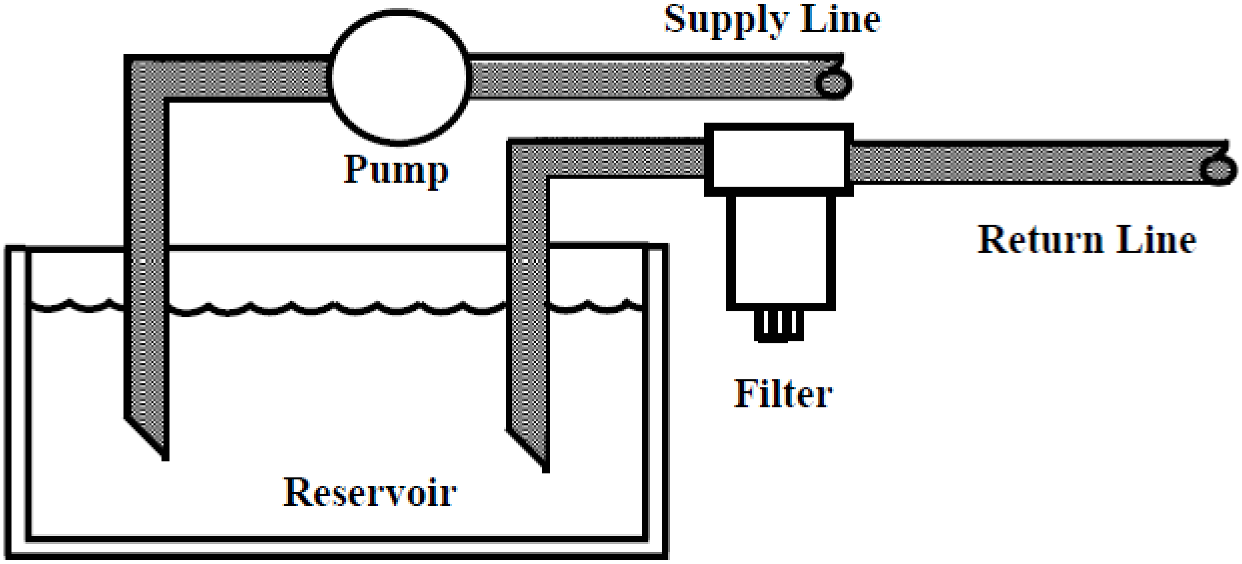
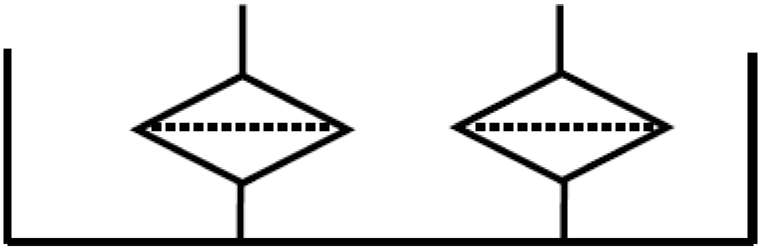


Fig below shows a typical configuration of connecting the supply and the return lines as well as the filter to the reservoir. The graphical symbol for a Reservoir and Filters is shown in Fig.



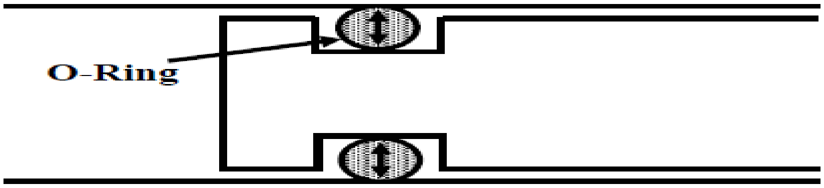
Connection Arrangement of Filter and Lines with a Reservoir



The graphical symbol for Reservoirs and Filters

Fittings and Seals

Various additional components are needed to join pipe or tube sections, create bends and also to prevent internal and external leakage in hydraulic systems. Although some amount of internal leakage is built-in, to provide lubrication, excessive internal leakage causes loss of pump power since high pressure fluid returns to the tank, without doing useful work. External leakage, on the other hand, causes loss of fluid and can create fire hazards, as well as fluid contamination. Various kinds of sealing components are employed in hydraulic systems to prevent leakage. A typical such component, known as the O-ring is shown below in Fig



Directional Control Valves, Switches and Gauges

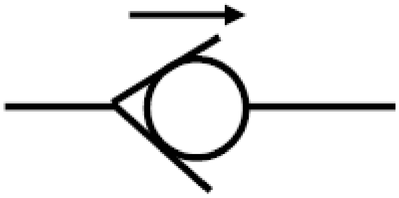
There are two basic types of hydraulic valves. Infinite position valves can take any position between fully open and fully closed. Servo valves and Proportional valves are in this category and are discussed in a separate lesson. Finite position valves can only assume certain fixed positions. In these positions the various inlet and outlet ports are either fully open or fully closed. However, depending on the position of the valve, particular inlet ports get connected to particular other outlet ports. Therefore flows in certain directions are established, while those in other directions are stopped. Since it is basically the directions of the flows that are controlled and not the magnitudes, these are called directional control valves.

Directional valves can be characterized depending on the number of ports, the number of directions of flow that can be established, number of positions of the valve etc. They are mainly classified in terms of the number of flow directions, such as one-way, two way or four-way valves. These are described below.

Directional valves are often operated in selected modes using hydraulic pressure from remote locations. Such mechanisms are known as pilots. Thus a valve that may be blocking the flow in a certain direction in absence of pilot pressure, may be allowing flow, when pilot pressure is applied. This enables one to build greater flexibility in the automation of system operation.

Check Valve

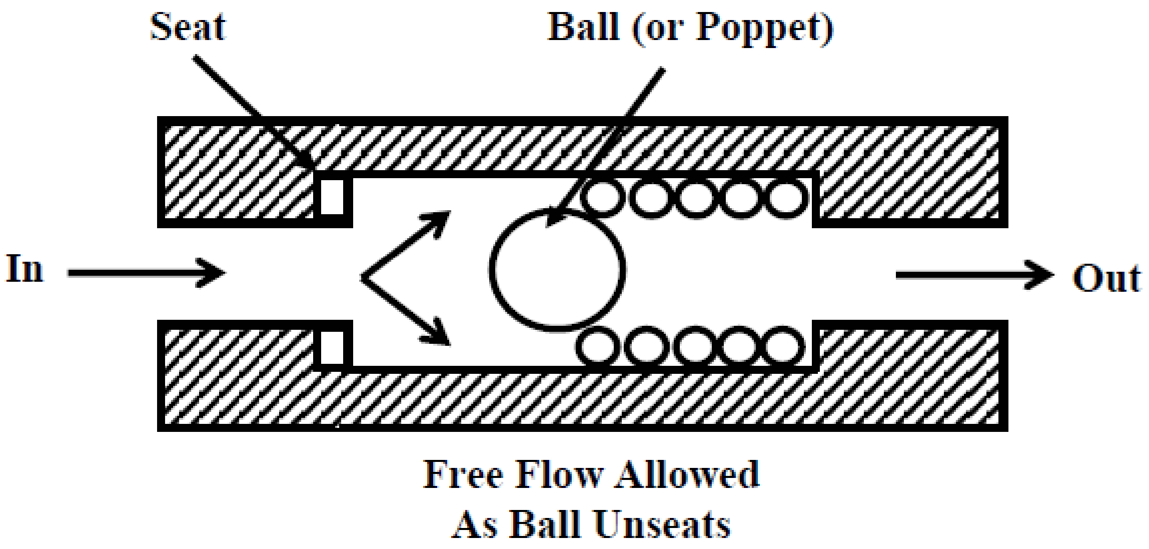
In its simplest form, a check valve is a one-way directional valve. It permits free flow in one direction and blocks flow in the other. As such is analogous to the electronic diode.



The Check Valve Symbol

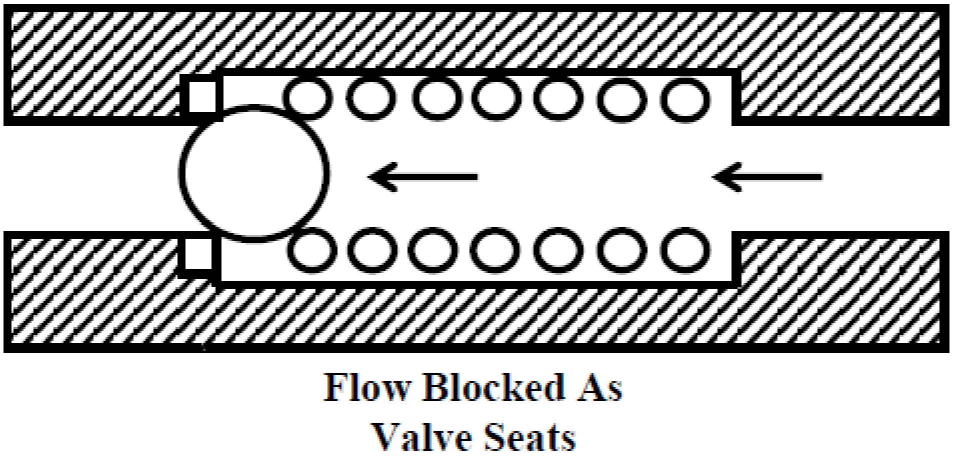
The simple ball-and-seat symbol shown in Fig. is used universally for denoting check valves although it is different from the way the other valves are denoted. The direction of the arrow shows the direction for free flow.

In its simplest form of construction, a check valve is realized as an in-line ball and spring as shown in Fig.



A check valve permits flow in one direction

Pressure from the left moves the ball from its seat so to permit unobstructed flow. Pressure from right pushes the ball tight on to the seat, and flow is blocked shown in Fig. In some valves a poppet is used in place of the ball. In some other construction, the valve inlet and outlet ports are made at right angles.

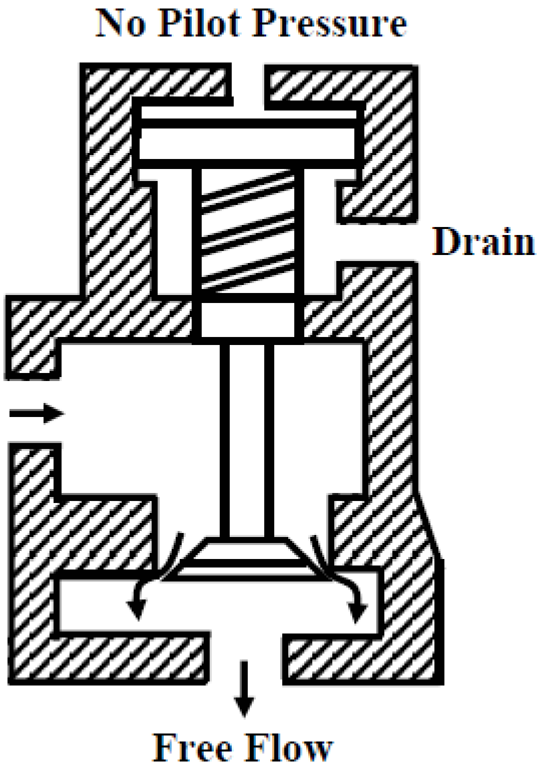


A Check Valve blocks flow in the reverse direction

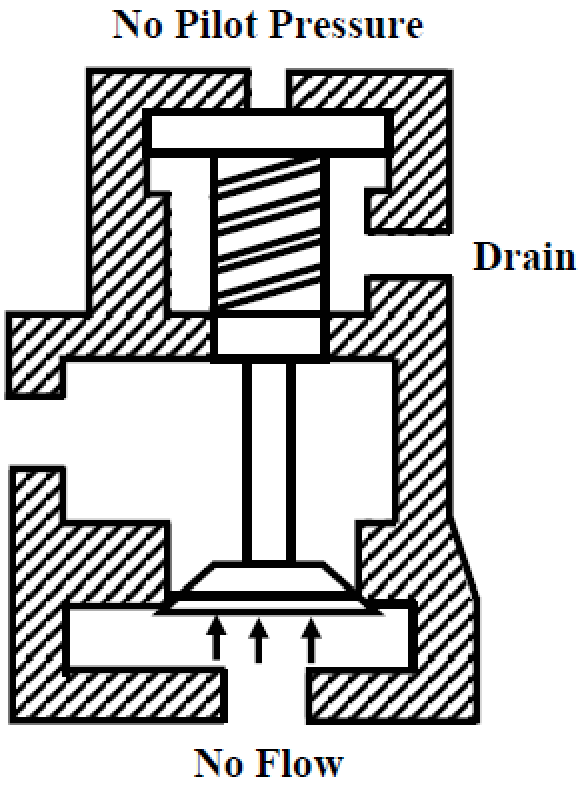
Pilot – operated Check Valves

Pilot-operated check valves are designed to permit free flow in one direction and to block return flow, unless pilot pressure is applied. However, under pilot pressure, flow is permitted in both directions. They are used in hydraulic presses as prefill valves – to permit the main ram to fill by gravity during the “fast approach” part of the stroke. They also are used to support vertical pistons which otherwise might drift downward due to leakage past the directional valve spool.

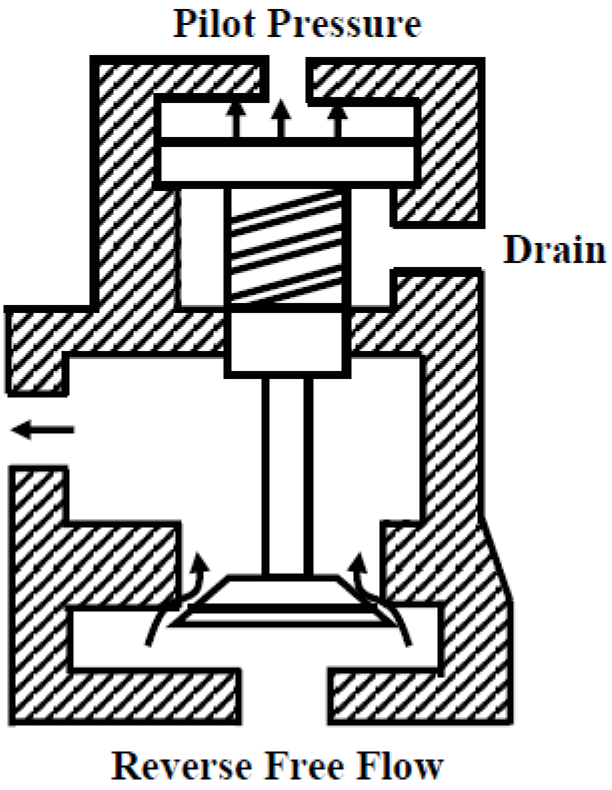
The construction of a pilot operated check valve is shown in Fig. With no pilot pressure, the valve functions as a normal check valve. Flow to bottom is permitted but the reverse is blocked. If pilot pressure is applied, the valve is open at all times, and flow is allowed freely in both directions as shown in Fig.



Unidirectional flow without pilot pressure



Unidirectional flow without pilot pressure



Reverse flow with pilot pressure

The check valve poppet has the pilot piston attached to the poppet stem. A light spring holds the poppet seated in a no-flow condition by pushing against the pilot piston. A separate drain port is provided to prevent oil from creating a pressure buildup on the underside of the piston. Reverse flow can occur only when a pressure that can overcome the pressure in the outlet chamber is applied.

Possible application of the valve can be to permit free flow to the accumulator, while blocking flow out of it. If the pilot is actuated the accumulator can discharge if the pressure at the inlet port is lower than the accumulator pressure.

UNIT – V

ELECTRICAL MACHINE DRIVES

Energy Savings with Variable Speed Drives

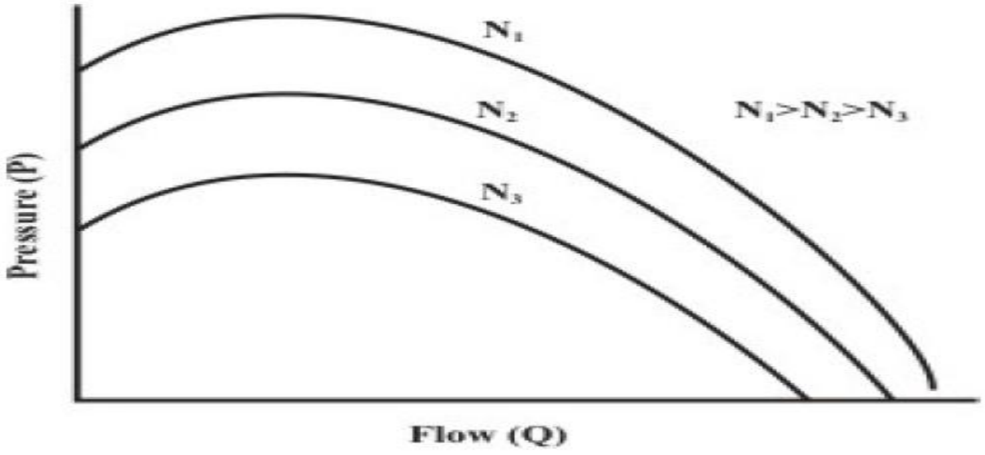
The AC induction motor is the major converter of electrical energy into mechanical and other useable forms. For this purpose, about two thirds of the electrical energy produced is fed to motors. Much of the power that is consumed by AC motors goes into the operation of fans, blowers and pumps. It has been estimated that approximately 50% of the motors in use are for these types of loads. These particular loads — fans, blowers and pumps, are particularly attractive to look at for energy savings. Several alternate methods of control for fans and pumps have been advanced recently that show substantial energy savings over traditional methods.

Basically, fans and pumps are designed to be capable of meeting the maximum demand of the system in which they are installed. However, quite often the actual demand could vary and be much less than the designed capacity. These conditions are accommodated by adding outlet dampers to fans or throttling valves to pumps. These control methods are effective, inexpensive and simple, but severely affect the efficiency of the system.

Others forms of control are now available to adapt fans and pumps to varying demands, which do not decrease the efficiency of the system as much. Newer methods include direct variable speed control of the fan or pump motor. This method produces a more efficient means of flow control than the existing methods. In addition, adjustable frequency drives offer a distinct advantage over other forms of variable speed control.

Fans: Characteristics and Operation

Large fans and blowers are routinely used in central air conditioning systems, boilers, drives and the chemical operations. The most common fan is the centrifugal fan that imparts energy in to air by centrifugal force. This results in an increase in pressure and produces air flow at the outlet.

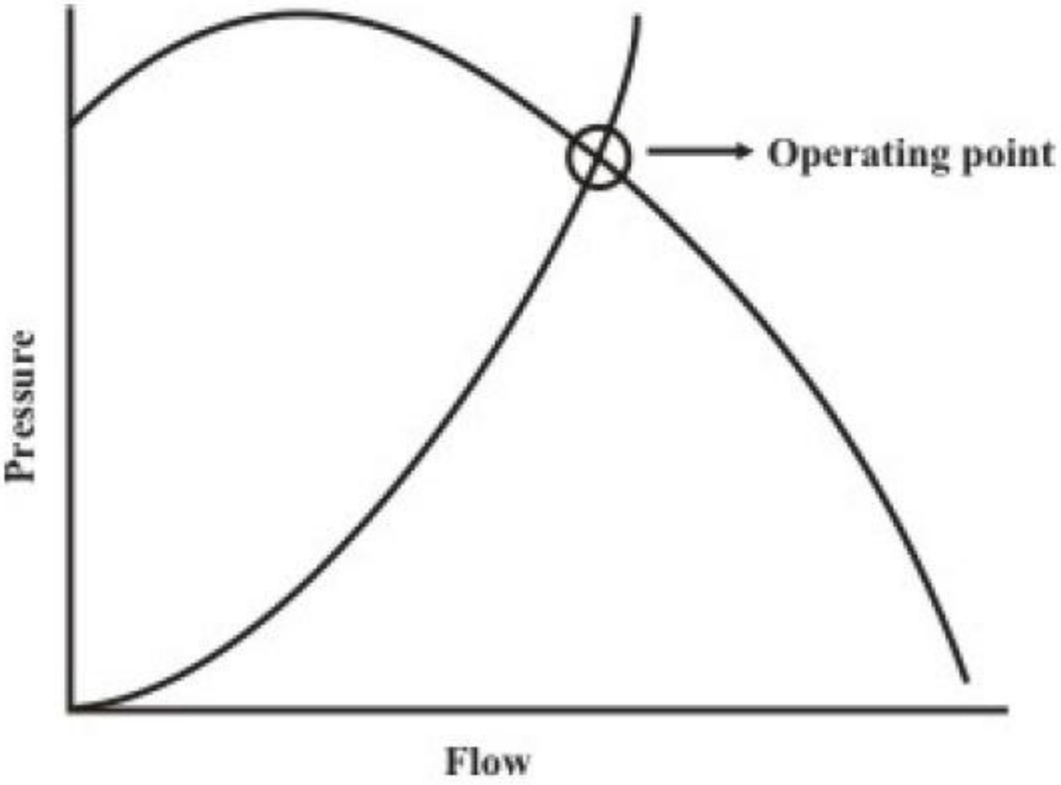


Fan Curve

Fig. is a plot of outlet pressure versus the flow of air of a typical centrifugal fan at a given speed. Standard fan curves usually show a number of curves for different fan speeds and include the loci of constant fan efficiencies and power requirements on the operating characteristics.

These are all useful for selecting the optimum fan for any application. They also are needed to predict fan operation and other parameters when the fan operation is changed. Appendix 1 gives an example of a typical fan curve for an industrial fan.

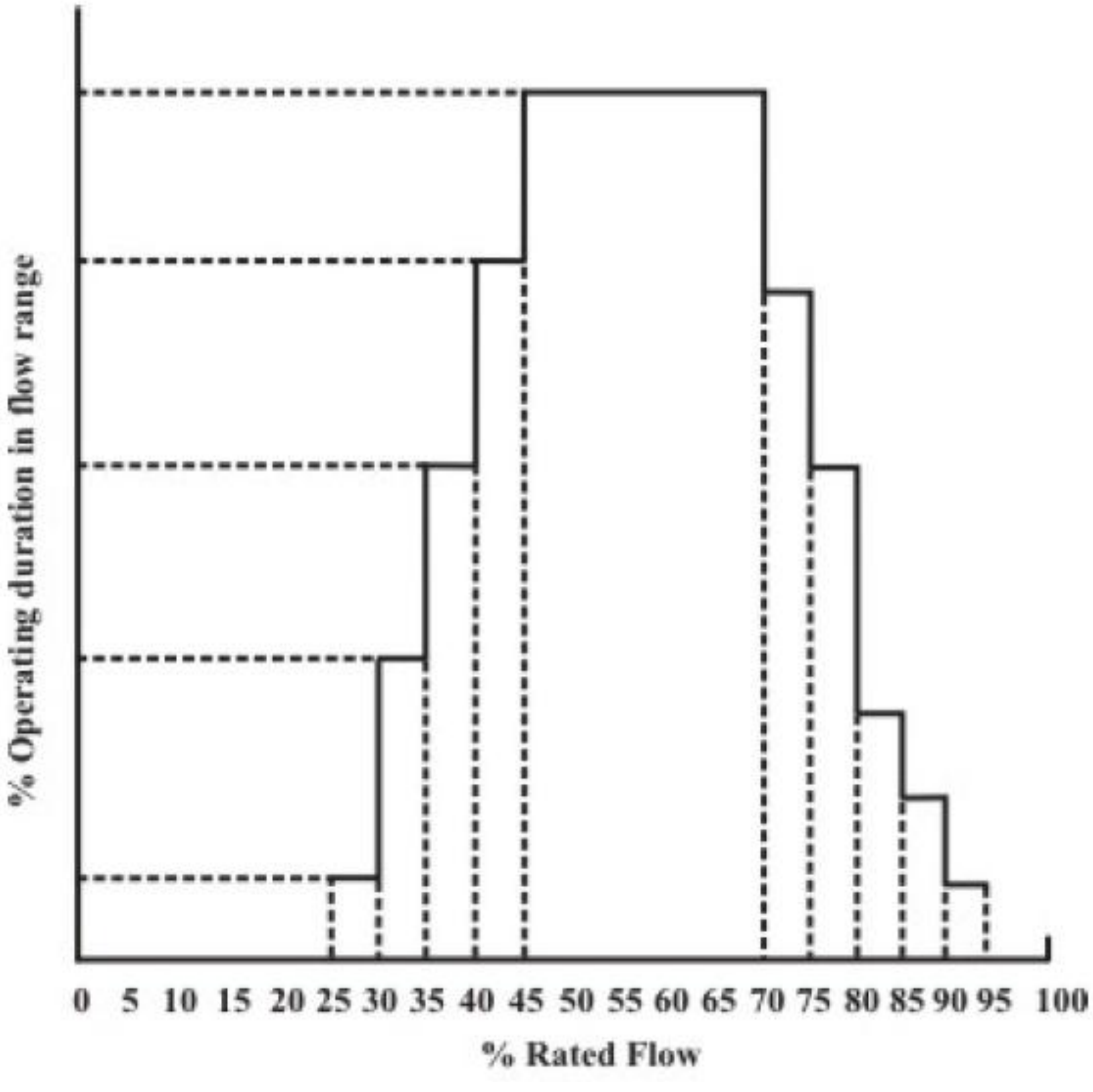
Fig. shows a typical system pressure-flow characteristics curve intersecting a typical fan curve.



System Curve

The system curve shows the requirements of the vent system that the fan is used on. It shows how much pressure is required from the fan to overcome system losses and produce airflow. The fan curve is a plot of fan capability independent of a system. The system curve is a plot of "load" requirement independent of the fan. The intersection of these two curves is the natural *operating point*. It is the actual pressure and flow that will occur at the fan outlet when this system is operated. Without external control, the fan will operate at this point.

Many systems however require operation at a wide variety of points. Fig. 31.3 shows a profile of the typical variations in flow experienced in a typical system.



Typical system flow-duration profile.

There are several methods used to modulate or vary the flow to achieve the optimum points. Apart from the method of cycling, the other methods affect either the system curve or the fan curve to produce a different natural operating point. In so doing, they also may change the fan's efficiency and the power requirements. Below these methods are explained in brief.

Step Motors: Principles, Construction and Drives

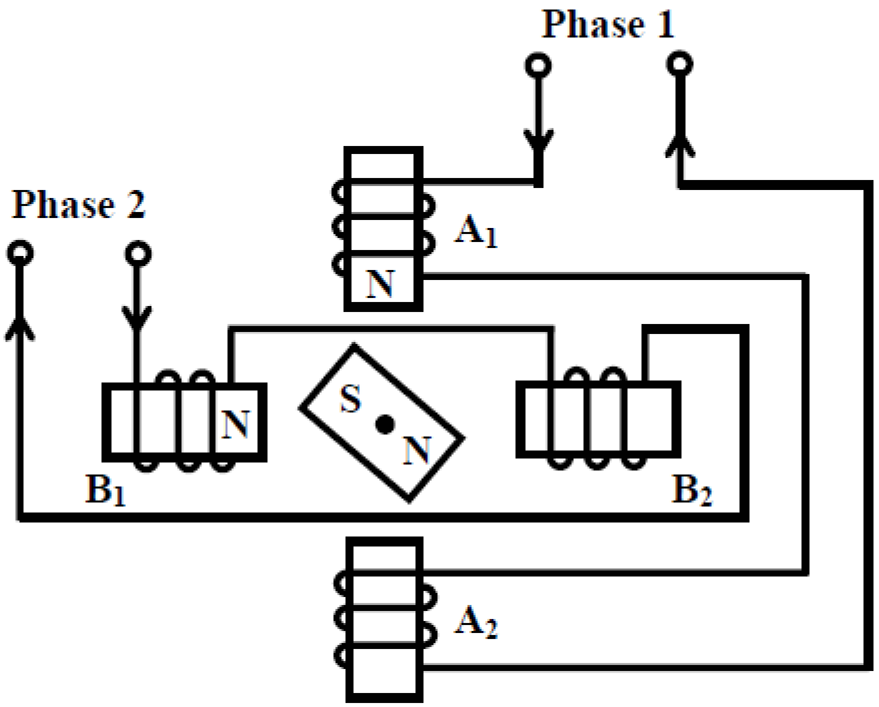
Step motors (often referred as stepper motors) are different from all other types of electrical drives in the sense that they operate on discrete control pulses received and rotate in discrete steps. On the other hand ordinary electrical a.c and d.c drives are analog in nature and rotate continuously depending on magnitude and polarity of the control signal received. The discrete nature of operation of a step motor makes it suitable for directly interfacing with a computer and direct computer control. These motors are widely employed in industrial control, specifically for CNC machines, where open loop control in discrete steps are acceptable. These motors can also be adapted for continuous rotation.

Construction

Step motors are normally of two types: (a) permanent magnet and (b) variable reluctance type. In a step motor the excitation voltage to the coils is d.c. and the number of phases indicates the number of windings. In both the two cases the excitation windings are in the stator. In a permanent magnet type step motor the rotor is a permanent magnet with a number of poles. On the other hand the rotor of a variable reluctance type motor is of a cylindrical structure with a number of projected teeth.

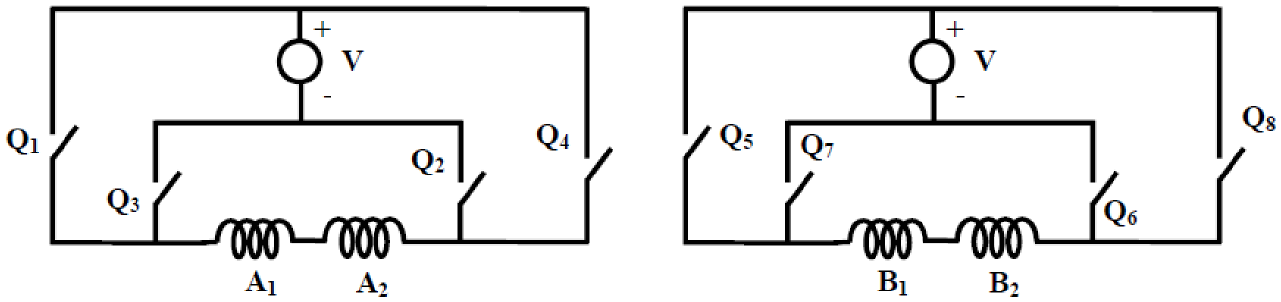
Permanent magnet step motor

The principle of step motor can be understood from the basic schematic arrangement of a small permanent magnet step motor is shown in Fig. This type of motor is called a two-phase two-pole permanent magnet step motor; the number of windings being two (phase 1 and phase 2) each split into two identical halves; the rotor is a permanent magnet with two poles. So winding A is split into two halves A_1 and A_2. They are excited by constant d.c. voltage V and the direction of current through A_1 and A_2 can be set by switching of four switches Q_1, Q_2, Q_3 and where four switches Q5-Q8 are used to control the direction of current as shown in Fig. The directions of the currents and the corresponding polarities of the induced magnets are shown in Fig. 1. Q_4 as shown in Fig. For example, if Q_1 and Q_2 are closed, the current flows from A_1 to A_2, while closing of the switches Q_3 and Q_4 sets the direction of current from A_2 to A_1. Similar is the case for the halves B_1 and B_2



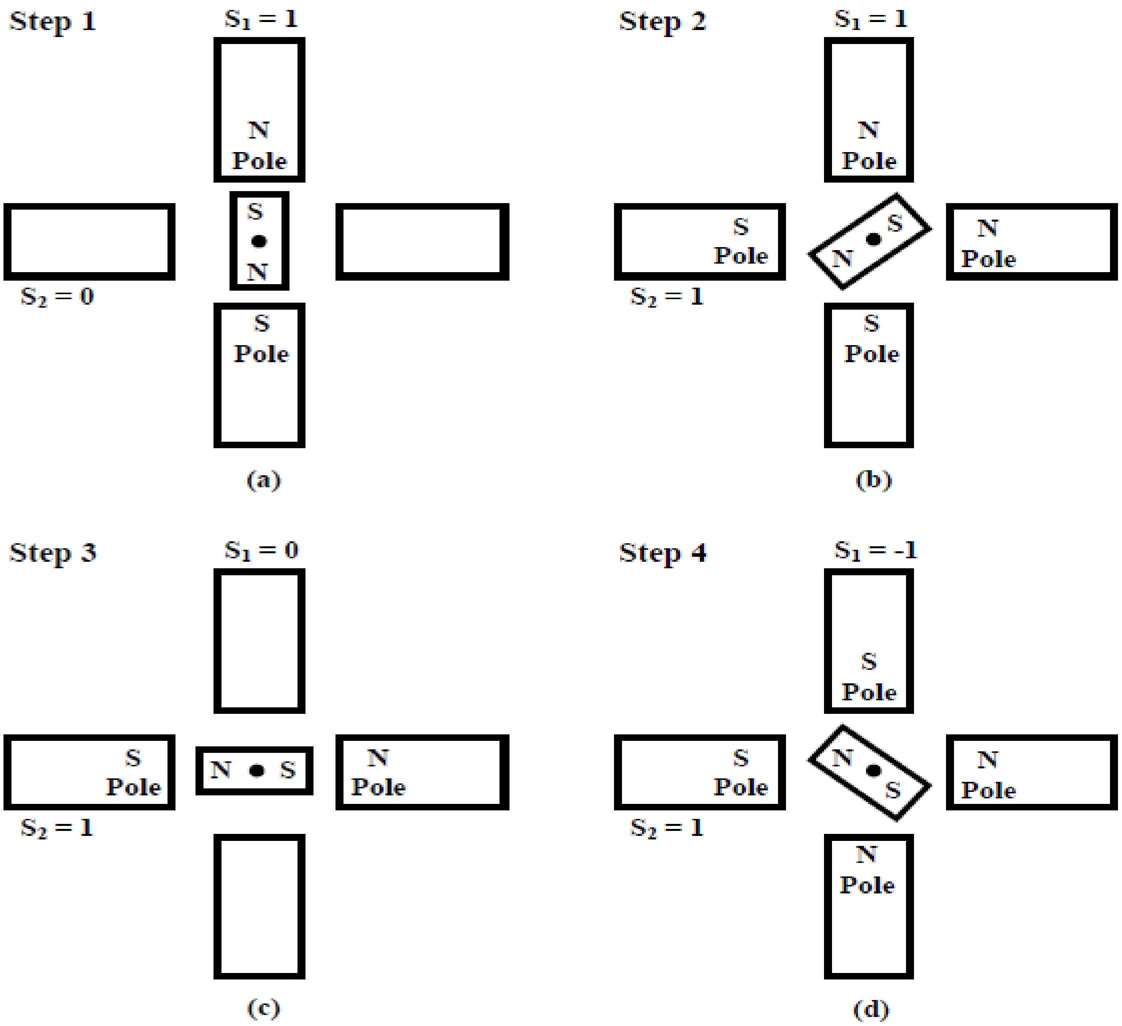
Schematic diagram of a two-phase two-pole permanent magnet stepper motor.

Switching Sequence



Now consider Fig.3 Let Winding A be energised and the induced magnetic poles are as shown in Fig. (we will denote the switching condition as S_1=1). The other winding B is not energised. As a result the moving permanent magnet will align itself along the axis of the stator poles as shown in Fig. In the next step, both the windings A and B are excited simultaneously, and the polarities of the stator poles are as shown in Fig.

We shall denote S_2=1, for this switching arrangement for winding B. The rotor magnet will now rotate by an angle of 45^o and align itself with the resultant magnetic field produced. In the next step, if we now make S_1=0 (thereby de-energising winding A), the rotor will rotate further clockwise by 45^o and align itself along winding B, as shown in Fig. In this way if we keep on changing the switching sequence, the rotor will keep on rotating by 45^o in each step in the clockwise direction. The switching sequences for the switches Q_1 to Q_8 for first four steps are tabulated in Table



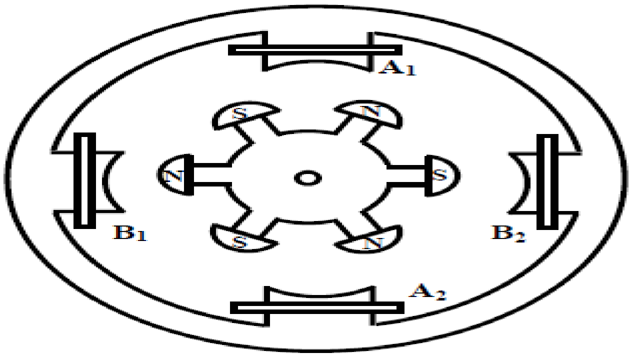
Stepping sequence (half-stepping) for a two-phase two-pole PM step motor for clockwise rotatio Table

Switching sequence corresponding to the movement shown in Fig.

<table>
<tr><th>Step 1</th><th colspan="2">Step 2</th><th colspan="2">Step 3</th><th>Step 4</th></tr>
<tr><td colspan="2">Q_1-Q_2</td><td colspan="2">ON
(S_1=1)</td><td colspan="2">ON
(S_1=1)</td></tr>
<tr><td colspan="3">Q_3-Q_4</td><td colspan="3">ON
(S_1= -1)</td></tr>
<tr><td colspan="6">Q_5-Q_6</td></tr>
<tr><td>Q_7-Q_8</td><td colspan="2">ON
(S_2=1)</td><td colspan="2">ON
(S_2=1)</td><td>ON
(S_2=1)</td></tr>
</table>

It is apparent from Table and Fig. that for this type of switching the step angle is 45° and it takes 8 steps to complete a complete revolution. So we have 8 steps / revolution. It can also be seen from Table 1 that a pair of switch (say Q_7-Q_8) remains closed during consecutive three steps of rotation and there is an overlap at every alternate step where both the two windings are energised. This arrangement for controlling the step motor movement is known as *half stepping*. The direction of rotation can be reversed by changing the order of the switching sequence.

It is also possible to have an excitation arrangement where each phase is excited one at a time and there is no overlapping where both the phases are energised simultaneously, though it is not possible for the configuration shown in Fig.1, since that will require the rotor to rotate by 90° in each step and in the process, may inadventedly get locked in the previous position. But *full stepping* is achievable for other cases, as for example for the two-phase six-pole permanent magnet step motor as shown in Fig. In this case, the stator pitch $\theta_s = 90^0$ and the rotor pitch $\theta_r = 60^0$; the full step angle is given by $\theta_{fs} = \theta_s \sim \theta_r = 30^0$ and the half step angle $\theta_{hs} = (\theta_s \sim \theta_r)/2 = 15^0$. The desired direction of rotation can be achieved by choosing the sequence of switching.



Two-phase six-pole permanent magnet stop motor.

The advantage of a permanent magnet step motor is that it has a *holding torque*. This means that due to the presence of permanent magnet the rotor will lock itself along the stator pole even when the excitation coils are de-energised. But the major disadvantage is that the direction of current for each winding needs to be reversed. This requires more number of transistor switches that may make the driving circuit unwieldy. This disadvantage can be overcome with a variable reluctance type step motor, as explained in the next section.

Another way of reducing the number of switches is to use unipolar winding. In unipolar winding, there are two windings per pole, out of which only one is excited at a time. The windings in a pole are wound in opposite direction, thus either N-pole or S-pole, depending on which one is excited.

Electrical Actuators: DC Motor Drives

Variable speed drives can be categorized into Adjustable Speed Drives and Servo Drives. In adjustable speed drives the speed set points are changed relatively infrequently, in response to changes in process operating pints. Therefore transient response of the drive system is not of consequence. In servo drives, as in CNC machines, set points change constantly (as in contouring systems).

While ac motors have replaced dc motors in most of the adjustable speed drive applications. For servo drive applications, dc motors are still used, although they are also being replaced by BLDC motors. In this lesson we discuss speed and position control with dc motors.

DC Servomotors

Direct current servomotors are used as feed actuators in many machine tool industries. These motors are generally of the permanent magnet (PM) type in which the stator magnetic flux remains essentially constant at all levels of the armature current and the speed-torque relationship is linear.

Direct current servomotors have a high peak torque for quick accelerations. A cross-sectional view of a typical permanent magnet dc servomotor is shown in Fig.

Mechanical Construction

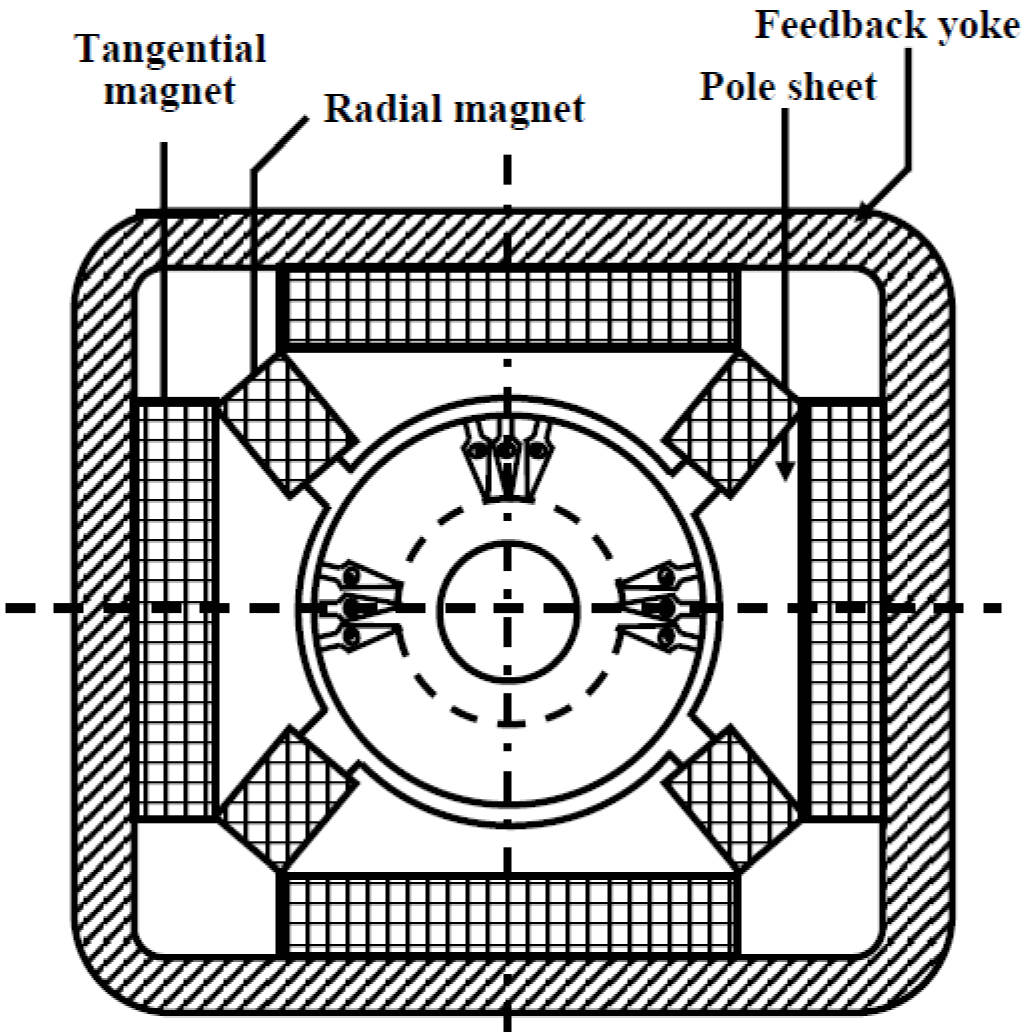
Stator consists of Yoke and Poles and provides mechanical support to the machine. The yoke provides a highly permeable path for magnetic flux. It is made of cast steel. Field poles are made of thin laminations stacked together. This is done to minimize the magnetic losses due to the armature flux. The cross sectional area of the field pole is less than that of the pole shoe. The pole shoe helps to establish a uniform flux density around the air gap.

Field winding: DC excitations are provided to field windings wound on pole shoes to create electromagnetic poles of alternating polarity. Depending on the connections of field windings DC motors may be termed as shunt, series, compound or separately excited. Shunt motors have field winding connected in parallel with the armature winding while series motors have the field winding connected in series with the armature winding. A compound dc machine may have both field windings wound on the same pole. Smaller DC servomotors generally have permanent magnets for poles.

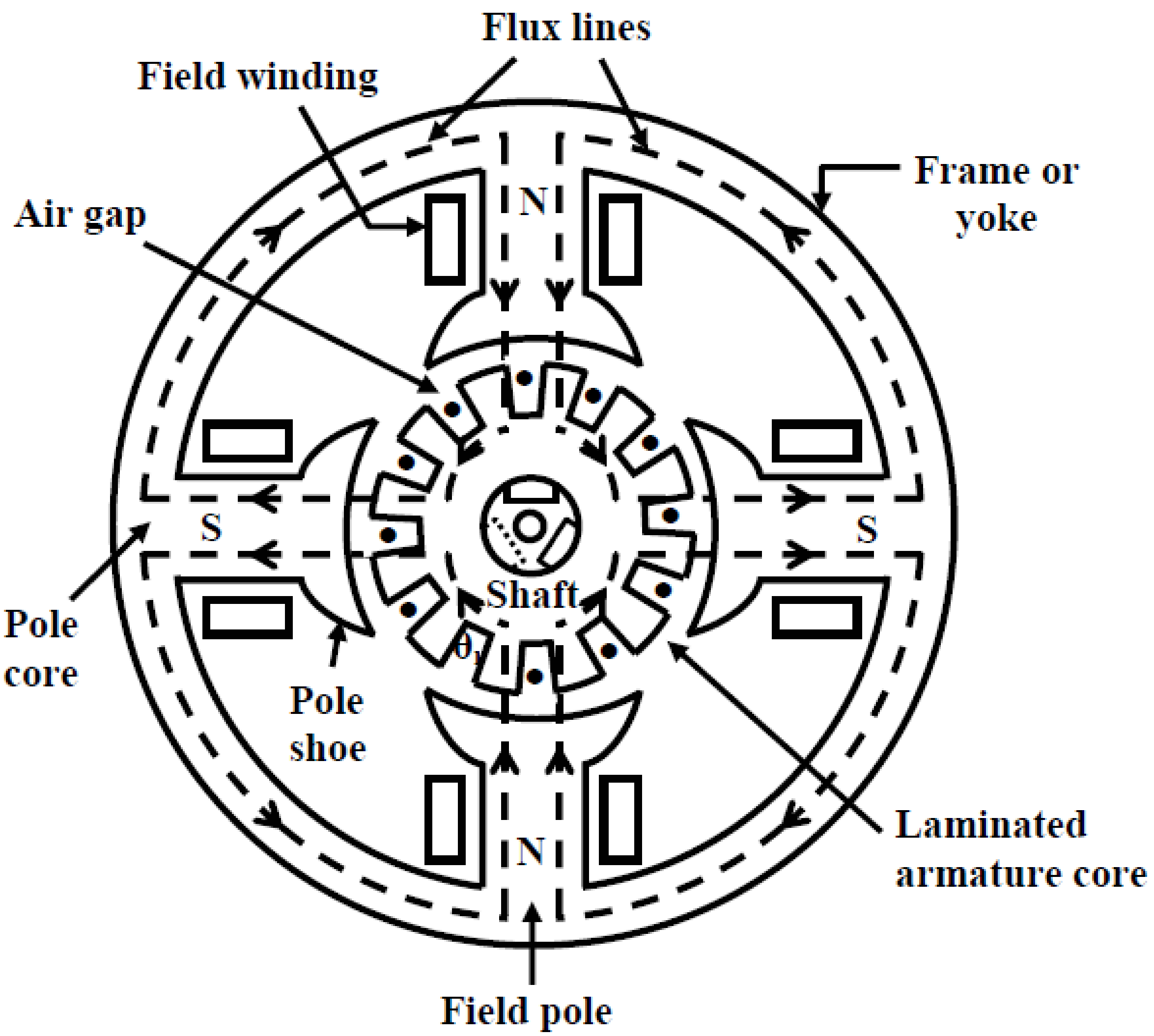
Armature – The rotating part of a dc machine is called the armature. The length of the armature is usually the same as that of the pole. It is made of thin, highly permeable, and electrically insulated circular steel laminations that are stacked together and rigidly mounted on the shaft. The laminations have axial slots on their periphery to house the armature coils. Insulated copper wires are typically used for the armature coils to achieve a low armature resistance.

Commutator – The commutator is made of wedge – shaped hard-drawn copper segments. Sheets of mica insulate the copper segments from one another. One end of the armature coil is electrically connected to a copper segment of the commutator. The commutators rotate with the armature keeping a sliding contact with the brushes, which remain stationary.

Brushes: Brushes are held in a fixed position by means of brush holders and remain in sliding contact with the commutator segments. An adjustable spring inside the brush holder exerts a constant pressure on the brush in order to maintain a proper contact between the brush and the commutator. The brushes are connected to the armature terminals of the machine. The material for the brush is normally carbon or carbon-graphite.



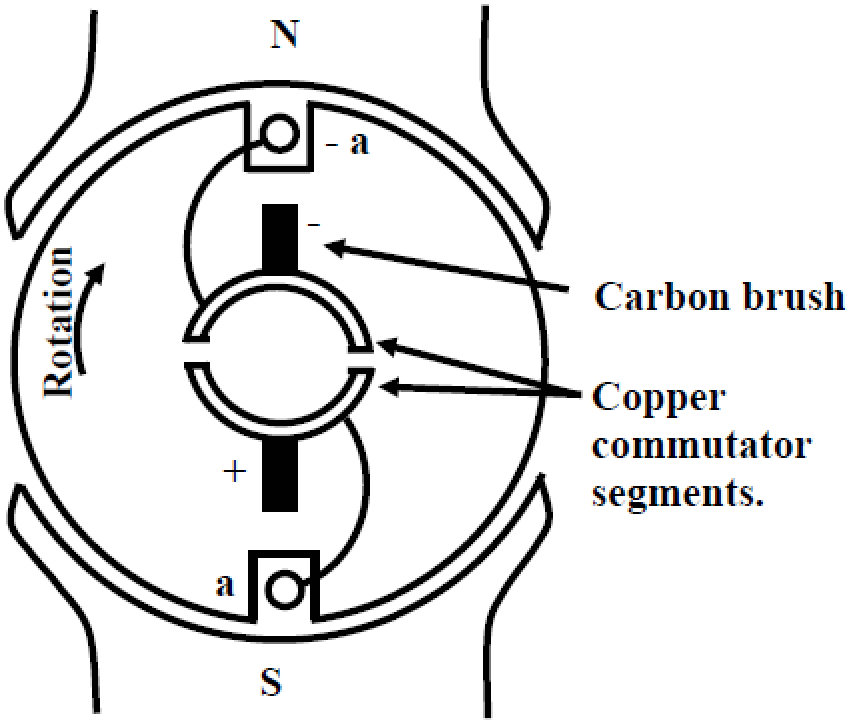
Cross-section of a permanent magnet-excited dc servomotor.



Diagrammatic sketch of a D.C. machine.

Principle of Operation

The cross-sectional view of a DC motor has been shown in Fig. 33.2. Consider a particular position in space between stator and rotor. Whichever conductor is present there, will have current flowing through it, which depends on the applied armature voltage. This current would produce a flux which would interact with the field flux to produce torque. In course of rotation of the armature adjacent conductors will occupy this position in space. No matter which conductor comes to that particular position at any given point of time, it will have same current flowing through it. This is true for all the positions although the magnitude and polarity of the torque produced by individual conductors in different positions may be different. The polarity of the torque is identical for conductor positions under north or south pole, since the direction of the current flowing through it at that position is unique, given the direction of rotation and the applied armature voltage due to the commutators slipping over the brushes, as shown in Fig.



Brush and commutator positions in a DC motor

Induction Motor Drives

For adjustable speed applications, the induction machine, particularly the cage rotor type, is most commonly used in industry. These machines are very cheap and rugged, and are available from fractional horsepower to multi-megawatt capacity, both in single-phase and poly-phase versions.

In cage rotor type induction motors the rotor has a squirrel cage-like structure with shorted end rings. The stator has a three-phase winding, and embedded in slots distributed sinusoidally. It can be shown that a sinusoidal three-phase balanced set of ac voltages applied to the three-phase.

stator windings creates a magnetic field rotating at angular speed ω_s = $4\pi f_s$ /P where f_s is the supply frequency in Hz and P is the number of stator poles.

If the rotor is rotating at an angular speed ω_r , i.e. at an angular speed (ω_s - ω_r) with respect to the rotating stator mmf, its conductors will be subjected to a sweeping magnetic field, inducing voltages and current and mmf in the short-circuited rotor bars at a frequency (ω_s - ω_r)$P/4\pi$, known as the slip speed. The interaction of air gap flux and rotor mmf produces torque. The per unit slip ω is defined as

$$S = \frac{\omega_s - \omega_r}{\omega_s}$$

Equivalent Circuit

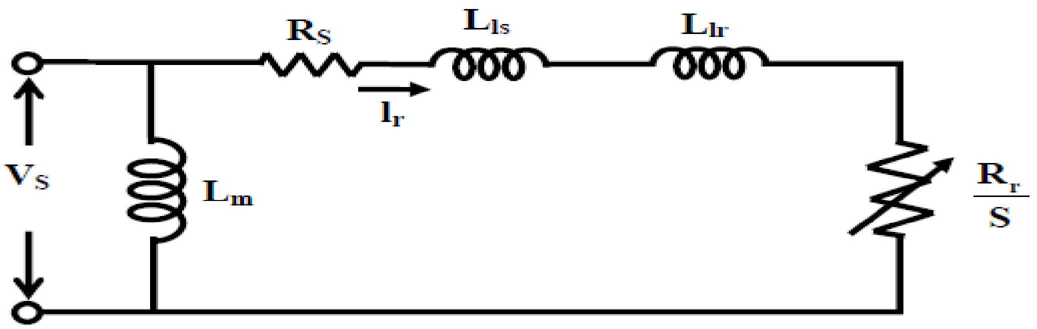
Figure shows the equivalent circuit with respect to the stator, where I_r is given as

$$I_r == \frac{V_m}{\left(\frac{R_r}{S}\right) + j\omega_e L_{lr}}$$

and parameters R_r and L_{lr} stand for the resistance and inductance parameters referred to to the stator.

Since the output power is the product of developed electrical torque T_e and speed ω_m. T_e can be expressed as

$$T_e = 3\left(\frac{P}{2}\right) I_r^2 \frac{R_r}{S\omega_e}$$



Approximate per phase equivalent circuit

In Figure , the magnitude of the rotor current I_r can be written as

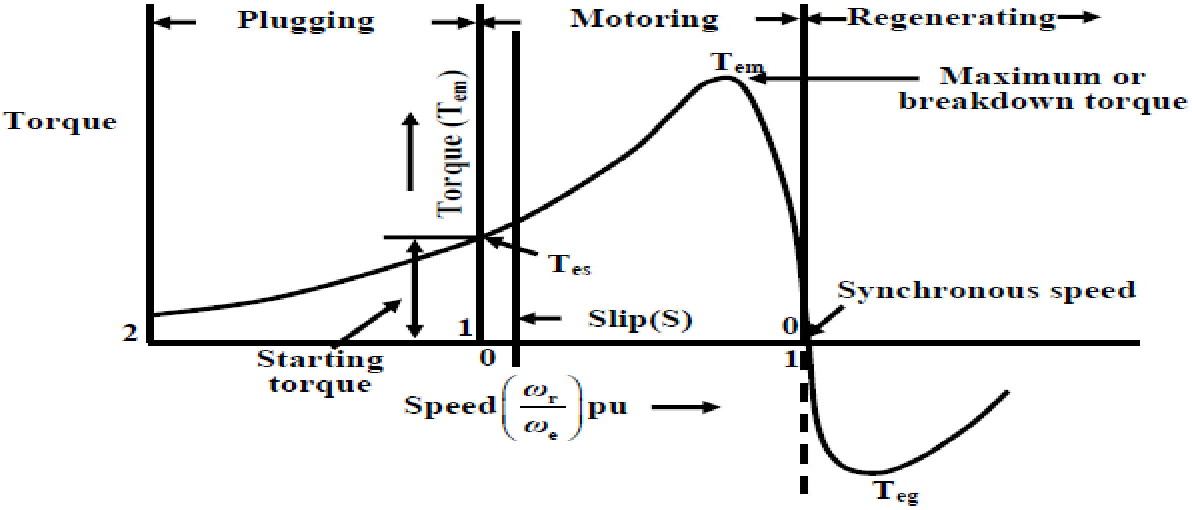
$$I_r = \frac{V_s}{\sqrt{(R_s + R_r/S)^2 + \omega_e^2 (L_{ls} + L_{lr})^2}}$$

This yields that,

$$T_e = 3\left(\frac{P}{2}\right)\frac{R_r}{S\omega_e}\cdot\frac{V_s^2}{(R_s + R_r/S)^2 + \omega_e^2 (L_{ls} + L_{lr})^2}$$

Torque-Speed Curve

The torque T_e can be calculated as a function of slip S from the equation 1. Figure shows the torque-speed $\left(\omega_r/\omega_e = 1 - S\right)$ curve. The various operating zones in the figure can be defined as plugging ($1.0 < S < 2.0$), motoring ($0 < S < 1.0$), and regenerating ($S< 0$). In the normal motoring region, $T_e = 0$ at $S = 0$, and as S increases (i.e., speed decreases), T_e increases in a quasi-linear curve until breakdown, or maximum torque T_{em} is reached. Beyond this point, T_e decreases with the increase in S.



Torque-speed curve of induction motor

In the regenerating region, as the name indicates, the machine acts as a generator. The rotor oves at supersynchronous speed in the same direction as that of the air gap flux so that the slip becomes negative, creating negative, or regenerating torque (Teg). With a variable-frequency power supply, the machine stator frequency can be controlled to be lower than the rotor speed ($\omega_e < \omega_r$) to obtain a regenerative braking effect.

Speed Control

From the torque speed characteristics in Fig. it can be seen that at any rotor speed the magnitude and/or frequency of the supply voltage can be controlled for obtaining a desired torque. The three possible modes of speed control are discussed below.

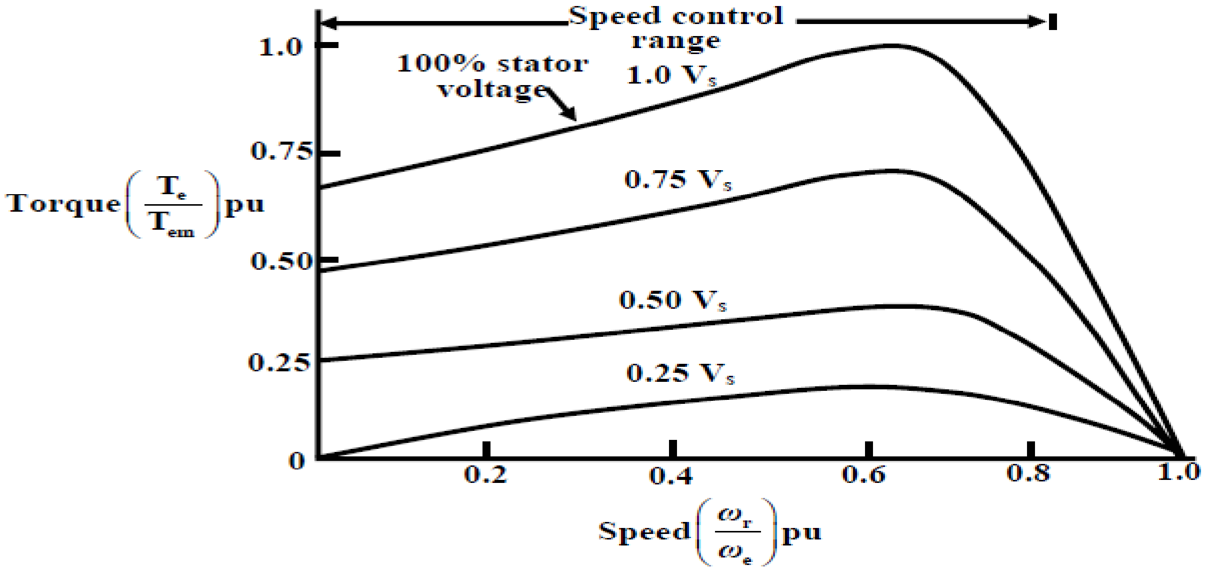
Variable-Voltage, Constant-Frequency Operation

A simple method of controlling speed in a cage-type induction motor is by varying the stator voltage at constant supply frequency. Stator voltage control is also used for “soft start” to limit the stator current during periods of low rotor speeds.

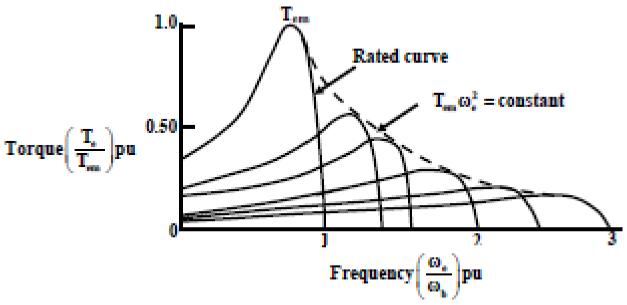
Figure shows the torque-speed curves with variable stator voltage. Often, low-power
motor drives use this type of speed control due to the simplicity of the drive circuit.

Variable-Frequency Operation

Figure shows the torque-speed curve, if the stator supply frequency is increased with constant supply voltage, where ω_e is the base angular speed. Note, however, that beyond the rated frequency ω_b, there is fall in maximum torque developed, while the speed rises.



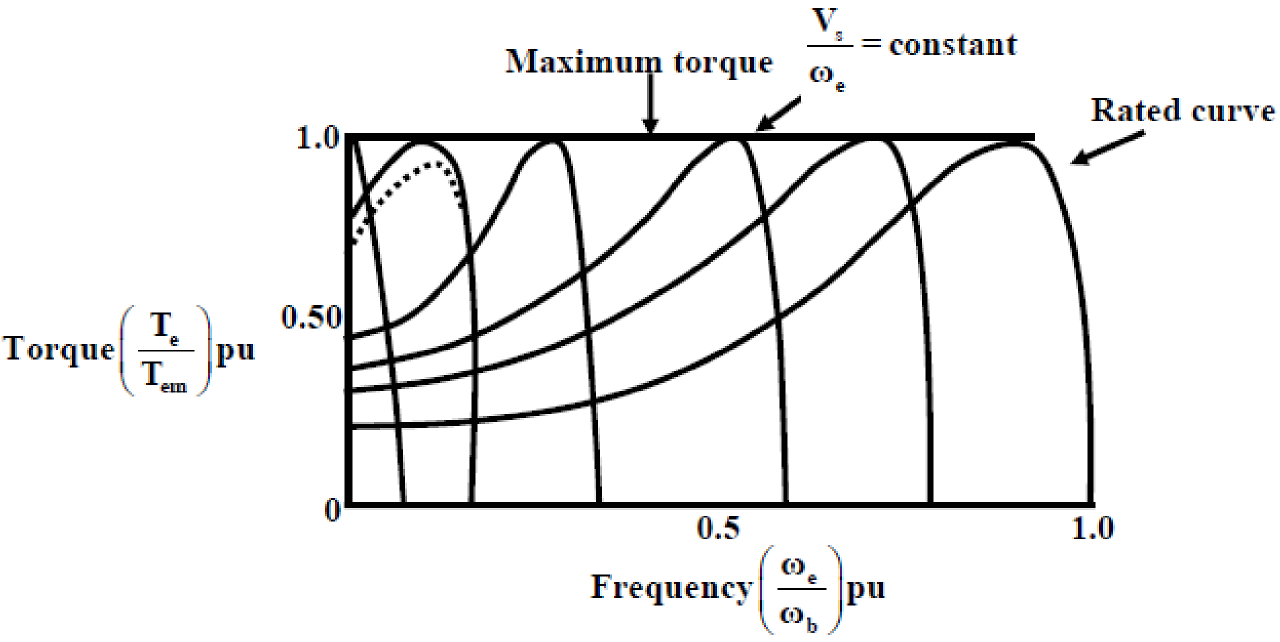
Torque-speed curves at variable supply voltage



Torque-speed curves at variable stator frequency

Variable voltage variable frequency operation with constant V/f

Figure shows the torque-speed curves for constant V/f operation. Note that the maximum torque T_{em} remains approximately constant. Since the air gap flux of the machine is kept at the rated value, the torque per ampere is high. Therefore fast variations in acceleration can be achieved by stator current control. Since the supply frequency is lowered at low speeds, the machine operates at low slip always, so the energy efficiency does not suffer.



Torque-speed curves at constant V/f

Majority of industrial variable-speed ac drives operate with a variable voltage variable frequency power supply.

BLDC Motor Drives

Brushless DC motors, rather surprisingly, is a kind of permanent magnate synchronous motor. Permanent magnet synchronous motors are classified on the basis of the wave shape of their induce emf, i.e, sinusoidal and trapezoidal. The sinusoidal type is known as permanent magnet synchronous motor; the trapezoidal type goes under the name of PM Brushless dc (BLDC) machine. Permanent magnet (PM) DC brushed and brushless motors incorporate a combination of PM and electromagnetic fields to produce torque (or force) resulting in motion. This is done in the DC motor by a PM stator and a wound armature or rotor. Current in the DC motor is automatically switched to different windings by means of a commutator and brushes to create continuous motion. In a **brushless motor**, the rotor incorporates the magnets, and the stator contains the windings. As the name suggests brushes are absent and hence in this case, commutation is implemented electronically with a drive amplifier that uses semiconductor switches to change current in the windings based on rotor position feedback. In this respect, the BLDC motor is equivalent to a reversed DC commutator motor, in which the magnet rotates while the conductors remain stationary. Therefore, BLDC motors often incorporate either internal or external position sensors to sense the actual rotor.

Advantage of Permanent Magnet Brushless DC Motor

BLDC motors have many advantages over brushed DC motors and induction motors. A few of these are:

- Better speed versus torque characteristics
- Faster dynamic response
- High efficiency
- Long operating life
- Noiseless operation
- Higher speed ranges

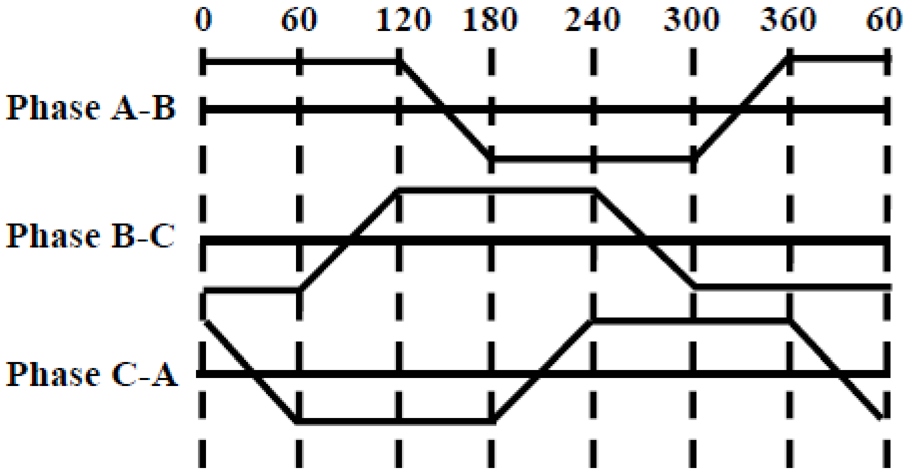
In addition, the ratio of torque delivered to the size of the motor is higher, making it useful in applications where space and weight are critical factors.

Structure of Permanent Magnet Brushless DC Motor

BLDC motors come in single-phase, 2-phase and 3-phase configurations. Corresponding to its type, the stator has the same number of windings. Out of these, 3-phase motors are the most popular and widely used. Here we focus on 3-phase motors.

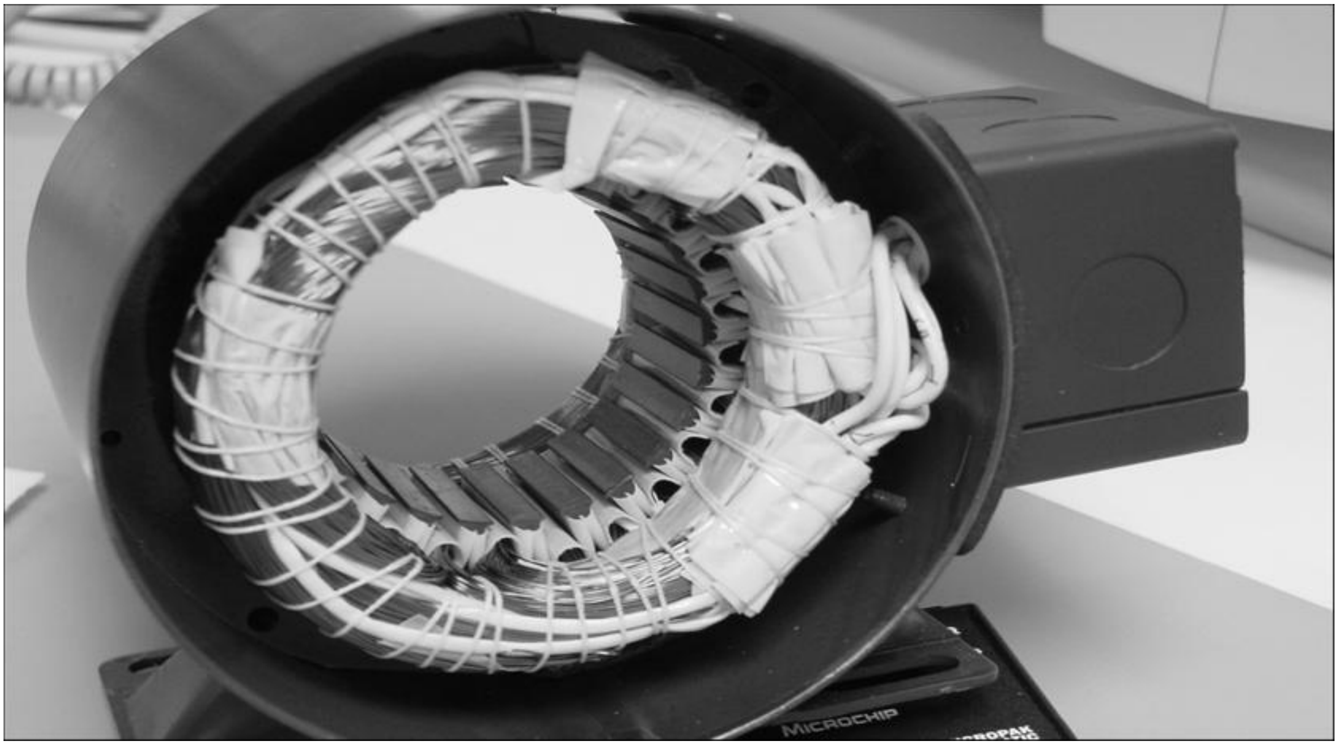
Stator

The stator of a BLDC motor consists of stacked steel laminations with windings placed in the slots that are axially cut along the inner periphery (as shown in Figure 2). Traditionally, the stator resembles that of an induction motor; however, the windings are distributed in a different manner. Most BLDC motors have three stator windings connected in star fashion. Each of these windings are constructed with numerous interconnected coils, with one or more coils are placed in the stator slots. Each of these windings are distributed over the stator periphery to form an even numbers of poles. As their names indicate, the trapezoidal motor gives a back trapezoidal EMF as shown in Figure.



Voltage phase diagram of a 3-phase BLDC motor.

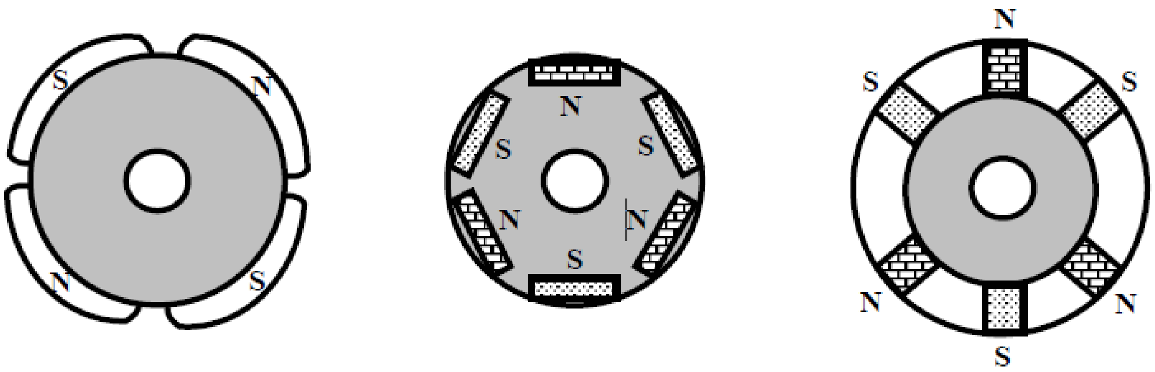
In addition to the back EMF, the phase current also has trapezoidal and sinusoidal variations in the respective types of motor. This makes the torque output by a sinusoidal motor smoother than that of a trapezoidal motor. However, this comes with an extra cost, as the sinusoidal motors take extra winding interconnections because of the coils distribution on the stator periphery, thereby increasing the copper intake by the stator windings. Depending upon the power supply capability, the motor with the correct voltage rating of the stator can be chosen. Forty-eight volts, or less voltage rated motors are used in automotive, robotics, small arm movements and so on. Motors with 100 volts, or higher ratings, are used in appliances, automation and in industrial applications.



Cross-sectional View of the BLDC motor stator.

Rotor

The rotor is made of permanent magnet and can vary from two to eight pole pairs with alternate North (N) and South (S) poles. Based on the required magnetic field density in the rotor, the proper magnetic material is chosen to make the rotor. Ferrite magnets were traditionally used to make the permanent magnet pole pieces. For new design rare earth alloy magnets are almost universal. The ferrite magnets are less expensive but they have the disadvantage of low flux density for a given volume. In contrast, the alloy material has high magnetic density per volume and enables using a smaller rotor and stator for the same torque. Accordingly, these alloy magnets improve the size-to-weight ratio and give higher torque for the same size motor using ferrite magnets. Neodymium (Nd), Samarium Cobalt (SmCo) and the alloy of Neodymium, Ferrite and Boron (NdFeB) are some examples of rare earth alloy magnets. Figure shows cross sections of different arrangements of magnets in a rotor.



Circular core with magnets on the periphery

Circular core with rectangular magnets embedded in the rotor

Circular core with rectangular magnets inserted into the rotor core

Cross-sections of different rotor cores.

Hall Sensors

Unlike a brushed DC motor, the commutation of a BLDC motor is controlled electronically. To rotate the BLDC motor, the stator windings should be energized in a sequence. It is important to know the rotor position in order to understand which winding will be energized following the energizing sequence. Rotor position is sensed using Hall effect sensors embedded into the stator. Most BLDC motors have three Hall sensors embedded into the stator on the non-driving end of the motor. Whenever the rotor magnetic poles pass near the Hall sensors, they give a high or low signal, indicating the N or S pole is passing near the sensors. Based on the combination of these three Hall sensor signals, the exact sequence of commutation can be determined.

Principle of operation and dynamic model of a BLDC Motor

The coupled circuit equations of the stator windings in terms of motor electrical constants are

$$v_{an} = R_a i_a + \frac{u}{dt}(L_{aa} i_a + L_{ba} i_b + L_{ca} i_c) + e_a$$

$$v_{bn} = R_b i_b + \frac{d}{dt}(L_{ab} i_a + L_{bb} i_b + L_{cb} i_c) + e_b$$

$$v_{cn} = R_c i_c + \frac{d}{dt}(L_{ac} i_a + L_{bc} i_b + L_{cc} i_c) + e_c$$

$R_a=R_b=R_c=R$

$L_{aa}=L_{bb}=L_{cc}=L_s$

$L_{ba}=L_{ab}=L_{ca}=L_{ac}=L_{bc}=L_{cb}=M$

$$\begin{bmatrix} v_{an}\\ v_{bn}\\ v_{cn}\end{bmatrix} = \begin{bmatrix} R & 0 & 0\\ 0 & R & 0\\ 0 & 0 & R\end{bmatrix}\begin{bmatrix} i_a\\ i_b\\ i_c\end{bmatrix} + \begin{bmatrix} L_s & M & M\\ M & L_s & M\\ M & M & L_s\end{bmatrix}\frac{d}{dt}\begin{bmatrix} i_a\\ i_b\\ i_c\end{bmatrix} + \begin{bmatrix} e_a\\ e_b\\ e_c\end{bmatrix}$$

Since, $i_a + i_b + i_c = 0$, and with $(L_s - M) = L$, we have,

$$\begin{bmatrix} v_{an}\\ v_{bn}\\ v_{cn}\end{bmatrix} = \begin{bmatrix} R & 0 & 0\\ 0 & R & 0\\ 0 & 0 & R\end{bmatrix}\begin{bmatrix} i_a\\ i_b\\ i_c\end{bmatrix} + \begin{bmatrix} L & 0 & 0\\ 0 & L & 0\\ 0 & 0 & L\end{bmatrix}\frac{d}{dt}\begin{bmatrix} i_a\\ i_b\\ i_c\end{bmatrix} + \begin{bmatrix} e_a\\ e_b\\ e_c\end{bmatrix}$$ where,

R : Stator resistance per phase, assumed to be equal for all phases

L_s : Stator inductance per phase, assumed to be equal for all phases.

M : Mutual inductance between the phases.

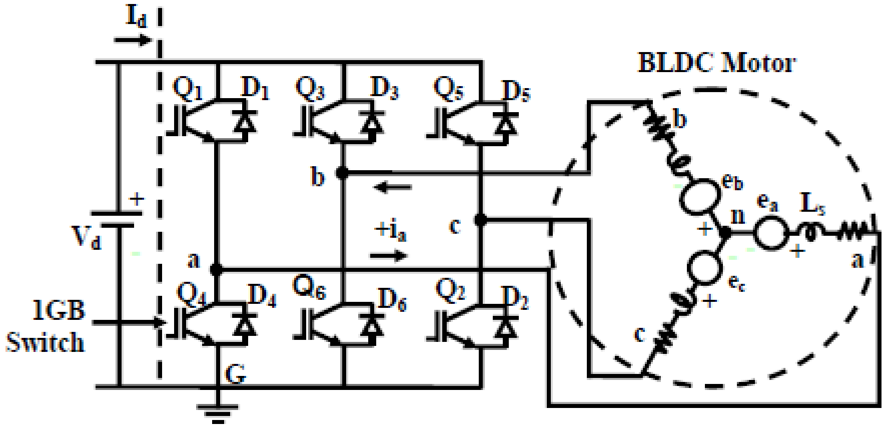
i_a, i_b, i_c : Stator current/phase.

$$e_a = f_a(\theta_r)\lambda_p\omega_m$$
$$e_b = f_b(\theta_r)\lambda_p\omega_m$$

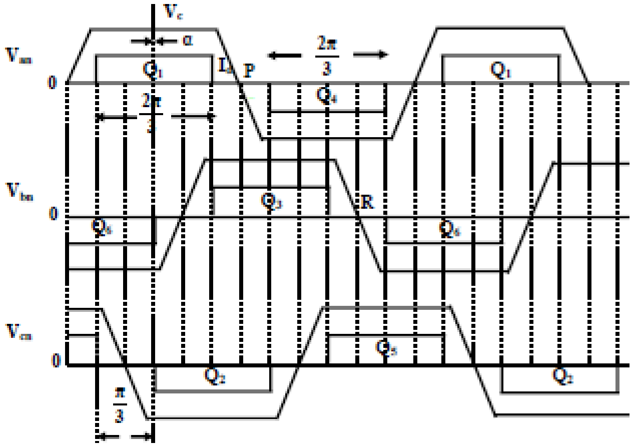
$$e_c = f_c(\theta_r)\lambda_p\omega_m$$

where, ω_m is the rotor mechanical speed and θ_r is the rotor electrical position.

The machine is represented in the figure by a three-phase equivalent circuit, where each phase consists of stator resistance R_s, equivalent self inductance L_s, and a trapezoidal CEMF wave in series. Figure shows the phase diagram of V_{an}, V_{bn}, V_{cn}.

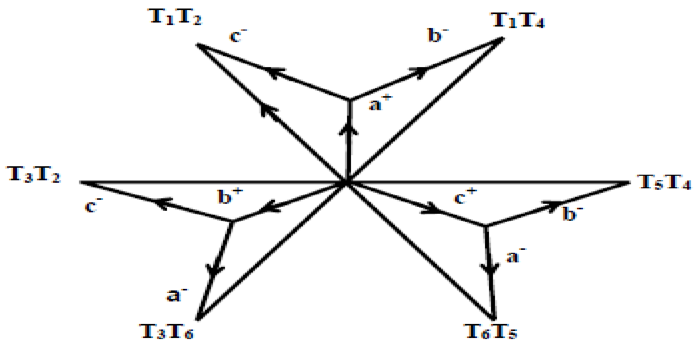


A 3-phase Equivalent Circuit of BLDC motor.



An Illustration of 3-phase switching sequence.

The spatial orientations of the stator MMF vector, under different switching phases of the inverter are shown below in figure 6. Therefore under a cyclic switching scheme one has a rotating stator MMF vector. If the switching can be synchronized with the rotor position, then an approximately fixed angle between the stator flux and the rotor flux can be maintained, while both rotate around the rotor axis. This is similar to the case of the DC motor, where the commutator brush arrangement maintains a fixed spatial direction of the armature flux aligned with the field flux, which is also fixed in space by construction. This is precisely what the inverter switching sequence shown in Fig.5 achieves. The switching instants of the individual transistor switches, $Q_1 – Q_6$ with respect to the trapezoidal emf wave is shown in the figure. Note that the emf wave is synchronized with the rotor. So switching the stator phases synchronously with the emf wave make the stator and rotor mmfs rotate in synchronism. Thus, the inverter acts like an electronic commutator that receives switching logical pulses from the rotor position sensor. This is why a BLDC drive is also commonly known as an electronically commutated motor (ECM).



Phasor diagram of stator MMF vector.

$$T_e = \lambda_p [\, f_a(\theta_r) i_a + f_b(\theta_r) i_b + f_c(\theta_r) i_c \,]$$

The equation of motion for simple system is,

$$T_e = J \frac{d\omega_m}{dt} + T_l + B\omega_m$$

where, J is the inertia of the motor and B is the friction coefficient.

$$\Rightarrow \frac{d\omega_m}{dt} = \frac{1}{J}(T_e - T_l - B\omega_m)$$

The relation between angular velocity and angular position (electrical) is given by

$$\frac{d\theta_r}{dt} = \frac{P}{2}\omega_m$$

where, P is the number of rotor poles. The state variable (θ_r), rotor position, is required to have the function $f_a(\theta_r)$, which is given as the trapezoidal function :

$$\begin{aligned}
f_a(\theta_r) &= 1 && 0 < \theta_r < \pi/3 \\
&= \left(\frac{\pi}{2} - \theta_r\right) * \frac{6}{\pi} && \pi/3 < \theta_r < 2\pi/3 \\
&= -1 && 2\pi/3 < \theta_r < \pi \\
&= -1 && \pi < \theta_r < 4\pi/3 \\
&= \left(\theta_r - 3 * \frac{\pi}{2}\right) * \frac{6}{\pi} && 4\pi/3 < \theta_r < 5\pi/3 \\
&= 1 && 5\pi/3 < \theta_r < 2\pi
\end{aligned}$$

Similarly

$$f_b(\theta_r) = f_a\left(\theta_r + 2\frac{\pi}{3}\right)$$

$$f_c(\theta_r) = f_a\left(\theta_r - 2\frac{\pi}{3}\right)$$

The induced emfs do not have sharp corners, as is shown in trapezoidal functions, but rounded edges. The emfs are the result of the flux linkages derivatives, and the flux linkages are continuous functions. Fringing also makes the flux density functions smooth with no abrupt edges. It is significant to observe that the phase-voltage equation is identical to the armature-voltage equation of a dc machine