651 (76 pages)

UNIT-1

FUNDAMENTALS OF ROBOTS

Robot: An industrial robot is a reprogrammable, multi-functional, manipulator design to move material, parts, tools or special devices through variable programmed motions for the performance of a variety of tasks.

Robotics: Robotics is the art, knowledge based which deals with designing, applying and using robots in human endeavors.

It is an interdisciplinary subject that benefits from mechanical engg, Electrical & Electronical Engg, computer science, biology and many other disciplines.

classification of Robots:

The classification of robots is done based on 3 institutes.

* JIRA (Japanese Industrial Robot Association)
* RIA (Robotics Institute of America)
* AFR (The Association Francaise deRobotique).

(1) According to JIRA, Robots are divided in to 6 classes.

class 1: Manual-Handling Device: A device with multiple degrees of freedom that is actuated by an operator.

class 2: Fixed / sequence Robot: A device that performs the successive stages of a task according to a predetermined unchanging method & is hard to modify.

class 3 : Variable - sequence robot : It is same as fixed sequence robot but easy to modify.

class 4 : Play Back Robot : A human operator performs the task manually by leading the robot, which records the motion for later playback. The robot repeats the same motion accorded to the recorded information.

class 5 : Numerical control Robot : The operator supplies the robot with a movement program rather than teaching it manually.

class 6 : Intelligent Robot : A robot with the means to understand its environment and the ability to successfony complete a task despite changes in the surrounding conditions under which it is to be performed.

(2) → The Robotics Institute of America (R.I.A) considers only 3-6 classes as robots.

(3) According to AFR, they considered 4 classifications

Type A : Handling devices with manual control to telerobotics

Type B : Automatic handling devices with predetermined cycles.

Type C : programmable, servo controlled robots with continous or point-to-point trajectories

Type D : It is same as type c, but with capability to acquire information from its environment.

Components of Robot :

(1) Manipulator or Rover

(2) End effector

(3) Actuators

(4) Sensors

(5) controller

(6) processor

(7) software

(1) **Manipulator :** manipulator is the main body of the robot and consists of the links, the joints and other structural elements of the body.

(2) **End effector :** This is the part that is connected to the last joint of a manipulator which generally handles objects, makes connection to other machines or performs the required tasks.

(3) **Actuators :** Actuators are the muscles of the manipulator. common types of actuators are servomotors, stepper motors, pneumatic & hydraulic cylinders. Actuators are controlled by the controller.

(4) sensors : Sensors are used to collect information about the internal state of the robot and to communicate with the outside environment.

The robots are often equipped with external sensory devices such as a vision system, touch and tactile sensors, speech synthesizers etc which enable the robot to communicate with the outside world.

(5) controller : The controller receives its data from the computer, controls the motions of the actuators and coordinates the motions with the sensory feedback information.

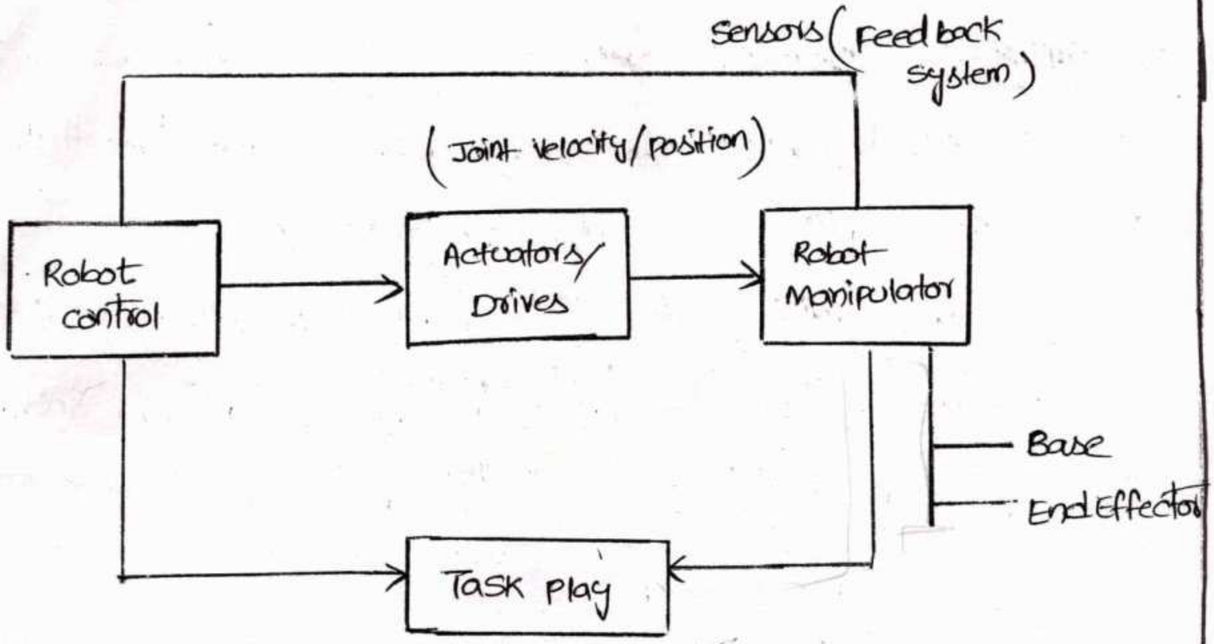
(6) processor : The processor is the brain of the robot. It calculates the motions of the robot's joints, determines how much and how fast each joint must move to achieve the desired location and speeds and oversees the coordinated actions of the controller and the sensors.

(7) software : Three groups of software are used in a robot. They are :

① operating system : It operates the computer.

② Robotic software : It calculates necessary motions of each joint based on kinematic equations of robot.

③ collection of routines and application programs.



Laws of Robotics:

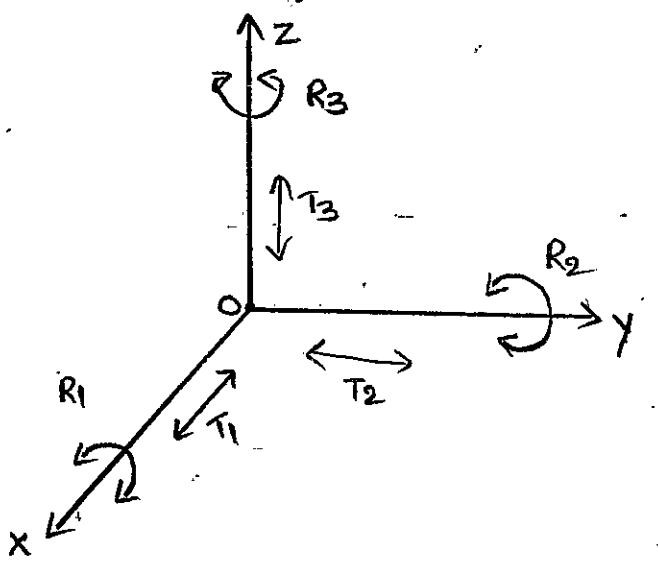
(1) A robot should not injure a human being or, through inaction, allow a human to be harmed.

(2) A robot must obey the orders given by human except when that conflicts with the first law

(3) A robot must protect its own existence unless that conflicts with first or second law.

Degree of freedom :

The number of independent movements that an object can perform in a 3-D space is called the number of degrees of freedom (DOF). Thus, a rigid body free in space has 6 DOF, three for position and three for orientation. These six independent movements are shown in the below figure.

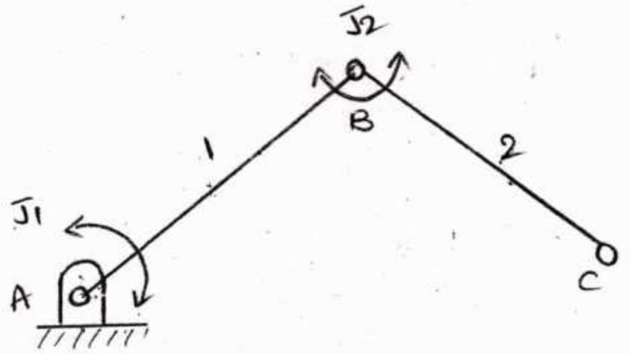


(i) Three translations (T_1, T_2, T_3) representing linear motions along three perpendicular axes, specifying the position of the body in space.

(ii) Three rotations (R_1, R_2, R_3) which represent angular motions about the three axes, specify the orientation of the body in space.

consider an open Kinematic chain of two links with revolute joints at A and B as shown in below fig. The first link is connected to the ground by a joint at A. Therefore, link 1 can only rotate about joint 1 (J_1) with respect to ground and contribute one independent Variable.

Link 2 can rotate about joint 2 (J_2) with respect to link 1 contributing another independent variable and so another DOF.



It is concluded that an open Kinematic chain with one end connected to the ground by a joint and the farther end of the last link free has as many degree of freedom as the number of joints in the chain. It is assumed that each joint has only one DoF.

The DOF is also equal to the no of links in the open Kinematic chain. In the above fig, the Kinematic chain manipulator with 2 DOF has 2 links & 2 Joints

Robot Joints :

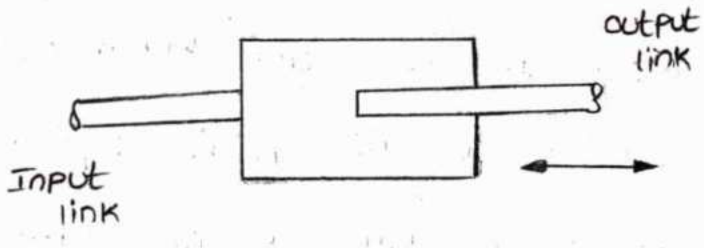
Robot Joints are classified in to two types ,

(1) Linear (or) prismatic
- (a) Translational Joint
- (b) orthogonal Joint

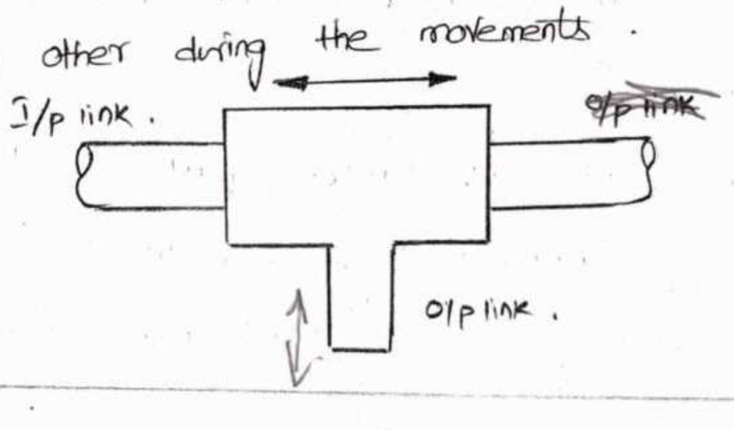
(2) Rotational
- (a) Rotational Joint
- (b) Twisting Joint
- (c) Revolving Joint

(1) Linear Joint : (prismatic Joints)

(a) Translation joint :- The relative movement between the input and output links is a translational sliding motion with the axis of the 2 links being parallel.

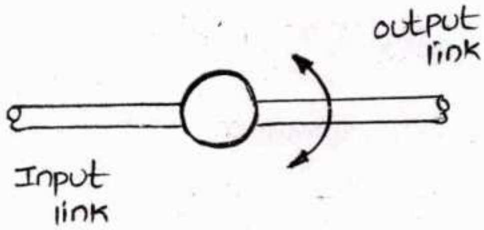


(b) orthogonal joint :- It is also a translational sliding motion but the input and output links are perpendicular to each other during the movements .

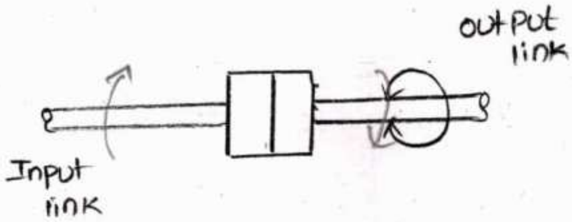


Rotational Joint :

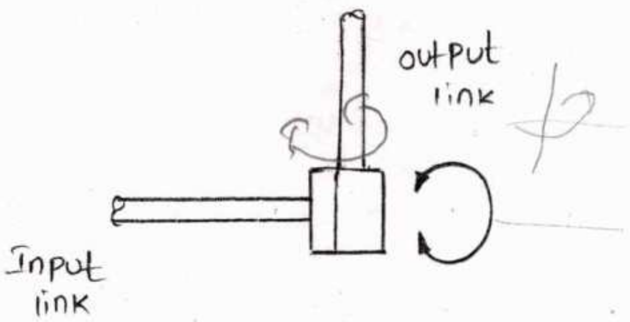
(a) Rotational Joint : This type provides rotational relative motion perpendicular to the axis of input and output links.



(b) Twisting Joint :- This joint also involves rotary motion but the axis of rotation is parallel to the axis of the two links.



(c) Revolving joint :- In this type of joint, the axis of input link is parallel to the axis of rotation and the axis of output link is perpendicular to the axis of rotation.

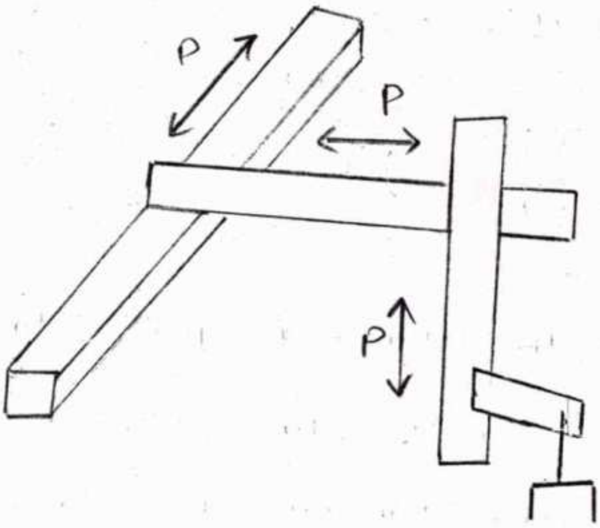


* **Robot configurations or Robot co-ordinates:**

These are of 4 types. They are:

(1) cartesian configuration (3P)
(2) cylindrical configuration (2P, R)
(3) Spherical or polar configuration (2R, P)
(4) ~~Art~~ Articulated or Jointed Arm configuration (3R) — Scara 3R

(1) **cartesian configuration (3P):** (Gantry robot)



Disadv.
→ complex in structure
→ occupies more space

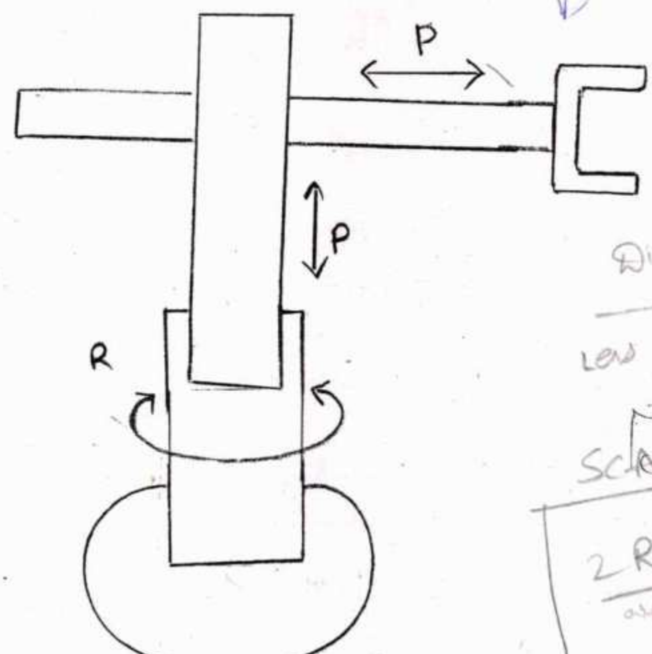
These type of robots have 3 degrees of rigid body freedom. They have three prismatic joints which produces three linear motions in X, Y, Z directions.

Examples: IBM RS-1 (Model 7565)

Advantages: (1) Rigid structure of Box frame type
(2) High repeatability with least error
(3) High load carrying capacity.

Cylindrical Robot configuration (R, 2P, ●):

These type of robots have two-prismatic joints and one revolute joint. Two prismatic joints give linear movement about any two axes and the third movement rotation is produced by the revolute Joint



Dis adv - less accurate.

SCARA

2R, 1P
are parallel

assembly

Example : GMF Model

Advantages : (1) High rigidity of the manipulater

(2) Higher load carrying capacity

(3) Geometrical advantage in specification.

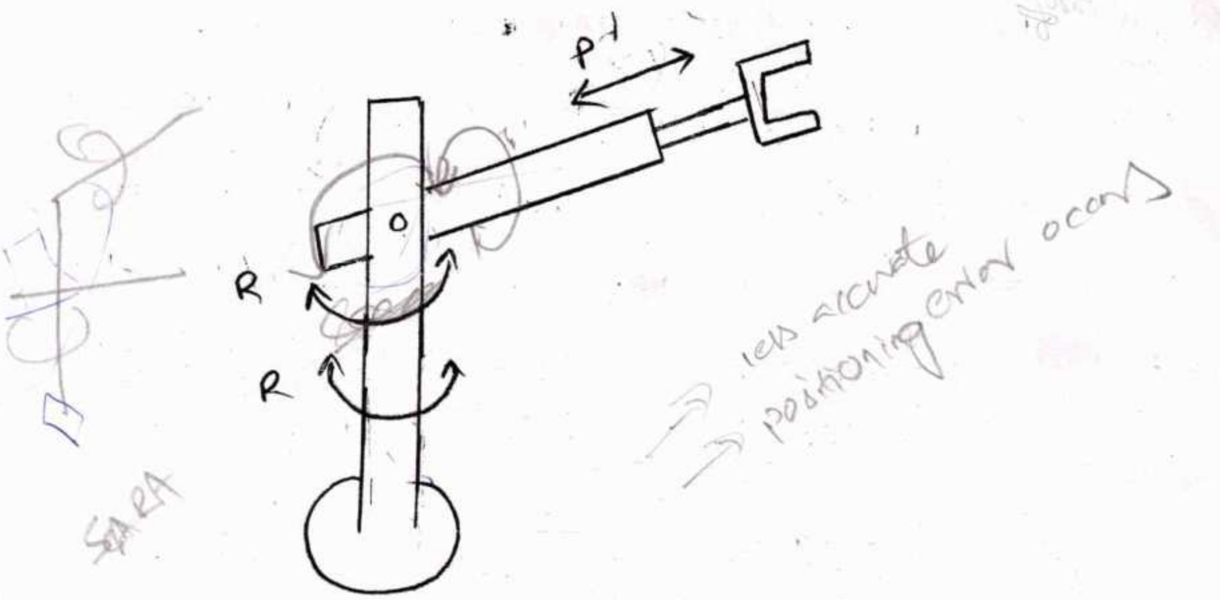
(4) less complex structure.

(3) spherical or polar configuration : (2R, P)

It uses a telescoping arm that can be raised or lowered about a horizontal pivot. It has 2 revolute joints and 1 prismatic joint.

Examples : Unimate 2000 series

MAKER 110.



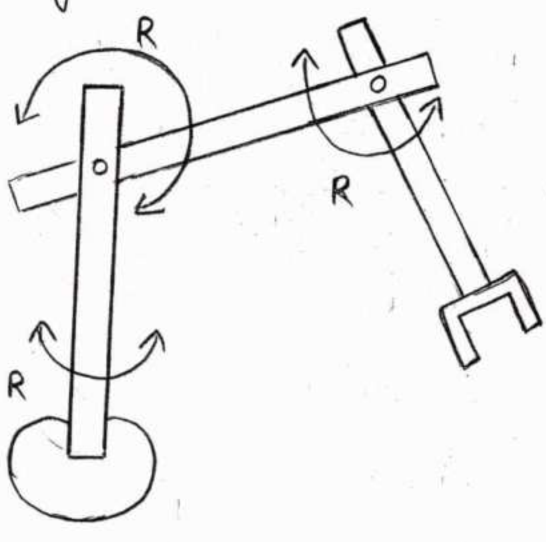
Advantages : (1) Higher reach from the base

(2) Geometrical advantage in specification

(3) Machine loading applications need this type.

(4) Jointed Arm (or) Articulated configurations (3R) :

In An articulated robot all joints are revolute, similar to human arm. These are the perhaps the most common configurations for industrial robot.



Examples : cincinnati milacron T3 (Model T76) robot

SCARA (selective compliance Assembly Robot Arm)

Advantages :

(1) Higher reach from the base

(2) useful in continous path generation applied to spray painting and arc welding.

(3) Reaching the conjested small openings without interference.

(4) H.

Reference frames :-

There are three types of reference frames attributed to the robot structures.

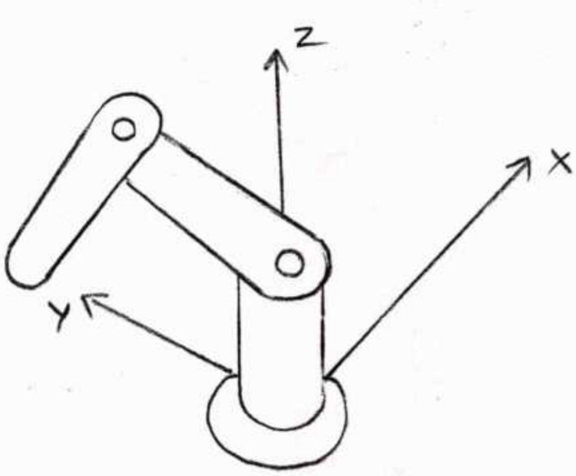
(1) Base reference frame (or) world reference frame

(2) Joint reference frame

(3) Tool refegence frame.

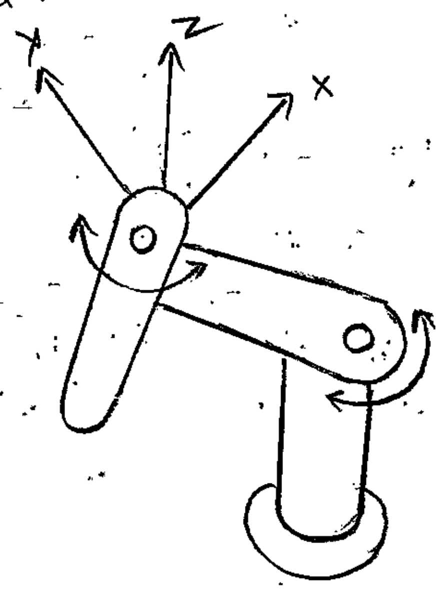
Base Reference frame :-

The basic x, y and z axes are the three axes of the base. The base may be fixed or rotate about the z-axis according to the need of the application. This is the universal reference frame.



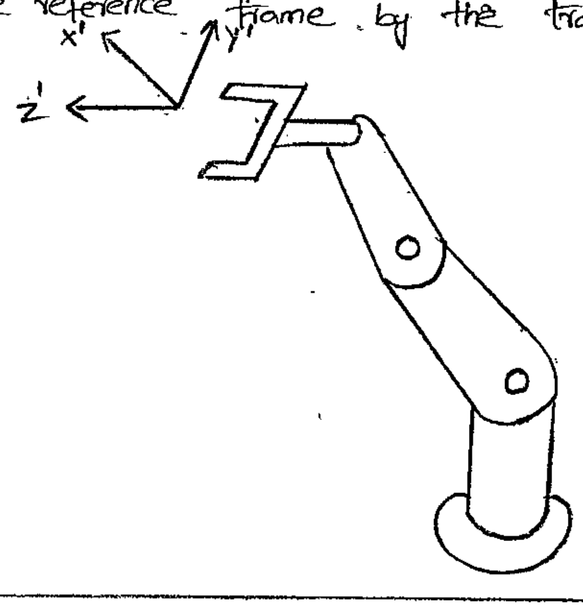
Joint Reference frame:

The reference axes defined at the joints of the robot are called the joint reference frame.
The joint can have both Translatory and rotational movements about it defined axes. In this case frame is not fixed.



Tool Reference frame:

This is the local frame of reference defined by the axes at the arm tip or the robot hand.
The tip or the tool reference frame is related to the base reference frame by the transformation of the coordinates



Payload → max load that can be lifted by a Robot
Precision → how accurately a robot can reach its point
Repeatability ⇒ if motion is repeated again & again
Reach max distance a robot can reach
Speed

Robot characteristics :-

The following are the characteristics of robot.

(1) Payload ⇒
(2) Reach
(3) precision
(4) Repeatability.
(5) speed (6) Accuracy. How closely it agrees the true value.

(1) Pay load :-

Pay load is the weight a robot can carry and still remain within its other specifications. For example, a robot's maximum load capacity may be much larger than its specified Payload, but at the maximum level, it may become less accurate, may not follow its intended path accurately or may have excessive deflection.

(2) Reach :-

Reach is the maximum distance a robot can reach within its work envelope. Reach is a function of the robot's joint lengths and its configuration.

(3) precision :-

Precision is defined as how accurately (correctly) a specified point can be reached. This is a function of the resolution of the actuators as well as its feedback devices. Most industrial robots can have precision of 0.001 inch or better.

(4) Repeatability :-

Repeatability is how accurately the same position can be reached if the motion is repeated many times. The radius of a circle that is formed by repeated motion is repeatability.

Programming Modes :-

The programming modes are :

(1) Physical setup.

(2) Lead Through or Teach mode

(3) Continous walk - Through mode

(4) Software mode.

(1) Physical Setup : In this mode, an operator sets up switches and hand stops that control the motion of the robot. This mode is used along with other devices, such as Programmable Logic controllers (PLC)

(2) Lead Through or Teach mode : In this mode, the robot's joints are moved with a teach pendant. When the desired location and orientation is achieved, the location is entered in to the controller. During playback, the controller will move the joints to the same locations & orientations.

(3) Continous walk - Through mode : In this mode, all the robot's joints are moved simultaneously, while the motion is continously sampled and recorded by the controller. During playback, the exact motion that was recorded is executed.

(4) Software mode : In this mode, a program is written off-line or on-line and is executed by the controller to control the motions.

Advantages of Robot :-

- Robotics and automation can, in many situations, increase productivity, safety, efficiency, quality and consistency of products.
- Robots can work in hazardous environments without the need for life support, comfort or concern about safety.
- Robots need no environmental comfort such as lighting, air conditioning, ventilation & noise protection.
- Robots work continously without experiencing fatigue.
- Robots have repeatable precision at all times, unless something happens to them.
- Robots can be much more accurate than humans.
- Robots and their accessories and sensors can have capabilities beyond that of humans.
- Robots can process multiple stimuli or tasks simultaneously.
- Robots replace human workers creating some economic problems such as lost salaries and social problems such as dissatisfaction and resentment among workers.
- Robots lack capability to respond in emergencies.

Disadvantages of Robot :-

- Robots lack capability to respond in emergencies, unless the situation is predicited & the response is included in the system.
- They lack decision-making power
- Robots are costly due to intial cost of equipment, installation costs, need for peripherials, training and programming.

Applications of Robots :-

(1) material handling and machine loading & unloading :

In this application, the robot's function is to move materials or parts from one location in the workcell to some other location.

(2) process application :- This category includes spot welding, arc welding, spray painting and other operations in which the function of the robot is to manipulate a tool to accomplish some manufacturing process in the workcell.

(3) Assembly and Inspection :- Robots are used in assembly operations. Inspection robots make use of sensors to guage and measure quality characteristics of manufactured product.

END EFFECTORS

End Effector :

An End effector is a "device that attaches to the wrist of the robot arm and enables the general purpose robot to perform a specific task". It is some-times referred to as the robot hand.

Types of End Effectors :

End effectors can be majorly divided in to two categories.

(1) Grippers

(2) Tools.

(1) Grippers :

Grippers are the end effectors used to grasp and hold the object or workpart that are to be moved by the robot.

Eg :- Parts handling applications including machine loading and unloading, picking parts from conveyors and arranging parts on to a pallet.

Grippers may be mechanical grasping devices, magnetic grippers, suction cups or other means for holding the objects.

Grippers are classified in to two types.

(1) single gripper

(2) Double gripper.

single gripper :

In single gripper one grasping device is mounted on the Robot's wrist.

Double gripper :

A double gripper has two grasping devices and attached to the wrist and is used to handle two seperate objects. The two gripping devices can be actuated independently. It is specially used for machine loading and unloading applications.

(2) Tools :-

Tools are endeffectors designed to perform work on the part rather than to grasp it. By, definition, the tool type end effector is attached to the robot's wrist.

Sometimes grippers are used to hold tools rather than work parts because the job requires several tools to be manipulated by the robot during the work cycle.

Eg: Deburring operation in which several different sizes and geometries of deburring tool must be held in order to reach all surfaces of the work part.

Gripper Mechanisms :

There are various ways of classifying mechanical grippers and their actuating mechanisms.

These are classified in to

(1) According to the type of finger movement used by gripper.

(2) According to the type of kinematic device used to actuate the finger movement.

(1) According to the type of finger movement used by gripper:

(i) pivoting movement

(ii) Linear or translational movement.

(i) pivoting movement :

In the pivoting movement, the fingers rotate about fixed pivot points on the gripper to open and close.

This motion is usally accomplished by some kind of linkage mechanism.

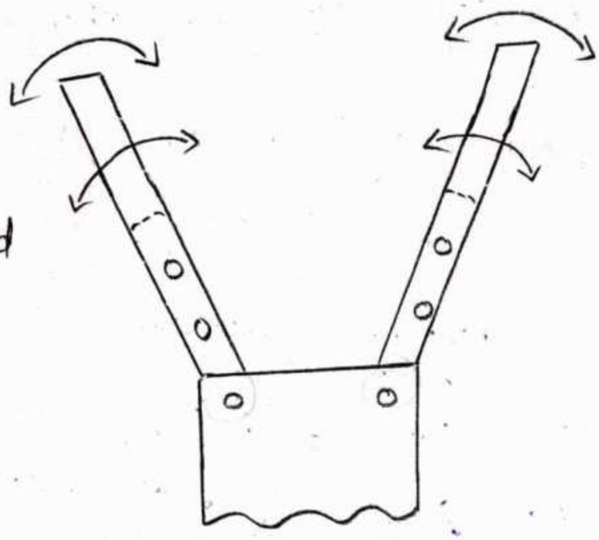


Fig rep's pivoting type of fingers.

(ii) Linear (or) Translational Movement :

In the linear movement, the fingers open and close by moving in parallel to each other. This is accomplished by means of guide rails or guide ways so that each finger base slides along a guide rail during actuation. This is accomplished by means of a linkage which would maintain the fingers in a parallel orientation to each other during actuation.

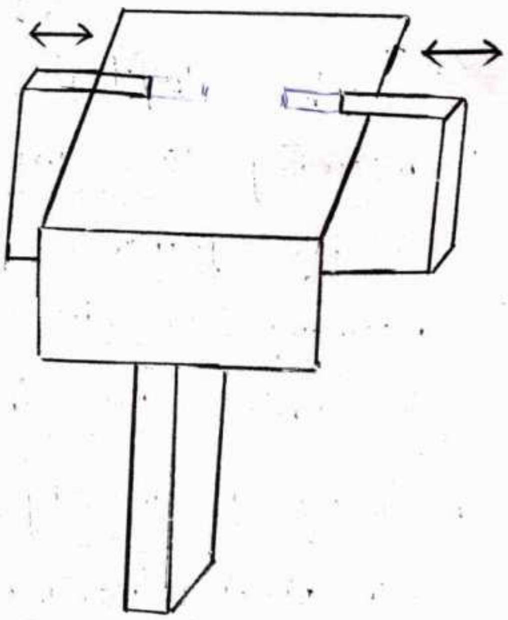


Fig rep's linear motion of fingers.

(2) According to the type of Kinematic device used to actuate the finger movement.

(i) Linkage actuation
(ii) Cam actuation
(iii) Gear and rack actuation
(iv) Screw actuation
(v) Rope and pulley actuation.

(i) Linkage actuation :

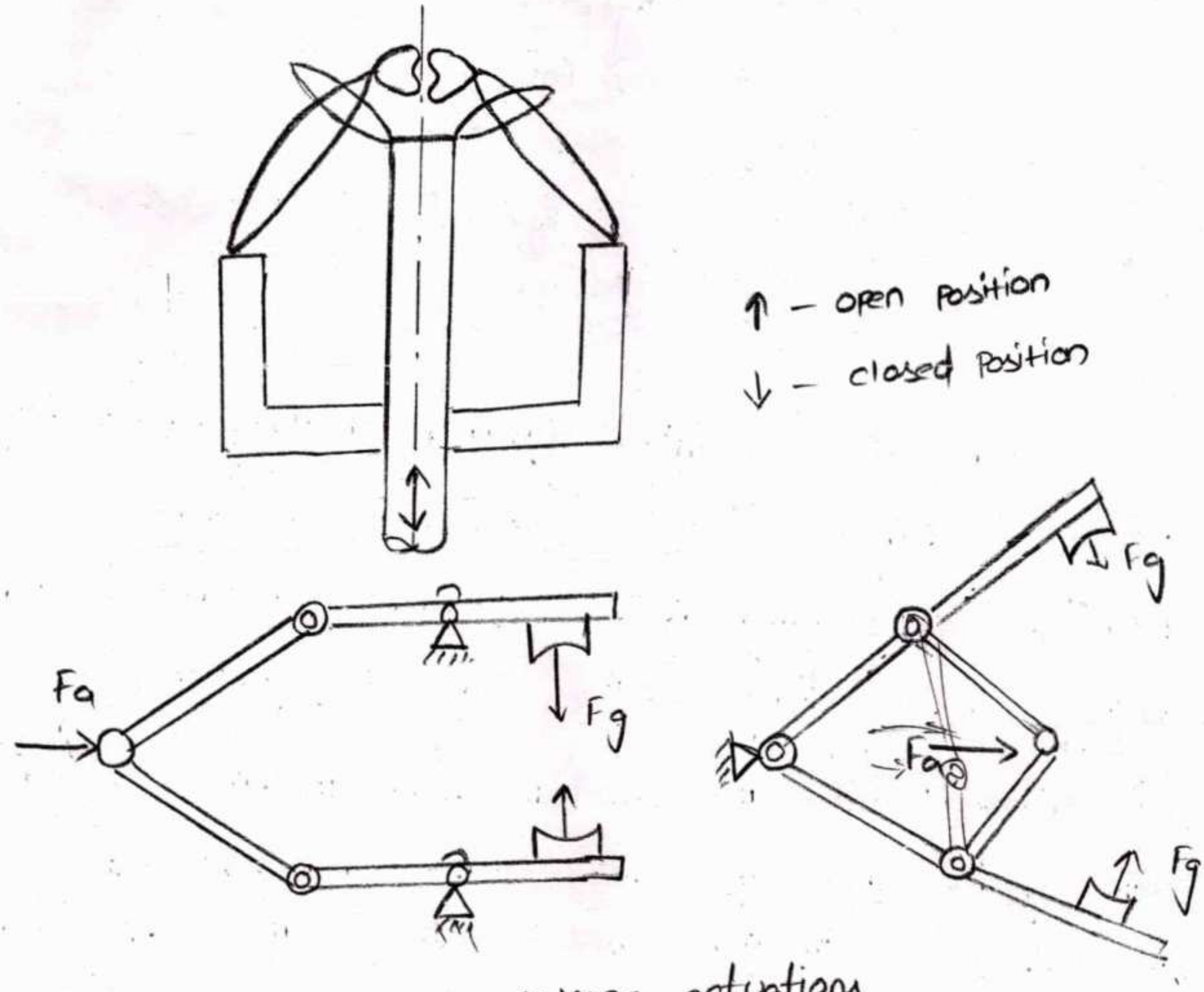
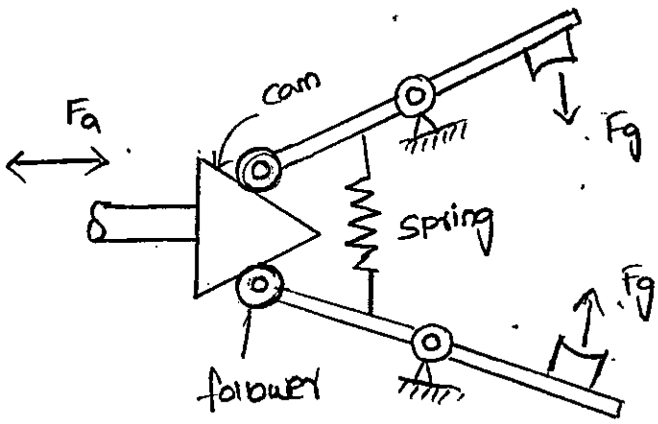


fig : Some linkage actuations

The linkage category covers a wide range of design possibilities to actuate the opening and closing of the gripper. The design of the linkage determines how the input force Fa to the gripper is converted in to the gripping force Fg applied by the fingers.

The linkage configuration also determines how wide the gripper fingers will open and how quickly the gripper will actuate.

(2) Cam actuation :



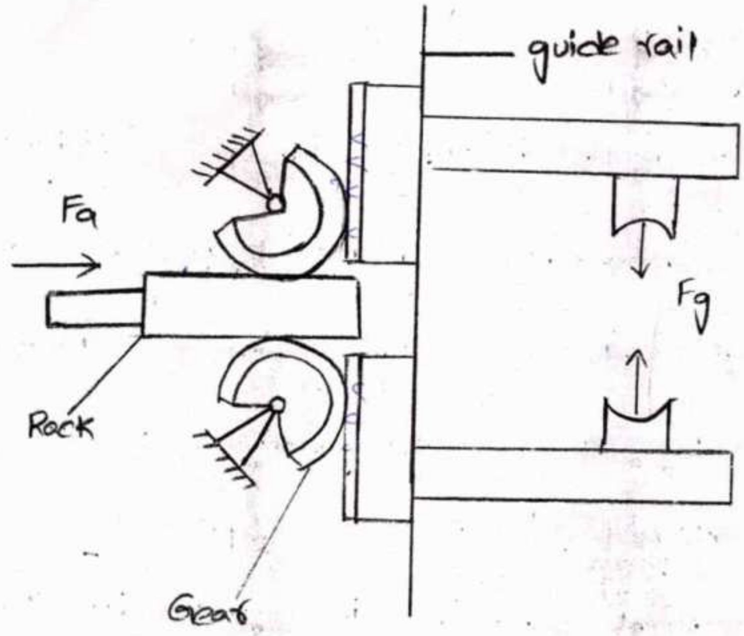
The cam actuated gripper includes a variety of possible designs. A cam and follower arrangement often uses a spring loaded follower, can provide the opening and closing action of the gripper.

Ex: Movement of the cam in one direction would force the gripper to open while the movement of cam in opposite direction would cause the spring to force the gripper to close.

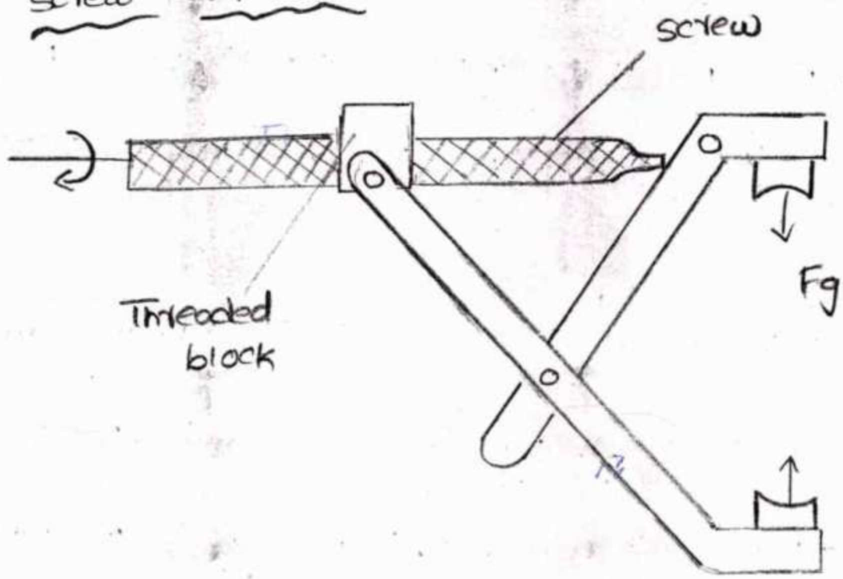
Advantage: spring actuation would accomodate different sized parts.

(3) Gear and Rack actuation :

The gear and rack actuation method of actuating the gripper fingers uses a a gear and rack configuration. The rack gear would be attached to a piston or some other mechanisms that would provide a linear motion. Movement of the rack would drive two partial pinion gears and these would inturn open and close the fingers.



(4) screw actuation:



The screw is turned by a motor usally accompained by a speed reduction mechanism. when the screw is rotated in one direction, this causes a threaded block to be translated in one direction. when a screw is rotated in the opposite direction, the threaded block moves in the opposite direction. The threaded block is in turn connected to the gripper fingers to cause the corresponding opening and closing action

(5) Rope and pulley actuation:

Rope and pulley mechanisms can be designed to open and close a mechanical gripper. Because of the nature of these mechanisms, some form of tension devices must be used to oppose the motion of the rope or cord in the pulley system.

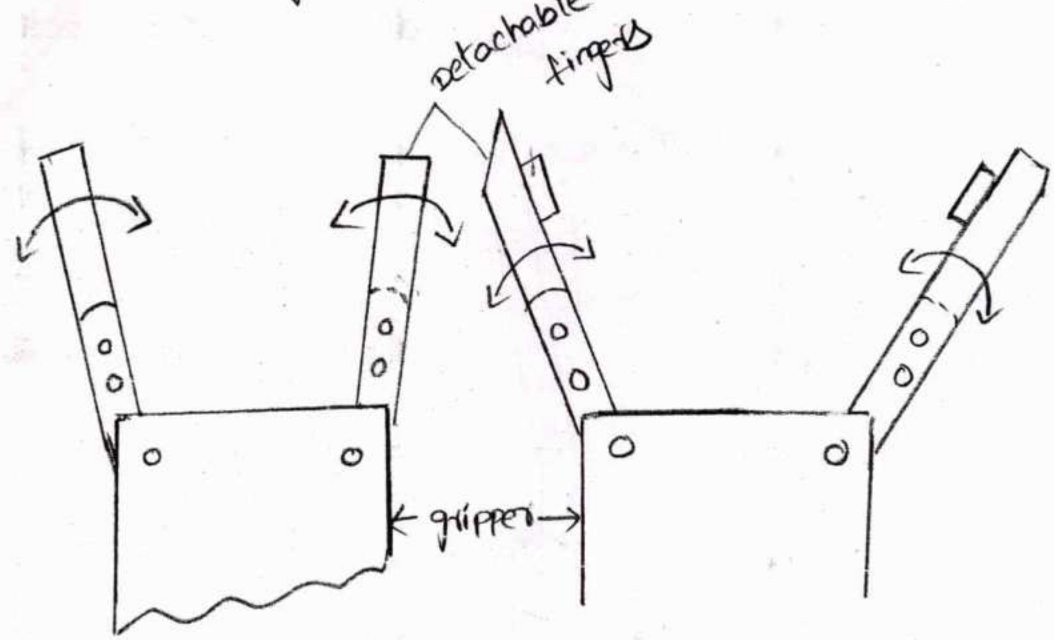
Ex: The pulley system might operate in one direction to open the gripper and the tension device would take up the slack in the rope and close the gripper when the pulley system operates in the opposite direction.

Mechanical gripper:

A mechanical gripper is an endeffector that uses mechanical fingers actuated by a mechanism to grasp an object. The finger some times called as Jaws, that may contact with the object.

The fingers are either attached to the mechanisms or an integral part of the mechanism.

The use of replacable fingers allows for wear and interchangeability.



An example of this interchangeability feature is illustrated in the above figure in which the gripper is designed to accomdate fingers of varying sizes.

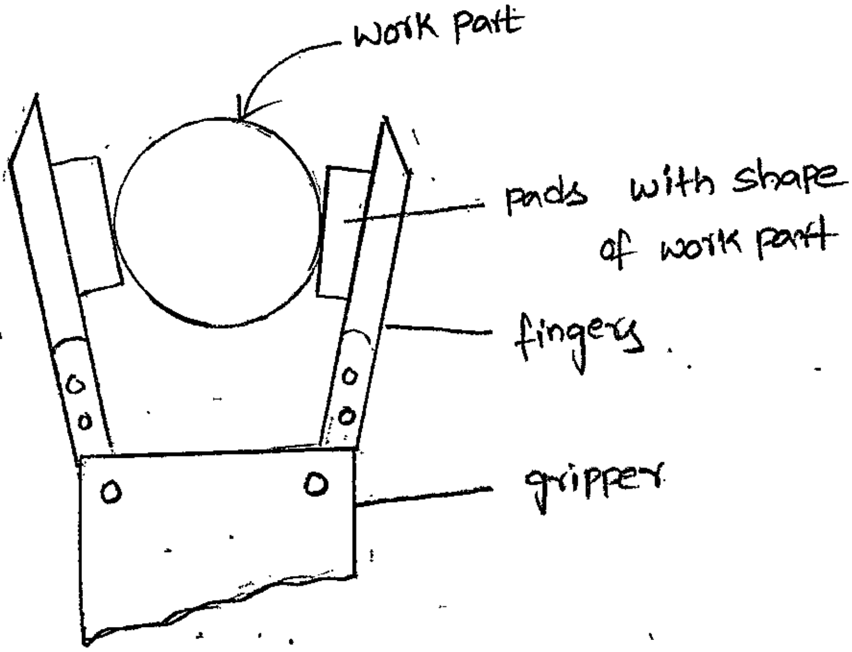
The function of the gripper mechanism is to translate some form of power input (from the robot) in to the grasping action of the fingers against the path.

There are 2 ways of constraining the path in the gripper.

(1) By physical constriction of the path width in the fingers.

(2) By friction between the fingers and work path.

(1) By physical constriction of the path. with in the fingers :-

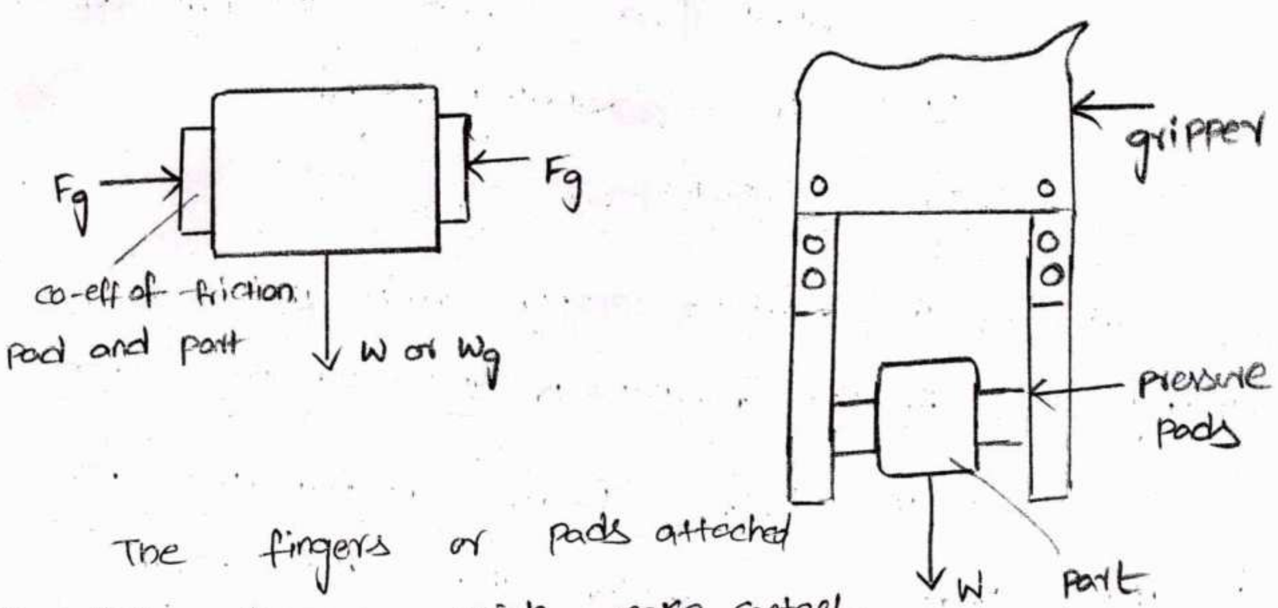


In this approach, the gripper fingers enclose the path to some extent, there by constraining the motion of the path.

This is actually accomplished by designing the contacting surfaces of the fingers to be in the approximate shape of the part geometry.

(2) By friction b/w the fingers and work path;

The second way of holding the part is by friction between fingers and the work path. with this approach the fingers must apply a force that is sufficient for friction to retain the path against gravity acceleration any other force that might arise during the holding portion of the work cycle.



The fingers or pads attached to the fingers which make contact with the part are generally soft. This tends to increase the coefficient of friction between the part and the contacting finger surface. It also serves to protect the path surface from scratching or other damage.

The friction method of holding the part results in a less complicated and therefore the gripper design is less expensive and it tends to ~~be~~ readily adaptable to a greater variety of work parts.

From fig. 1

$$W = n_f \cdot u \cdot f_g$$

where w = weight of the part

μ = coefficient of friction of the finger contact surface against the part surface.

n_f = No of contacting fingers

F_g = gripper force

This equation would apply when the force of gravity is directed parallel to the contacting surfaces.

If the force tending to pull the part out of the fingers is greater than the weight of the object then,

$$\boxed{wg = n_f \cdot \mu F_g}$$

where g = the g factor

The g factor takes the combined effect of gravity and acceleration.

→ $g = 3$, if the deceleration force is applied in the same direction as the gravity force.

→ $g = 1$ if the acceleration is applied in opposite direction to gravity force.

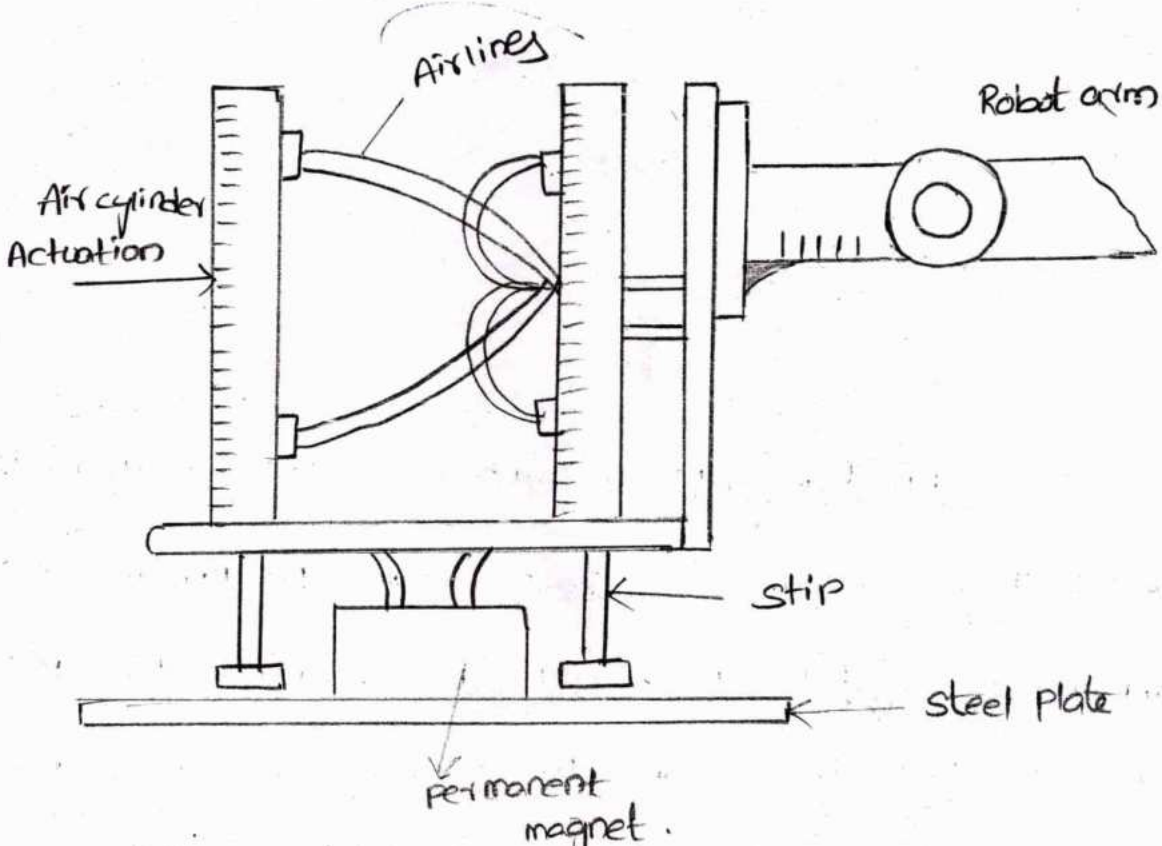
→ $g = 2$ if the acceleration is applied in a horizontal direction to gravity force.

Magnetic Gripper :

Magnetic grippers are a feasible means of handling ferrous materials. magnetic grippers are classified in to two types. They are :

(1) Electro magnet gripper
(2) permanent magnet gripper.

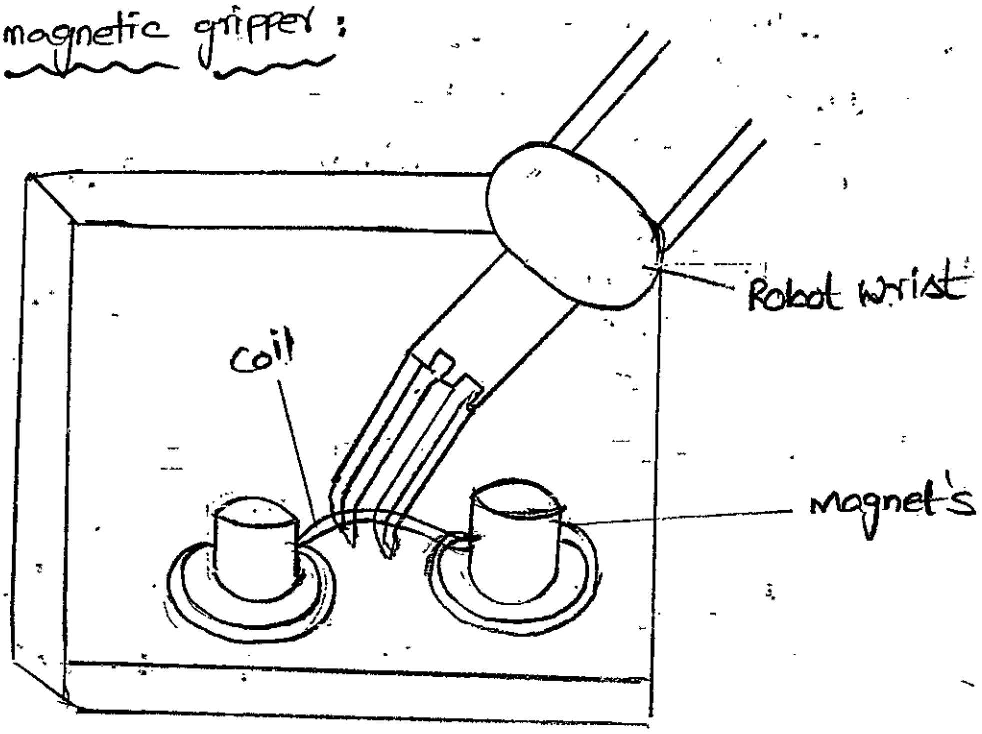
(1) ~~Electro~~ permanent magnetic gripper :-



~~Electro magnet~~

permanent magnets have the advantage of not requiring an external power source to operate the magnet. These are often considered for handling tasks in hazardous environment requiring explosion proof apparatus. The fact that no electrical circuit is needed to operate the magnet is that it reduces the danger of spark which might cause ignition in such an ignition.

(2) Electro magnetic gripper:



Electromagnetic grippers are easier to control but they require a source of DC power and an appropriate control unit. It is easier to ~~part se~~ release the part at the end of the handling cycle with electromagnetic gripper and also easier to accomplish with an electromagnet than with permanent magnet.

Advantages of magnetic gripper:

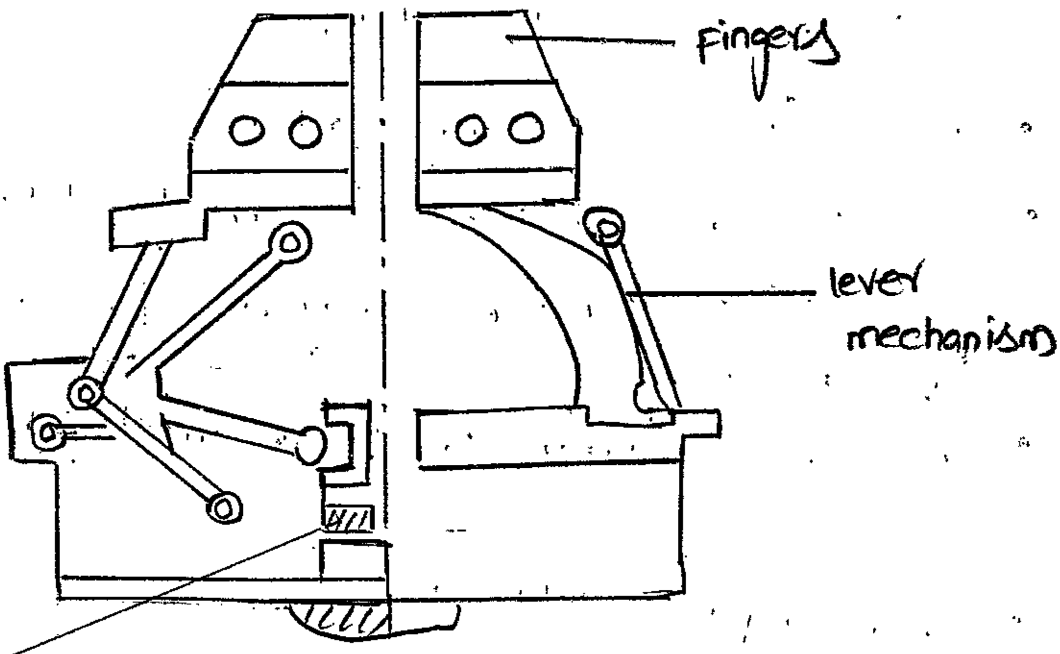
- pick up times are very fast
- Variations in part size can be tolerated. The gripper does not have to be designed for one particular work part.
- They have ability to handle metal parts with holes
- They require only one surface for gripping.

Disadvantages of magnetic gripper:

- The residual magnetism remained in the workpiece may cause a problem in subsequent handling.
- The Possible side slippage and other errors which limit the precision of this means of handling.
- The problem of picking up only one sheet from a stack.

pneumatic grippers:

The pneumatic gripper is equipped with roller membrane cylinder with a rolling motion replacing conventional piston cylinder. The motion is transmitted to gripper fingers by means of level mechanism.



The grippers are actuated by switching values in the circuit. The finger stroke is limited by end stops or the workpiece to be gripped. The griping force is determined by the pressure of air applied and the leverage.

Tools as End effector:

In many applications, the robot is required to manipulate a tool rather than a workpart. In a limited number of these applications the end effector is a gripper that is designed to grasp and handle the tool. The reason for using a gripper in these applications is that there may be more than one tool to be used by the robot in the workcycle. The use of a gripper permits the tools to be exchanged during the cycle, and thus facilitates this multitool handling function.

In most of the robot applications in which a tool is manipulated, the tool is attached directly to the robot wrist. In these cases the tool is the endeffector. Some examples of tools used as endeffector in robot applications include

- spot welding tools
- Arc welding torch

- Spray painting nozzle
- Rotating spindles for operations such as
  - drilling
  - Routing
  - wire brushing
  - grinding
- Heating torches
- Water Jet cutting tool

Design considerations for grippers (or) end effectors:

Part specification	Performance specification	Source specification	Position specification	Environmental specification	Material specification.
• weight of the part • size and shape of component • Tolerance on the part size • change of shape and size during processing • surface finish of the part. • care for delicacy of the part to be handled.	• coefficient of friction between part & object • speed & acceleration during motion • Accuracy & repeatability of the robot • lead time and cycle time. • Interchange-ability of fingers • memory capacity of the controller	• Pneumatic : Air pressure & discharge and cylinder size • Electrical : The power rating & specification of actuator. • Hydraulic ; The oil pressure, volume flow rate and power pack. • mechanical ; Linkage design & transmission design specification	• Holding methods • physical difficulty. • Length of fingers. • The object and tool orientation. • product design changes • spare parts specifications	• Heat & temp of object and atmosphere • Humidity & moisture. • Dirty & safety • Hazardness of chemicals used	• strength and rigidity • Durability & fatigue factors • Friction properties • Factor of safety • Design standards • compatibility with the environment.

10

Selection consideration of gripper :

Actuation selection	Drive selection	Protection selection	Process selection
• Mechanical or friction gripping methods • pad shape selection • Vaccum actuation • Magnetic grasping • Adhesive gripping • Expandable bladder type actuation.	• pneumatic drive systems • Hydraulic drive system for heavy duty of operation • Electrical drive for light duty applications • Speed reduction of mechanical transmission.	• Heat shield for sensors and actuators • Forced cooling by air and water to take away the heat. • selection of heat resistance material for fingers and components of grippers • shield for hazardous chemicals.	• Accurate processing methods for fingers. • Leak prevention for pneumatic / hydraulic actuators. • Ease of assembly of fingers and linkages.

Homogeneous Transformation matrix : [HT - matrix]

Rotation 3×3	Position 3×1
prospective matrix 1×3	stretch 1×1 (scale factor)

4×4

A point vector $v = ai + bj + ck$ can be represented in three dimensional space by the column matrix.

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

where $a = x/w$, $b = y/w$, $c = z/w$ and w is a scaling factor.

A vector can be translated or rotated in space by means of a transformation. The transformation is accomplished by 4×4 matrix 'H'. For instance the vector 'v' is transformed in to the vector u by the following operation

$$\boxed{U = HV}$$

The transformation to accomplish a translation of a vector in space by a distance 'a' in the x direction, 'b' in the 'y' direction and 'c' in the z direction

1

is given by

$$H = \text{Trans}(a, b, c)$$

$$= \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation of a vector about each of the three axes by angle 'θ' can be accomplished by rotation transformation.

About the x-axis $R(x, \theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

About the y-axis $R(y, \theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ -0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

About the z-axis $R(z, \theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

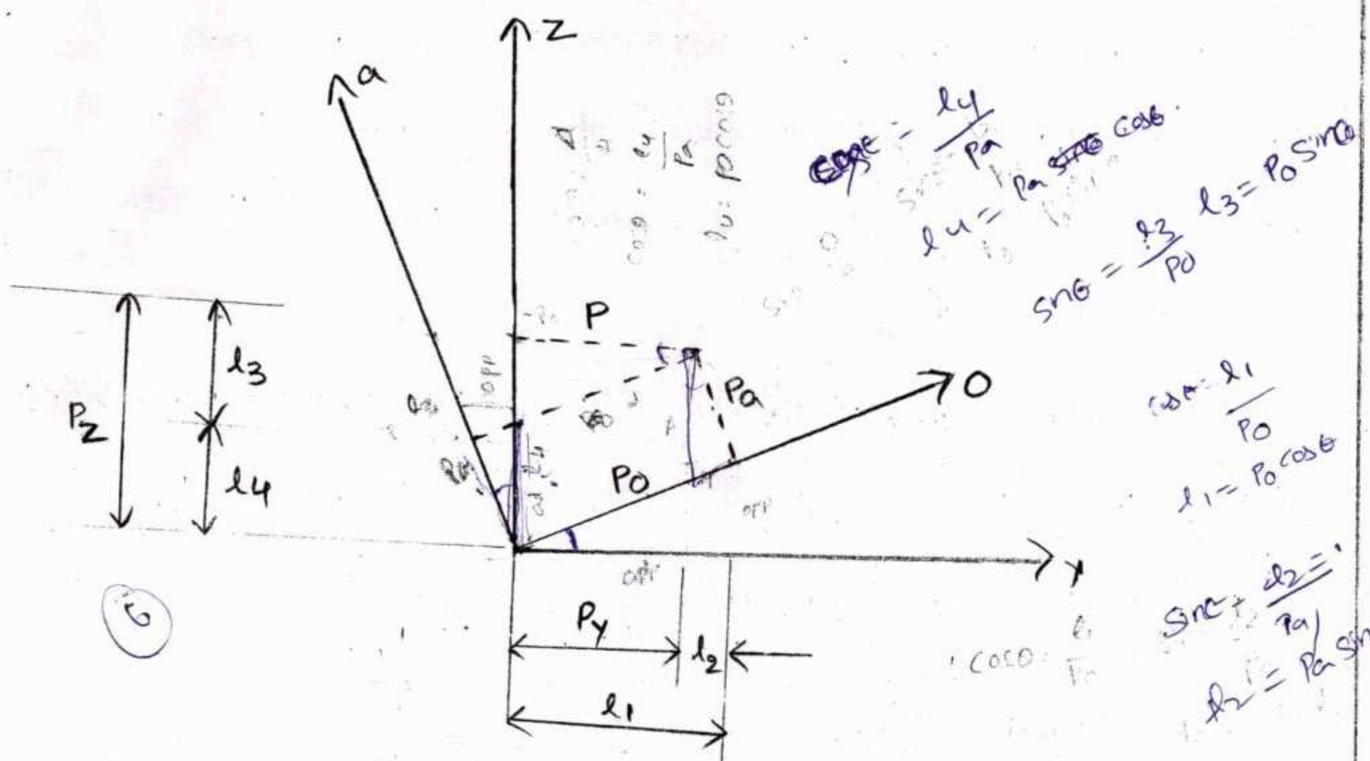


Fig: co-ordinates of a point relative to the reference frame and rotating frame as viewed from X-axis

In the above figure we can notice that the value of P_x does not change as the frame rotates about the X-axis, but the values of P_y and P_z do changes.

$$P_x = P_n$$

$$P_y = l_1 - l_2$$

$$= P_o\cos\theta - P_a\sin\theta$$

$$P_z = l_3 + l_4$$

$$= P_o\sin\theta + P_a\cos\theta$$

which in the matrix form as,

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} P_n \\ P_o \\ P_a \end{bmatrix}$$

This means that the coordinates of the point (or vector) p in the rotated frame must be premultiplied by the rotation matrix, as shown to get the coordinates in reference frame.

This rotation matrix is only for a pure rotation about the x-axis of the reference frame is denoted as

$$P_{xyz} = Rot(x, \theta) \times P_{noa}$$

||ly $$P_{xyz} = Rot(y, \theta) \times P_{noa}$$

$$P_{xyz} = Rot(z, \theta) \times P_{noa}.$$

$$Rot(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix}$$

$$Rot(y, \theta) = \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix}$$

$$Rot(z, \theta) = \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Representation of transformations :

Representation of a pure translation :

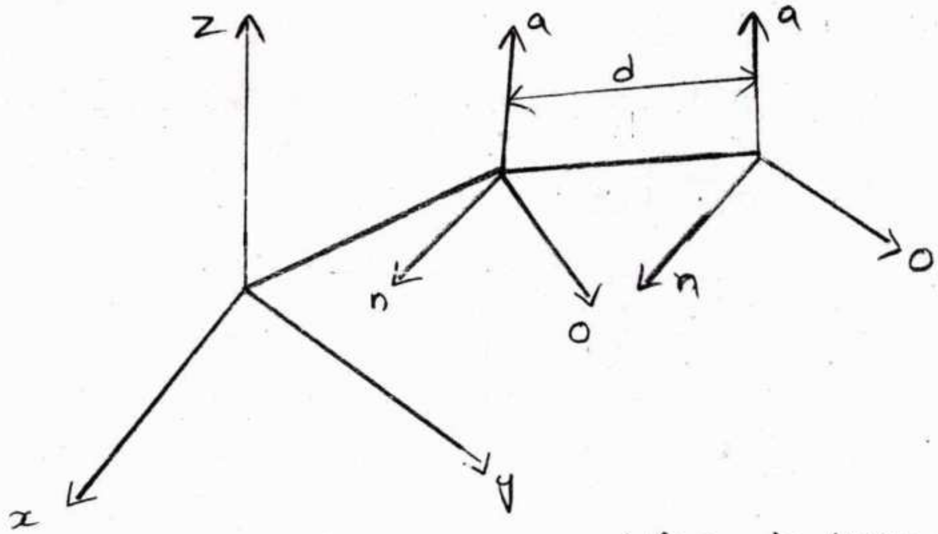


fig : pure Translation in space.

If a frame moves in space without any change in its orientation, the transformation is a pure translation.

$$T = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where d_x, d_y, d_z are the three components of a pure translation vector $\bar{d}$ relative to the x-, y- and z-axes of the reference frame.

$$F_{old} = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

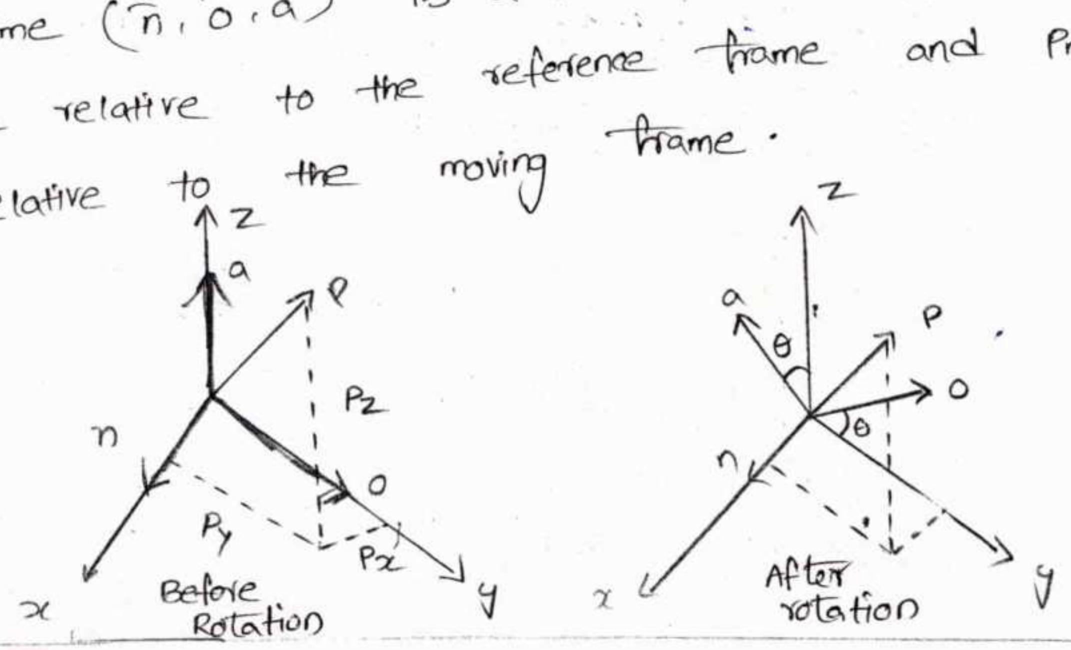
$$F_{new} = Trans(dx, dy, dz) \times F_{old}$$

$$F_{new} = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} n_x & o_x & a_x & P_x + dx \\ n_y & o_y & a_y & P_y + dy \\ n_z & o_z & a_z & P_z + dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$ $n_x * o_x$

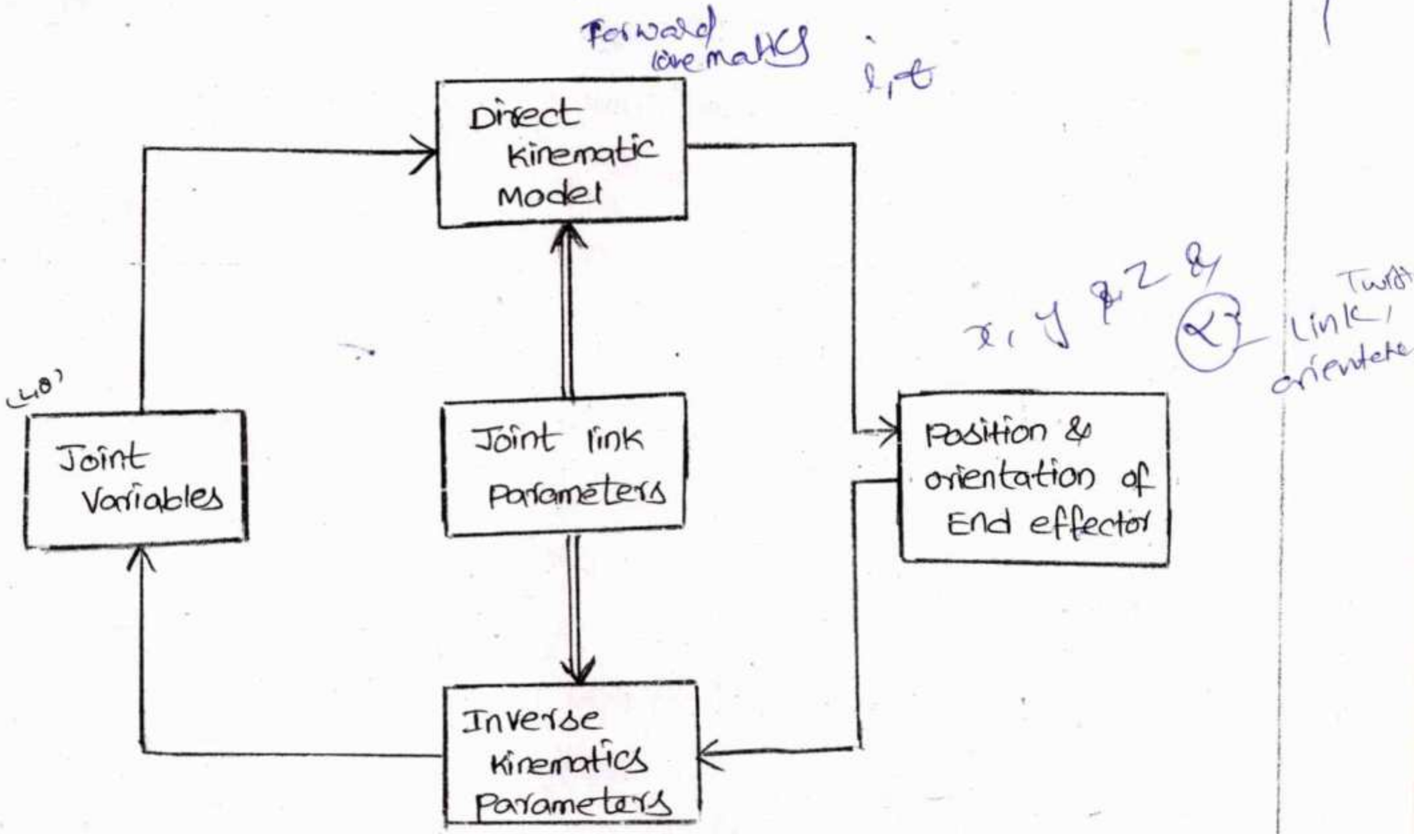
Representation of pure rotation about an axis:

Let's assume that a frame $(\bar{n}, \bar{o}, \bar{a})$ located at the origin of the reference frame $(\bar{x}, \bar{y}, \bar{z})$ will rotate through an angle of 'θ' about the x-axis of the reference frame. And also assume that attached to the rotating frame $(\bar{n}, \bar{o}, \bar{a})$ is a point 'P' with coordinates P_x, P_y & P_z relative to the reference frame and P_n, P_o, P_a relative to the moving frame.



ROBOT KINEMATICS

Forward and Inverse Kinematics of Robot :-



Forward Kinematics :

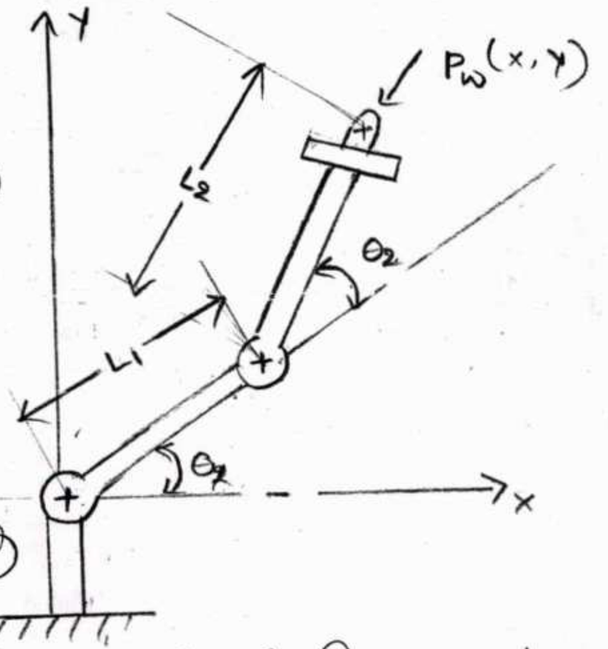
Calculating the position and orientation of the hand of the robot by knowing the link length and joints angle is known as forward kinematics

Inverse Kinematics :

If the position and orientation of the hand of the robot are known, then the lengths & joints of link can be calculated. This is inverse kinematics of a robot.

Forward Transformations of a 2 DOF Arm :-

We can determine the position of the end of the arm in world space by defining a vector for link 1 and another for link 2.



For first link :

$r_1 = [L_1 \cos\theta_1 \quad L_1 \sin\theta_1]$ — (1)

For second link :

$r_2 = [L_2 \cos(\theta_1 + \theta_2),$
$L_2 \sin(\theta_1 + \theta_2)]$ — (2)

The above vectors addition of (1) & (2) yields the coordinates x and y of the end of the arm pivot P_w in world space.

$$x = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \quad — (3)$$

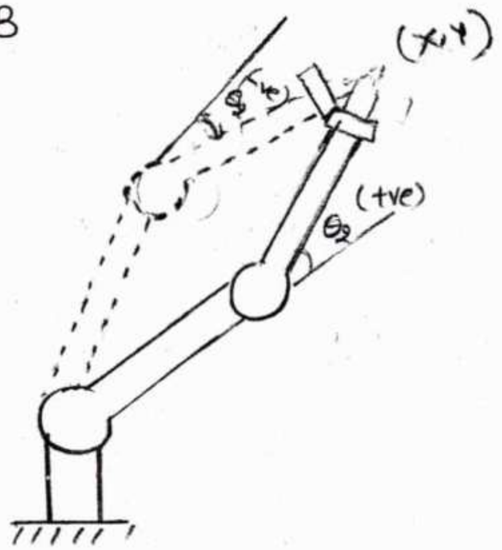
$$y = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \quad — (4)$$

Reverse Transformation of a 2DOF :

We need to derive the joint angles for the give end arm position in world space.

$\cos(A+B) = \cos A \cos B - \sin A \sin B$

$\sin(A+B) = \sin A \cos B + \cos A \sin B$



$x = L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2)$

$x^2 = \left[L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2)\right]^2$

$x^2 = L_1^2 \cos^2\theta_1 + L_2^2 \cos^2(\theta_1+\theta_2)$

$\quad + 2 L_1 L_2 \cos\theta_1 \cdot \cos(\theta_1+\theta_2)$

$y^2 = L_1^2 \sin^2\theta_1 + L_2^2 \sin^2(\theta_1+\theta_2) + 2 L_1 L_2 \sin\theta_1 \cdot \sin(\theta_1+\theta_2)$

$x^2 + y^2 = L_1^2(\cos^2\theta_1 + \sin^2\theta_1) + L_2^2(\cos^2(\theta_1+\theta_2))$

$\quad + \sin^2(\theta_1+\theta_2) + 2 L_1 L_2 \cos\theta_1 \cos(\theta_1+\theta_2) +$

$\quad 2 L_1 L_2 \sin\theta_1 \sin(\theta_1+\theta_2)$.

$= L_1^2 + L_2^2 + 2 L_1 L_2 (\cos\theta_1 \cos(\theta_1+\theta_2)) +$

$\quad \sin\theta_1 \cdot \sin(\theta_1+\theta_2))$

$x^2 + y^2 = L_1^2 + L_2^2 + 2 L_1 L_2 (\cos(\theta_1 - (\theta_1+\theta_2)))$

$x^2 + y^2 = L_1^2 + L_2^2 + 2 L_1 L_2 \cos\theta_2$

$$\boxed{\cos\theta_2 = \frac{x^2 + y^2 - L_1^2 - L_2^2}{2 L_1 L_2}}$$

//

Defining α and β angles :

β = Total angle

α = Twisting angle → (orientation of tool)

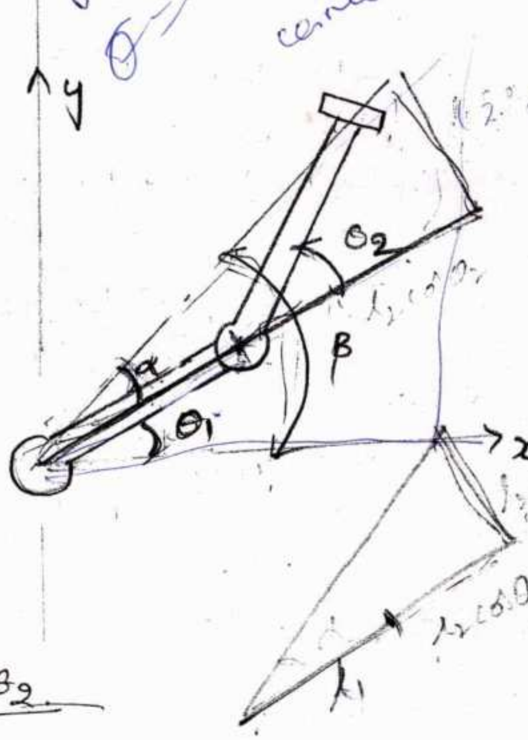
θ = Joint angle → connecting

$$\tan \beta = y/x$$

$$\beta = \alpha + \theta_1$$

$$\beta - \alpha = \theta_1$$

$$\tan(\beta - \alpha) = \tan\theta_1$$



$$\tan \alpha = \frac{L_2 \sin\theta_2}{L_1 + L_2\cos\theta_2}$$

$$\tan(\beta - \alpha) = \frac{\tan\beta - \tan\alpha}{1 + \tan\alpha \tan\beta}$$

$$= \frac{\frac{y}{x} - \frac{L_2 \sin\theta_2}{L_1 + L_2 \cos\theta_2}}{1 + \left(\frac{L_2 \sin\theta_2}{L_1 + L_2\cos\theta_2}\right)(y/x)}$$

$$\tan\theta_1 = \frac{y(L_1 + L_2 \cos\theta_2) - L_2 \sin\theta_2 \cdot x}{x(L_1 + L_2 \cos\theta_2) + y(L_2 \sin\theta_2)}$$

$$\cos^2\theta_2 = 1 - \sin^2\theta_2$$

$$\sin^2\theta_2 = 1 - \cos^2\theta_2$$

$$\sin\theta_2 = \pm\sqrt{1 - \cos^2\theta_2}$$

$$\tan\theta_2 = \frac{\sin\theta_2}{\cos\theta_2} = \frac{\pm\sqrt{1 - \cos^2\theta_2}}{\cos\theta}$$

Let $\cos\theta_2 = D$

$$\tan\theta_1 = \pm \frac{\sqrt{1-D^2}}{D}$$

Forward and Inverse Kinematics for Position :

The position of the origin of a frame attached to a rigid body has 3 degrees of freedom. The following are the possibilities for the position kinematics.

(a) Cartesian (LLL) coordinates
(b) cylindrical (2L, R) coordinates
(C) spherical (2R, L) coordinates
(d) Jointed Arm (RRR) coordinates.

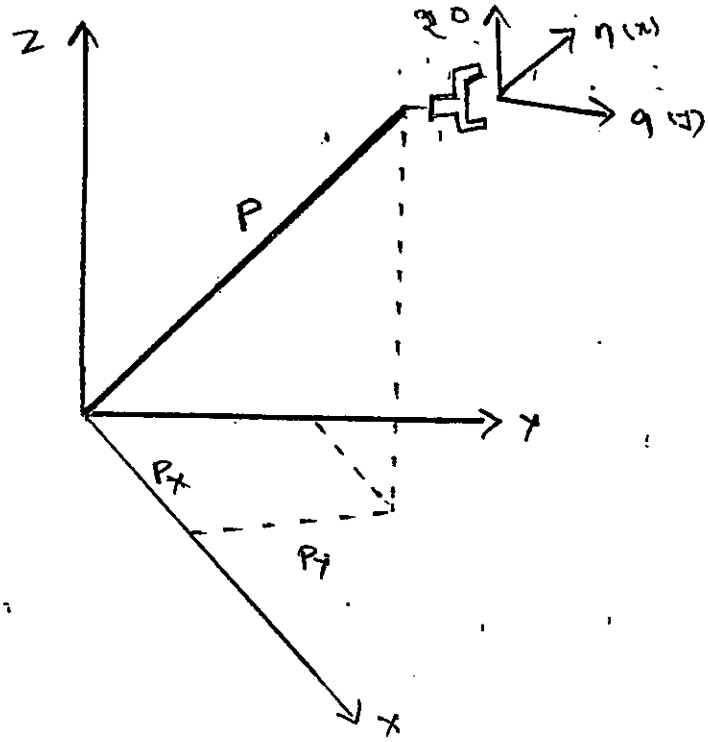
(a) cartesian coordinates

In this type, there will be 3 linear movements along 3 major x-, y-, z- axes. Here all actuators are linear (such as a hydraulic ram or a linear power screw) and the positioning of the hand of the robot is accomplished by moving 3 linear joints along the 3 axes.

$$R.T_p = T_{cart} = \begin{bmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

RTp – Transformation b/w reference frame & origin of hand 'p'

T_{cart} – cartesian Transformation matrix.



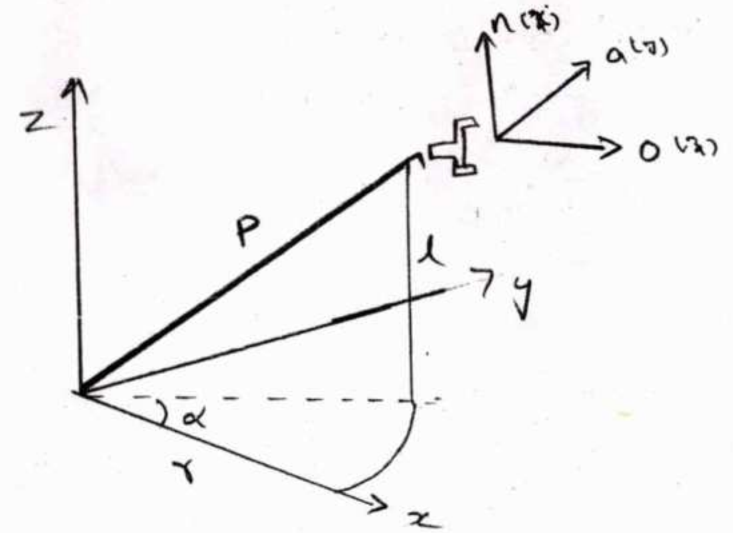
(b) <u>cylindrical coordinates</u>: The cylindrical coordinate system includes two linear translations and one rotation

$$RT_P = Trans(0,0,l)\ Rot(z,\alpha)\ Trans(r,0,0).$$

$$RT_P = T_{cyl}(r,\alpha,l)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\alpha & -S\alpha & 0 & 0 \\ S\alpha & C\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & r \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$RT_P = T_{cyl} = \begin{bmatrix} C\alpha & -S\alpha & 0 & rC\alpha \\ S\alpha & C\alpha & 0 & rS\alpha \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

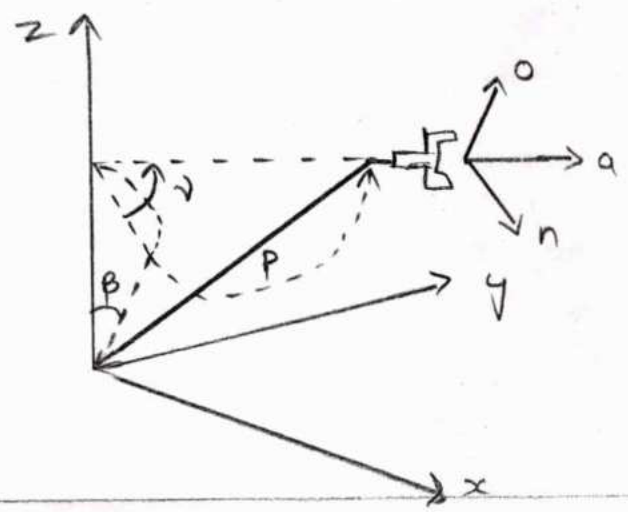


(c) Spherical coordinates :

$$ {}^{R}T_{P} = T_{sph}(r, \beta, \gamma) = Rot(z, \gamma)\, Rot(y, \beta)\, Trans(0, 0, r) $$

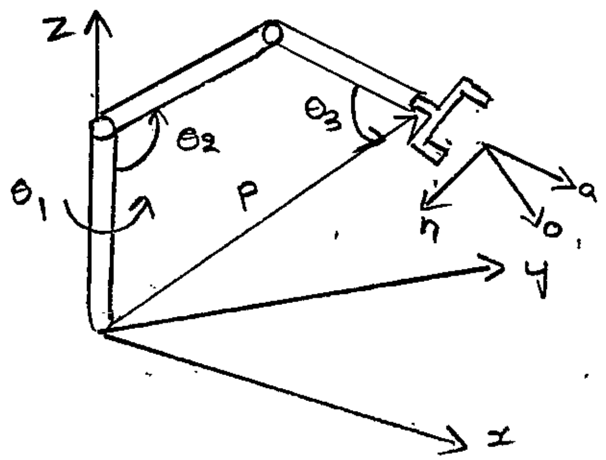
$$ {}^{R}T_{P} = \begin{bmatrix} C\gamma & -S\gamma & 0 & 0 \\ S\gamma & C\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\beta & 0 & S\beta & 0 \\ 0 & 1 & 0 & 0 \\ -S\beta & 0 & C\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{R}T_{P} = T_{sph} = \begin{bmatrix} C\beta \cdot C\gamma & -S\gamma & S\beta \cdot C\gamma & r S\beta \cdot C\gamma \\ C\beta \cdot S\gamma & C\gamma & S\beta \cdot S\gamma & r S\beta \cdot S\gamma \\ -S\beta & 0 & C\beta & r C\beta \\ 0 & 0 & 0 & 1 \end{bmatrix} $$



(d) Articulated coordinates :

Articulated robots consists of three rotations. The matrix representation for this is discussed in D-H representation.



Forward and Inverse Kinematic Equations for orientation:

The moving frame attached to the hand of the robot has already moved to a desired position but is still parallel to the reference frame or that is an orientation other than what is desired. The next step will be to rotate the frame appropriately in order to achieve a desiered orientation without changing its position. Thee common configurations are:

(a) Roll, pitch, Yaw (RPY) angles

(b) Euler angles

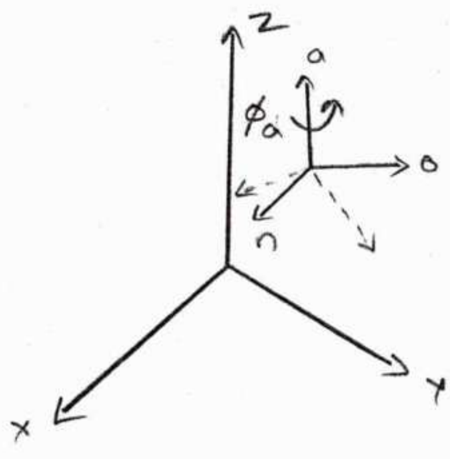
(c) Articulated joints

(a) Roll, pitch, Yaw (RPY) angles:

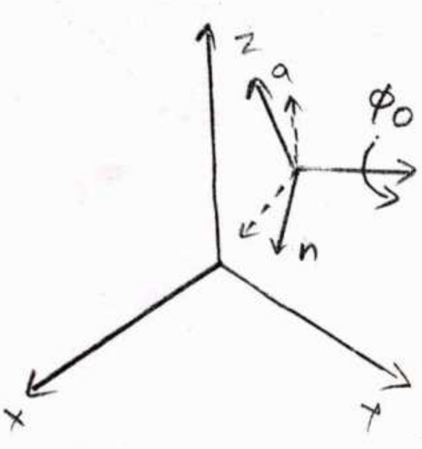
This is a sequence of three rotations about current $\bar{a}, \bar{o}, \bar{n}$ axes respectively, which will orientate the hand of the robot to a desired orientation. The assumption here is that the current frame is parallel to the reference frame and thus its orientation is the same as the reference frame before the RPY movement.

R - roll — (Rotation or swivel movement in a plane ⊥ to end of arm
P - pitch — (" or bending movement in vertical to the arm
Y - Yaw (" or Twisting movement horizontal to the arm)

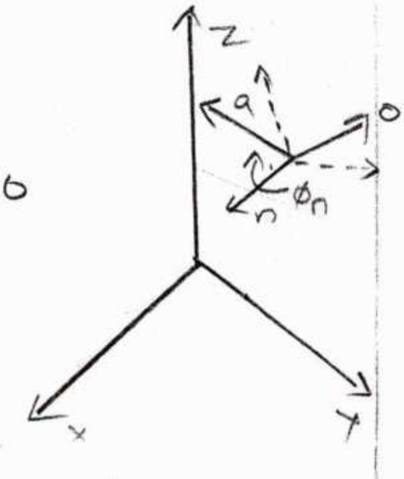
If the current moving frame is not parallel to the reference frame, then the final orientation of the hand will be a combination of the previous orientation.



Rot^n about Z-axis



Rot^n about Y-axis



Rot^n about X-axis

Rotation of ϕ_a about the $\bar{a}$ axis.

[z axis of moving frame] — roll

Rotation of ϕ_o about the $\bar{o}$ axis

[y-axis of moving frame] — pitch

Rotation of ϕ_n about the $\bar{n}$ axis

[x-axis of moving frame] — Yaw.

The matrix representation of the RPY orientation will be

$$RPY(\phi_a, \phi_o, \phi_n) = Rot(a, \phi_a)\, Rot(o, \phi_o)\, Rot(n, \phi_n)$$

$$= \begin{bmatrix} C\phi_a & -S\phi_a & 0 & 0 \\ S\phi_a & C\phi_a & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi_o & 0 & S\phi_o & 0 \\ 0 & 1 & 0 & 0 \\ -S\phi_o & 0 & C\phi_o & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\phi_n & -S\phi_n & 0 \\ 0 & S\phi_n & C\phi_n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C\phi_a C\phi_o & C\phi_a S\phi_o S\phi_n - S\phi_a C\phi_n & C\phi_a S\phi_o C\phi_n + S\phi_a S\phi_n & 0 \\ S\phi_a C\phi_a & S\phi_a S\phi_o S\phi_n + C\phi_a C\phi_n & S\phi_a S\phi_o C\phi_n - C\phi_a S\phi_n & 0 \\ -S\phi_o & C\phi_o S\phi_n & C\phi_o C\phi_n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

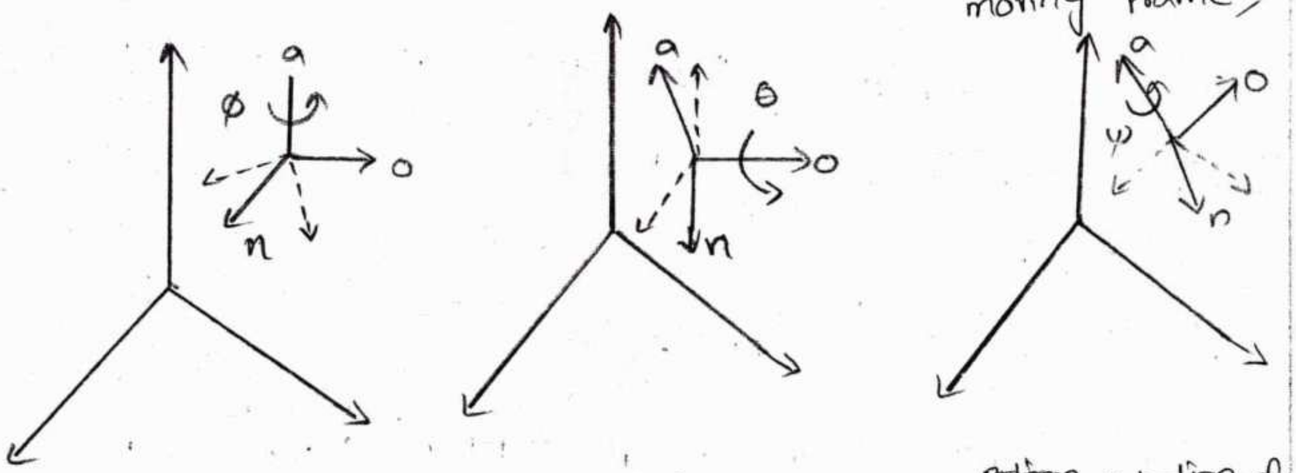
(b) Euler Angles :-

Euler angles are very similar to RPY, except that the last rotation is also about the current a-axis

Rotation of ϕ about the $\bar{a}$ axis – (z axis of the moving frame)

Rotation of θ about the $\bar{o}$-axis – (y axis of the moving frame)

Rotation of ψ about the $\bar{a}$-axis – (z axis of the moving frame)



Rotation of ϕ about the $\bar{a}$-axis   Rotation of θ about $\bar{o}$-axis   Rotation of ψ about $\bar{a}$-axis

The matrix representation of the euler angle orientation $\text{Euler}(\phi, \theta, \psi) = \text{Rot}(a, \phi)\,\text{Rot}(o, \theta)\,\text{Rot}(a, \psi)$

$$= \begin{bmatrix} C\phi & -S\phi & 0 & 0 \\ S\phi & C\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta & 0 \\ 0 & 1 & 0 & 0 \\ -S\theta & 0 & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 & 0 \\ S\psi & C\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C\phi C\theta C\psi - S\phi S\psi & -C\phi C\theta S\psi - S\phi C\psi & C\phi S\theta & 0 \\ S\phi C\theta C\psi + C\phi S\psi & -S\phi C\theta S\psi + C\phi C\psi & S\phi S\theta & 0 \\ -S\theta C\psi & S\theta S\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Forward and Inverse Kinematic equations for Position and orientation:

The matrix representing the final location and orientation of the robot is a combination of the preceding equations. The robot is made of a cartesian and RPY set of Joints

$$RT_H = T_{cart}(P_x, P_y, P_z) \times RPY(\phi_a, \phi_o, \phi_n)$$

$$= \begin{bmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi_a C\phi_o & C\phi_a S\phi_o S\phi_n - S\phi_a C\phi_n & C\phi_a S\phi_o C\phi_n + S\phi_a S\phi_n & 0 \\ S\phi_a C\phi_o & S\phi_a S\phi_o S\phi_n + C\phi_a C\phi_n & S\phi_a S\phi_o C\phi_n - C\phi_o S\phi_n & 0 \\ -S\phi_o & C\phi_o S\phi_n & C\phi_o C\phi_n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} c\phi_a c\phi_o & c\phi_a s\phi_o s\phi_n - s\phi_a c\phi_n & c\phi_a s\phi_o c\phi_n + s\phi_a s\phi_n & P_x \\ s\phi_a c\phi_o & s\phi_a s\phi_o s\phi_n + c\phi_a c\phi_n & s\phi_a s\phi_o c\phi_n - c\phi_a s\phi_n & P_y \\ -s\phi_o & c\phi_o s\phi_n & c\phi_o c\phi_n & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

if the robot is designed based on spherical (position) coordinates and Euler angles (orientation),

$$^R T_H = T_{sph}(r, \beta, \gamma) \times Euler(\phi, \theta, \psi)$$

$$= Rot(z, \gamma)\ Rot(y, \beta)\ Trans(0, 0, r)\ Euler(\phi, \theta, \psi)$$

$$= \begin{bmatrix} c\beta \cdot c\gamma & -s\gamma & s\beta \cdot c\gamma & r c\beta \cdot c\gamma \\ c\beta \cdot s\gamma & c\gamma & s\beta \cdot s\gamma & r s\beta \cdot s\gamma \\ -s\beta & 0 & c\beta & r c\beta \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\phi c\theta c\psi - s\phi s\psi & -c\phi c\theta s\psi - s\phi c\psi & c\phi s\theta & 0 \\ s\phi c\theta c\psi + c\phi s\psi & -s\phi c\theta s\psi + c\phi c\psi & s\phi s\theta & 0 \\ -s\theta c\psi & s\theta c\psi & c\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

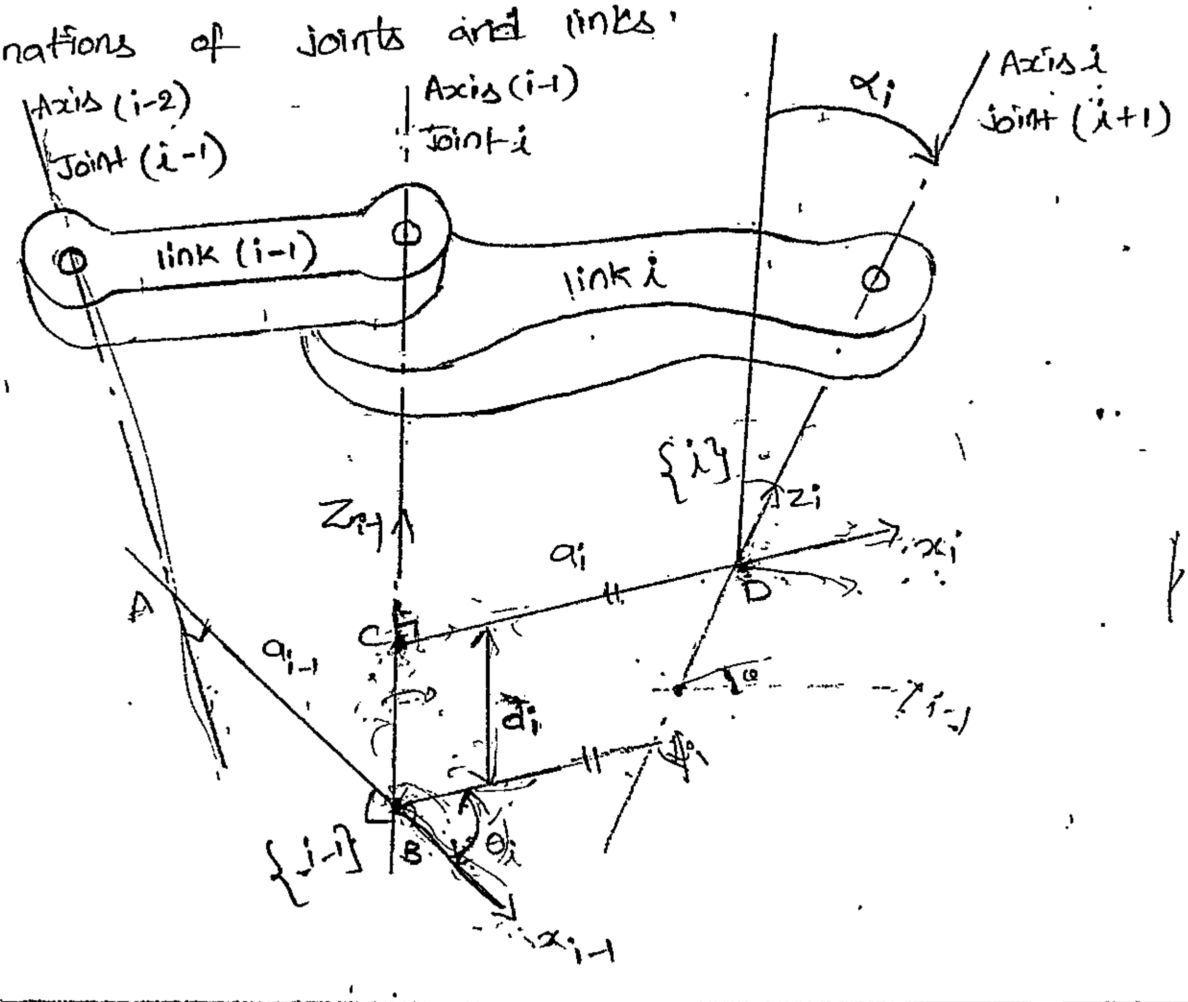
The forward and inverse kinematics solutions for these cases are not developed here since many different combinations are possible.

Instead in complicated designs, the Denavit-Hartenberg representation is recommended.

DENAVIT-HARTENBERG Representation of forward Kinematic Equations of Robots:

Denavit-Hartenberg (D-H) model of representation is a very simple way of modeling robots links and joints that can be used for any robot configuration regardless of its sequence or complexity.

It can also be used to represent transformation in any coordinates such as cartesian, cylindrical, spherical, Euler & RPY. It can be used for representation of all revolute articulated robots, SCARA robots or any possible combinations of joints and links.



The above figure shows a pair of adjacent links, links $(i-1)$ and link i, their associated joints, joints $(i-1)$, i and $(i+1)$ and axes $(i-2)$, $(i-1)$ and i respectively.

with respect to frame $\{i-1\}$ and frame $\{i\}$ the four DH parameters arise. They are:

(1) Two link parameters (a_i, α_i)

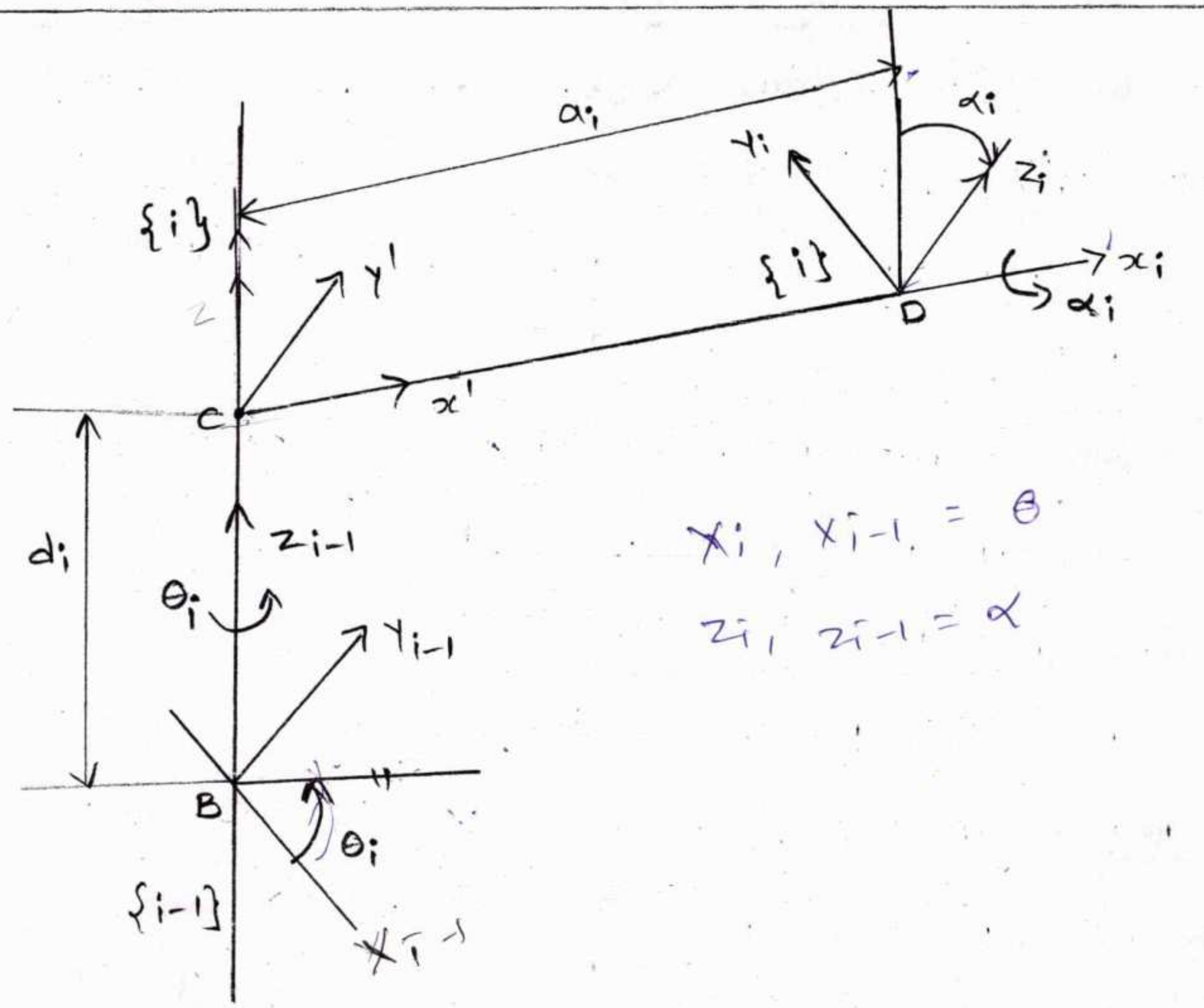
(2) Two joint parameters (d_i, θ_i)

(a) Link length (a_i): Distance measured along x_i-axis from the point of intersection of x_i-axis with z_{i-1} axis to the origin of frame $\{i\}$. (dist CD)

(b) Link Twist (α_i): Angle between z_{i-1} and z_i axes measured about x_i axis in the right hand sense

(c) Joint Distance (d_i): Distance measured along z_{i-1} axis from the origin of frame $\{i-1\}$ to the intersection of x_i-axis with z_{i-1} axis BC — offset dist

(d) Joint Angle (θ_i): Angle between x_{i-1} and x_i axes measured about the z_{i-1} axis in the right hand sense.



The transformation of frame {i-1} to frame {i} consist of four basic transformations as

(a) A rotation about z_{i-1} axis by an angle θ_i

(b) Translation along z_{i-1} axis by distance d_i

(c) Translation by distance a_i along x_i axis

(d) Rotation by an angle α_i about x_i axis.

$Rot(z,\theta)_i \; T(z, d_i) \; \sout{Rot}(x, a_i) \; Rot(x, \alpha_i)$

$$ {}^{i-1}T_i = T_z(\theta_i)\, T_z(d_i)\, T_x(a_i)\, \sout{T_x}(\alpha_i) $$

$$ = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

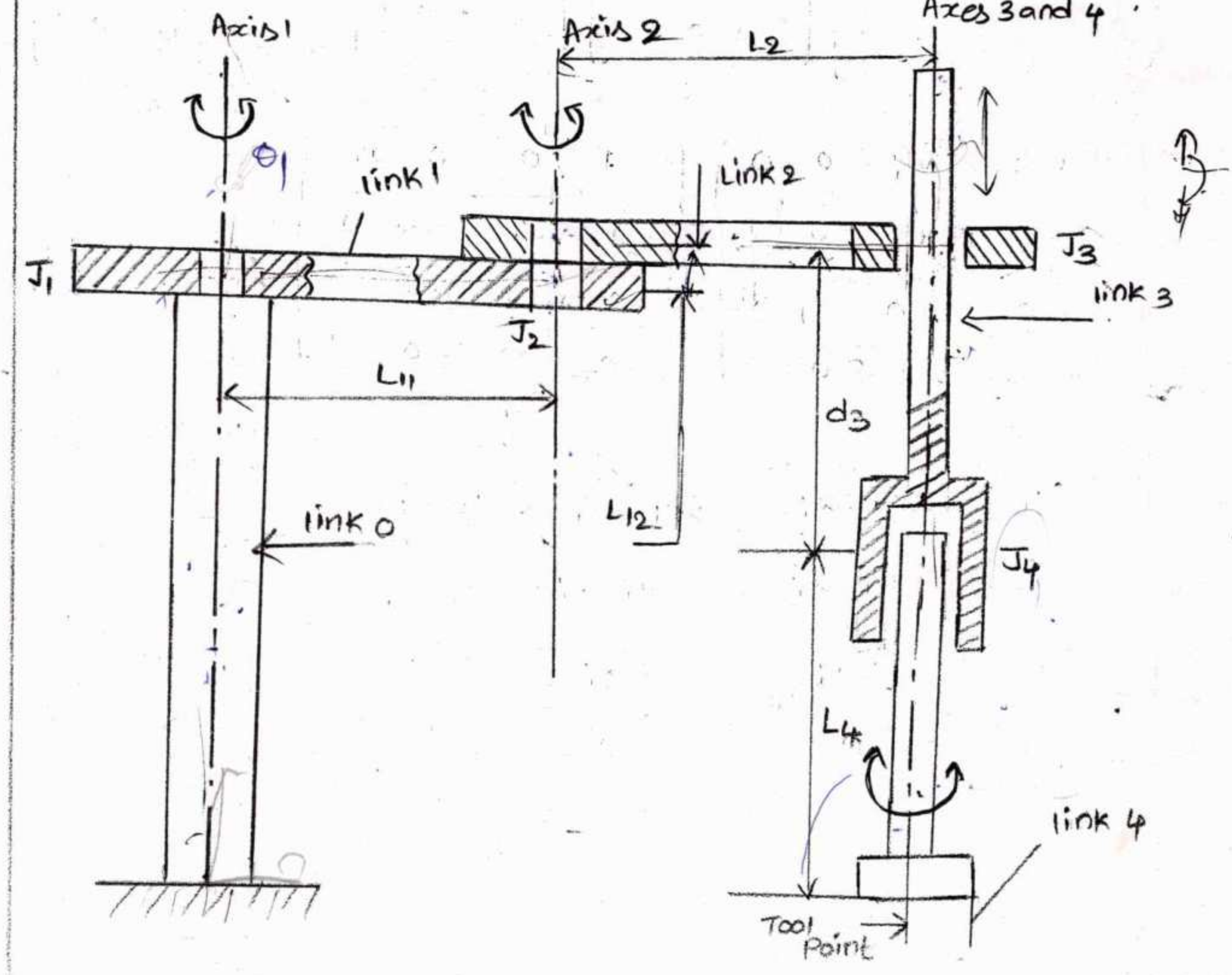
$$ = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

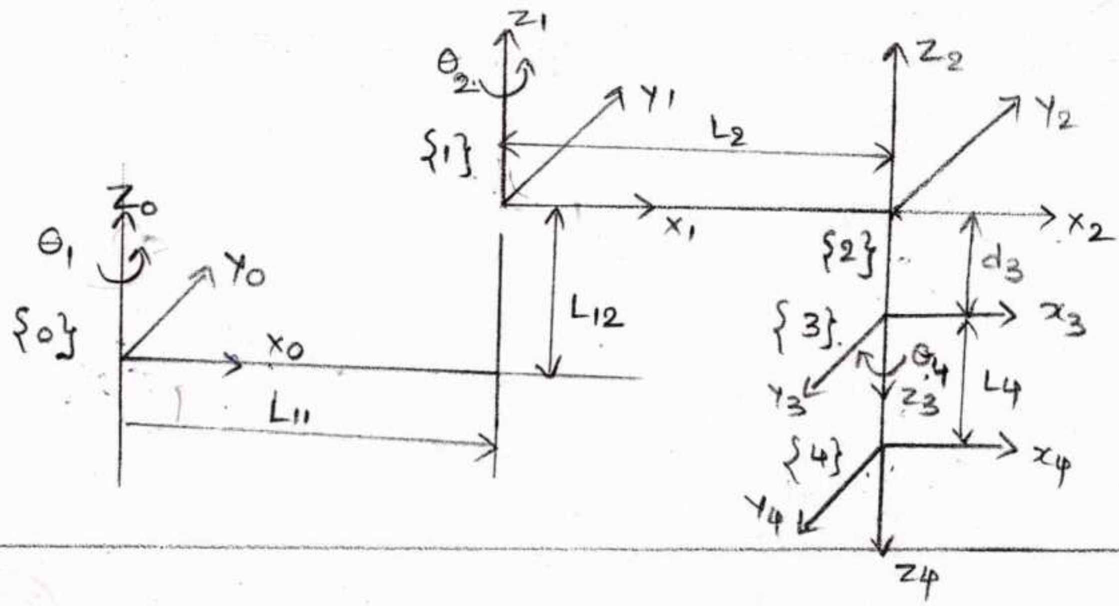
$$ = \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

selective compliance Assembly Robotic arm

SCARA Robot (or) SCARA Manipulator Kinematics :- (4 DOF) (RRTR)

The SCARA manipulator is a widely used industrial robot for assembly operation.





Link i	a_i	α_i	d_i	θ_i
1	L_{11}	0	L_{12}	θ_1
2	L_2	0	0	θ_2
3	0	180	d_3	0
4	0	0	L_4	θ_4

$$ {}^{0}T_{1} = \begin{bmatrix} C_1 & -S_1 & 0 & L_{11}C_1 \\ S_1 & C_1 & 0 & L_{11}S_1 \\ 0 & 0 & 1 & L_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

X_i & X_{i-1} = θ_i

Z_i, Z_{i-1} = α_i

$$ {}^{1}T_{2} = \begin{bmatrix} C_2 & -S_2 & 0 & L_2C_2 \\ S_2 & C_2 & 0 & L_2S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}T_{3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

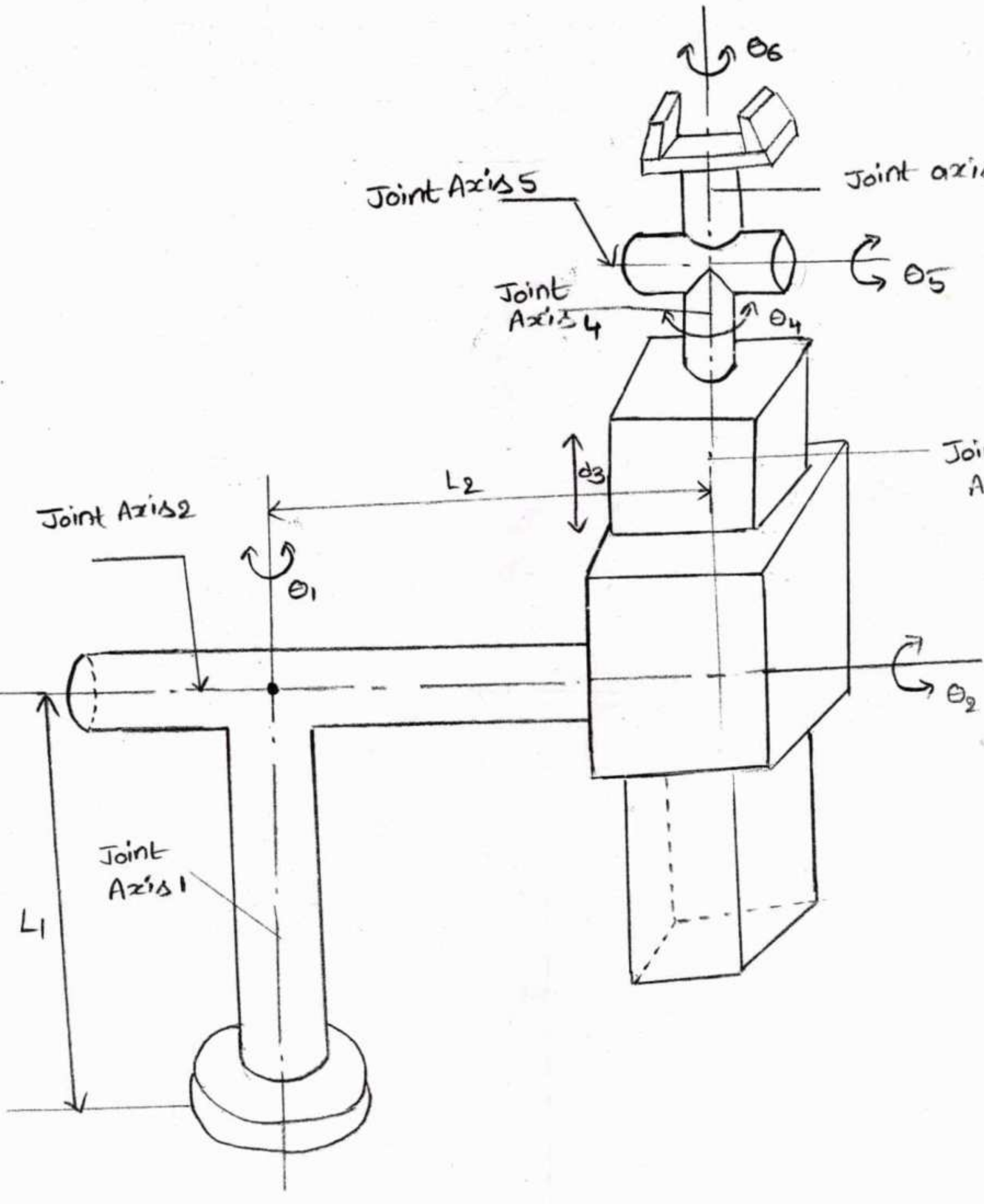
$$ {}^{3}T_{4} = \begin{bmatrix} C_4 & -S_4 & 0 & 0 \\ S_4 & C_4 & 0 & 0 \\ 0 & 0 & 1 & L_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

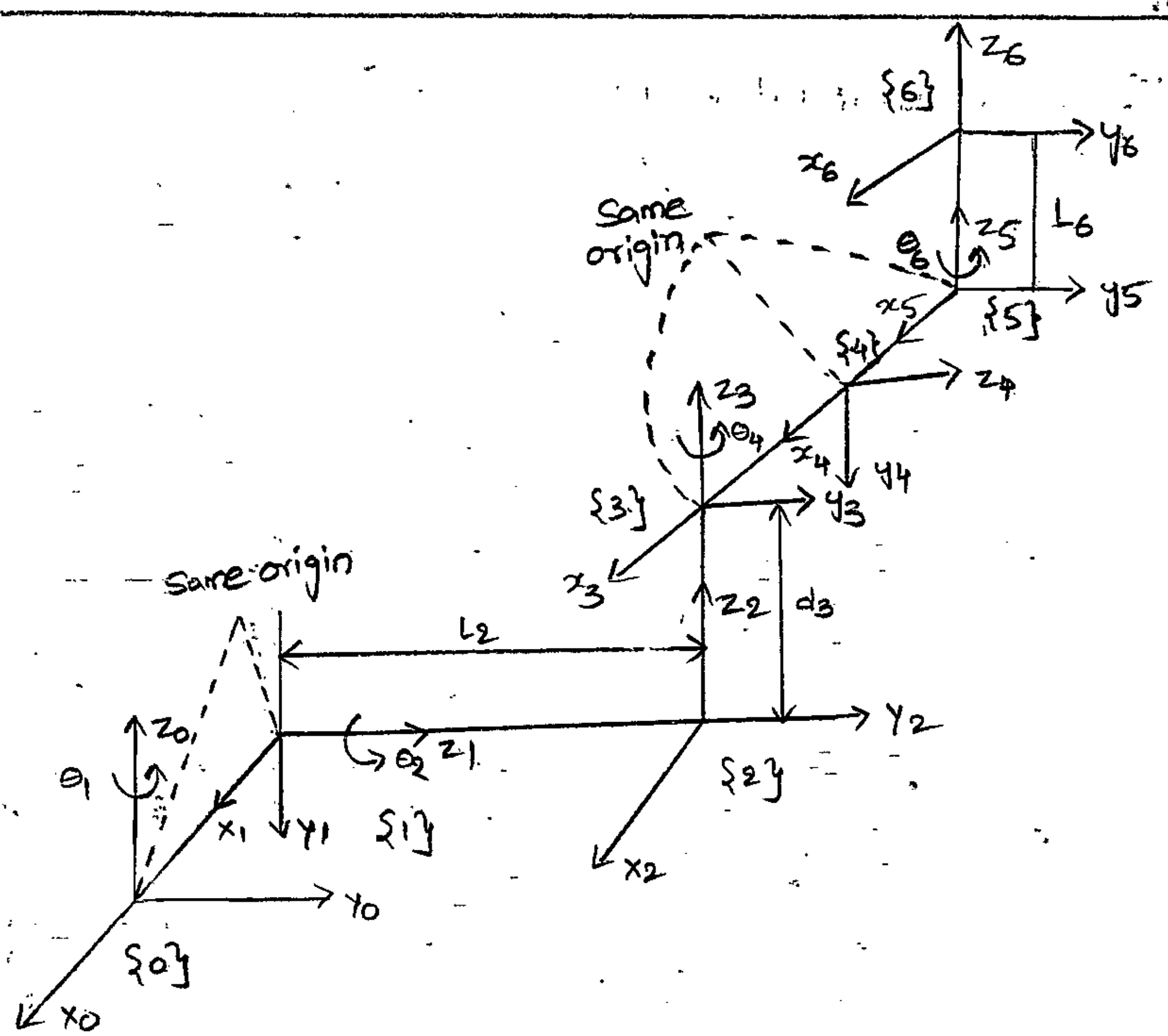
$$^0T_4 = {}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot {}^3T_4$$

$$= \begin{bmatrix} c_1 & -s_1 & 0 & L_{11}c_1 \\ s_1 & c_1 & 0 & L_{11}s_1 \\ 0 & 0 & 1 & L_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_2 & -s_2 & 0 & L_2c_2 \\ s_2 & c_2 & 0 & L_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & L_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^0T_4 = \begin{bmatrix} c_{124} & s_{124} & 0 & L_2c_{12} + L_{11}c_1 \\ s_{124} & -c_{124} & 0 & L_2s_{12} + L_{11}s_1 \\ 0 & 0 & -1 & L_{12} + d_3 - L_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Stanford Manipulator Kinematics :





Link i	a_i	α_i	d_i	θ_i
1	0	-90	0	θ_1
2	0	90	L_2	θ_2
3	0	0	d_3	0
4	0	-90	0	θ_4
5	0	90	0	θ_5
6	0	0	L_6	θ_6

$$ {}^{0}T_6 = {}^{0}T_1 \cdot {}^{1}T_2 \cdot {}^{2}T_3 \cdot {}^{3}T_4 \cdot {}^{4}T_5 \, {}^{5}T_6 $$

$$ {}^{0}T_1(\theta_1) = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{1}T_2(\theta_2) = \begin{bmatrix} C_2 & 0 & S_2 & 0 \\ S_2 & 0 & -C_2 & 0 \\ 0 & 1 & 0 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}T_3(\theta_3) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{3}T_4(\theta_4) = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{4}T_5(\theta_5) = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad {}^{5}T_6(\theta_6) = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & L_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}T_1 \cdot {}^{1}T_2 = \begin{bmatrix} C_1C_2 & -S_1 & C_1S_2 & -S_1L_2 \\ S_1C_2 & C_1 & S_1S_2 & C_1L_2 \\ -S_2 & 0 & C_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}T_3 \cdot {}^{3}T_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$
{}^{4}T_5 \cdot {}^{5}T_6 = \begin{bmatrix} c_5 c_6 & -c_5 s_6 & s_5 & s_5 L_6 \\ s_5 L_6 & -s_5 s_6 & -c_5 & -c_5 L_6 \\ s_6 & L_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{0}T_6 = \begin{bmatrix}
\begin{array}{l} c_1 c_2 c_4 c_5 c_6 \\ - s_1 s_4 c_5 c_6 \\ - c_1 s_2 s_5 c_6 \\ - c_1 c_2 s_4 s_6 \\ - s_1 c_4 s_6 \end{array} &
\begin{array}{l} c_1 c_2 c_4 c_5 s_6 \\ + s_1 s_4 c_5 s_6 \\ + c_1 s_2 s_5 s_6 \\ - c_1 c_2 s_4 c_6 \\ - s_1 c_4 c_6 \end{array} &
\begin{array}{l} c_1 c_2 c_4 c_5 \\ - s_1 s_4 s_5 \\ + c_1 s_2 c_5 \end{array} &
\begin{array}{l} c_1 c_2 c_4 s_5 L_6 \\ - s_1 s_4 s_5 L_6 \\ + c_1 s_2 c_5 L_6 \\ + c_1 s_2 d_3 - \\ s_1 L_2 \end{array} \\
\begin{array}{l} - s_1 c_2 c_4 c_5 c_6 \\ + c_1 s_4 c_5 c_6 \\ - s_1 s_2 s_5 s_6 \\ - s_1 c_2 s_4 s_6 \\ + c_1 c_4 s_6 \end{array} &
\begin{array}{l} - s_1 c_2 c_4 c_5 s_6 \\ - c_1 s_4 c_5 s_6 \\ + s_1 s_2 s_5 s_6 \\ - s_1 c_2 s_4 s_6 \\ + c_1 c_4 c_6 \end{array} &
\begin{array}{l} s_1 c_2 c_4 s_5 \\ + c_1 s_4 s_5 \\ + s_1 s_2 c_5 \end{array} &
\begin{array}{l} s_1 c_2 c_4 c_5 L_6 \\ + c_1 s_4 s_5 L_6 \\ + s_1 s_2 c_5 L_6 \\ + s_1 s_2 d_3 \\ + c_1 L_2 \end{array} \\
\begin{array}{l} s_2 c_4 c_5 c_6 \\ - c_2 s_5 c_6 \\ + s_2 s_4 s_6 \end{array} &
\begin{array}{l} s_2 c_4 c_5 s_6 \\ + c_2 s_5 s_6 \\ + s_2 s_4 c_6 \end{array} &
\begin{array}{l} - s_2 c_4 s_5 \\ + c_2 c_5 \end{array} &
\begin{array}{l} - s_2 c_4 s_5 L_6 \\ + c_2 c_5 L_6 \\ + c_2 d_3 \end{array} \\
0 & 0 & 0 & 1
\end{bmatrix}
$$

Let $T = \begin{bmatrix} n_x & o_x & a_x & d_x \\ n_y & o_y & a_y & d_y \\ n_z & o_z & a_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^{0}T_{6}$

using the direct kinematic model for the home position,

$\theta_1 = \theta_2 = \theta_4 = \theta_5 = \theta_6 = 0$. and $d_3 = L_3$.

and assume L_3 is minimum size of prismatic link, the end effector position and orientation can be computed as

$$T = \left[\begin{array}{c|ccc} n_x & o_x & a_x & d_x \\ n_y & o_y & a_y & d_y \\ n_z & o_z & a_z & d_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L_2 \\ 0 & 0 & 1 & L_3+L_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

UNIT - IV

Differential motions & velocities

Jacobian of a Robot system:

The Jacobian is a representation of the geometry of the elements of a mechanism in time. It allows the conversion of differential motions or velocities of individual joints to differential motions or velocities of points of interest.

$$x = l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2)$$

$$y = l_1 \sin\theta_1 + l_2 \sin(\theta_1 + \theta_2)$$

$$dx_B = -l_1 \sin\theta_1 \, d\theta_1 - l_2 \sin(\theta_1+\theta_2)(d\theta_1 + d\theta_2)$$

$$dy_B = l_1 \cos\theta_1 \, d\theta_1 + l_2 \cos(\theta_1+\theta_2)(d\theta_1+d\theta_2)$$

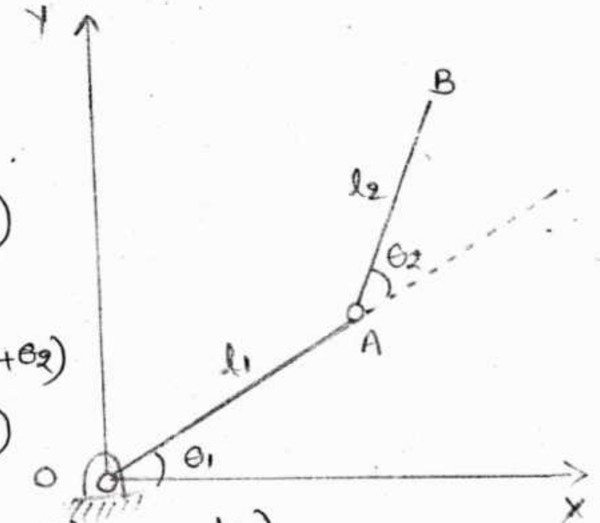


Fig: A 2-DOF planar mechanism

$$\begin{bmatrix} dx_B \\ dy_B \end{bmatrix} = \begin{bmatrix} -l_1 \sin\theta_1 - l_2 \sin(\theta_1+\theta_2) & -l_2 \sin(\theta_1+\theta_2) \\ l_1 \cos\theta_1 + l_2 \cos(\theta_1+\theta_2) & l_2 \cos(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} d\theta_1 \\ d\theta_2 \end{bmatrix}$$

{Differential motion of point B} = {Jacobian matrix} {Differential motion of Joints}

Suppose if it is for a set of equations Y_i in terms of a set of variables x_j

$Y_1 - x_1$

$$Y_i = f_i(x_1, x_2, x_3, \ldots x_j)$$

The differential change in Y_i, for a differential change x_i is

$$\begin{cases} \delta Y_1 = \dfrac{\partial f_1}{\partial x_1}\delta x_1 + \dfrac{\partial f_1}{\partial x_2}\delta x_2 + \cdots + \dfrac{\partial f_1}{\partial x_j}\delta x_j \\ \delta Y_2 = \dfrac{\partial f_2}{\partial x_1}\delta x_1 + \dfrac{\partial f_2}{\partial x_2}\delta x_2 + \cdots + \dfrac{\partial f_2}{\partial x_j}\delta x_j \\ \vdots \\ \delta Y_i = \dfrac{\partial f_i}{\partial x_1}\delta x_1 + \dfrac{\partial f_i}{\partial x_2}\delta x_2 + \cdots + \dfrac{\partial f_i}{\partial x_j}\delta x_j \end{cases}$$

The above equations can be represented in matrix form as

$$\begin{bmatrix} \delta Y_1 \\ \delta Y_2 \\ \vdots \\ \delta Y_i \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_j} \\ \dfrac{\partial f_2}{\partial x_1} & \cdots & & \\ \vdots & & & \\ \dfrac{\partial f_i}{\partial x_1} & & & \dfrac{\partial f_i}{\partial x_j} \end{bmatrix} \begin{bmatrix} \delta x_1 \\ \delta x_2 \\ \vdots \\ \delta x_j \end{bmatrix}$$

(or)

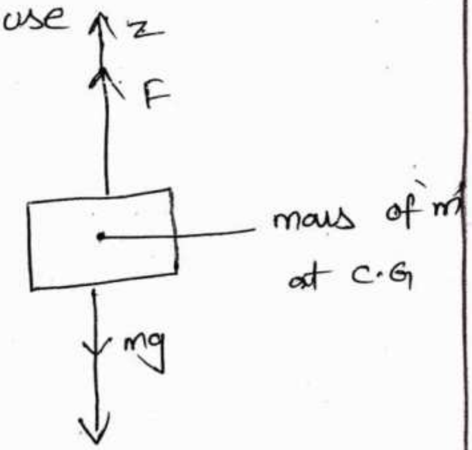
Continue on next page (2)

UNIT - V

DYNAMIC ANALYSIS & FORCES

Lagrangian mechanism :

Lagrangian mechanics is based on the differentiation of the energy terms with respect to the system's variables and time. For simple cases, it may take longer to use this technique than Newtonian mechanics. However as the complexity increases, lagrangian method becomes relatively simpler to use



According to Newtons eqn,

$$F = m\ddot{y} + mg$$

$$m\ddot{y} = F - mg \quad \text{—— (1)}$$

L.H.S :

$$m\ddot{y} = \frac{d}{dt}(m\dot{y}).$$

$$= \frac{d}{dt} \cdot \frac{d}{d\dot{y}}\left(\frac{1}{2} m \dot{y}^2\right)$$

$$= \frac{d}{dt}\left(\frac{dK}{d\dot{y}}\right)$$

$K = \frac{1}{2} m v^2$

$v = \dot{y}$

$K = \frac{1}{2} m \dot{y}^2$

R.H.S :

$$mg = \frac{d}{dy}(mgy)$$

$$= \frac{dP}{dy}$$

The Lagrangian can be defined as

$$\boxed{L = K - P}$$

$$L = \frac{1}{2} m\dot{y}^2 - mgy$$

$$\frac{\partial L}{\partial \dot{y}} = m\dot{y}$$
$$= \frac{\partial K}{\partial \dot{y}} \quad \text{---(2)}$$

$$\frac{\partial L}{\partial y} = -mg$$
$$= -\frac{\partial P}{\partial y} \quad \text{---(3)}$$

From eqn (1)

$$m\ddot{y} = F - mg$$

$$\frac{\partial}{\partial t}\left(\frac{\partial K}{\partial \dot{y}}\right) = F - \frac{\partial P}{\partial y}$$

$$\frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{y}}\right) = F + \left(\frac{\partial L}{\partial y}\right)$$

$$\boxed{F = \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{y}}\right) - \left(\frac{\partial L}{\partial y}\right)}$$

The above equation is used for linear Joints.

$$\boxed{T = \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta}}$$

This equation is used for rotary Joints.

x
y
z

pose

A
B

$$[R \mid T]$$

$$\left[\delta y_i\right] = \left[\frac{\partial f_i}{\partial x_j}\right]\left[\delta x_j\right]$$

diff. motion of hand from position x, y, z — global

$$\begin{bmatrix} dx \\ dy \\ dz \\ \delta x \\ \delta y \\ \delta z \end{bmatrix} = \begin{bmatrix} \text{Robot} \\ \text{Jacobian} \end{bmatrix} \begin{bmatrix} d\theta_1 \\ d\theta_2 \\ d\theta_3 \\ d\theta_4 \\ d\theta_5 \\ d\theta_6 \end{bmatrix}$$

→ diff. move joint.

$$[D] = [J]\,[D_\theta]$$

dx, dy, dz represents the differential motions of the hand along x, y, z-axes and δx, δy, δz represents the differential rotations of the hand along x-, y- and z-axes.

Jacobian relates the individual joints motions to over all mechanism motions. Jacobian is time related because all the elements magnitude vary with time.

Differential motions of a frame :

The differential motions of a frame can be divided in to the following :

- Differential translations
- Differential rotations
- Differential transformations.

(1) Differential Translation :

A differential translation is a translation of a frame at different values. Thus, it can be represented by Trans (dx , dy , dz). This moves that the frame has moved at a differential amount along the three axes.

$$= \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(2) Differential Rotation :

A differential rotation is a small rotation of the frame. It is generally represented by the Rot (K, dθ) which means that the frame has rotated an angle of dθ about an axis K.

Specifically differential rotations about the x-, y-, z- axes are defined by δx, δy and δz respectively.

$$Rot(x, \delta x) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\delta x & -\sin\delta x & 0 \\ 0 & \sin\delta x & \cos\delta x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Since the rotations are small we can use the following approximations.

$$\sin\delta x = \delta x$$

$$\cos\delta x = 1$$

$$Rot(x, \delta x) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -\delta x & 0 \\ 0 & \delta x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot(y, \delta y) = \begin{bmatrix} \cos\delta y & 0 & \sin\delta y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\delta y & 0 & \cos\delta y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & \delta y & 0 \\ 0 & 1 & 0 & 0 \\ -\delta y & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Rot(z, \delta z) = \begin{bmatrix} \cos\delta z & -\sin\delta z & 0 & 0 \\ \sin\delta z & \cos\delta z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -\delta z & 0 & 0 \\ \delta z & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Differential Rotation about a general axis $\bar{K}$:

The differential rotation about a general axis $\bar{K}$ is composed of three differential rotations about the three axes, in any order. Thus, a differential motion about any general axis $\hat{K}$ can be expressed as

$$Rot(K, d\theta) = Rot(x, \delta x)\, Rot(y, \delta y)\, Rot(z, \delta z)$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -\delta x & 0 \\ 0 & \delta x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & \delta y & 0 \\ 0 & 1 & 0 & 0 \\ -\delta y & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\delta z & 0 & 0 \\ \delta z & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -\delta z & \delta y & 0 \\ \delta x\delta y+\delta z & -\delta x\delta y\delta z+1 & -\delta x & 0 \\ -\delta y+\delta x\delta z & \delta x+\delta y\delta z & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Neglecting all the higher order differentials, we get

$$Rot(K, d\theta) = Rot(x, \delta x)\, Rot(y, \delta y)\, Rot(z, \delta z)$$

$$= \begin{bmatrix} 1 & -\delta z & \delta y & 0 \\ \delta z & 1 & -\delta x & 0 \\ -\delta y & \delta x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Differential Transformations of a frame:

The differential transformation of a frame is a combination of differential translations and rotations. If we denote original frame as T and assume that the change in the frame T as a result of a differenti transformation is expressed as dT, then

$$[T + dT] = [Trans(dx, dy, dz), Rot(K, d\theta)][T]$$

(or) $$[dT] = [Trans(dx, dy, dz, Rot(K, d\theta) - I][T]$$

where T is original frame.

dT is change in frame as a result of differential transformation.

$$[dT] = [\Delta][T]$$

where Δ is differential operator and yields change in frame

$$[\Delta] = [\text{Trans}(dx, dy, dz) \times \text{Rot}(K, d\theta) - I]$$

$$= \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\delta z & \delta y & 0 \\ \delta z & 1 & -\delta x & 0 \\ -\delta y & \delta x & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -\delta z & \delta y & dx \\ \delta z & 1 & -\delta x & dy \\ -\delta y & \delta x & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & -\delta z & \delta y & dx \\ \delta z & 0 & -\delta x & dy \\ -\delta y & \delta x & 0 & dz \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Differential changes between frames:

The equation for differential transformation of a frame is

$$[dT] = [\Delta][T]$$

In the above relation, 'Δ' is called the differential operator. The differential operator represents a differential operator relative to the fixed reference frame and is technically $^{U}\Delta$.

It is possible to define another differential operator, this time relative to the current frame itself. The differential operator relative to the frame (T_Δ) is relative to a current frame

$$[dT] = [T]\,[T_\Delta]$$

$$[T]^{-1}[\Delta][T] = [T]^{-1}[T][T_\Delta]$$

$$\boxed{[T_\Delta] = [T]^{-1}[\Delta][T]}$$

This equation can be used to calculate the differential operator relative to the frame, T_Δ.

$$T^{-1} = \begin{bmatrix} n_x & n_y & n_z & -p\cdot n \\ o_x & o_y & o_z & -p\cdot o \\ a_x & a_y & a_z & -p\cdot a \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Delta = \begin{bmatrix} 0 & -\delta z & \delta y & dx \\ \delta z & 0 & -\delta x & dy \\ -\delta y & \delta x & 0 & dz \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$[T^{-1}][\Delta][T] = {}^{T}\Delta = \begin{bmatrix} 0 & -{}^{T}\delta z & {}^{T}\delta y & {}^{T}dx \\ {}^{T}\delta z & 0 & -{}^{T}\delta x & {}^{T}dy \\ -{}^{T}\delta y & {}^{T}\delta x & 0 & {}^{T}dz \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

${}^{T}\Delta$ is made to look exactly like the Δ matrix but all elements are relative to the current frame.

$${}^{T}\delta x = \bar{\delta}\cdot\bar{n}$$
$${}^{T}\delta y = \bar{\delta}\cdot\bar{o}$$
$${}^{T}\delta z = \bar{\delta}\cdot\bar{a}$$
$${}^{T}dx = \bar{n}\cdot[(\bar{\delta}\times\bar{p}) + \bar{d}]$$
$${}^{T}dy = \bar{o}\cdot[(\bar{\delta}\times\bar{p}) + \bar{d}]$$
$${}^{T}dx = \bar{a}\cdot[(\bar{\delta}\times\bar{p}) + \bar{d}]$$

Relation between Jacobian and the differential operator :

The equation used to represent the differential motions is

$$[D] = [J][D_\theta] \quad \text{--- (1)}$$

The equation used to represent the differential transformation of a frame is

$$[dT] = [\Delta][T] \quad \text{--- (2)}$$

suppose if robot's joints are moved at differential amount, using eqn (1) and knowing the Jacobian we can calculate the [D] matrix which contains values of $dx, dy, dz, \delta x, \delta y, \delta z$ (differential motions of the hand) These values are substituted in ~~equation~~ the equation to calculate Δ.

$$\Delta = \text{Trans}(dx, dy, dz) \times \text{Rot}(k, d\theta) - \underline{I}$$

From the above eqn, differential operator (Δ) is calculated and it is substituted in eqn (2) to calculate [dT]

$$[d]_{new} = [d]_{old} + [dT]$$

Inverse Jacobian:

To calculate the differential motions (or velocities) needed at the joints of the robot for a desired hand differential motion (or velocity), we need to calculate the inverse of the Jacobian and use it in the following equation.

$$[D] = [J][D_\theta]$$

$$[J^{-1}][D] = [J^{-1}][J][D_\theta]$$

$$[J^{-1}][D] = [D_\theta]$$

$$\Rightarrow \boxed{[D_\theta] = [J^{-1}][D]}$$

This means if we know the inverse of the Jacobian, one can calculate how fast each joint must move, so that the robot's hand will yield a desired differential motion or velocity.

Effective moments of Inertia:

To simplify the equations of motion, the equations can be rewritten in symbolic form as

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} D_{ii} & D_{ij} \\ D_{ji} & D_{jj} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_i \\ \ddot{\theta}_j \end{bmatrix} + \begin{bmatrix} D_{iii} & D_{ijj} \\ D_{jii} & D_{jjj} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \end{bmatrix}$$

$$= \begin{bmatrix} D_{iij} & D_{iji} \\ D_{jij} & D_{jji} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \dot{\theta}_2 \\ \dot{\theta}_2 \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} D_i \\ D_j \end{bmatrix}$$

The above equation is written for a two degree-of-freedom systems; a coefficient in the form of D_{ii} is known as effective inertia at joint i. This effective inertia at joint i creates an acceleration at joint i which causes a torque at joint i equal to $D_{ii}\ddot{\theta}_i$ where, as a coefficient in the form D_{ij} is known as coupling inertia between joints i and j as an acceleration at joint i or j causes a torque at joint j or i equal to $D_{ij}\ddot{\theta}_i$ or $D_{ji}\ddot{\theta}_j$.

$D_{ijj}\theta_j^2$ terms represent centripetal forces acting at joint i due to a velocity at joint j.

All terms with $\dot{\theta}_1\dot{\theta}_2$ represent coriolis accelerations and when multiplied by corresponding inertia they represent coriolis forc

Dynamic Equations for multiple degree of freedom Robots:

The dynamic equations for a 2 DOF system is much more complicated than a 1 DOF system. Similarly these equations for a multiple degree of freedom robot are very long and complicated but can be found by calculating the K.E and P.E of the links and joints by defining the Lagrangian and by differentiating it with joint variables.

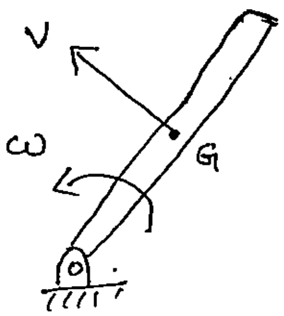
The kinetic energy of a rigid body with motion in three dimensions is

$$K = \frac{1}{2} m \bar{V}^2 + \frac{1}{2} \bar{\omega} \cdot \bar{h}_G$$

where $\dot{h}_G$ is the angular momentum of the body about G

The KE of a rigid body in planar motion simplifies to

$$K = \frac{1}{2} m\bar{V}^2 + \frac{1}{2} \bar{I} \omega^2$$



A rigid body in 3 dimensional motion.

The velocity of a point about a robot's link can be defined by differentiating the position eqn of the point.

The transformations between the hand frame and the base frame of the robot can be represented as

$$^RT_H = {^RT_1}\, {^1T_2}\, {^2T_3} \cdots {^{n-1}T_n} = A_1 A_2 A_3 \cdots A_n$$

For 6-axis robot, the eqn can be written as

$$^0T_6 = {}^0T_1\, {}^1T_2\, {}^2T_3 \ldots {}^5T_6 = A_1 A_2 A_3 \ldots A_6$$

Derivative of an A_i matrix for a revolute joint w.r.t joint variable θ_i is

$$\frac{\partial A_i}{\partial \theta_i} = \frac{\partial}{\theta_i} \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & \alpha_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -S\theta_i & -C\theta_i C\alpha_i & C\theta_i S\alpha_i & a_i S\theta_i \\ C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The above matrix can be broken in to constant matrix Q_i and the A_i matrix

$$= \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\frac{\partial A_i}{\partial \theta_i} = Q_i A_i$$

similarly for prismatic joints,

$$\frac{\partial A_i}{\partial d_i} = \frac{\partial}{\partial d_i}\begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The above matrix can also be broken in to Q_i & A_i

$$\frac{\partial A_i}{\partial \theta_i} = Q_i \cdot A_i$$

$$\therefore Q_{(revolute)} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$Q_{(prismatic)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

we can express position of point 'P' as

$$P_i = {}^{R}T_i\, r_i = {}^{0}T_i\, r_i$$

Differentiating w.r.t Joint Variables

$$v_i = \frac{d}{dt}\left({}^{0}T_i r_i\right)$$

$$= \sum_{j=1}^{i}\left(\frac{\partial({}^{0}T_i)}{\partial q_j}\frac{dq_j}{dt}\right) r_i = \sum_{j=1}^{i}\left(U_{ij}\frac{dq_j}{dt}\right) r_i$$

The kinetic energy of an element of mass m on a link is, $dK_i = \frac{1}{2}\left(\dot{x}_i^2 + \dot{y}_i^2 + \dot{z}_i^2\right) dm$

$$v_i v_i^T = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \end{bmatrix} \begin{bmatrix} \dot{x}_i & \dot{y}_i & \dot{z}_i \end{bmatrix} = \begin{bmatrix} \dot{x}_i^2 & \dot{x}_i\dot{y}_i & \dot{x}_i\dot{z}_i \\ \dot{y}_i\dot{x}_i & \dot{y}_i^2 & \dot{y}_i\dot{z}_i \\ \dot{z}_i\dot{x}_i & \dot{z}_i\dot{y}_i & \dot{z}_i^2 \end{bmatrix}$$

$$\text{Trace}\left(v_i v_i^T\right) = \text{Trace}\begin{bmatrix} \dot{x}_i^2 & \dot{x}_i\dot{y}_i & \dot{x}_i\dot{z}_i \\ \dot{y}_i\dot{x}_i & \dot{y}_i^2 & \dot{y}_i\dot{z}_i \\ \dot{z}_i\dot{x}_i & \dot{z}_i\dot{y}_i & \dot{z}_i^2 \end{bmatrix} = \dot{x}_i^2 + \dot{y}_i^2 + \dot{z}_i^2$$

combining the above eqns yields the following eqn for KE,

$$dK_i = \frac{1}{2}\,\text{Trace}\left[\left(\sum_{p=1}^{i}\left(U_{ip}\frac{dq_p}{dt}\right) r_i\right)\left(\left(\sum_{r=1}^{i}\left(U_{ir}\frac{dq_r}{dt}\right) r_i\right)^T\right] dm_i$$

where p and r represents the different Joint members

$$K_i = \int dK_i = \frac{1}{2}\,\text{Trace}\left[\sum_{p=1}^{i}\sum_{r=1}^{i} U_{ip}\left(\int r_i r_i^T dm_i\right) U_{ir}^T\, \dot{q}_p \dot{q}_r\right]$$

The pseudo Inertia matrix representing the $\int r_i r_i^T$ terms can be written as

$$J_i = \begin{bmatrix} -(I_{xx}+I_{yy}+I_{zz})/2 & I_{xy} & I_{xz} & m_i\bar{x}_i \\ I_{xy} & I_{xx}-I_{yy}+I_{zz}/2 & I_{yz} & m_i\bar{y}_i \\ I_{xz} & I_{yz} & I_{xx}+I_{yy}-I_{zz}/2 & m_i\bar{z}_i \\ m_i\bar{x}_i & m_i\bar{y}_i & m_i\bar{z}_i & m_i \end{bmatrix}$$

since this matrix is independent of joint angles and velocities, it must be evaluated only once.

$$K_i = \frac{1}{2} \sum_{i=1}^{n} \sum_{p=1}^{i} \sum_{r=1}^{i} \mathrm{Trace}\left(U_{ip} J_i U_{ir}^T\right) \dot{q}_p \dot{q}_r$$

The total kinetic energy of the robot

$$K_i = \frac{1}{2} \sum_{i=1}^{n} \sum_{p=1}^{i} \sum_{r=1}^{i} \mathrm{Trace}\left(U_{ip} J_i U_{ir}^T\right) \dot{q}_p \dot{q}_r + \frac{1}{2} \sum_{i=1}^{n} I_{i(act)} \dot{q}_i^2$$

The potential energy of the system is the sum of the potential energies of each link

$$P = \sum_{i=1}^{n} P_i = \sum_{i=1}^{n} \left[-m_i g^T \cdot \left({}^0T_i \bar{r}_i \right) \right]$$

The Lagrangian is then

$$L = K - P$$

$$= \frac{1}{2} \sum_{i=1}^{n} \sum_{p=1}^{i} \sum_{r=1}^{i} \mathrm{Trace}\left(U_{ip} J_i U_{ir}^T\right) \dot{q}_p \dot{q}_r + \frac{1}{2} \sum_{i=1}^{n} I_{i(act)} \dot{q}_i^2$$

$$- \sum_{i=1}^{n} \left[-m_i g^T \cdot \left({}^0T_i \bar{r}_i \right) \right]$$

TRAJECTORY PLANNING

Trajectory planning:

The trajectory or the path refers to a time history of position, velocity and acceleration of all possible movements of the manipulator. The internal functions and the computing systems represent the path or trajectory generation.

Path or trajectory planning refers to the way a robot is moved from one location to another in a controlled manner.

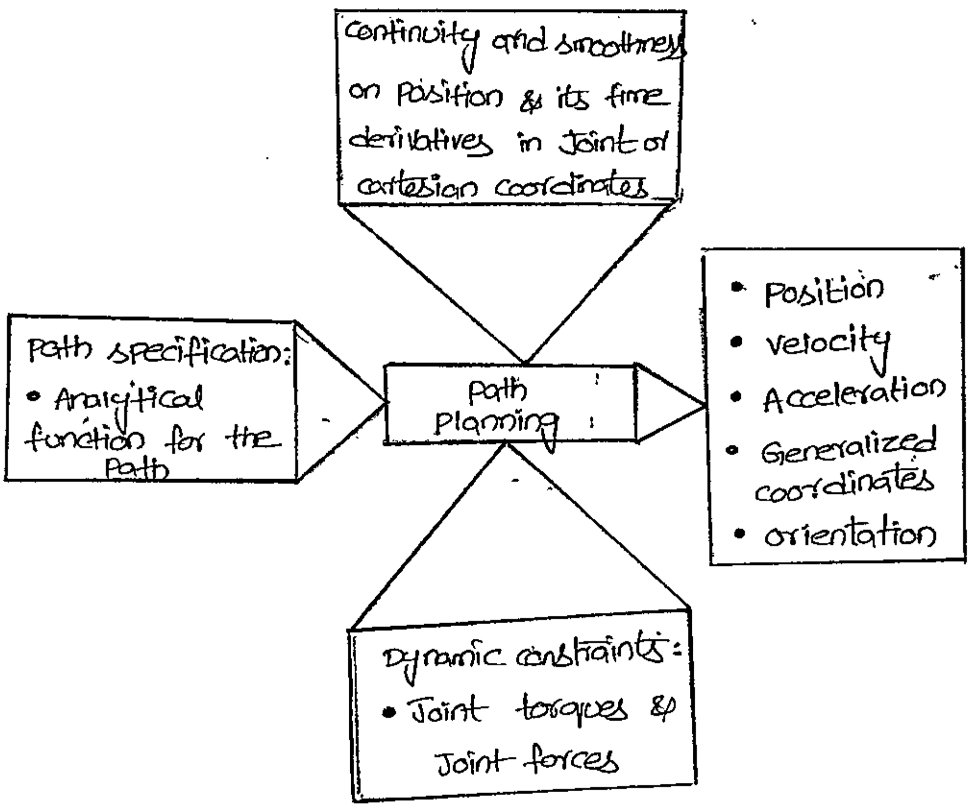


Fig: Path planning Block diagram

Path control modes: (Modes to avoid obstacles in trajectory)

The movement of the arm of a robot to be controlled has the following schemes to be considered under two possible constraints (1) Trajectory constraints
(2) obstacle constraints

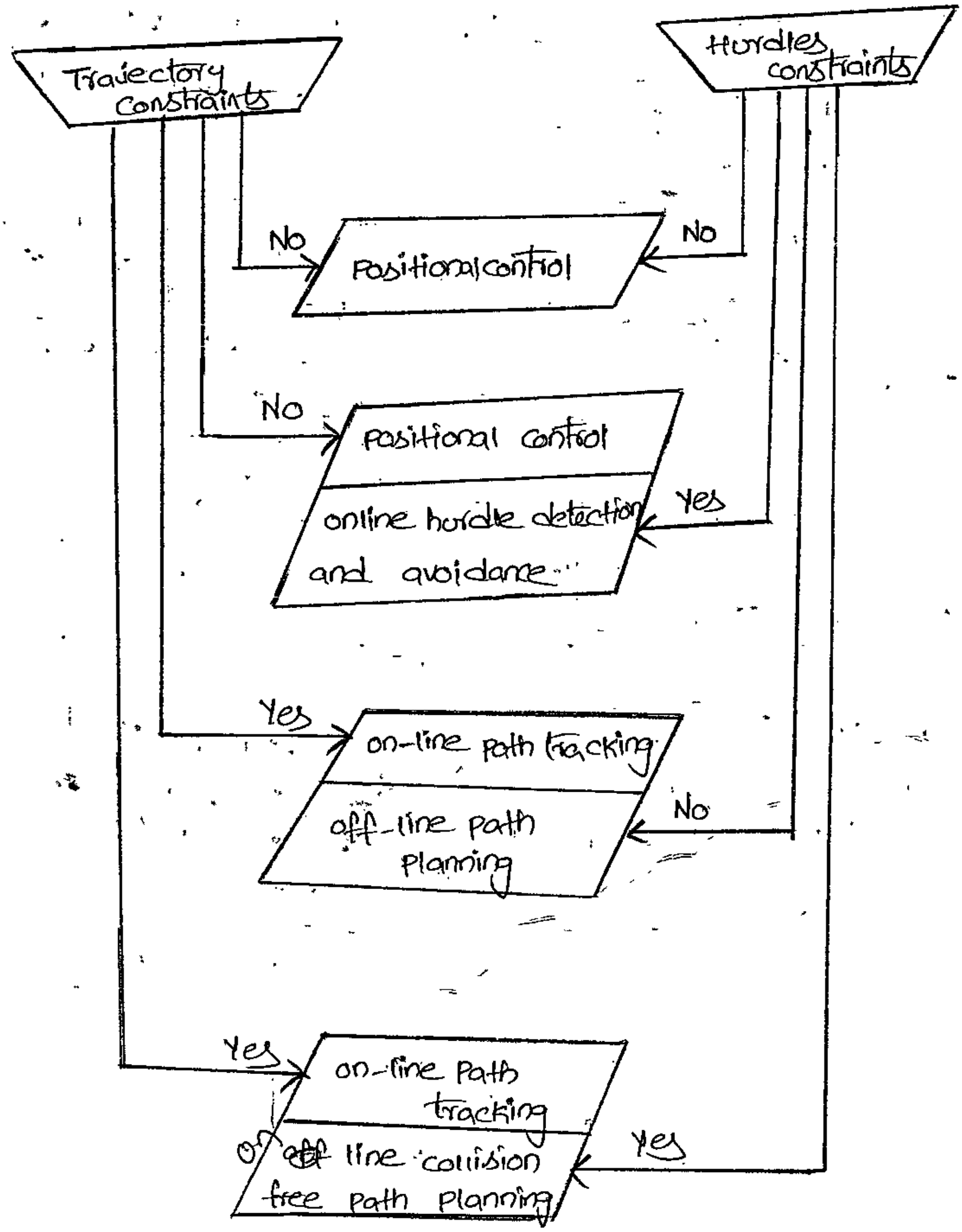


Fig: Path control schemes

Mode 1 : when there is no obstacle and there is no constraint on the path to be followed the positional control is sufficient.

Mode 2 : when there is certain obstacle and no path constraint the positional control is to be employed with hurdle detection and avoidance.

Mode 3 : when there is no obstacle in the path and tool has to move in certain given path, the off line path planning with on-line path tracking has to be carried out.

Mode 4 : The presence of a hurdle with a definite constraint on path needs off line collision free path planning coupled with on-line path tracking.

General considerations in Trajectory Planning :

The general points to be considered while planning the manipulator trajectories are :

- **Frame Basis :**

It is advantangeous to describe and generate path by specifying robot motions in terms of tool frame with respect to the base frame.

- Specification of Path :

The decoupled motion description of the tool frame from a particular robot or end effector resorts in module formation. This helps in usage of the same description to different manipulators or different sizes of tool.

- Movements of manipulator :

The manipulator motion is identified and relative to the initial position and final position of the tool frame with respect to the base frame.

- Sequence of Intermediate Points :

This involves in specifying sequence of intermediate points between the end positions of the tool frame with respect to station frame. The tool has to pass through these intermediate points describing the path.

- Elapse Time :

The specifying of elapsed time between the intermediate points in the description of the path gives temporal attributes of the manipulator motion.

- Smooth traverse :

By constraining the spatial and temporal qualities of the path between the points it is possible to ensure the smoothness of operation. To describe the motion the path has to be described by a well defined smooth function which is also continous with two of its time derivatives being continous.

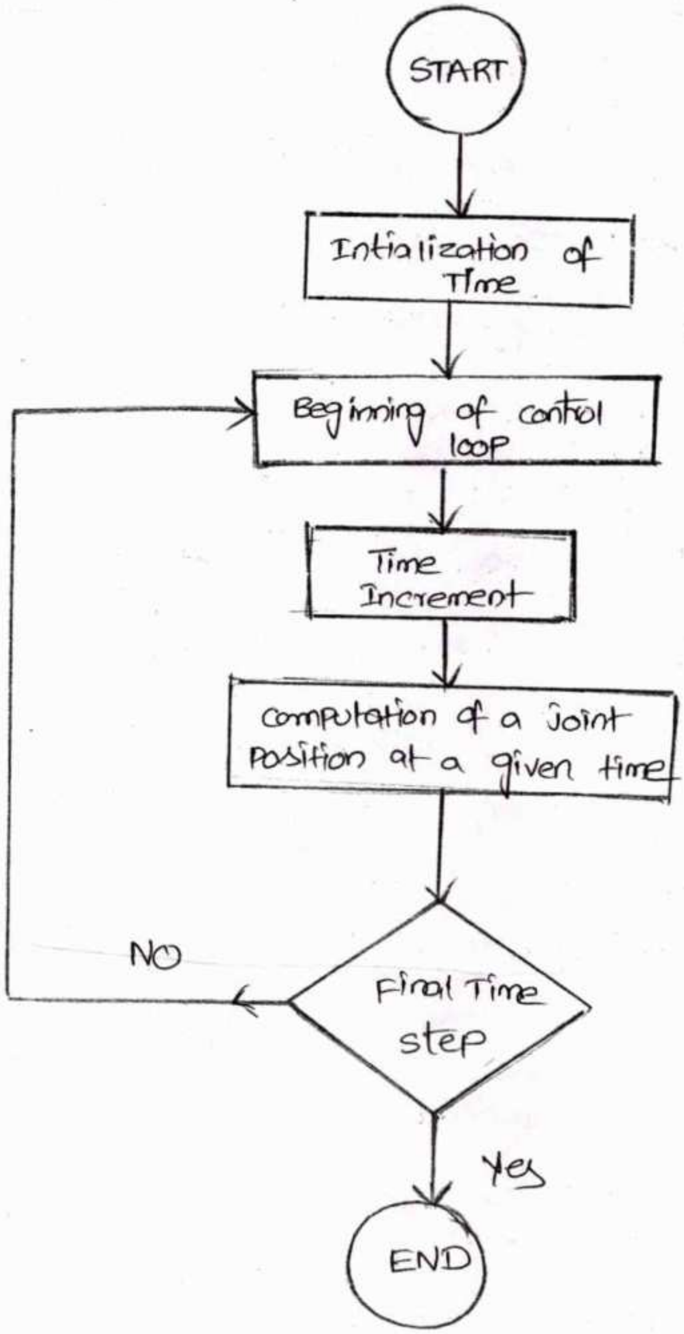
Schemes of path generation :

There are two methods of path generation :

(1) The Joint space scheme

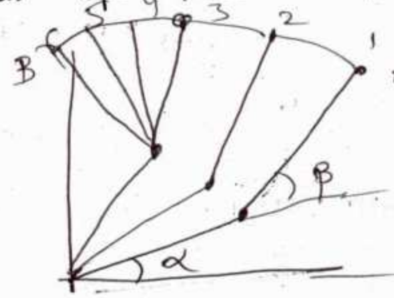
(2) The cartesian space scheme.

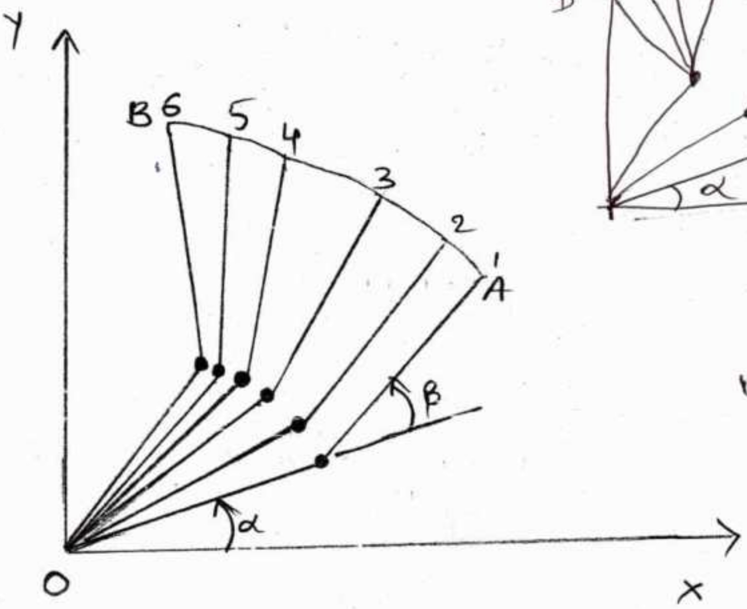
(1) Joint space scheme :



Flow chart for Joint space scheme

In the Joint space scheme, the time history of all joint variables and their first two time derivatives are used for planning the motion of the manipulator.





10°/sec

α	B
20	30
30	40
40	50
40	60
40	70
40	80

Fig: Joint-space normalized movements of a robot with two degrees of freedom.

In the above figure, the only calculation needed was the joint values for the destination i.e, to move the robot from point A to point B.

Using the inverse kinematic equations of the robot, the total joint displacements are calculated. The description of the motion made by the robot by its joint values is called Joint space scheme.

Advantages of Joint space scheme:

- Planning of trajectory from control variables in direct terms
- Real time planning of trajectory.
- Ease of generating trajectory planning.
- uses low degree polynomials to interpolate joint hinge points
- computationally faster
- Dealing of the manipulator dynamic constraints is relatively easy.

Disadvantages:

- Difficulty in determining the location of the joints, links and hand during motions
- obstacle avoidance in the path is very difficult.
- Less accurate along the cartesian path.

(2) cartesian space scheme:

In this method the manipulator's hand position, velocity and acceleration time history are used in planning and using these, joint space variables are computed.

In cartesian space scheme all segments of the motion must be calculated. The basic difference for joint space & cartesian space is for cartesian space, the joint values must be repeatedly calculated through the inverse kinematic eqns of the robot.

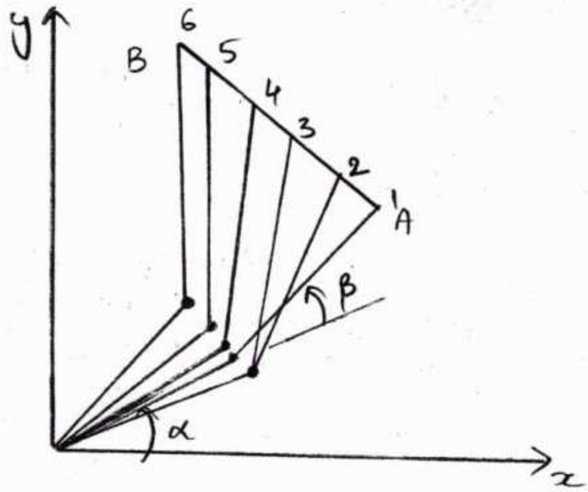
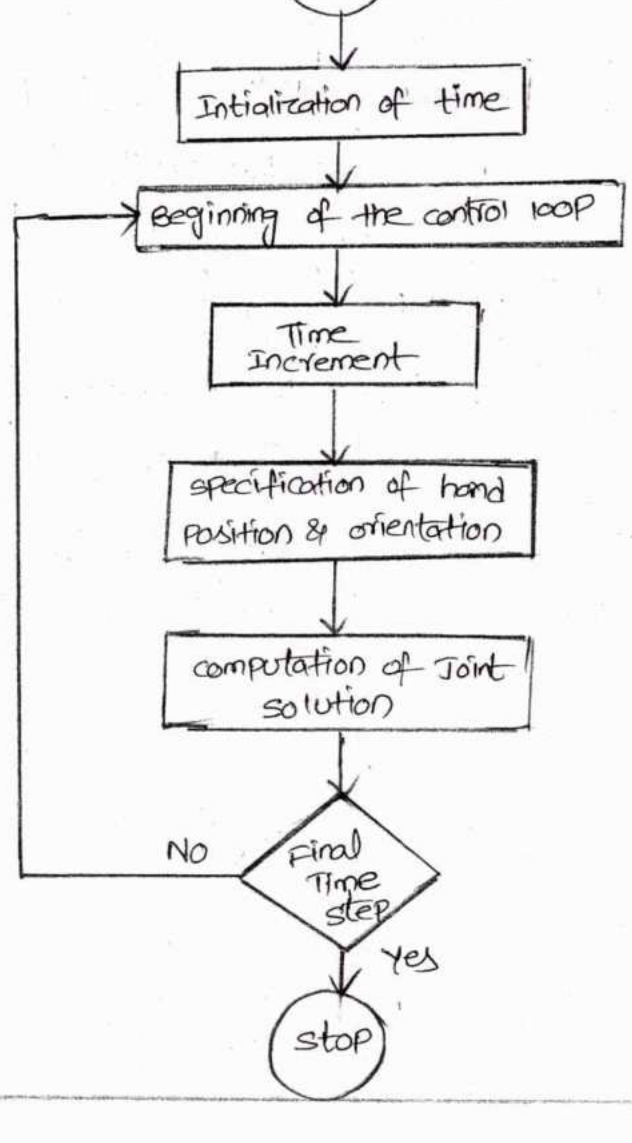
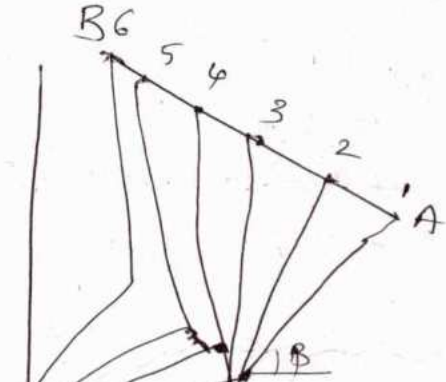


Fig : cartesian space movements of a two-degree freedom of a robot

Flow chart for cartesian space scheme :





α	β
20	30
14	55
16	69
21	77

Advantages of cartesian space scheme :

- This method is straight forword.
- obstacles can be easily avoidable.
- The assured accuracy along the straight line path,
- Easy determination of link and its locations during motion.

Disadvantages :

- It is computationally slow
- Longer control intervals.
- mapping the hand coordinates in to joint coordinates is not properly defined.

Trajectory planning with 3rd order polynomial :

Position : $\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3$

Velocity : $\dot{\theta}(t) = c_1 + 2 c_2(t) + 3 c_3 t^2$

Acceleration : $\ddot{\theta}(t) = 2 c_2 + 6 c_3 t$

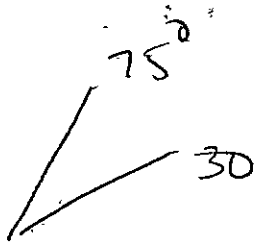
5th order polynomial :

position : $\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5$

velocity : $\dot{\theta}(t) = a_1 + 2 a_2 t + 3 a_3 t^2 + 4 a_4 t^3 + 5 a_5 t^4$

Acceleration : $\ddot{\theta}(t) = 2 a_2 + 6 a_3 t + 12 a_4 t^2 + 20 a_5 t^3$

(Pb) It is desired to have the first joint of a six-axis robot to go from a initial angle of 30° to a final angle of 75° in 5 seconds. Using a third order polynomial calculate the joint angles at 1, 2, 3, 4 seconds?



5 sec

(Sol) Given:

$\theta(t_i) = 30°$

$\theta(t_f) = 75°$

$t_i = 0$ sec

$t_f = 5$ sec

①, ②, 3, 4

Position eqn :— $\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3$

Velocity eqn :- $\dot\theta(t) = c_1 t \cdot 2c_2 t + 3c_3 t^2$.

Acceleration eqn :- $\ddot\theta(t) = 2c_2 + 6c_3 t$

Initial position: $\theta(t_i) = 30°$ ($\Rightarrow t_i = 0$ sec)

$\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3$

$\theta(0) = 30°$

$30 = c_0 + c_1(0) + c_2(0)^2 + c_3(0)^3$

$\boxed{c_0 = 30}$ —— (1)

Final position: $\theta(t_f) = 75°$ ($t_f = 5$ sec)

$\theta(5) = 75°$

$75 = c_0 + c_1(5) + c_2(5)^2 + c_3(5)^3$

$c_0 + 5c_1 + 25c_2 + 125c_3 = 75$

$30 + 5c_1 + 25c_2 + 125c_3 = 75$

$$\boxed{5c_1 + 25c_2 + 125c_3 = 45} \quad \text{--- (2)}$$

Initial Velocity:

$f'(x) = 0$

$f''(x) > 0$

$\theta'(t_i) = 0$

$\theta'(t) = c_1 + 2c_2t + 3c_3t^2$

$0 = c_1 + 2c_2(0) + 3c_3(0)^2$

$$\boxed{c_1 = 0} \quad \text{--- (3)}$$

Final Velocity:

$\theta'(t_f) = 0$

$\theta'(5) = c_1 + 2c_2t + 3c_3(t)^2$

$0 = c_1 + 2c_2(5) + 3c_3(5)^2$

$$\boxed{10c_2 + 75c_3 = 0} \quad \text{--- (4)}$$

Sub $c_1 = 0$ in eqn (2)

$5c_1 + 25c_2 + 125c_3 = 45$

$5(0) + 25c_2 + 125c_3 = 45$

$$\boxed{25c_2 + 125c_3 = 45} \quad \text{--- (5)}$$

$$25c_2 + 125c_3 = 45$$

$$10c_2 + 75c_3 = 0$$

solving the above eqns we get

$$\boxed{c_3 = -0.72}$$

$$\boxed{c_2 = 5.4}$$

Joint angles :

Position eqn : $\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3$

$$= 30 + 0 + 5.4t^2 - 0.72t^3$$

$$\boxed{\theta(t) = 30 + 5.4t^2 - 0.72t^3}$$

$$\theta(1) = 30 + 5.4(1)^2 - 0.72(1)^3 = 34.68^\circ$$

$$\theta(2) = 30 + 5.4(2)^2 - 0.72(2)^3 = 45.84^\circ$$

$$\theta(3) = 30 + 5.4(3)^2 - 0.72(3)^3 = 59.16^\circ$$

$$\theta(4) = 30 + 5.4(4)^2 - 0.72(4)^3 = 70.32^\circ$$

solving the eqns (4) & (5),

we get $c_3 = -0.72$

$c_2 = 5.4$

To find Joint angles:

$$\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3$$

$$= 30 + 0 + 5.4t^2 - 0.72t^3$$

$$\theta(t) = 30 + 5.4t^2 - 0.72t^3$$

$$\theta(1) = 30 + 5.4 - 0.72 = 34.68°$$

$$\theta(2) = 30 + 5.4(2)^2 - 0.72(2)^3 = 45.84°$$

$$\theta(3) = 30 + 5.4(3)^2 - 0.72(3)^3 = 59.16°$$

$$\theta(4) = 30 + 5.4(4)^2 - 0.72(4)^3 = 70.32°$$

To find velocity:

$$\dot{\theta}(t) = c_1 + 2c_2(t) + 3c_3 t^2$$

$$= 0 + 2(5.4)t + 3(-0.72)t^2$$

$$\dot{\theta}(t) = 10.8t - 2.16t^2$$

$$\dot{\theta}(1) = 10.8(1) - 2.16(1)^2 = 8.64$$

$$\dot{\theta}(2) = 10.8(2) - 2.16(2)^2 = 12.96°$$

$$\dot{\theta}(3) = 10.8(3) - 2.16(3)^2 = 12.96°$$

$$\dot{\theta}(4) = 10.8(4) - 2.16(4)^2 = 8.64°$$

To find acceleration :

$$\ddot{\theta}(t) = 2c_2 + 6c_3 t$$

$$= 2(5.4) + 6(-0.72)t$$

$$\boxed{\ddot{\theta}(t) = 10.8 - 4.32t}$$

$$\ddot{\theta}(1) = 10.8 - 4.32 = 6.48^\circ$$

$$\ddot{\theta}(2) = 10.8 - 4.32(2) = 2.16^\circ$$

$$\ddot{\theta}(3) = 10.8 - 4.32(3) = -2.16^\circ$$

$$\ddot{\theta}(4) = 10.8 - 4.32(4) = -6.48^\circ$$

(Pb) A fifth order polynomial is to be used to control the motions of the joints of a robot in joint space. Find the coefficient of fifth order polynomial that allow a joint to go from initial angle of 0° to a final joint angle of 75° in 3 seconds, while the initial and final velocities are zero and initial acceleration and deacceleration are 10°/sec².

Sol. Given :

$$\theta(t_i) = 0^\circ$$

$$\theta(t_f) = 75^\circ$$

$$t_i = 0\,sec$$

$$t_f = 3\,sec$$

$$\ddot{\theta}(t_i) = 10^\circ/sec^2$$

$$\ddot{\theta}(t_f) = -10^\circ/sec^2$$

$$\dot{\theta}(t_i) = \dot{\theta}(t_f) = 0$$

Initial boundary conditions:

$\theta(0) = 0$, $\dot{\theta}(0) = 0$, $\ddot{\theta}(0) = 10$

$\theta(t_i) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5$

$0 = c_0 + c_1(0) + c_2(0)^2 + c_3(0)^3 + c_4(0)^4 + c_5(0)^5$

$\boxed{c_0 = 0}$

$\dot{\theta}(t_i) = 0$

$\dot{\theta}(t_i) = c_1 + 2c_2 t + 3c_2 t^2 + 4c_4 t^3 + 5c_5 t^4$

$= c_1 + 2c_2(0) + 3c_2(0)^2 + 4c_4(0)^3 + 5c_5(0)^4$

$\boxed{c_1 = 0}$

$\ddot{\theta}(t_i) = 10$

$\ddot{\theta}(0) = 2c_2 + 6c_3 t + 12c_4 t^2 + 20c_5 t^3$

$10 = 2c_2 + 6c_3(0) + 12c_4(0)^2 + 20c_5(0)^3$

$2c_2 = 10$

$\boxed{c_2 = 5}$

Final boundary conditions:

$\theta(3) = 75^\circ$, $\dot{\theta}(3) = 0$ $\ddot{\theta}(3) = -10^\circ/sec^2$

$\theta(3) = 75$

$\theta(3) = c_0 + c_1(3) + c_2(3)^2 + c_3(3)^3 + c_4(3)^4 + c_5(3)^5$

$c_0 + 3c_1 + 9c_2 + 27c_3 + 81c_4 + 243c_5 = 75$

$$9(5) + 27c_3 + 81c_4 + 243c_5 = 75$$

$$\boxed{27c_3 + 81c_4 + 243c_5 = 30}$$

$$\dot{\theta}(3) = 0$$

$$\dot{\theta}(3) = c_1 + 2c_2 t + 3c_3 t^2 + 4c_4 t^3 + 5c_5 t^4$$

$$0 = c_1 + 2c_2(3) + 3c_3(3)^2 + 4c_4(3)^3 + 5c_5(3)^4$$

$$0 = 0 + 6c_2 + 27c_3 + 108c_4 + 405c_5$$

$$0 = 6(5) + 27c_3 + 108c_4 + 405c_5$$

$$\boxed{27c_3 + 108c_4 + 405c_5 = -30}$$

$$\ddot{\theta}(t) = -10$$

$$2c_2 + 6c_3(t) + 12c_4(t)^2 + 20c_5 t^3 = -10$$

$$2(5) + 6c_3(3) + 12c_4(3)^2 + 20c_5(3)^3 = -10$$

$$10 + 18c_3 + 108c_4 + 540c_5 = -10$$

$$\boxed{18c_3 + 108c_4 + 540c_5 = -20}$$

Solving the above equations we get

$$c_3 = 190/9 = 21.11$$

$$c_4 = -100/9 = -11.11$$

$$c_5 = 40/27 = 1.48$$

Finally the position, velocity & acceleration equations are

$$\theta(t) = 5t^2 + 21.11t^3 - 11.11t^4 + 1.48t^5$$

$$\dot{\theta}(t) = 10t + 63.33t^2 - 44.44t^3 + 7.4t^4$$

$$\ddot{\theta}(t) = 10 + 127.32t - 133.32t^2 + 29.6t^3$$

UNIT - VII

SENSORS

sensor :

The interaction of the robot with environmental setups needs mechanisms known as sensor.

classification of sensor :

Robot sensors

- Internal sensor
  - Detects arm position, velocities, acceleration
  - used in Robot control
- External sensor (used in)
  - Robot guidance
  - object identification
  - object handling
  - Non contact type.
    1. Range sensors
    2. proximity sensors
       - a) Inductive sensor
       - b) Hall effect sensor
       - c) ultrasonic sensor
       - d) optical sensor
  - contact type
    1. Touch sensor
    2. slip sensor
    3. Force / Torque sensor

position sensors
- Potentiometer
- Resolvers.
- Encoder.

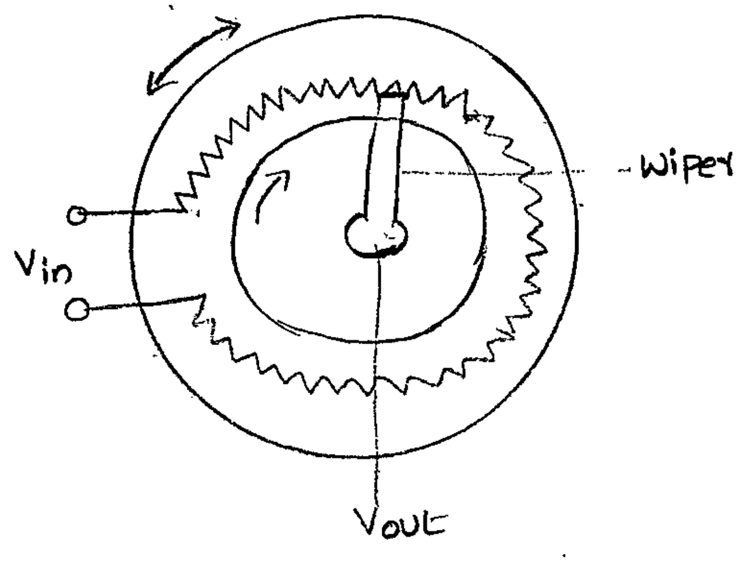
velocity sensors
- Tachometer.
- Encoder
- Differentiation of position signal

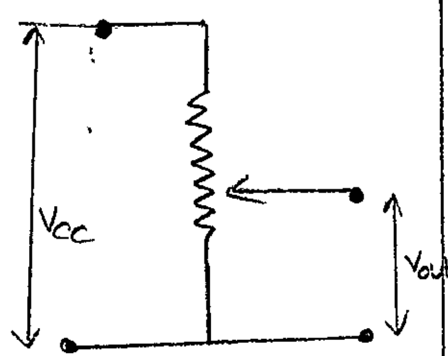
Position sensors :

Position sensors are used to measure displacement in both linear and rotary as well as movements.

(1) potentiometers :

A potentiometer converts position information in to a variable voltage through a resistor. The sweeper on the resistor moves due to a change in position, the proportion of the resistance before or after the point contact with the sweeper compared with the total resistance varies.





Since in this capacity the potentiometer acts as a voltage divider. The output will be proportional to resistance.

$$V_{out} = V_{cc} \frac{R_1}{R}$$

$$= V_{cc} \times \frac{Q_{actual}}{Q_{total}}$$

potentiometers can be rotary or linear and thus can measure linear or rotary motions. Rotary potentiometers can also be multiple-turn, enabling the user to measure many revolutions of motion.

potentiometers are either wire wound or thin film deposit, which is a deposit of a thin film of resistive material on a surface. The major benefit of thin film potentiometer is that their output is continous and thus less noisy.

Potentiometers are generally used as internal feed back sensors in order to report the position of joints and links. potentiometers are used alone or together with other sensors such as encoders.

The potentiometer reports the start up position. Encoders reports the current position. The combination of sensors allows for minimal input requirement but the maximum capacity.

(2) Encoders :

An encoder is a simple device that can output a digital signal for each small portion of a movement. To do this, an encoder wheel or strip is divided into small sections. A light source on one side provides a beam of light, to the other side of the encoder wheel or strip, where it is seen by an other light-sensitive sensor such as a phototransistor.

If the wheel's angular position is such that the light can go through, the sensor on the opposite side will be turned on and will have a high signal. If the angular position of the wheel is such that the light is stopped, the sensor will be off and its output will be low. As the wheel rotates, it can continuously send signals. If the signals are counted, the approximate total angular displacement of the wheel can be measured at any time.

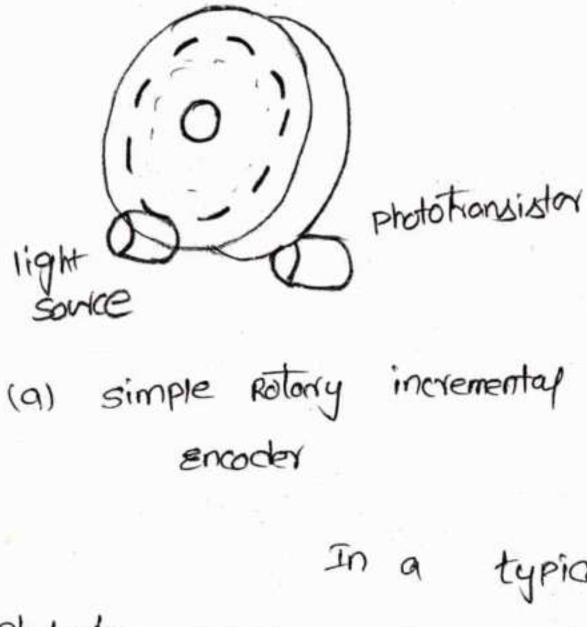
There are 2 basics types of encoders,

(1) Incremental Encoder.

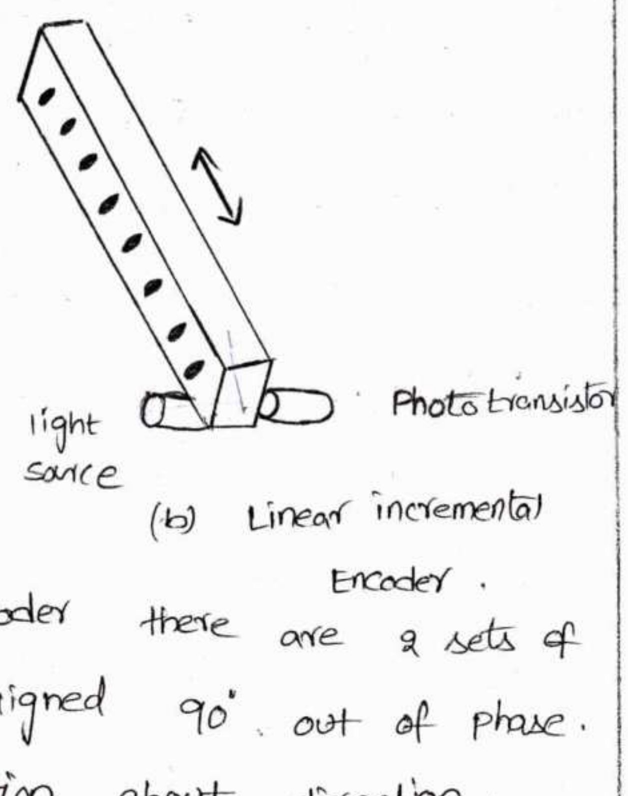
(2) Absolute Encoder

(i) Incremental encoder :

It consists of a glass disk marked with alternating transparent and opaque stripes aligned radially. A phototransmitter (a light source) is located on one side of the disk and a photo receiver is on the other side. As the disk rotates, the light beam is alternately completed & broken. The output from the photoreceiver is a pulse train whose frequency is proportional to the speed (N) of rotation of the disk.



(a) simple Rotary incremental encoder



(b) Linear incremental Encoder.

In a typical encoder there are 2 sets of Phototransmitter and receivers aligned 90° out of phase. This phasing provides information about direction. If the signal A leads signal B by 90°, the encoder disk rotates in one direction. If B leads signal A by 90° then it rotates in other direction.

By counting the pulses and by adding or subtracting based on the sign, it is possible to use the encoder to provide position information w.r.to a known stationary location.

(ii) Absolute Encoder:

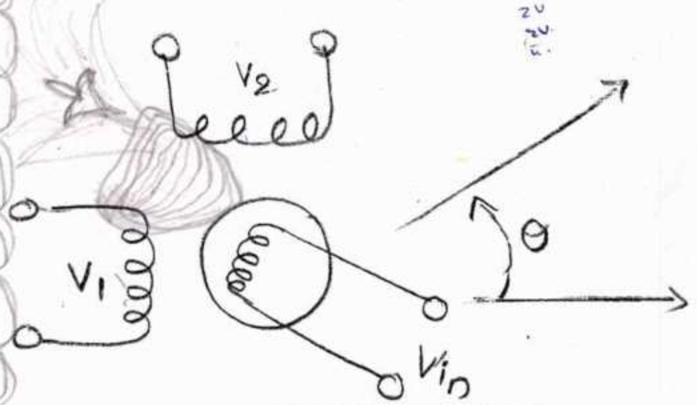
It is used to find the position of the object. Absolute encoders empoly the same basic construction as incremental encoder except that there are more tracks of stripes & a corresponding no of receivers and transmitters. The stripes are arranged to provide a binary number to shaft angle. 1st track will have 2 stripes, 2nd track will have 4 stripes, 3rd will have 8 and so on. Angle can be read directly from the encoder. The resolution of an absolute encoder is dependent on the no of tracks and is given by.

$$\boxed{\text{Resolution} = 2^n}$$ n — no of tracks on disk.

(3) Resolvers :

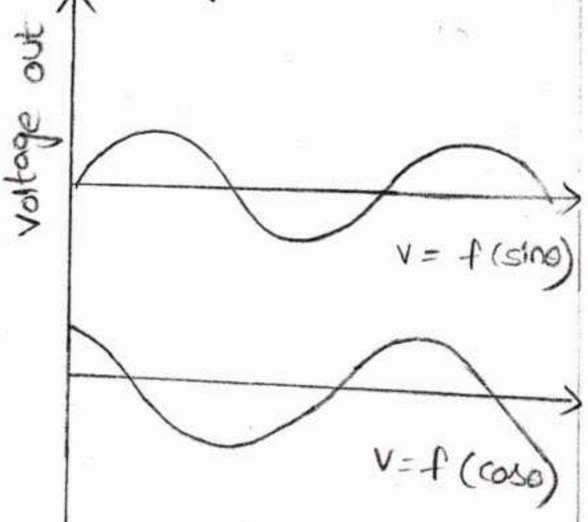
Resolvers are very similar to LVDT's in principle, but are used to measure an angular motion. A resolver is also a transformer where the primary coil is connected to the rotating shaft and carries an alternating current through slip rings. There are 2 secondary coils placed 90° apart from each other. As the rotor rotates, the flux it develops also rotates with it.

When the primary coil in the rotor is parallel to the either of the two secondary coils, the voltage induced in that coil is maximum, while the other secondary coil that is perpendicular to it does not developed any voltage. As the rotor rotates, eventually the voltage in the first secondary coil goes to zero, while the second coil develops its maximum voltage.



$V_1 = A \sin \omega(t) \sin\theta$

$V_2 = A \sin \omega t \cos\theta$



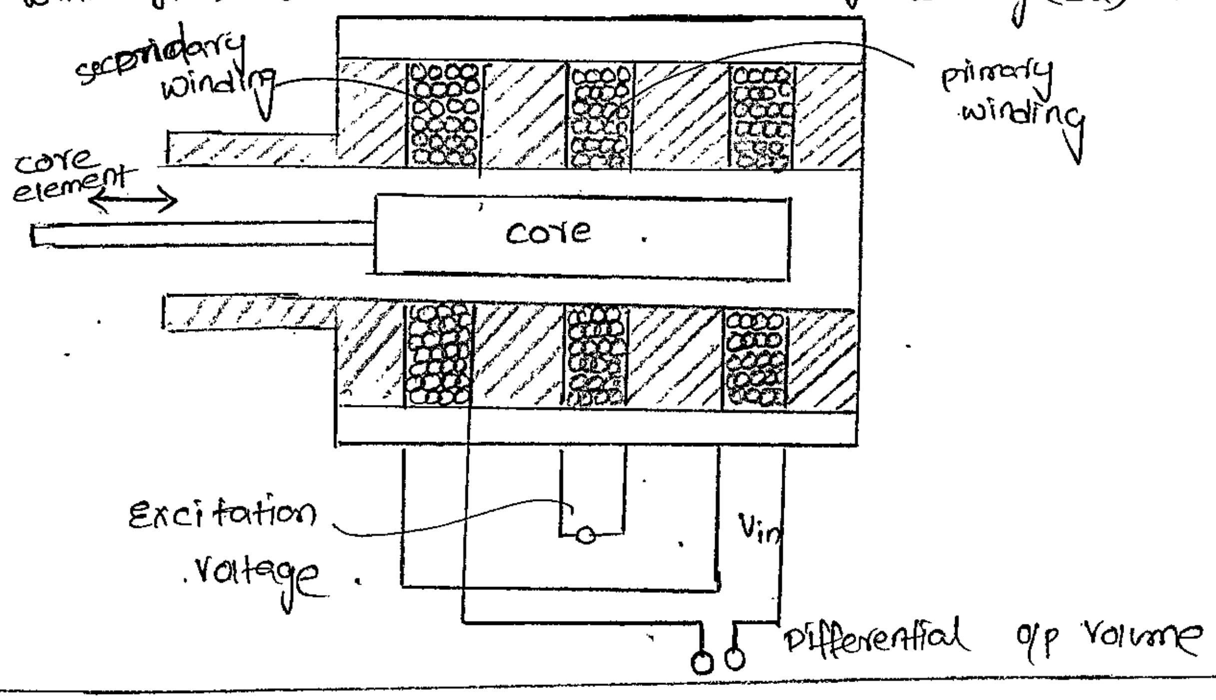
where $A \sin(\omega t)$ = Excitation voltage

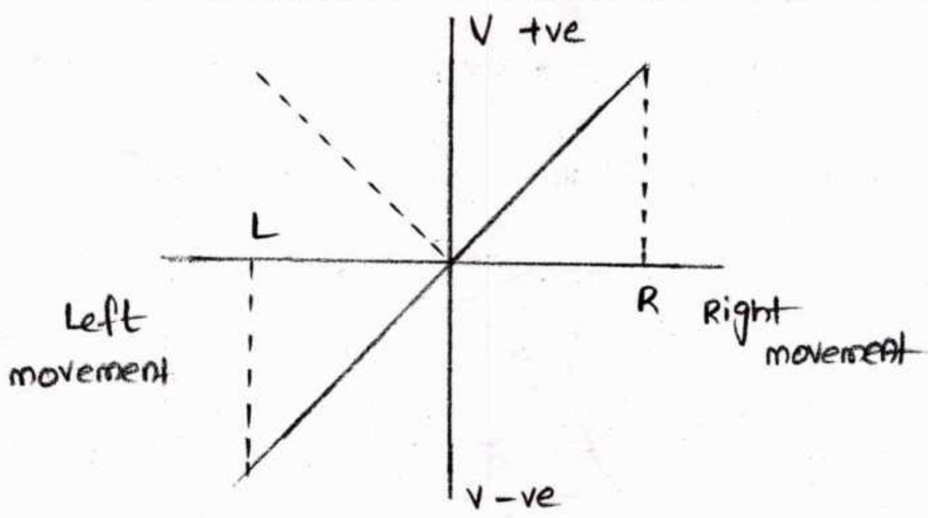
θ = Angle of rotation w.r.t stator.

For all other angles in between the two secondary coils develops a voltage proportional to the sine and cosine of the angle b/w the primary and the two secondary coils. Although the output of a resolver is analog, it is equal to the sine and cosine of the angle, eliminating the necessity to calculate these values later. Resolvers are reliable, robust and accurate.

(4) Linear variable differential Transformer : (LVDT).

LVDT is another type of position sensors where construction is shown in below fig. It consists of an primary winding and two secondary windings and one movable core. The primary is excited with an AC source. when the core is in its exact central location, the amplitude of voltage induced in primary winding(S1) will be same as secondary winding(S2).

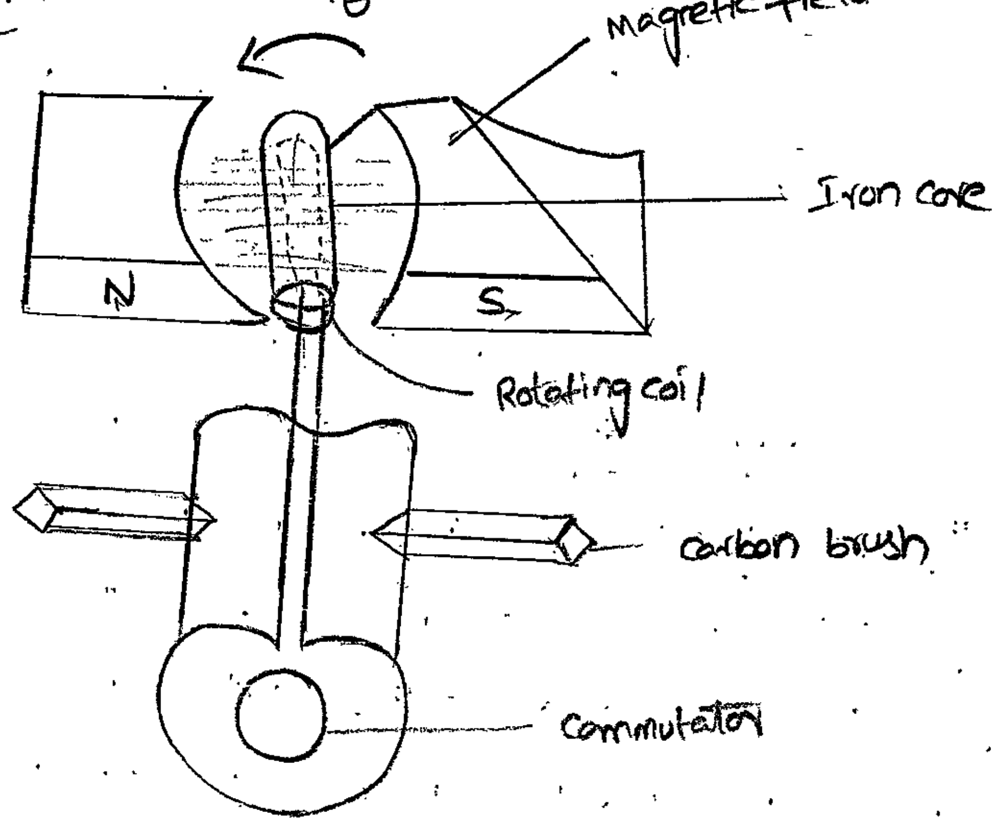




The secondary windings are connected in such a way that the output voltage is equal to the difference b/w S_1 & S_2 ($\because V_{out} = S_1 \sim S_2$). So that the output voltage will be zero at the central location.
The figure illustrates the nature of o/p voltage as the core is moved to the left or to the right.
The magnitude of the o/p voltage is a linear of the core position and the phase is determined by the side of the null position on which the core is located.
Finally the AC o/p of LVDT can be converted in to DC using rectifiers.

Velocity sensors :

Tachometer :



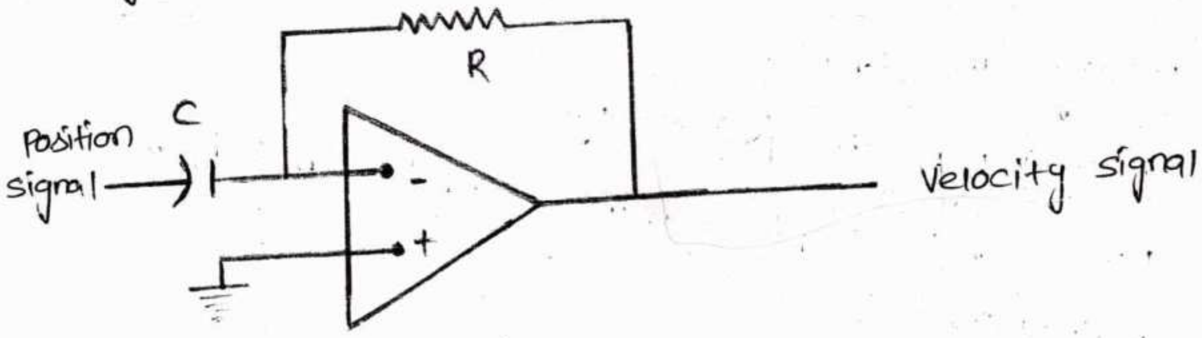
Analog Tachometer.

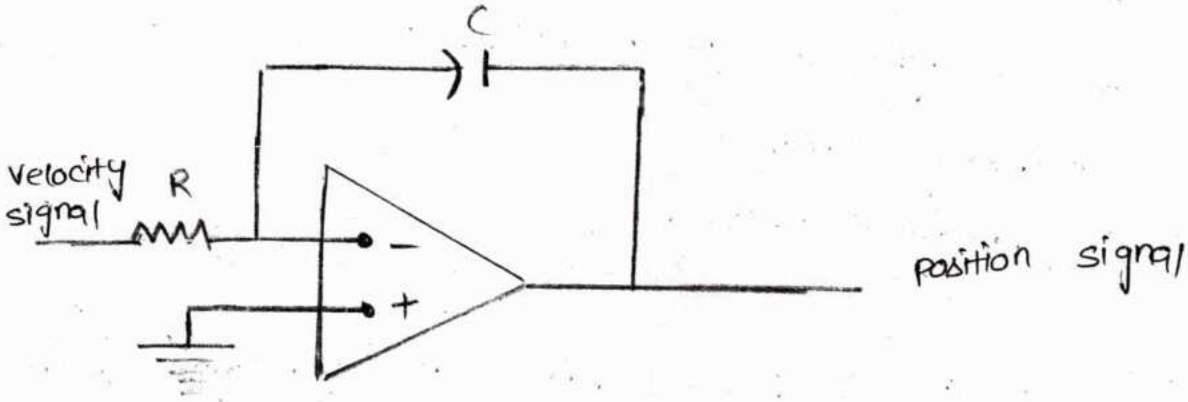
A tachometer is a generator that converts mechanical energy in to electrical energy. Its output is an analog voltage proportional to the input angular speed. It may be used along with potentiometer to estimate velocities. $V_o(t) = K_t(t)\,\omega$

Differentiation of position signal :

If the position signal is clean, it is actually possible and even simple to differentiate the position signal and to convert it to velocity signal.

To do this it is necessary that the signal be as continous as possible in order to prevent creation of large impulses in velocity signal. Thus, it is recommended that thin film plastic resistors be used for position measurement. However differentiation of a signal is always noisy and should be done carefully.





The figure shows a R-C circuit with an op-amp that can be used for differentiation, where the velocity signal is

$$V_{out} = -RC \frac{dV_{in}}{dt}$$

similarly the velocity signal can be integrated to yield position signals

$$V_{out} = \frac{-1}{RC} \int V_{in} \, dt$$

Touch and Tactile sensors :

Touch sensors :

The touch sensors gather the information established by the contact between the parts to be handled and the fingers in the manipulator endeffectors. The signals of touch informations are useful in

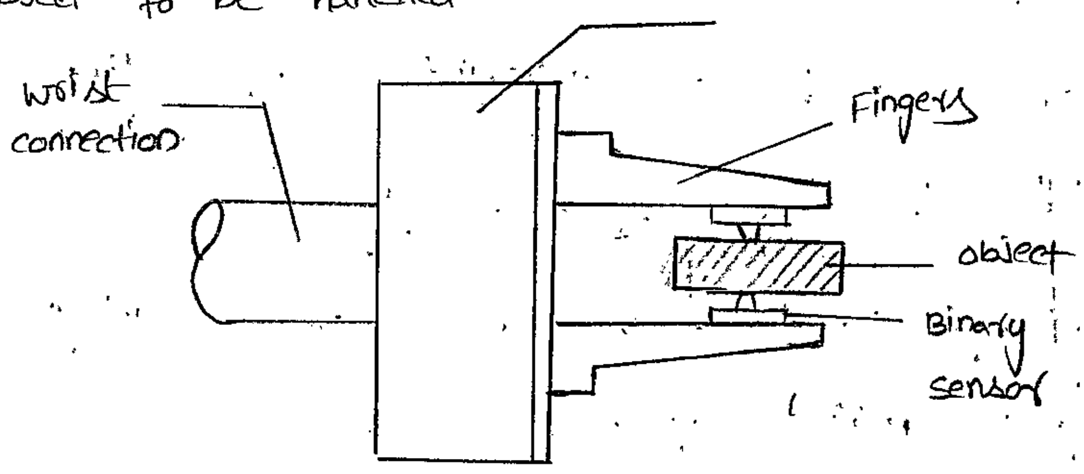
→ locating the objects
→ Recognising the object type
→ Force and torque control needed for task manipulation.

There are 2 types of touch sensors.

(1) Binary sensors
(2) Analog sensors.

Binary sensors :

It is used to detect the existence of the object to be handled.

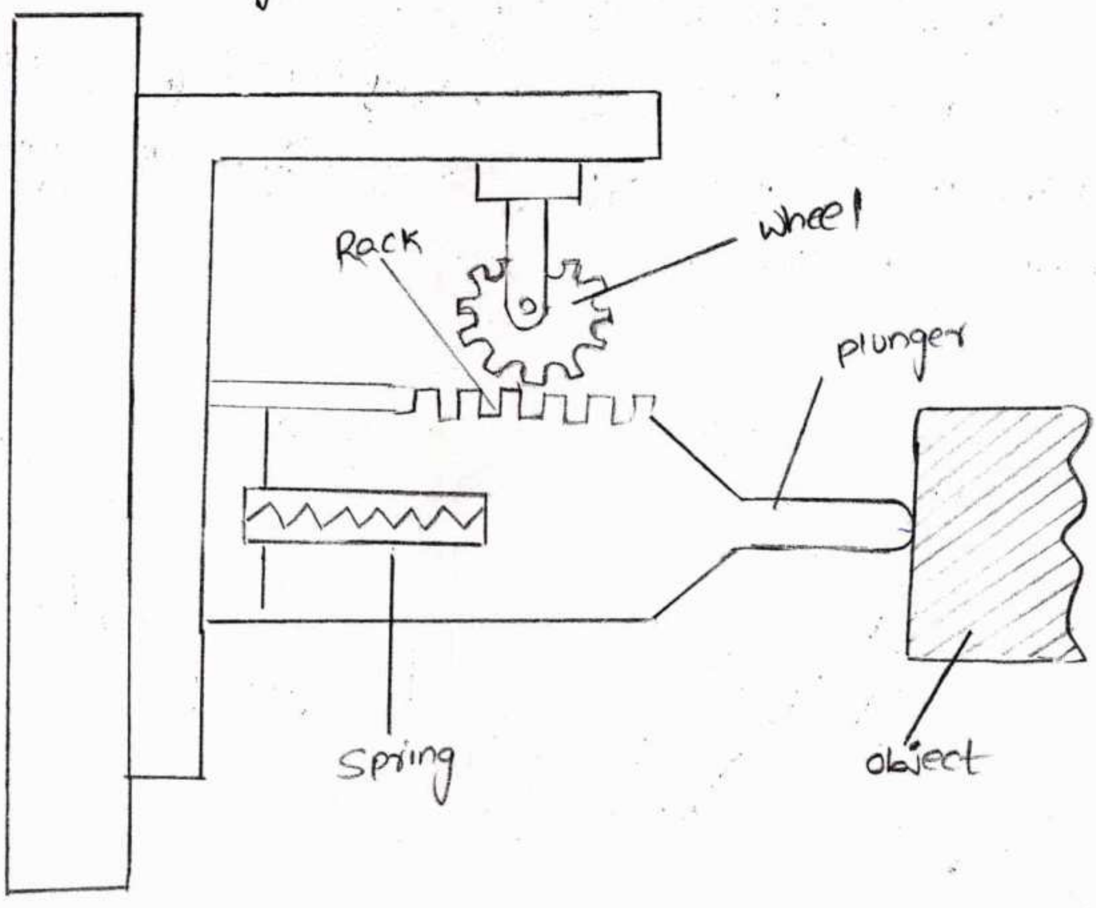


Hand with Binary sensor.

For example, micro switches and limit switches. The devices that deliver sensing signal by contact at two gripping points are termed the binary sensors.

The fingers as shown in above figure accomodate the binary sensors. The contact with the parts results in deflection and this information is sufficient to determine the presence of object between the finger. The proper grasping and manipulation of the object in the work envelope can be easily achieved through centering of the fingers assisted by the information given by binary sensors.

Analog sensor : It is used to produce proportional output signal for the force exerted locally. For example, a code wheel with a plunger.



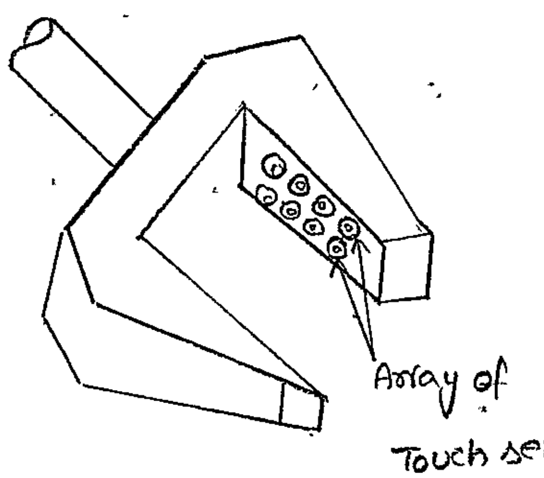
This type of sensors are featered by spring actuated plunger connected to a code wheel. The deflection of the plunger rod by the action of contact force results in rotation of the wheel which gives an output proportional to the sensors force.

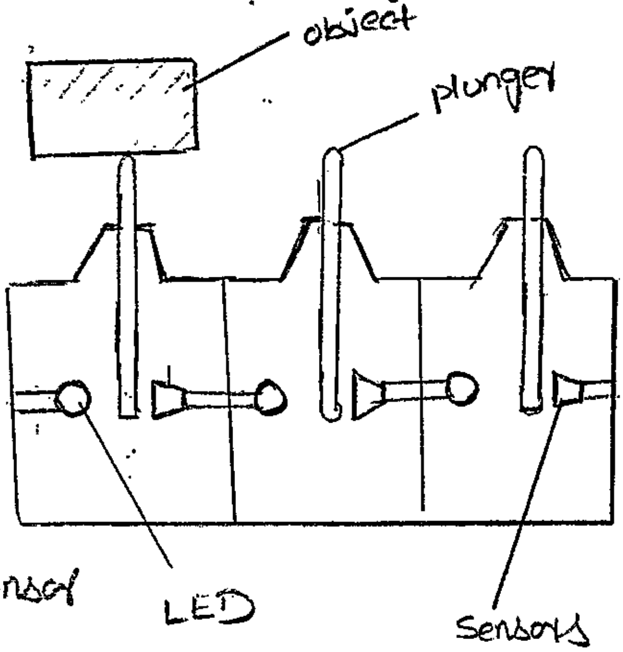
$$\boxed{F = K\delta}$$

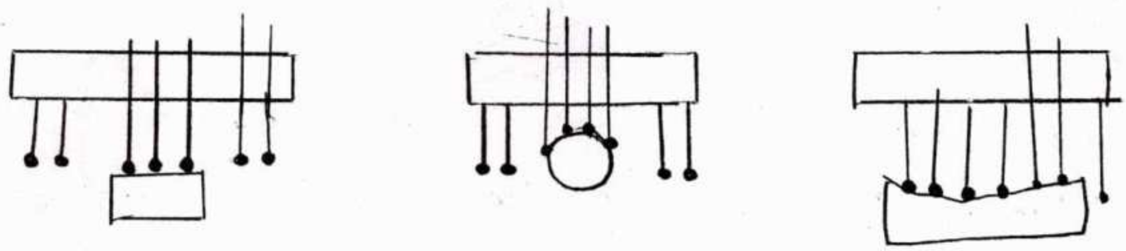
K = spring rate
δ = deflection of plunger recorded.

Tactile sensors:

An array of touch sensors arranged systematically to provide information about the contact of fingers with the object is called tactile sensors. The special tactile sensors also provide additional information like shape, size and the type of material of the objects.







proximity sensors :

The output of the proximity sensors gives an indication of the presence of an object with in the vicinity job operation. In robotics these sensors are used to generate information of object grasping and obstacle avoidance.

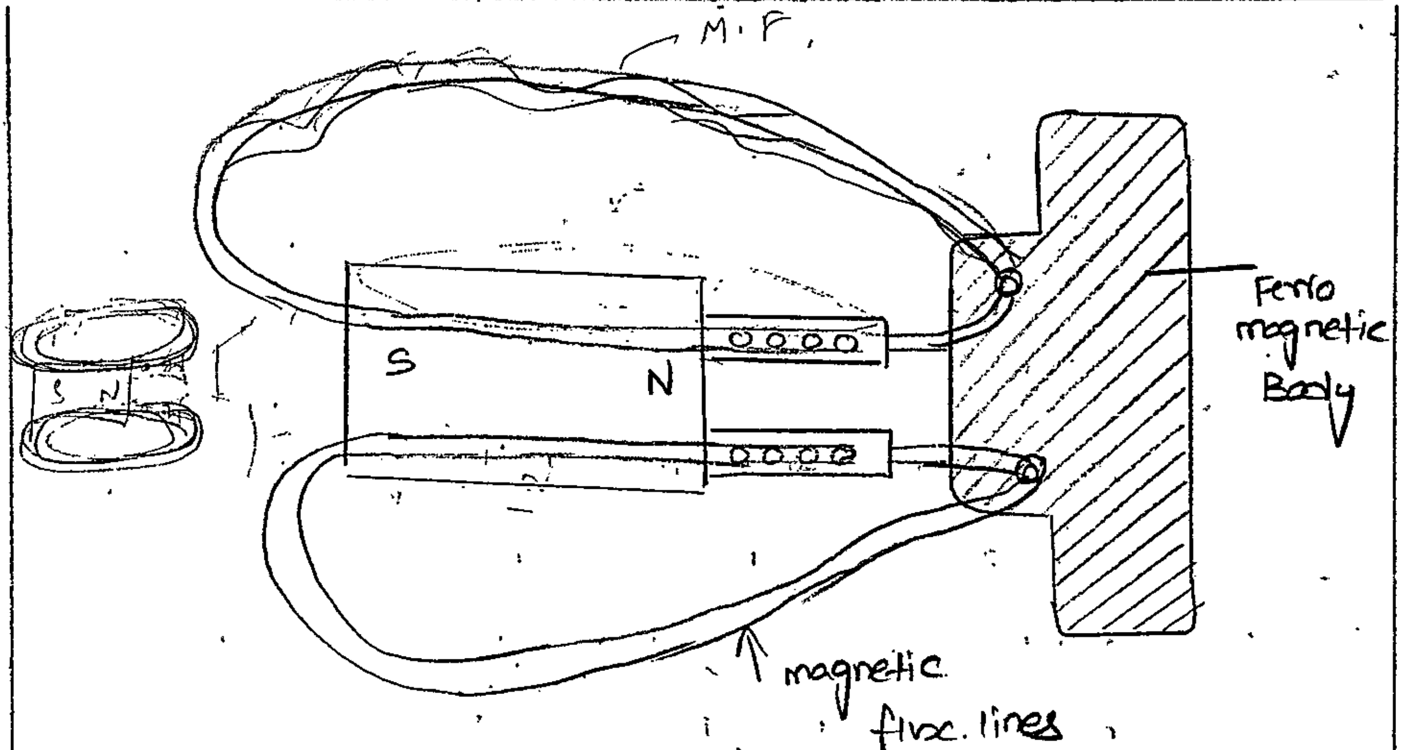
There are four types of proximity sensors.

They are : (1) Inductive sensor
(2) Hall Effect sensor
(3) ultrasonic proximity sensor
(4) optical sensor.

(1) Inductive sensor :

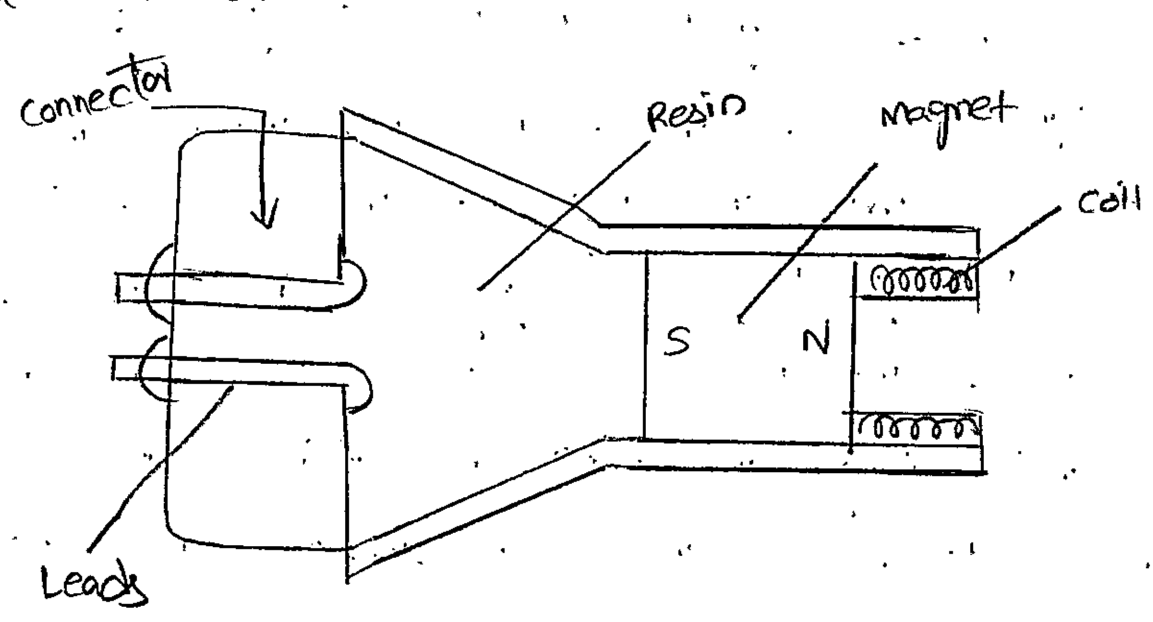
principle :

The ferromagnetic material when brought close to this type of sensor results in change in position of the flux lines of the permanent magnet leading to no change in inductance of the coil. The induced current pulse in the coil with change in amplitude and shape is proportional to rate of change of flux lines in magnet.

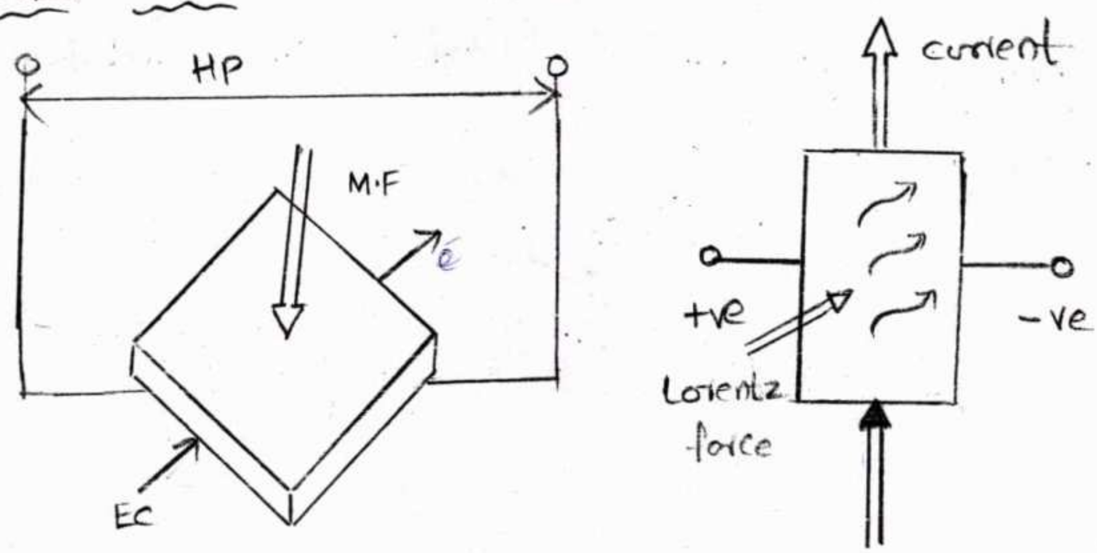


construction:

The proximity inductive sensor basically consists of wound coil located infront of a permanent magnet encased inside a rugged housing. The leads from the coil, embedded in resin is connected to the display through a connector.



Hall Effect sensor :

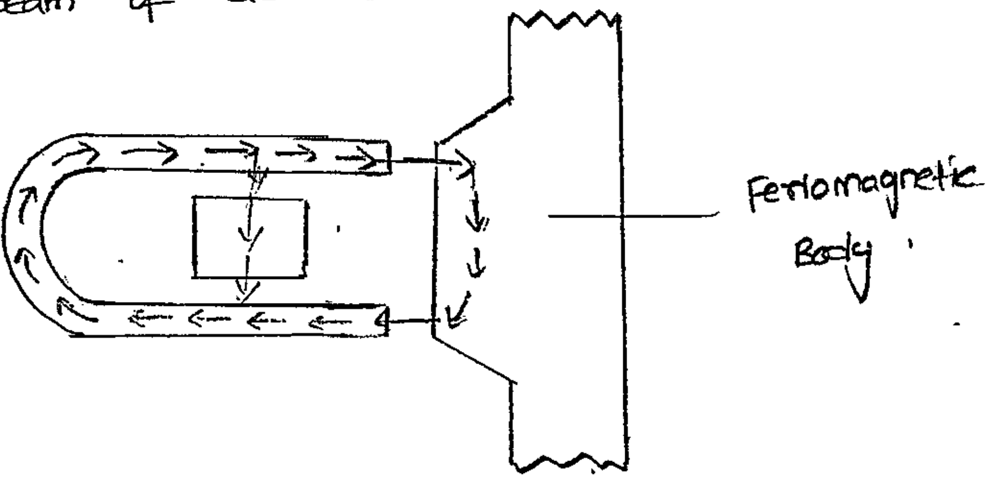


HP → Hall potential
MF → Magnetic field
EC → Electric current

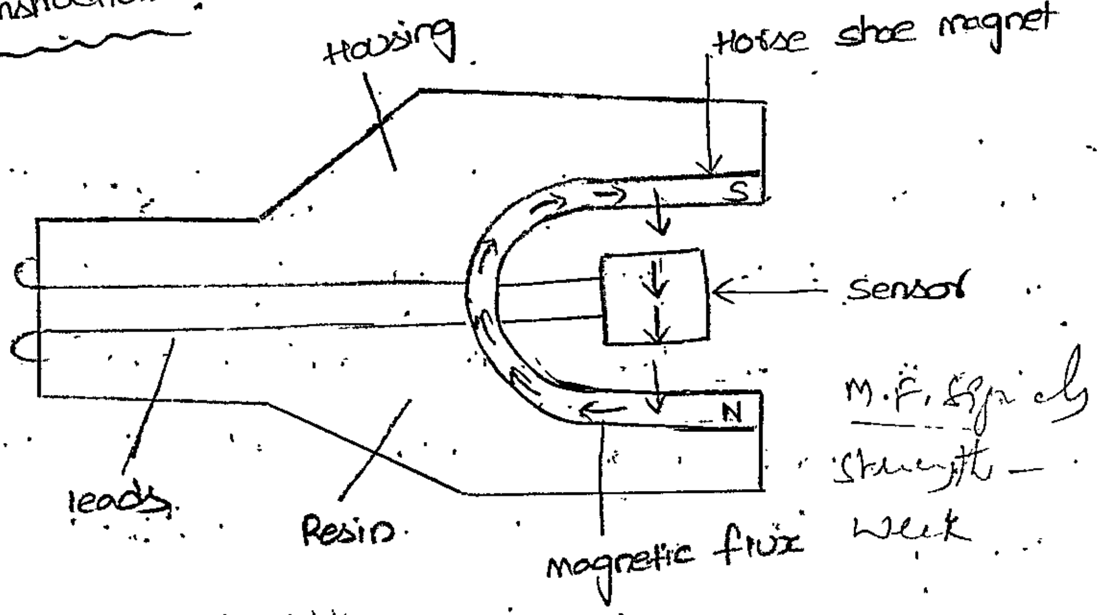
Principle :

Hall effect deals with the voltage between the two points in a conductor which changes by the near field of the magnetised or ferromagnetic material. The sensor experiences a weakend magnetic field in the close proximity of a ferromagnetic materials due to the bending of the flux lines of the magnet through approaching object.

Electrons are made to pass through a plate which is rectangular in shape and a magnetic field is applied at right angle to the plane of plate The electrons are deflected towards one side of the plate making that side negatively charged and other side positively charged. The force due to applied magnetic field is known as Lorentz force. The mechanism of deflection is governed by balance of Lorentz force and force on the beam of electrons.



construction :



A sensor element is stationed between the poles of a horse shoe magnet constructed inside a container.

The decrease in the strength of the magnetic field resulting due to the proximity of the object field reduces the voltage across the sensor. The sensor gives binary output for the decision making devices of control for forther actions

The silicon makes the ideal selection for a semiconductor interms of size, strength and capacity to electrical interference prevention.

Advantages :

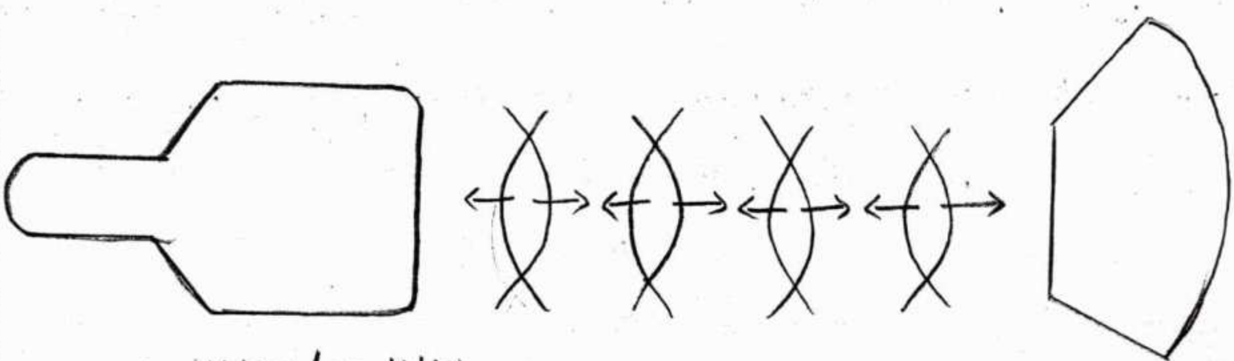
- They can operate as switches at high frequency.
- They cost less than electromechanical devices
- They are free from contact bounce problem.
- They can be used under severe environmental service conditions as they are immune to environme-ntal contaminations.
- They can be used as proximity, position and displacement sensors.

ultrasonic proximity sensor :

principle :

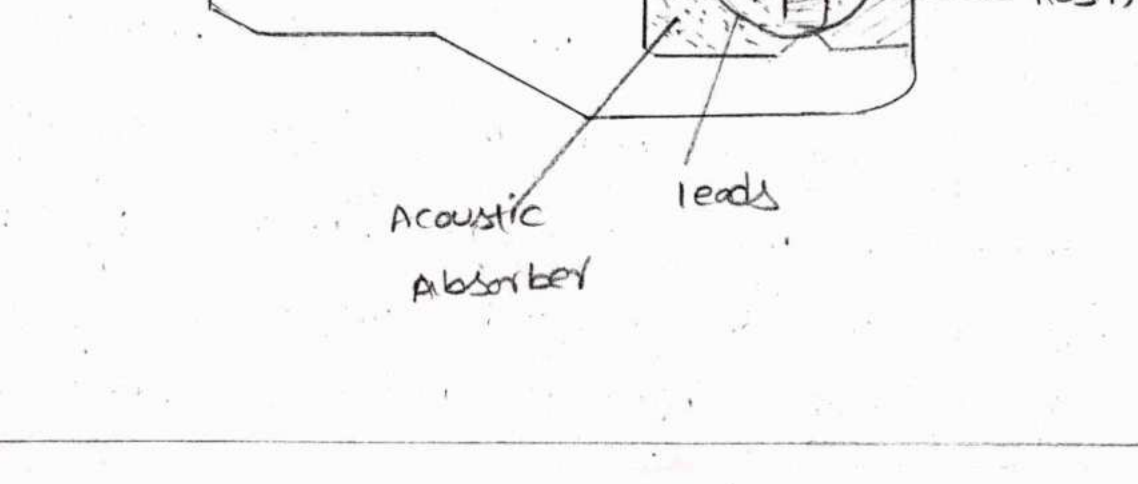
The acoustic waves emitted by the sensors reach the object and get reflected and receivers the waves to generate the information about the presence of the object. This type of operation is the echomode type. When the sensor acts only as the transmitter the waves get blocked by the presence of the object and the receiver gets no no signal. This type is known as opposed mode

Application → Submarines
→ Speed of ~~speed~~ wind
→ ~~Amount of liquid from tank~~



Construction :

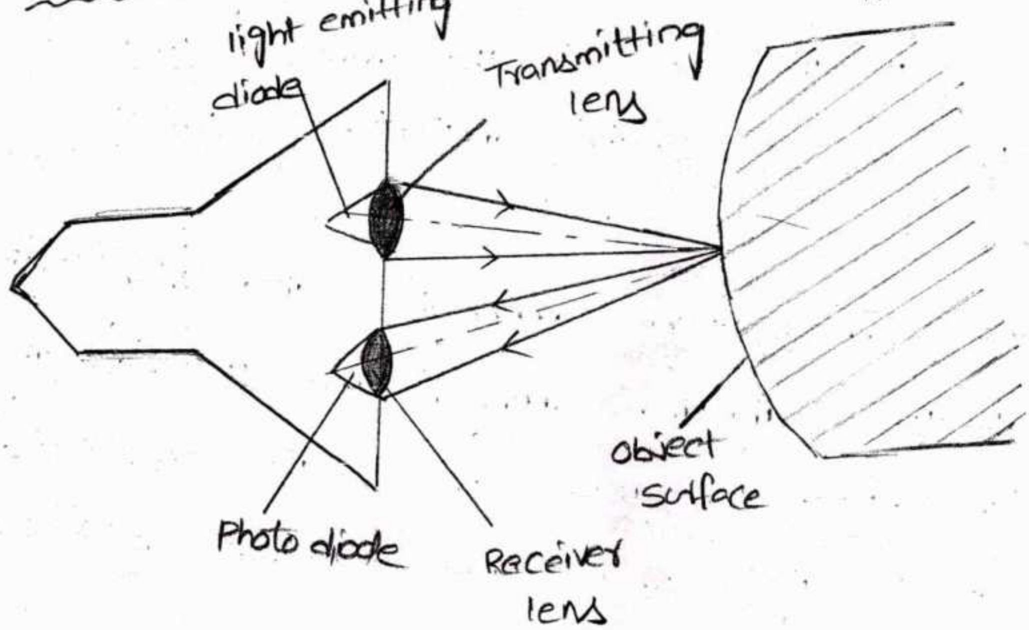
The main part in this type of sensor is the Transducer which can act both as transmitter and receiver. The sensor is covered by a resin block which protects from dust and humidity. For acoustic damping, absorber material is provided. Finally a metallic housing gives general protection.



Optical sensor:

Optical sensors are similar to ultrasonic sensors. The proximity of the object is detected by the action of the travelling light wave as it propagates from the transmitter and reflected by object towards the receiver.

Construction:



The light emitted by a diode is focussed by the transmitter lens on to the object surface. The reflected light waves travel back and received by the solid state photo diode, through a receiver lens. When the object is with in the range of the sensor it is possible to detect the presence of the receiver. The range is defined by the position and orientation of the object and the focal length of the sensor lens.

Range sensors :-

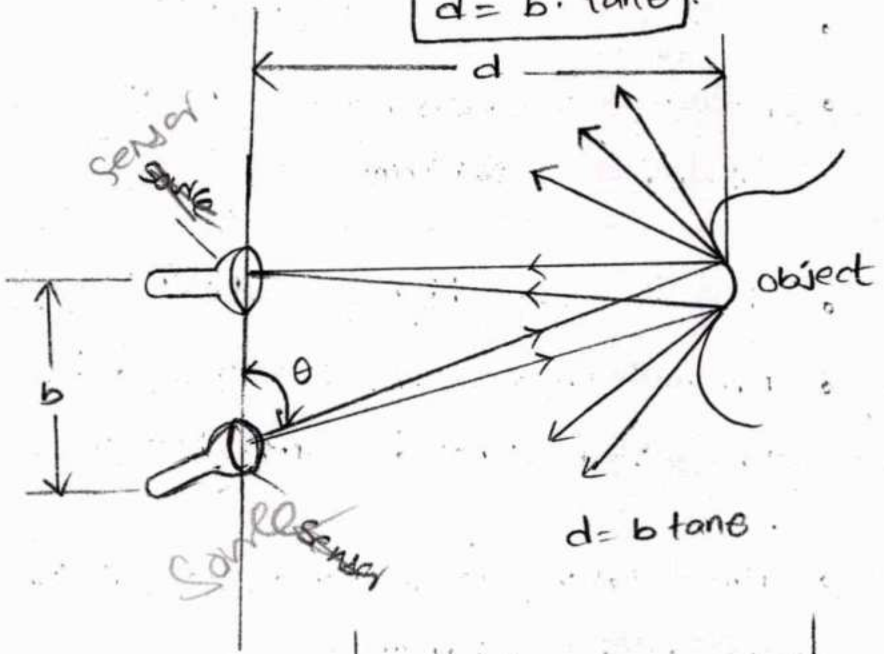
The distance between the object and the robot hand is measured using the range sensors within its range of operation. the calculation of the distance is by visual processing / use: Range sensors find use in robot navigation and avoidance of the obstacles in the path. The exact location and the general shape characteristics of the part in the work envelope of the robot is done by special applications for the range sensors.

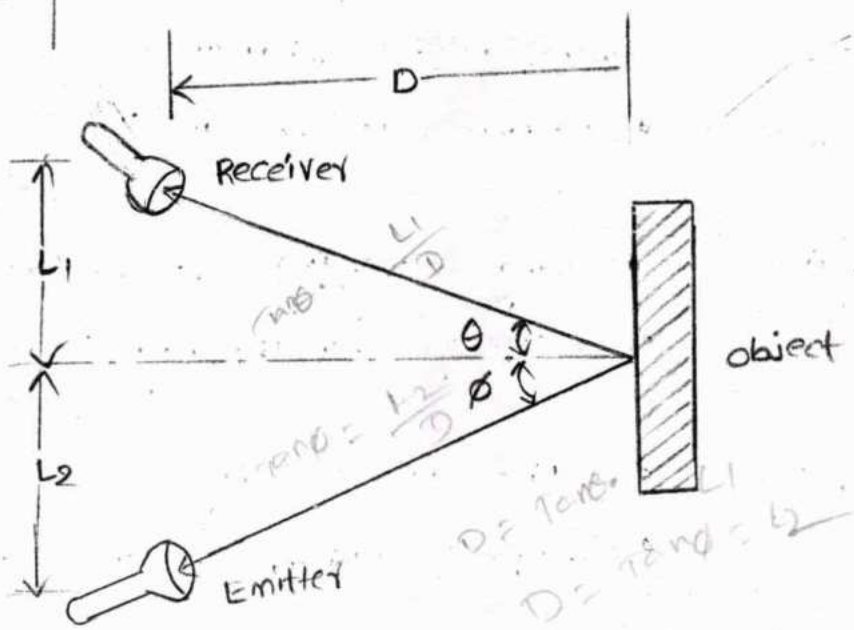
There are several approaches like Triangulation method, structured lighting approach and time-of-flight range finders etc. In these cases the source of illumination can be light-source, laser beam or based on ultrasonics.

Triangulation Method :

This is the simplest of the techniques, which is easily demonstrated in the figure. The object is swept over by a narrow beam of sharp light. The sensor focused on a small spot of the object surface detects the reflected beam of light. If 'θ' is the angle made by the illuminating source and 'b' is the distance between source and the sensor, the distance 'd' of the sensor on the robot is given as

$$d = b \cdot \tan\theta$$





$$D = \frac{(L_1 + L_2)\tan\theta \cdot \tan\phi}{\tan\theta + \tan\phi}$$

Desirable features for sensors and Transducers :-

Features	Functions
(1) precision	• should be as high as possible • Deviation in measurement reading should be minimum.
(2) Accuracy	• should be as high as possible • Error between sensed value and actual value should approach zero
(3) Speed of Response	• Time taken to respond to variation should be minimum • Response to be ~~instantaneous~~ instaneous
(4) operating range	• Range operating to be wide • Accuracy over the range to be acceptable
(5) Reliability	• Life to be high • frequent failures are not acceptable
(6) Calibration	• Should be easy to calibrate • Drift to be minimum • Should take less time to calibrate without much trouble.
(7) Cost and Ease	• The cost for Purchase should be low. • Installation & operation should be easy & less costly

Application of sensors in Robot :

(1) object identification

(2) Recognising the size, shape, material of the object

(3) Locating the object.

(4) Motion control variable detection

(5) Robot guidance without obstruction.

(6) Force and torque control needed for the task manipulator

(7) Safety monitoring.

(8) part inspection for quality control.

(9) Determining the position and related information about the object in the robot cell.

//

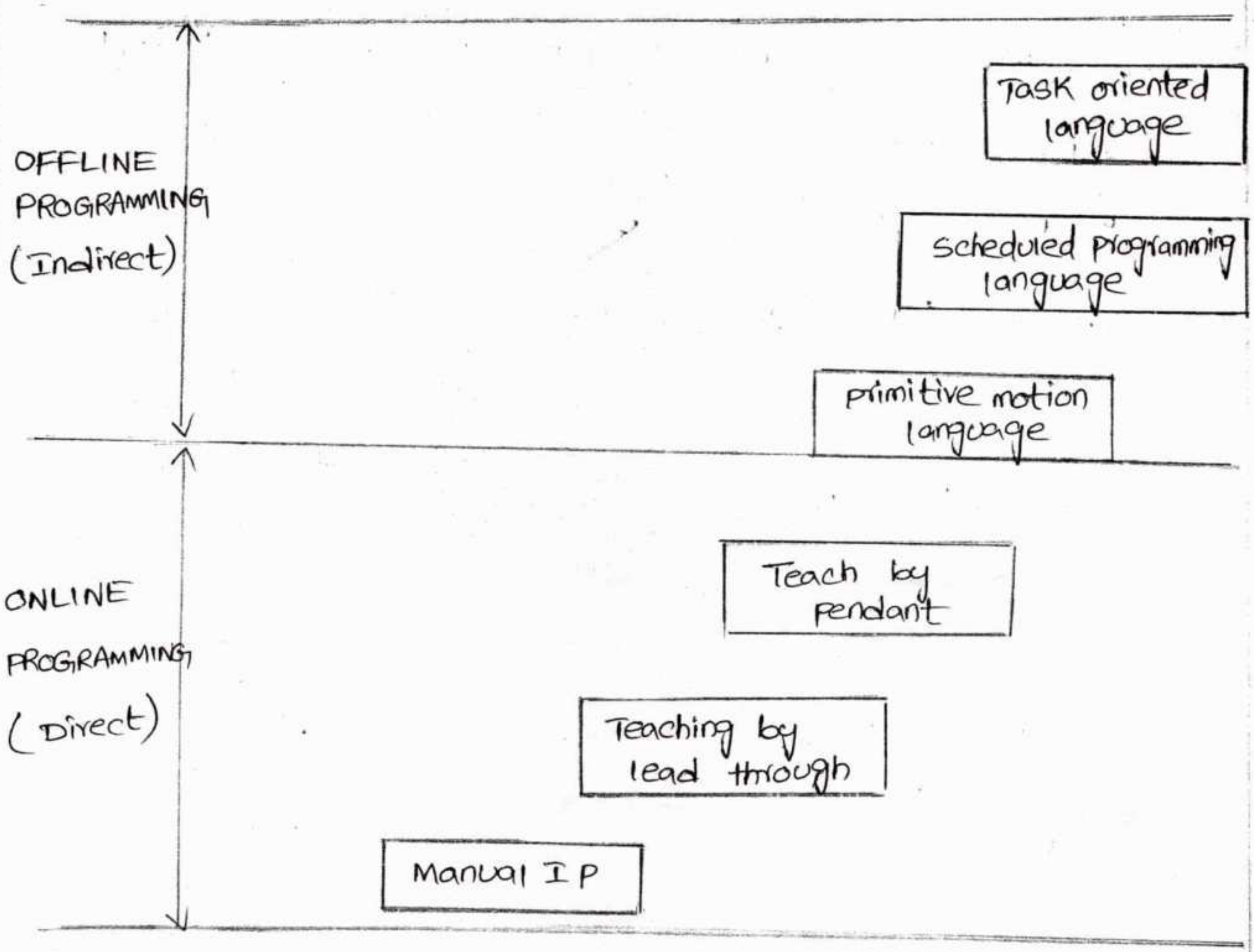
UNIT - VIII

ROBOT PROGRAMMING

Robot programming:

A robot program can be defined as a path in space to be followed by the manipulator combined with the peripheral actions that support the work cycle.

The process of robot programming includes teaching it the task to be performed, storing the program, executing the program and debugging it. Robot programming is similar to real-time programming in that the programs must be interrupt driven and take account of limited resources.



Programming types :

(1) Lead through programming (ON LINE)

(a) manual lead through programming

(b) powered lead through programming

(2) Textual languages (OFF LINE)

(1). Lead Through Programming : (on-line)

Lead-through programming requires the operator to move the robot arm through the desired motion path by a teach procedure, thereby entering the program in to the controller memory. There are two methods of performing the lead-through teach procedure

- Powered lead - through
- Manual lead - through

(a) powered lead through programming :-

The powered lead through method makes use of "teach pendant" to control the various joints, motions and to power drive the robot arm and wrist through a series of points in space. Each point is recorded in to the memory for subsequent Playback during the work cycle.

The teach pendant is usally a small hand held control box with a combinations of toggle switches, dials and buttons to regulate the robot physical movements and programming capabilities.

Among the various robot programming method the powered lead through method is the most common today. It is largely limited to point-to-point motion rather than continous motion because of the difficulty in using the teach pendant to regulate geometric motion in space.

Applications :

- Transfer task
- Machine loading and onloading
- Spot welding

(b) Manual lead through programming method :

It is also called as walk through method. It is more readily used for continous path programming where the motion cycle involves smooth complex curve linear movements of the robot arm.

This programming method requires the operator to physically grasp the end-of-arm or tools attached to the arm and manually move through the motion sequence, recording the path in to the memory.

Applications :

- spray painting .
- Arc welding
- continous operations .

Advantages of ON-LINE programming :

* Easy and quick to implement .
* operator is in continous touch with the working environment . So it is easy to avoid obstacles .
* No need of skilled operator .

Disadvantages :-

- Not possible to integrate with CAD/CAM , CIM and FMS systems .
- Not suitable for more no of motions .

(2) Textual languages (OFF LINE)

Robot languages have been developed for easy control of motion of robot having different structure and geometrical capabilities . Some of the robot languages have been developed by modifying the existing general purpose computer languages and some of them are written in a completely newstyle.

Types of Robot languages :

ACL : The advanced command language (ACL) is a robot language that employs a user friendly conversational command environment . Yaskawa robots use it .

APT : The Automatically programmed Tools (APT) language is a computer language dealing with motion . It was developed by Electronic systems Labaratory of MIT in 1956 .

AL : The AL (Arm Languge) high level programming language was developed at the robotics research centre of stanford university .

AML : The AML high level programming language was developed for use with the IBM RS 1 / assembly robot . A mfg language

HELP : HELP is a high-level programming language developed for use with General Electric's Allegro assembly robot .

Wave : Wave was the first high level language created for programming a robot . stanford Artificial labaratory developed it in 1973 .

VAL : VAL (Victor Assembly language) is a high-level programming language developed for PUMA lines of robot.

MML : MML was a model-based mobile robot language which was developed at the university of California. It is a high level offline programming language.

MCL : MCL is short for manufacturing control language and was developed by McDonnell Douglas for the U.S. Air Force's ICAM project.

RAIL : RAIL is a high level programming language developed by Automatrix for use with robots and vision systems.

IBL : IBL (Instruction Based Learning) is a method to train robots using natural language instructions. IBL uses unconstrained language

Some of the robot languages are VAL, AML, MCL, AL, APT, HELP

APT – Automatically programmed Tools
AL – Assembly language
AML – A manufacturing language
MCL – manufacturing control language.
VAL – Victor's Assembly language.
AUTOPASS – a IBM language (Instruction Based Learning)

Advantages of OFFLINE programming:

- Easy to stimulate
- Verification of program through simulation and visualization.
- Reuse of existing CAD data

Disadvantages:

- Needs skilled person and extensive training.
- No chance of avoidance of obstacles in the middle of the program execution because operator may not be in continous contact with the environment.

VAL commands :

(1) Motion control statements :

APPRO P1, Z1 — command to approach the point P1 in the z direction by Z1 distance above the object.

MOVE P1 — command to move the arm from the present position to point P1.

MOVE P1 VIA P2 — Asks the robot to move to point P1 through point P2.

DMOVE (J1 ΔX) — moves the Joint J_1 by an increment of ΔX (linear)

DMOVE (J1, J2, J3) (dα, dβ, dθ) — command to move joints J_1, J_2 and J_3 by incremental angles of dα, dβ, dθ respectively.

(2) Speed control commands :

SPEED V IPS — The speed of the end effector is to be V inch per second at the time of program execution.

SPEED R — command to operate the arm endeffector at R percent of the normal speed at the time of program execution.

(3) Position control commands :

HERE P1 — Defining the name of a point as P1 .

DEFINE P1 = POINT ($x, y, z, w_\alpha, w_\beta, w_\theta$) — The command defines the point P1 with x, y, z coordinates and $w_\alpha, w_\beta, w_\theta$ the wrist rotation angles

Path control : DEFINE PATH 1 = PATH (P_1, P_2, P_3) — The path of the end effector is defined by the connection between points P_1, P_2 and P_3 in series .

MOVE PATH 1 — Movement of the end effector along path 1 .

Frame definition :

DEFINE FRAME 1 = FRAME (P_1, P_2, P_3) — Assigning variables name to FRAME 1 defined by points P_1, P_2 and P_3 . P_1 - origin , P_2 - point along x axis and P_3 - point along XY plane .

MOVE ROUTE : FRAME 1 — Defines the movement in the path for frame 1 .

(4) End effector operation commands :

OPEN — opens the gripper fingers

CLOSE 50 MM — Informs gripper to close keeping 50 mm width between the fingers .

CLOSE 5 LB — Applies 5 Lb gripper force .

CENTER - closes the gripper slowly till the establishment of contact with the object to be gripped.

OPERATE TOOL - (SPEED N RPM) positioning and operating the powered tool. Here the EE is replaced by servo powered tool.

(5) operation of the sensors:

SIGNAL 4, ON - The command actuates the output port 4 and turns at certain stage of the program.

SIGNAL 5, OFF — The output port 5 is turned off.

WAIT 13, ON The device gives a feed back signal indicating that it is on.

REACT 16, SAFETY - The change in signal (if any) in the input line 16, should be deviated to the sub-routine SAFETY.

Robot programming motions : (Manipulator path control)

In controlling the manipulator we are not only interested in end points reached by the robot joint but also in the path followed by the arm in traverling from one point to another in the work space.

There are 3 common types of motion that a robot manipulator can make in travelling from point to point. They are (1) slew motion

(2) Joint interpolated motion

(3) straight line motion.

(1) Skew motion :

It represents the simplest types of motions. As the robot is commanded to travel from point A to point B, each axis of the manipulator travels as quickly as possible from its respective initial position to its required final position. Therefore all axes begin moving at the same time, but each axis ends its motion in an ellapsed time that is proportional to the product of its distance moved and its top speed.

This motion generally resorts in unnecessary wear on the joints and often leads to unanticipated resorts in terms of the path taken by the manipulator.

(2) Joint - Interpolated motion :

Joint interpolated motion requires the robot controller to calculate the amount of time it will take for each joint to reach its destination at the commanded speed. It then selects the maximum time among these and uses this value as the time for all the axes. This means that a seperate velocity is calculated for each axis.

The advantage of Joint Interpolated motion over slew motion is that the joints are generally driven at less than their respective maximum velocities thus reducing maintenance problems for the robot. Also the path that is followed by the manipulator is repeatable and predictable regardless of the total time and commanded velocity.

(3) Straight - line motion :

In straight line motion it requires the end of the manipulator to travel along straight path defining cartesian coordinates. This is the most demanding type of motion for the controller to execute for a cartesian coordinate (LLL) robot.

For manipulators with rotational joints, most straight line motions are unnatural and the controller must compute the sequence of incremental joint required for the end of the arm to move in a linear fashion.

Applications of straight line motion is useful in operations such as arc welding, laying adhesives along a straight path, spray painting and inserting a peg in to a hole in an assembly operation.

WAIT, SIGNAL AND DELAY COMMANDS:

All industrial robots can be instructed to send signals or wait for signals during execution of the program. These signals are sometimes called interlocks. The most common form of interlock signal is to actuate the robot's end effector.

To control the gripper, robots are typically coordinated with other devices in the cell also. To accomplish this coordination, we introduce two commands that can be used during the program. The first command is SIGNAL M.

SIGNAL M: It instructs the robot controller to output a signal through line M where M is one of several output lines available to the controller.

The second command is WAIT N.

WAIT N : It indicates that the robot should wait at its current location until it receives a signal on line N, where N is one of several input lines available to the robot controller.

A WAIT instruction can be programmed to accomplish the feed back.

The other alternative command is DELAY X SEC

DELAY X SEC : This command is used to cause the robot to delay before proceeding to the next step. The robot would be programmed to wait for a specified amount of time to ensure that the operation had taken place.

This command has a length of time as its argument rather than an input line.
This command mainly indicates that the robot should wait X seconds before proceeding to the next step in the program.

94 A3 | 322 327 328 330 | 305 350
A0 B1 A1

Branching :

Most controllers for industrial robots provide a method of dividing the program in to one or more branches. Branching allows the robot program to be sub divided in to convenient segments that can be executed during the program.

A branch can be thought of as a subroutine that is called one or more times during the program. The subroutine can be executed either by branching to it at a particular place in the program or by testing an input signal line to branch to it.

Branching method is frequently used on spray painting robots which have been programmed to paint a limited variety of parts moving past the workstation of a conveyor.

photoelectric cells are frequently employed to identify the part to be sprayed by distinguishing between the geometric features of the different parts.

A relocatable branch allows the programmer to specify a branch involving a set of incremental points in space that are performed relative to some defined starting point for the branch.

① ✓ Shift from teacher centric to learner centric

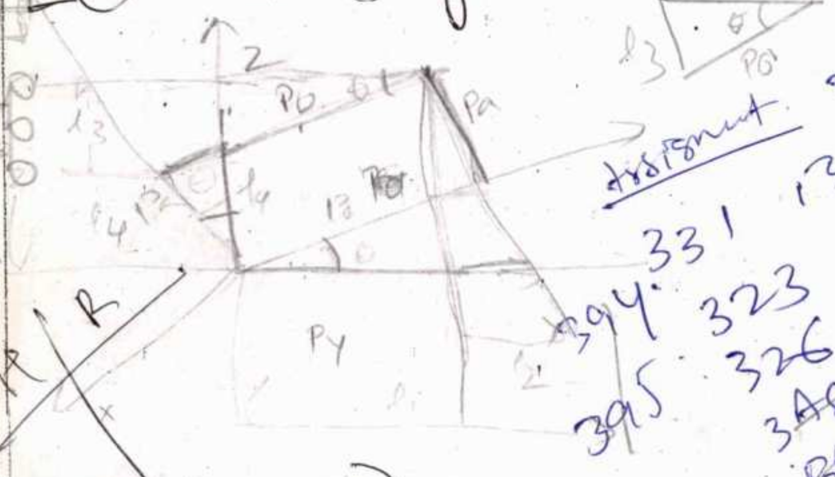
✓ Teachers changing to-tech

② the mythic 25th hour

③ meeting synchronous (either face-to-face election)

④ User story

90 91 92 95 A3 A8 320 325 228 329 330 332 333 350 325



assignment 398 391 390 389 331 323 326 394 395 3A0 3A1 A3 3A8 3A6 399 331, 370, 333 330 315 303 391 307 3A4 328

14/10 (AN)

(FN)

B2 B3 B4 B6 D0 D6 D7 D9 C0 C2 E3 E6 E9 C4 C5 C7

B2 B3 B4 B6 C2 C3 C5 D6 D7 C4 330 D4 C6 322 E9 C7 F0 D0 E9 335 338

LE-325 332

shah 328

$l_3 = P_0 \sin\theta$

$l_1 = P_0 \cos\theta$

$l_2 = P_a \sin\theta$

$P_y = l_1 - l_2$

$= P_0 \cos\theta - P_a \sin\theta$

$P_z = l_3 + l_4$

$= P_0 \sin\theta + P_a \cos\theta$

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$