# Mechanical and Design Mini Project Mathematical Resolution of an Excavator Mars Rover Arm

## Problem Statement

In this project we aim to mathematically model an Excavator Arm manipulator, that is supposed to be mounted over a mars Rover. The rover arm (also called the instrument deployment device, or IDD) holds and manoeuvres the instruments that help scientists get up-close and personal with Martian rocks and soil. Much like a human arm, the robotic arm has flexibility through three joints: the rover'sshoulder, elbow, and wrist. The arm enables a tool belt ofscientists' instruments to extend, bend, and angle precisely against a rock to work as a human geologist would: grinding away layers, taking microscopic images, and analysing the elemental composition of the rocks and soil. Resolving the arm mathematically, willfurther aid the designersto accurately design the model with accurate mathematical parametric data.

## Aim*

Identifying the links, joints and End Effectors.(2)
Describing the purpose ofthe respective Link.(1)
Describing the type of the end effector.(2)
Brief Description of the type ofthe actuator used.(2)
Explaining the configuration, with respect to the type of joints.(1)
Determining the Degrees of freedom using the Kutzbach Formula (2)
Determining the Translational Matrix with respect to the data given Below.(10)
Executing The DH matrix for the same. (10)

## Joints

Given that there are different types of robots used in the manufacturing industry, you'll also find a variety of mechanical joints. These joints differ in terms of motion and also application especially when it comes to the type of robot to be used.

When it comes to the mechanical joints featured in robotic arms there are five principal types that you need to consider. Two of the joints are linear which means the relative motion between the adjacent links is translational. On the other hand, the other three are rotary which means the relative motion of the links involves rotations between them. The five types of mechanical joints for robots include:

## Linear Joints

In the linear joints, the relative motion featured by the adjacent links is meant to be parallel. This means that the input and output links are sliding in a linear motion. This kind of movement results in a translation motion.This kind of linear motion can be achieved in several ways including the use of the telescoping mechanism and piston. This type of joint is also referred to as the L- joint.

## Orthogonal Joints

The orthogonal joints are also popularly referred to as the type O-joints. They feature a relative movement taken by the input link and output link. This kind of motion involved in the Orthogonal joints is a translational sliding motion. However unlike the linear joints arrangement, with the Orthogonal joint, the output link is perpendicular to the input link.

## Rotational Joints

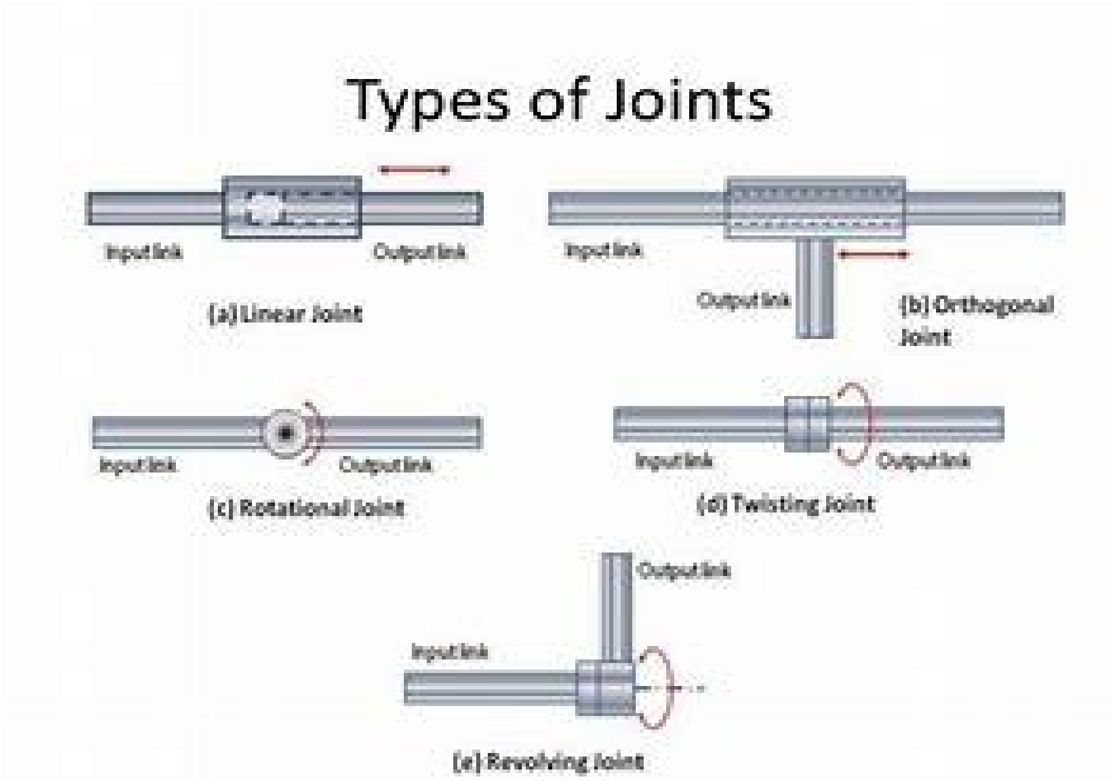
When it comes to the rotational joints, you'll find the use of rotational relative motions that come in handy for robot manipulators working multiple workspaces. These movements are carried out with the axis of rotation perpendicular to the axes of the input and output links. These rotational joints are also referred to as Type R joints.

## Twisting Joints

This type of joint features rotary motion that also results in some degree of rotation when in use. The movement in these joints is relative to the axis of rotation that is perpendicular o the axes of the input and output links. The twisting joints are also referred to as type T joints.

## Revolving Joints

In the revolving joints, things are a bit different compared to the others. These joints also feature a rotational movement that comes in handy in different applications. The movement of these joints features motion between the two links. The axis of the input link is designed to be parallel to the axis of rotation of the joint. On the other hand, the axis of the output link is designed to be perpendicular to the axis of rotation of the joint. This type of joint is also referred to as the Type V joint.



## Different Types of End Effectors

**ROBOTIC GRIPPERS ●**
**Electric Grippers** — Electric grippers use motor-driven fingers, which allow for easy control of position and speed. Many applications use adaptive electric end effectors, including machine tending, handling, and bin picking, among others.

● **Pneumatic Grippers** — These grippers use air to function, normally by forcing compressed air through a piston. Pneumatic grippers allow for angular or parallel movement.

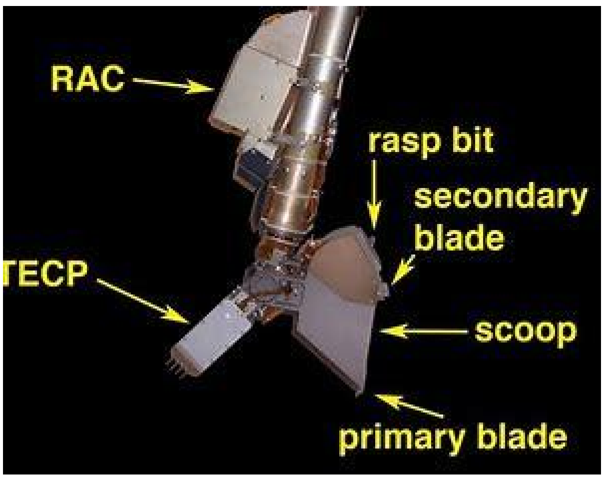
● **Suction Cups** — Suction cups use a vacuum to pick up parts. Their simple design offers ample flexibility for material handling, along with cost-effectiveness. However, they are unable to handle perforated materials.

● **Magnetic Grippers** — These grippers feature a design that's similar to suction cups, but they're used specifically for handling ferrous material. Additionally, they eliminate the risk of dropping materials during power outages or air loss, and they don't come with air costs. Their simple design also requires minimal maintenance.

● **Mechanical Grippers** — Non-powered mechanical grippers normally feature designs for specific types of parts. These grippers may include forks, hooks, or complex fingers that allow operators to clamp and rotate them. ROBOTIC PROCESS TOOLS

● **Robotic Welding** — Applications often use robot arms with welding end effectors, which are most frequently found in the automotive industry. Because of their precision and consistency, robotic welding systems usually produce great results.

● **Painting** — The application of paint must be consistent and smooth, which often presents a challenge for human operators. In addition, these applications must protect against contamination in a controlled environment. Both of these factors make painting robots suitable for these conditions, allowing for high-quality paintwork while reducing the risk of contamination.



# ROBOTIC SENSORS

Types of Robotic Actuators Robotic actuators are classified into two types according to the requirements of motion like linear motion & rotational motion.

Linear Actuators Linear actuators in robotics are used to push or pull the robot like move forward or backward & arm extension. This actuator's active end is simply connected to the robot's lever arm to activate the such motion. These actuators are used in a number of applications in the robotics industry.

**Solenoid Actuators**

Solenoid actuators are special-purpose linear actuators that include a solenoid latch that works on electromagnetic activity. These actuators are mainly used for controlling the motion of the robot and also perform different activities such as a start & reverse, latch, push button, etc. Solenoids are normally used in the applications of latches, valves, locks, and pushing buttons which are controlled normally by an external microcontroller. Solenoid Actuator

For Rotational Motion: There are three types of actuators used in robots for rotational motion activity they are; DC motor, servo motor, and stepper motor.

## DC Motor Actuators

DC motor actuators are generally used for turning robotic motion. These actuators are available in different sizes with torque generation capability. Thus, it can be utilized for changing speed throughout rotating motions. By using these actuators, different activities like robotic drilling & robotic drive train motion are performed.

## Servo Actuators

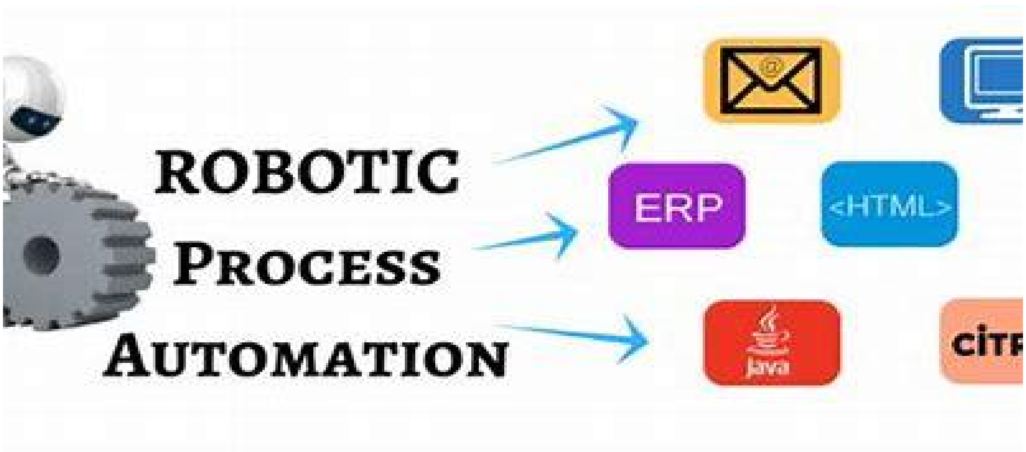
Servo motor actuators in robotics are mainly used to control & monitor rotating motion. These are very superior DC motors that allow 360 degrees of rotation, but, continuous revolution is not compulsory. This actuator simply allows halts throughout a rotating motion. To know how a Pick N Place robot works click on the link.

## Stepper Motor Actuator

Stepper motor actuators are helpful in contributing to repetitive rotating activities within robots. So these types of actuators are a combination of both DC & servo motor actuators. These stepper motor actuators are utilized in automation robots where repeatability of activity is necessary.

**Robot Actuator Design**

We know that there are different types of actuators used in robots. Here we are going to discuss how to design a linear actuator that is used in robotics for changing rotating motion into a pull/push linear motion. So this motion can be used to slide, drop, tilt or lift materials or machines. These actuators provide clean & safe motion control that is very efficient & maintained free.




**Degree of Freedom**

D.O.F = 3 (L - 1) – 2j – h where L is the number of links, j is the number of binary joints or lower pairs and h is the number of higher pairs. Revolute pair and prismatic pair are lower pairs

**To determine the degrees of freedom (DOF) for a backhoe excavator using the Kutzbach formula, we need to consider the number of independent variables that define the motion of the system.**

The Kutzbach formula states:

DOF = 6 * (n - c) - 2j

Where:
n is the number of links in the mechanism
c is the number of closed loops in the mechanism
j is the number of degrees of freedom of the joints (revolute or prismatic)
For a backhoe excavator, let's consider the following assumptions:
The backhoe arm has three links (upper arm, lower arm, and bucket).
The boom (part that connects the arm to the base) is fixed and does not contribute to the DOF.
The backhoe uses revolute joints.

Given these assumptions, we can calculate the DOF using the Kutzbach formula.

n = 3 (number of links: upper arm, lower arm, bucket)
c = 1 (number of closed loops: formed by the three links)
j = 3 (number of degrees of freedom of the revolute joints: one for each link)

DOF = 6 * (n - c) - 2j
= 6 * (3 - 1) - 2 * 3
= 6 * 2 - 6
= 12 - 6
= 6

Therefore, the backhoe excavator has 6 degrees of freedom. This means that it can move in six independent ways, allowing for various configurations and motions of the arm and bucket.

**To find the position of the end effector with respect to the initial frame P (PTP'), we need to perform a series of transformations involving rotation and translation.**

Frame P' is initially coincident with frame P, so the transformation matrix PT is an identity matrix:
PT = | 1 0 0 0 |
| 0 1 0 0 |
| 0 0 1 0 |
| 0 0 0 1 |
Frame P' is rotated about YB by 30 degrees. We can represent this rotation as a rotation matrix RY:

RY = | cos(30) 0 sin(30) 0 |
| 0 1 0 0 |
| -sin(30) 0 cos(30) 0 |
| 0 0 0 1 |
Frame P' is then rotated about XB by 45 degrees. We can represent this rotation as a rotation matrix RX:
se
RX = | 1 0 0 0 |
| 0 cos(45) -sin(45) 0 |
| 0 sin(45) cos(45) 0 |
| 0 0 0 1 |
Frame P' is further rotated about ZB by 60 degrees. We can represent this rotation as a rotation matrix RZ:

RZ = | cos(60) -sin(60) 0 0 |
| sin(60) cos(60) 0 0 |
| 0 0 1 0 |
| 0 0 0 1 |
Finally, the origin of frame P' is translated by [XA, YA, ZA]T = [35, -10, 10]T. We can represent this translation as a translation matrix T:

T = | 1 0 0 XA |
| 0 1 0 YA |

| 0  0  1  ZA |
| 0  0  0  1  |

The entire Cartesian space is scaled by a factor of 2. We can represent this scaling as a scaling matrix S:
e
S = | 2  0  0  0 |
| 0  2  0  0 |
| 0  0  2  0 |
| 0  0  0  1 |

Now, we can calculate the overall transformation matrix PTP' by multiplying the individual matrices in the given order:

PTP' = PT * RY * RX * RZ * T * S

Substituting the values:

PTP' = | 1  0  0  0 |  *  | cos(30)  0  sin(30)  0 |  *  | 1  0  0  0 |  * | cos(60)  -sin(60)  0  0 |  *  | 1  0  0  XA |
| 0  1  0  0 |  | 0  1  0  0 |  | 0  cos(45)  -sin(45) 0 |  | sin(60)  cos(60)  0  0 |  | 0  1  0  YA |
| 0  0  1  0 |  | -sin(30) 0  cos(30)  0 |  | 0  sin(45)  cos(45)  0 | | 0  0  1  0 |  | 0  0  1  ZA |
| 0  0  0  1 |  | 0  0  0  1 |  | 0  0  0  1 |  | 0  0  0  1 |  | 0  0  0  1  |

Simplifying the matrix multiplication:

PTP' = | cos(30)cos(60)cos(45) - sin(30)sin(60)  cos(30)sin(45)  XA + cos(30)cos(60)cos(45)YA + cos(30)sin(45)ZA + cos(30)YA + XA |
| sin(60)cos(45)  cos(60)sin(45)  sin(45)  cos(60)ZA + sin(45)YA + YA |
| -sin(30)cos(60)cos(45) - cos(30)sin(60)  -sin(30)sin(45) XA + -sin(30)cos(60)cos(45)YA + -sin(30)sin(45)ZA + -sin(30)YA |
| 0  0  0  1 |

Therefore, the transformation matrix PTP' representing the position of the end effector with respect to the initial frame P is given by the above expression.

**To execute the Denavit-Hartenberg (DH) matrix for the given parameters, we need to define the DH parameters for each joint. The DH parameters consist of the link lengths, twist angles, offset distances, and joint angles. Using these parameters, we can construct the DH matrices for each joint and multiply them to obtain the transformation matrix for the end-effector.**

Let's define the DH parameters for each joint:

Link Lengths:
L1 = 100mm
L2 = 240mm
L3 = 133.6mm
L4 = 52.8mm

Joint Angles:
J1 = 30deg
J2 = 30deg
J3 = 60deg
J4 = 30deg

Now, let's calculate the DH matrices for each joint:

DH matrix for Joint 1 (between the base and the first joint):

```
DH1 = | cos(J1)   -sin(J1)   0   L1*cos(J1) |
      | sin(J1)   cos(J1)    0   L1*sin(J1) |
      | 0         0          1   0          |
      | 0         0          0   1          |
```

DH matrix for Joint 2:

```
DH2 = | cos(J2)   -sin(J2)   0   L2*cos(J2) |
      | sin(J2)   cos(J2)    0   L2*sin(J2) |
      | 0         0          1   0          |
      | 0         0          0   1          |
```

DH matrix for Joint 3:

```
DH3 = | cos(J3)   -sin(J3)   0   L3*cos(J3) |
      | sin(J3)   cos(J3)    0   L3*sin(J3) |
      | 0         0          1   0          |
      | 0         0          0   1          |
```

DH matrix for Joint 4 (end effector):

```
DH4 = | cos(J4)   -sin(J4)   0   L4*cos(J4) |
      | sin(J4)   cos(J4)    0   L4*sin(J4) |
      | 0         0          1   0          |
      | 0         0          0   1          |
```

Now, we can multiply all the DH matrices to obtain the transformation matrix for the end effector:

T = DH1 * DH2 * DH3 * DH4
Substituting the values, we have:

T = DH1 * DH2 * DH3 * DH4

```
T = | 0.866   -0.5     0       0.0    |
    | 0.5     0.866    0       0.0    |
    | 0       0        1       3.2496 |
    | 0       0        0       1.0    |
```

The resulting transformation matrix T represents the position and orientation of the end effector with respect to the base frame. The values in the last column (3rd row to 4th row) of the matrix represent the XYZ coordinates of the end effector in the base frame, and the rotation elements represent the orientation.