DISSERTATION

on

Experimental and Analytical Modelling and Control of Precision Surgery Manipulator

Submitted in partial fulfilment of the requirement for award of the degree
of

Master of Engineering

in

CAD/CAM Engineering

Submitted By

AASHISH KUMAR

Roll No. 801181001

Under the supervision of

Dr. Tarun Kumar Bera
and
Mr. Devender Kumar
Assistant Professor
Department of Mechanical Engineering
Thapar University, Patiala

Mr. Sanjeev Verma
Principal Scientist

Biomedical Instrumentation Division
CSIR-Central Scientific Inst. Org.
Chandigarh

DEPARTMENT OF MECHANICAL ENGINEERING
THAPAR UNIVERSITY
PATIALA-147004, INDIA,
JULY 2014

This Thesis is dedicated to

my parents

DECLARATION

I hereby declare that the thesis entitled **"Experimental and Analytical Modelling and Control of Precision Surgery Manipulator"** in the partial fulfilment of the requirements for award of **Master of Engineering** degree in **CAD/CAM Engineering** in **Mechanical Engineering Department, Thapar University, Patiala,** is an authentic record of work carried out under the supervision and guidance of **Dr. Tarun Kumar Bera and Mr. Devender Kumar, Assistant Professor** of **Mechanical Engineering Department, Thapar University** and **Mr. Sanjeev Verma, Principal Scientist** of **CSIR-CSIO, Chandigarh.** This matter embodied in this report has not been submitted in part or full to any other university or institute for the award of any degree.

(Aashish Kumar)

This is to certify that above declaration made by the student concerned is correct to the best of our knowledge and belief.

Mr. Devender Kumar
Assistant Professor
Thapar University, Patiala

Mr. Sanjeev Verma
Principal Scientist
Biomedical Inst. Div.
CSIR-Central Scientific Inst. Org.
Chandigarh

Dr. Tarun Kumar Bera
Assistant Professor
Thapar University, Patiala

Countersigned by:

Head of the ME Department
Thapar University, Patiala

Dean of Academic Affairs
Thapar University, Patiala

ACKNOWLEDGEMENT

I would like to express a deep sense of gratitude and thank profusely to my supervisors **Dr. Tarun Kumar Bera, Mr. Devender Kumar and Mr. Sanjeev Verma** for their sincere & invaluable guidance, suggestions and attitude, which inspired me to submit thesis in the present form. Their dynamism and diligent enthusiasm have been highly instrumental in keeping my spirits high. Their flawless and forthright suggestions blended with an innate intelligent application have crowned my task with success.

I am also thankful to Head, Department of Mechanical Engineering for his encouragement and inspiration for execution of the thesis.

AASHISH KUMAR

Roll No. 801281001

ABSTRACT

In this fast growing world everyone is trying to be perfect and want to reach anywhere with as much as less time it takes. So, in the medical field also every patient want to get best treatment with high precision and accuracy. A robotic arm can provide both of them which can be controlled by computers and the inputs given by a specialise doctor sitting on another table or may be somewhere else in the world. The robotic surgery enables surgeons to perform even the most complex and delicate procedures through very small incisions. A robotic surgery provide patient less time to recover, less blood loss.

- Designing of manipulator for the considerations of functions it performs, its limitations and motor housing.
- Complete robotic manipulator's construction in Solid Works software with all the attachment of motors and how the motion transferred to different joints by the use of ropes or belts is designed.
- Forward modelling and inverse modelling using Bond Graph of robotic manipulator systems are developed.
- The forward kinematics and inverse kinematics of the robotic manipulator by using Denavit-Hartenberg (D-H) representation and Newton-Rapson method by using Mat lab tool of serial manipulator.
- Comparison of the results comes from both methods *i.e.,* Bond Graph approach and D-H approach.
- Its back hand system which is to transfer motions to joints and by which motor is used in here with the all construction of it in Solid Works and by use of Rapid Prototyping Machine.

Keywords: Surgical Robot, Serial Manipulator, D-H Algorithm, Bond Graph, Fabrication.

LIST OF ABBREVATIONS

ASIMO	Advanced Step in Innovative Mobility
C	Capacitance
De	Effort Detector
Df	Flow Detector
D-H	Denavit-Hartenberg
DOF	Degree of Freedom
ED&H	Extended Denavit and Hartenberg
F	Force
FPGA	Field Programmable Gate Array
GY	Gyrator
HD	High Definition
I	Inductance
J	Jacobian
KUKA	Keller und Knappich Augsburg.
MGY	Modulated Gyrator
MIS	Minimal Invasive Surgery
PID	Proportional Integrated Derivative
PUMA	Programmable Universal Manipulation Arm
R	Resistance
RUR	Rossum's Universal Robots
SCARA	Selective Compliance Assembly Robot Arm
SE	Source of Effort
SF	Source of Flow
TF	Transformer
TOPIO	TOSY Ping Pong Playing Robot
USB	Universal Serial Bus

NOMENCLATURE

a_{i-1}	Link length
$c\theta_i$ or c_i	cosine(θ_i)
dist	Distance of second joint
dist2	Distance of third joint
d_i	Joint distance
inr_s	Inertia of motor shaft
i_m	Current in motor
m	Mass of body
mH	High Gain
mL	Low gain
r1	Radius of manipulator shaft
r_m	Resistance of motor
s_s	Stiffness of motor shaft
$s\theta_i$ or s_i	sine(θ_i)
α_{i-1}	Link twist angle
θ_i	Joint angle
λ	Lambda
μ_m	Motor constant
υ_m	Voltage to motor
ω	Angular velocity

LIST OF FIGURES

LIST OF TABLES

INDEX

CONTENTS **Page No.**

CHAPTER 1 INTRODUCTION

The research on robotic surgery is a new field to develop and also challenging which inspires to make it. In robotic surgery, different types of manipulator are required at different usage, such as serial manipulators, parallel manipulators, and hybrid manipulators. All these manipulators are used to hold the tool and to perform the task. To do all these functions for surgery the manipulators should be precise and accurate in motion as per the command. Here, the manipulator is serial and it will perform in the patient's body while operating. In robotic surgery, less time is required for operations, smaller incision is possible and less staff is required.

1.1 BACKGROUND AND MOTIVATION

At present, the medical field is revolutionizing its technology in the development of robotic devices and complex image processing. This all is done to remove disadvantages due to direct operation of the patients by the surgeons. A few years back when surgeons operate the surgery, it gives lots of cuts and pain to patient, but now, tissues inside the body are operated with the help of manipulators by minimum invasive surgery (generally about 1cm) due to which very small cuts are required for surgery, less blood loss and fast recovery is achieved. Robotic surgery allows the surgeons to perform complex surgical operations through small incisions using the robotic manipulator technology. Surgical robots are self-powered, computer-controlled devices that can be programmed to aid in the positioning and manipulation of surgical instruments. This provides surgeons with better accuracy, control and flexibility. These operations are guided by viewing (*i.e.,* endoscope) equipment. It is not wrong to say that computer-assisted and robotic surgeries are categorized under minimally invasive surgery.

Robots cannot actually replace human, but it can improve their ability to operate through smaller incisions by programming the devices, creating proper algorithms, accurate sensors, and improved user interfaces. Technology is now becoming more and more integrating into the medical system as from imaging systems to pre-programmed robots. In this work, the working of a particular serial manipulator which is going to be used in the project of shared controlled system robotic surgery is explored.

1.2 MANIPULATORS

Industry related robots perform many tasks like picking and placing objects or part, motion is taken by observing similar manual tasks are handled by a fully working human arm. These robotic arms are also called robotic manipulators (Fig. 1.1) which are automatically controlled, reprogrammable, multipurpose manipulator programmable in three or more axes, which may be fixed or mobile. In robotics, a manipulator is a device which is used to manipulate tools or subject without any direct contact of human being. The application was originally to deal with radioactive and bio-hazardous materials, using robotic arms and they can also used in inaccessible places at different places. In recent developments, they are used in applications like robotically assisted surgery and also in space. It is a type of arm like mechanism that consisting of a series of parts, generally sliding or jointed, which hold and move subjected to a number of degrees of freedom.

Fig.1.1 Robotic Manipulator [**1**]

Robot Manipulators consist of rigid links, which are connected through joint actuators that create relative motion of neighbouring links. Joints are attached to sensors to read joints position and its speed. Adjacent links are connected to each other by joint. There are mainly two types of joint primitives.

1.2.1 *Types of Joints*

- **Prismatic joint (P):** These joint exhibit translational or sliding displacement along an axis. It means one link slides on the other along a straight line. It has only one degree of freedom.

- **Revolute joint (R):** These joint exhibit rotary motion or angular displacement about an axis. It is frequently consulted to as a hinge, articulated or rotational joints. It has only one degree of freedom.

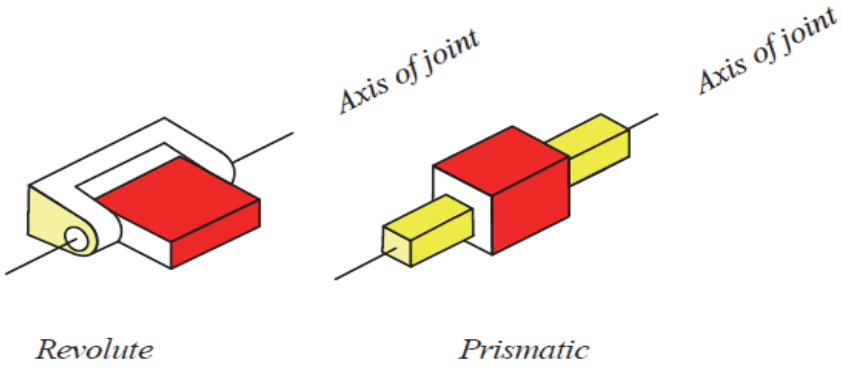


Fig. 1.2 Types of Joints [**2**]

These Prismatic and Revolute joints are the most common joints that are utilized in serial manipulators. These are also called active joints and as much these joints are there in a manipulator. Passive joints are lower pair joints that provide surface contact and some of those are cylindrical, spherical, screw and planar. These joints also provide same function or to get extra degree of freedom in manipulator. When a wrist, gripper and the control system are attached to the manipulator, it becomes a robot.

- **Wrist:** It is the kinematic range of mountains in between forearm and end-effectors which can take in multiple sticks in it. Mostly, it is used for spherical wrists; by this three revolute joint's axis intersect at a single spot which is called wrist point. The orientation of the manipulator depends upon the wrist articulation and the level of freedom of wrist is one or two or three according to the application.
- **End-effectors:** It is the voice of the manipulator which is climbed on the last tie to execute the required work from the manipulator. There are many types of end-effectors, but the simplest one is gripper which is generally capable of doing the opening and closing action. Its purpose is to curb and to depart. The actual work is done by end-effectors or tool mounted on it because the arm and wrist are used to give position and to orient the end-effectors. Many researches are done on it for different kind of functions like to grab, to scoop, to ploy, roller claws *etc*.
- **Control System:** It is having mainly three parts which are actuators, sensors and control unit. Actuators are used to supply power for mechanical structure to work

against gravity, inertia or whatever other outside powers. It is used for altering the geometric positions of the manipulator hand. Sensors are used to detect and collect information about environmental and internal states. It gives velocity, acceleration, joint position and force to the control unit. Control Unit has three functions which collect information from sensors of manipulator, take decision according to the desired output and the actual output relation and finally communicate between environment and manipulator. The assembly of wrist and end-effectors are also called hand.

1.2.2 *Types of Manipulators*

Manipulators are of primarily two types:

- Serial manipulator
- Parallel manipulator

1.2.3 *Serial Manipulators*

These are the most common manipulators used in industrial robots. These are designed or manufactured as a series of links connected by motor controlled joints that extend from a base joint to an end-effectors joint. Usually, they have an anthropomorphic arm structure which means like a human arm so it describes as consisting of a shoulder, an elbow and a wrist. Serial manipulators (robots) generally have six joints, because it requires at least 6 degrees of freedom to place a manipulated tool or end-effectors in an arbitrary position and orientation in the workspace of the manipulator or robot. A very popular application for serial manipulators or robots in now a day's industries is the pick and place assembly manipulator or robot, called a SCARA (Selective Compliance Assembly Robot Arm) robot which has 4 degrees of freedom.

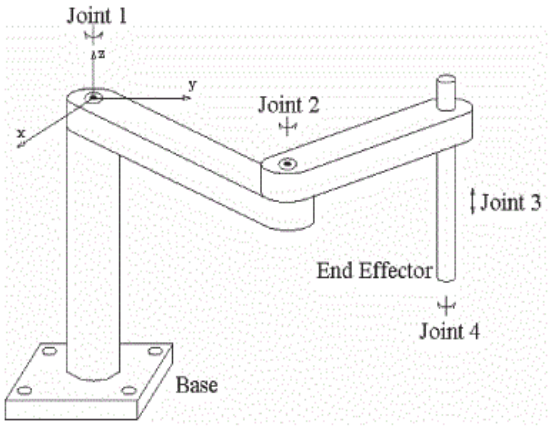


Fig. 1.3 SCARA Robot [**1**]

- **Structure**

It is the most general form, a serial manipulator or robot consists of a number of rigid links connected with the help of joints. Simplicity is considered in manufacturing and control which lead the manipulator or robots with only revolute or prismatic joints and orthogonal, parallel or wrist means intersecting joint axes. 321 kinematic structures are made by Pieper D.L. which is the first practically relevant result in this context. The Inverse kinematics of serial manipulators with 6 revolute joints, and with 3 consecutive joints intersecting, it can be solved in closed form, *i.e.,* analytically. This result has a highly positive influence on the design of industrial manipulators or robots.

The most important advantage of a serial manipulator is having a large workspace with respect to the size of the robot and also the floor space it occupies. The main disadvantages of these manipulators or robots are:

- Open kinematic structure gives the low stiffness inherent to it
- Errors are summed up or accumulated and amplified from link to link
- They cannot carry and move large weight by the actuators
- They manipulate relatively low effective load

- **Kinematics**

The position and orientation of a manipulator's or the robot's end-effectors are derived from the joint positions by means of a geometric model of the manipulator or robot arm. For the serial manipulators or robots, the designing from joint positions to end-effectors pose is easy; the inverse designing is more difficult. So, in most of the industrial manipulators or robots have special designs that reduce the complexity of the inverse designing.

- **Workspace**

The accessible workspace of a manipulator's or the robot's end-effectors are the manifold of reachable frames. The dextrous manipulator's workspace or frame consists of the points of the accessible or the reachable workspace where the robot can generate velocities that can span the complete tangent space at that particular point, *i.e.*, it can translate the manipulated tool with 3 (three) degrees of freedom, and rotate the tool to 3 degrees of freedom of rotation. The relationships between the joint space and Cartesian space coordinates of the tool taken by the manipulator or robot are usually multiple valued: the same position and orientation can be attained or reached by the serial manipulator or arm in different ways, each with a different

set of joint coordinates. So, the accessible or the reachable workspace of the robot is divided in different configurations (also known as assembly modes), in which the kinematic chain relationships are locally one to one.

- **Singularity**

The singularity is a configuration for a serial manipulator in which the joint parameters do not completely define the position and orientation of the tool or end-effectors. Singularities always occur in configurations only when the joint axis are aligning in a way that will reduce the ability of the manipulator or arm to position the end-effectors. For example, boundary singularity occurs when a serial manipulator is fully extended.

At a singularity, one or more degrees of twist can be obtained by the end-effectors (instantaneously, these directions cannot be moved by the end-effectors). The serial manipulators or robots having less than 6 independent joints are always called singular in the sense that they can never create span of a 6-dimensional twist space. This is usually called an architectural singularity. A singularity is usually a sub-manifold, not an isolated point in the workspace of the robot.

- **Redundant manipulator**

The manipulator or robot having more than 6 degrees of freedom is called a redundant manipulator which means that it has the additional joint parameters that permit the configuration of the manipulator or robot to change while it can hold its end-effectors in the fixed position and orientation.

A manipulator having seven joints is a typical redundant manipulator, for example, three at the wrist, one elbow joint and three at the shoulder. This manipulator can get motion in its elbow around a circle while maintaining a particular position and orientation of its tool or end-effectors. A hyper-redundant is having many more than 6 degrees of freedom and is generally called as snake robot.

1.2.4 *Parallel manipulator*

It is a mechanical system that uses many computer controlled serial chains to support a single platform, tool or end-effectors. However, a manipulator which is formed from 6 linear actuators which support a movable base for many devices like flight simulators is the best known parallel manipulator. This device is known as the Stewart platform or the Gough-

Stewart platform in the recognition of the engineers who was first designed and implemented it.

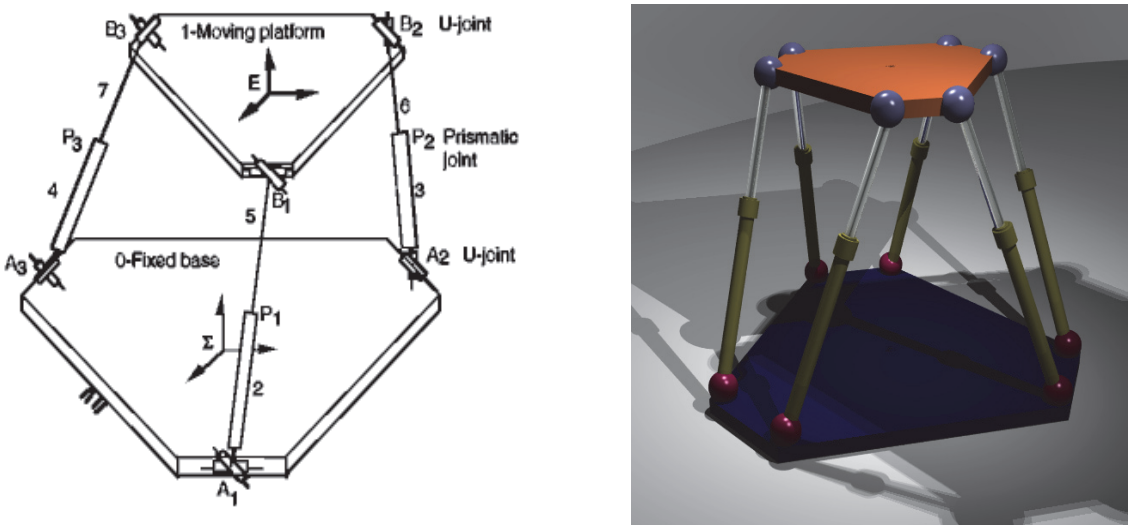


Fig. 1.4 Parallel Manipulator [**1**]

The main disadvantage of parallel manipulators or robots, in comparison to serial manipulators or robots, is their limited workspace. As in case of serial manipulators, the workspace is limited by the mechanical and geometrical limits of the design (collisions between links or legs minimal and maximal lengths of the links or legs). The workspace can also limited by the existence of the singularities, which are positions where, for some of the trajectories movement, the variability or variation of the lengths of the links or legs is infinitely smaller than the variation of the position. In contrast, at a singular position, a force (such as gravity) applied to the tool or end-effectors induce or implement infinitely large constraints on the links or legs, which might result in a kind of "explosion" of the manipulator or robot. The determination of the singular positions is quite difficult (for a usual parallel manipulator, this is an open problem). The workspaces of the parallel manipulators or robots are, generally, artificially limited or restricted to a small region where one knows that there is no singularity.

One more drawback of parallel manipulators is their non-linear behaviour. The command or order which is required for getting a circular or a linear movement of the tool or end-effectors depends dramatically on the location in the workspace and makes changes linearly during the movement. Because of this difficulty of like a non-linear command, the parallel manipulators are still not used in high precision machining, despite their good mechanical properties (speed and precision).

1.3 ROBOTICS

It is the branch of science and technology which deals with design, construction, operation and implementation or application of Robots. Robotics also deals with computer systems for their control, sensory feedback and processing of information. In dangerous environment or manufacturing processes, robotics technology deals these areas with automated machines that can replace human because they resemble human in appearance, behaviour or cognition. By field of bio-inspired robotic many of the robots are inspired.

Now a day, robotics are very fast growing field, in every direction as technologies advance continues, design, research and building new robots which serves many practical purposes as commercially, domestically or militarily. Many tasks which are hazardous to people like defusing bombs, exploring shipwrecks and mines are done by robots. However, robotics field started in classical times to operate autonomous machines and its concept, but till 20th century did not grow substantially in the research of its functionality and potential uses. As throughout history, robotics were usually seen to mimic human behaviour and generally manages similar tasks as human. Three laws of Robotics proposed by Isaac Asimov and later the zeroth law is added which are following:

- **Law 0:** A robot may not injure humanity or, through inaction, allow humanity to come to harm.
- **Law 1:** A robot may not injure a human being or, through inaction, allow a human being to come to harm, unless this would violate a higher order law.
- **Law 2:** A robot may not obey orders given to it by human beings, except where such orders would conflict with a higher order law.
- **Law 3:** A robot must protect its existence as long as such protection does not conflict with a higher order law.

▪ Robotics research

Usually the research in robotics gives attention on new types of robots investigation not on specific industrial tasks, different ways to think of designing robots and new ways to construct them but other investigations also should be done. Open-sourcing of robotic projects can be a new innovation in robot designing. Hans Moravec predicted generations of robots in 1997, robot future use:

- First generation robots should have the intellectual capacity capable to perhaps a lizard and it should available by 2010.
- Second generation robots, as disadvantage of first generation robots that they are incapable of learning will be improved in this by 2020 with intelligence which might be comparable to a mouse.
- Third generation robots should have intelligence comparable to that of monkeys and
- Fourth generation robots should be comparable to a human being and this will happen by 2050.

1.4 ROBOTS

A virtual or mechanical artificial agent, generally an electro-mechanical machine that is guided by an electronic circuitry or computer program is known as the Robot. Mr. Karel Capek introduced this word "Robot" to the public in his play R.U.R. (Rossum's Universal Robots), published in 1920. The play is related to the factory system and in this play artificial people called robots are used.

Robots can be semi-autonomous or autonomous and its range from the industrial robots to humanoids like as Honda's ASIMO (Advanced Step in Innovative Mobility) and TOSY's TOPIO (TOSY Ping Pong Playing Robot), collectively programmed swarm robots or may be microscopic nano robots.

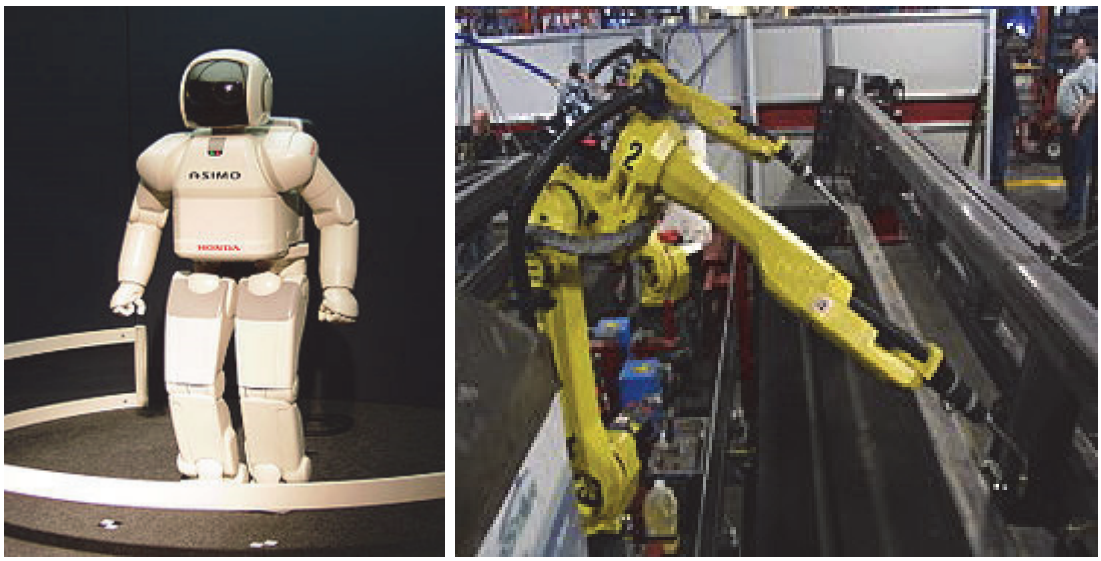


Fig. 1.5 (a) ASIMO Robot and (b) Welding Robot [**1**]

Robots have replaced the humans in the helping or assisting in performing those dangerous and repetitive tasks which humans prefer not to do, or are unable to do due to size

limitations, or even those like as in outer space or at the places where survival of humans is not possible like in extreme environments of the bottom of the sea. There is no one definition or description of the robot that satisfies everyone and most of people have their own. The two ways that robots differ from actual human beings are, simply stated, in the domain of biological form, and in the domain of cognition. The usual consensus is that a "robot" is a machine and not a being simply because it is not intelligent (it requires computer programming to function), regardless of how human-like it might appear. Conversely, an imaginary "robotic machine" or "artificial life form" that may think near or above human intelligence, and had a sensory body, would no longer said to be a "robot" but it will be some kind of "artificial being".

Any automatically operated machine that replaces human being effort, though it might not resemble like human beings in look or appearance or perform functions in a human being like manner is called as a robot. Also a robot as a machine that performs various complex acts (like walking or talking) and looks or appears like a human being, or a device that can do repetitive tasks even though it's complicated automatically, or a mechanism guided by automatic controls. Today Robots are used in many fields, some of those robots are as follows [**3**]:

➢ Mobile Robots	➢ Industrial Robots
➢ Military Robots	➢ Educational Robots
➢ Modular Robots	➢ Mining Robots
➢ Research Robots	➢ Health care Robots

1.5 DA-VINCI SURGICAL SYSTEM

The da Vinci Surgical System is a tool that utilizes advanced, robotic technology to assist the surgeon in the operation. It does not act on its own and its movements are controlled by the surgeon. This da Vinci Surgical System has a 3-Dimensional high definition vision system, computer software and special instruments that allow the surgeon to operate with enhanced vision, dexterity, precision and control. The da Vinci Surgical System has instruments like mechanical wrists that can bend and rotate to mimic the movements of the human being wrist, which allows the surgeon to make small precise movements inside the patient's body.

The 3-Dimensional HD image can be magnified up to 10 times so that can allows surgeon have a close-up view of that part of the patient.

The Da-Vinci robotic system provides surgeons with an alternative to both conventional laparoscopy and traditional open surgery, putting a surgeon's hands at the controls of a state of the art robotic system or platform. The Da Vinci System allows or enables surgeons to perform even the most delicate and complex procedures through very small incisions with high precision. Da Vinci offers most hysterectomy patients faster return work, less blood loss and other activities.

Advantages of da Vinci surgery are:

- To Comparatively shorter recovery time
- Less blood loss during surgery
- It involves less scarring
- Faster return to work and other daily activities
- Better clinical outcomes in most of the cases

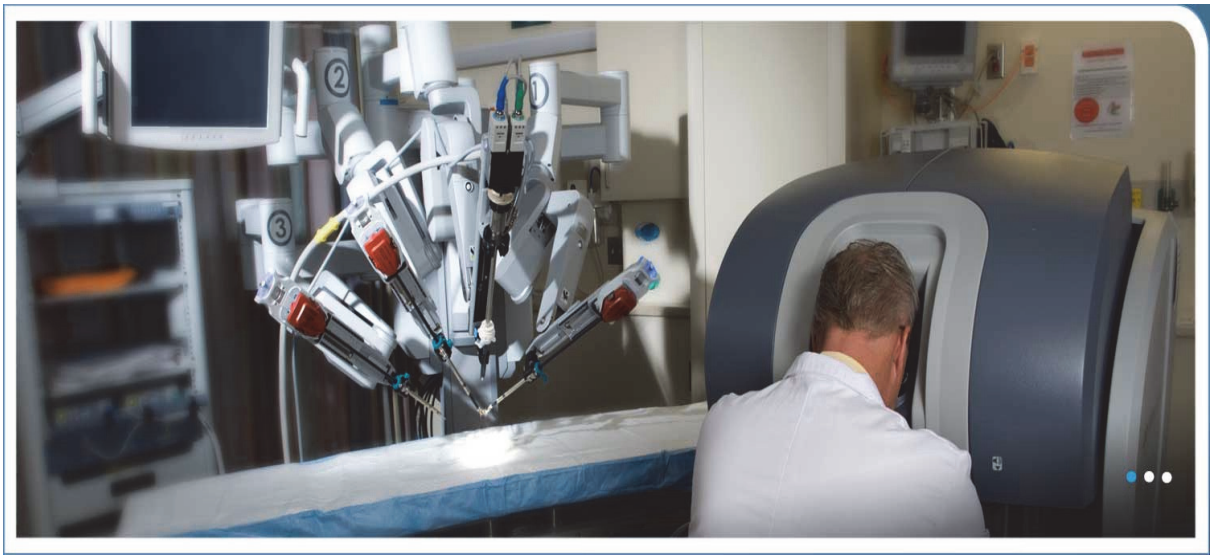


Fig. 1.6 Da Vinci Surgical System [**1**]

The robot seamlessly translates the surgeon's hand, wrist and finger movements into precise, real-time movements of surgical instruments inside the patient's body. If there is any emergency or any robotic part not working than the surgeon can immediately leave the robot and work directly with the patient if necessary. It's a dramatic improvement over conventional laparoscopy, in which the doctor is guided by a video monitor while working above the patient with handheld, long-shafted instruments. The da Vinci Surgical System is

designed to overcome the limitations of manual laparoscopy with enhanced capabilities, including high definition 3D vision and a magnified view.

It is usually called a “robot”, da Vinci cannot act on its own. The surgery is performed entirely by the surgeon or doctor. Together, the da Vinci System and Single-Site instruments allow the surgeon or doctor to perform gallbladder surgery through a single incision. Due to this, patients may be able to get back to your life without the recovery or scars that usually follow major surgery. By da Vinci Surgical System range of operations can be done which are commercially available for use in the following specialities:

➢ General Surgery	➢ Thoracic Surgery
➢ Cardiac Surgery	➢ Head and Neck Surgery
➢ Urologic Surgery	➢ Gynaecologic Surgery

1.6 ROBOTIC SURGERY

Robotically assisted surgery and computer assisted surgery are terms of technological developments that use robotic surgery systems to aid in surgical procedures. Robotically assisted surgery was developed to remove the limitations of minimally invasive surgery and to increase the capabilities of surgeons performing open surgery.

In the robotically assisted minimally invasive surgery, instead of directly moving or handling the instruments, the surgeon uses one of five methods to control the instruments, either through computer control or a direct tele-manipulator. A tele-manipulator is a remote manipulator that permits or allows the surgeon or doctor to perform the normal movements related or associated with the surgery whilst the robotic arms or manipulators carry out those movements using tools or end-effectors and manipulators to perform the actual surgery on the patient's body. In computer controlled surgical systems the surgeon uses a computer to control the robotic arms or manipulators and its tools or end-effectors, though these systems can also still use the tele-manipulators for its input. The advantage or benefit of using the computerised method for surgery is that the surgeon does not have to be present at the same place as of patient, but can be anywhere in the world, leading to the possibility for remote surgery. The da Vinci senses the surgeon’s hand movements and translates them electronically into scaled-down micro-movements to manipulate the tiny proprietary instruments. It also detects and filters out any tremors in the surgeon's hand movements, so

that they are not duplicated robotically. The camera used in the system provides a true stereoscopic picture transmitted to a surgeon's console.

In traditional open heart surgery, the surgeon or doctor makes a ten to twelve-inch incision, then gains access to the heart by splitting the sternum (breast bone) and spreading open the rib cage. The patient's heart is then placed on a heart-lung machine and the heart is stopped for some period of time during the operation. This approach can be associated with prolonged time to complete recovery and postoperative infection and pain. So as patient recovery after robotic assisted heart surgery is quicker, the hospital stay is very less. Reduced recovery times are not only better for the patient, they also reduce the number of staff needed during surgery, nursing care required after surgery, and, therefore, the overall cost of hospital stays.

Fig. 1.7 Robotic Surgery pictorial view [**1**]

Compared with some other minimally invasive surgery techniques or approaches, robot assisted surgery gives the surgeon much better control over the surgical instruments and a much better view of the surgical site of a patient's body. Even, surgeons no longer have to stand throughout or complete the surgery and do not tire as quickly. In the last, the surgical robot system can continuously be used by rotating surgery teams.

1.7 CONTRIBUTION OF THESIS

The main contribution of the thesis is as follows:

- The detailed forward model and inverse model in bond graph of robotic manipulator systems are developed.
- Geometric modification/modelling of the back end of the manipulator in Solid Works for the attachment of motors and the motion transferred to different joints by the use of ropes.
- Solution of the forward kinematics and inverse kinematics by using the Denavit-Hartenberg (D-H) representation and Newton-Rapson method by using a Mat lab tool of serial manipulator.
- Design considerations of manipulator and motor housing.

1.8 ORGANISATION OF THESIS

There are total five chapters are included in my thesis. Brief view of all the information in these chapters is given as:

A comprehensive literature review is presented in Chapter 2. The literature is pertaining to the Da Vinci System, Denavit-Hartenberg (D-H) representation and its algorithm. The literature on Robotic manipulators used in surgical systems. In last objective of my work is given.

In Chapter 3, detailed review of the bond graph tool with its different energy domains and all basic bond graph elements are included. The principle of work by use of bond graph and development of the serial manipulator system in bond graph and by Denavit-Hartenberg (D-H) Algorithm solution for the particular robotic manipulator with the help of Mat lab and in the last comparison of the results and different graphs comes from bond graph.

The design consideration and its limits for the construction of this robot serial manipulator is given in Chapter 4. It includes the making of complete manipulator in Solid Works software having motor position and how it transforms energy from motors to joints with the help of pulley's. In this Chapter 4, RPT manufacturing is also defined.

In Chapter 5, conclusions are presented. This Chapter also presents lists of references and the scope of future work.

CHAPTER 2 LITERATURE REVIEW

2.1 INTRODUCTION

In this Chapter, the literature review of Da Vinci Surgical System, Denavit-Hartenberg (D-H) Algorithm and its calculation of parameters. The Bond Graph modelling used for robotic serial manipulators and some robotic arms and how accurate results it gives is useful. The Bond Graph modelling represents a different part of the whole system and it calculates both forward and inverse kinematics of the manipulators or robots.

2.2 DA VINCI SYSTEM

For minimally invasive surgical procedures the enhanced dexterity of surgical telerobots for doctors is described. Computational architecture and recent clinical applications are presented for a 7 (seven) degree of freedom (DOF) master-slave system [**4**]. By use of this surgical teleoperator system, high quality stereo visualisation and man-machine interface by which surgeon's hand is directly connected with the motion of his surgical tool tips in the patient's body [**5–6**].

Fig.2.1 Surgical Tool [**4**]

In Da Vinci surgical system, two subsystems are there; first one is surgeon's console, the surgeon's user interface, housing the display system, the surgeon's handle and the electronic controller. Another one is patient side cart which consists of surgical endoscope, the tool manipulator, the camera manipulator, fully sterilised tools and the assistant's user interface. The electronic controller is manufactured as an system with speed, reliability and fails safe operation performance. It has 48-DOF and its rates up to 1000 cycles/sec. Its clinical applications are many like orthopaedic (orthoscopy), hernia repair, gynaecology, intra-rectal surgery, general surgery (laparoscopy) *etcetera*. To use this system surgeon's is well trained before any operation.

For minimally invasive surgery this paper designed a robotic control system for the parallel, serial robot [**7**]. For 6-degree of freedom serial portion of the mechanical arm taken

us from the robot, the system controls it based on Field Programmable Gate Array (FPGA) and with a transmission unit which based on Universal Serial Bus (USB) and the parallel portion having 4 (four) degree of freedom structure is controlled by the Programmable Multi Axis controller (PMAC) based on Proportional Integrated Derivative (PID) parameters. In this position testing accuracy is developed for serial manipulator or robot with the help of the FPGA. Here data transformation speed is highly increased with USB based transformation. Fast response and high precision design achieved for parallel manipulator or robot by using PID parameters in the PMAC. It is the research for minimally invasive surgical robots.

In this paper Da Vinci system simulator discussed its design and development for 3-Dimensional virtual reality in learning and training environment for Minimal Invasive Surgery (MIS) [**8**]. Here, for development and design the Denavit-Hartenberg (D-H) parameters are defined for forward kinematics and inverse kinematics is also resolved.

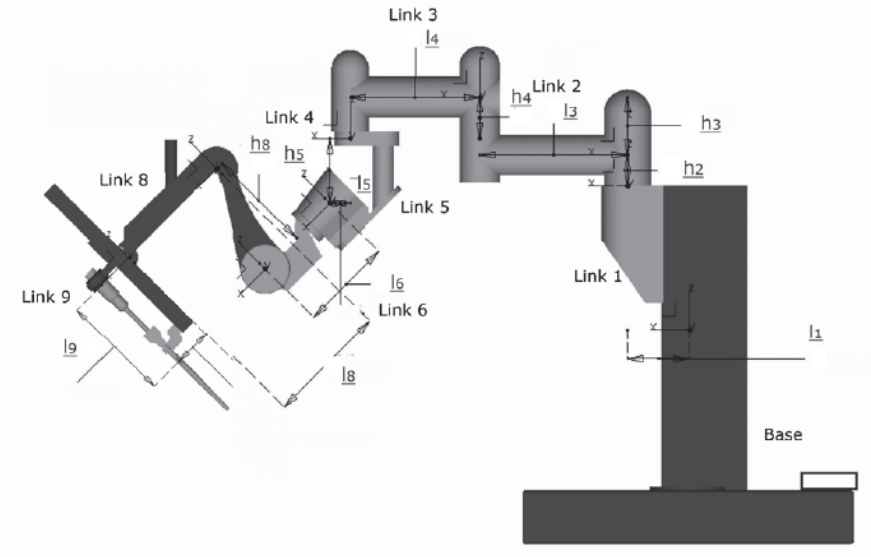


Fig. 2.2 Arm with endoscope holder in Da Vinci Surgical System [**8**]

It is now programmed for the system in simulator according to the requirements and the same results behaviour is observed. This simulator is used for training in operation rehearsals. The movement of the robot and endoscopes under kinematic constraints are correctly generated. A gripper is developed to replace standard stylus to opening movements, orientation, clone positions which are inspired by Da Vinci surgeon console.

A new computer based simulator is developed for Da Vinci surgical system which provides two handed interface control to Da Vinci robot and it is much simpler for practice [**9**]. This is in its beginning stages, but as per the author, its promising research for the science. The simulation kernel ensured that its motion control is more effective and provides a basic collision framework. It's composed of three basic parts as rendering, device

management and computational control. This simulator benefits the user to understand the robot's limitations while controlling its instruments.

Efficacy comparison of the traditional manually assisted laparoscopic techniques and the Da Vinci surgical robotic system using both 3-Dimensional and 2-Dimensional views in performing standardised exercises. [**10**]. Some inexperienced medical students are selected for surgical exercises. Each student had performed four exercises two times each, these are placing a ring over receptacles, running a suture, grasping a free hanging suture and cut-in three pieces of it, last is to perform a surgical knot.

Table 2.1 Mean Time needed to perform exercises [**10**]

Exercises	**Da Vinci 3D(sec)**	**Da Vinci 2D(sec)**	**Manual(sec)**
Ring Grasping	9.5 ± 8.7	12.0 ± 3.6	19.0 ± 5.7
Suture Cutting	27.5 ± 7.9	68.0 ± 26.7	64.0 ± 19.7
Suture Passing	60.5 ± 39.2	90.5 ± 29.0	356.5 ± 112.6
Knot Tying	51.0 ± 23.2	82.6 ± 23.0	184.5 ± 55.0

It is observed by these experiments that inexperienced medical students performed different standardised tasks much quicker with very less errors when assisted by Da Vinci robot as compared to the traditional manually assisted laparoscopic surgery. So, it concludes that with the help of a robotic system to perform difficult laparoscopies interventions become easier and efficient.

Predictive control or soft computing alternatives in modern control methods are investigated, which are applicable effectively in telesurgery [**11**]. By the help of this technology remotely operated devices are effectively allowed to run many healthcare systems. It is also used in space applications. Many control schemes are developed and tested to deal with transparency, latency and bandwidth. Satellite communication is used for medical assistance and medical education. [**12**–**14**].

2.3 SERIAL MANIPULATOR

The problem of desired workspace with given constraints is resolved for the manipulators with certified solutions. It determines the approval serial chain topologies like that a set of provided constraints is guaranteed to be satisfied within all points in the predefined workspace [**15**]. The arrangements of linkages and joints forming the mechanism are referred

here as topology. An interval based methodology is introduced to provide an automated finding of the design parameter space to find out the ranges of values in the manipulator or robot design. The parameters are expressed in a bounded range of continuous data values. Further in this paper an interval based inversely kinematic process is introduced for a serial manipulator without any explicitly defined topology. For anthropomorphic serial manipulator, the design process and workspace verification are also presented. As that of Denavit-Hartenberg parameters like as offset angles and link lengths which completely define a manipulator are taken. As a result of anthropomorphic arm with spherical wrist the joint displacement for all the revolute joints are set at $\pm 30^{\circ}$. It's taken first by only considering first three joints and secondly by all six degree of freedom addressing both position and orientation. It is effective for all variations of kinematic topology in a serial manipulator.

For control design of the serial link mechanical manipulator some fundamental properties are investigated in this paper. Revolute and Prismatic joints are used in the manipulator [**16**]. The important contribution of this paper are mainly threefold. First, it explores the upper bound of the inertia property. Second, it explores the upper bound of gravitational term and lastly, it explores the upper bound of the Coriolis and centrifugal terms. The basis of this a robust control is proposed and these are generic properties can be used for other control designs. All the terms made here are applicable for any serial manipulator. The properties used for both globally and local performance analysis.

In a general serial manipulator's method of deflection analysis, gravity, end-effectors and inertial loads are described. It is general model and applicable to any number of serial degree of freedom having a combination of revolute and prismatic joints [**17**]. The quasi-static approach is used to analyse end-effectors error and development of end-effectors spring is completely described. For kinematic and geometric analysis of joint loads, kinematic and geometric influence coefficients are used [**18**–**19**]. Here, geometry of manipulators, deformation modelling in local coordinates, small displacement influence, force and moment transformations, load contribution due to end-effectors, gravity and due to inertia, end-effectors deflection due to compliant links and joints, generalised end-effectors spring. This work is also demonstrated computationally in simulation software [**20**]. The development of generalised spring is fully described. By model referencing the deformations are reduced up to 90% by real time error compensation.

The accuracy of the manipulator by acting of kinematic calibration procedure on the controller is improved. Here error compensation, estimation of the numerical value of the manipulator's parameters, kinematic model of a general serial manipulator and with an

example of experimental data is presented [**21**]. Extended Denavit and Hartenberg (ED&H) approach, Incremental approach, Modified Incremental approach, Pose measuring approach, Pose matching approach and elimination of redundant parameters are used. An ED&H is adopted to describe the robot structure and error by variations of these parameters. The pose measuring and pose matching approach are used to find the vector of errors. For the estimation of structure parameters three procedures are followed as are, a non linear optimisation, an iterative linearization of the equations and an extended Kalman filter. After applying ED&H approach and Incremental approach they are compared. Joint rotations are measured by giving predefined poses to the robot practically and forced to reach the desired pose. It concludes that modifies approach is much easier to implement an easily automates but of this joint offsets are not explicitly included in the parameters set. Practical calibration for different estimation algorithms and their performances of measuring robot are well described.

A singularity robust method is proposed and implemented on PUMA 560 robot. Complete treatment of singularities, from identifications of its singular configurations to the handling algorithm and discontinuity issues [**22**]. Here a main focus is made on discontinuity problems with the singularity robust methods with removal of degenerate components. For providing continuous control on the end-effectors over the boundary of singular region the null motion is proposed.

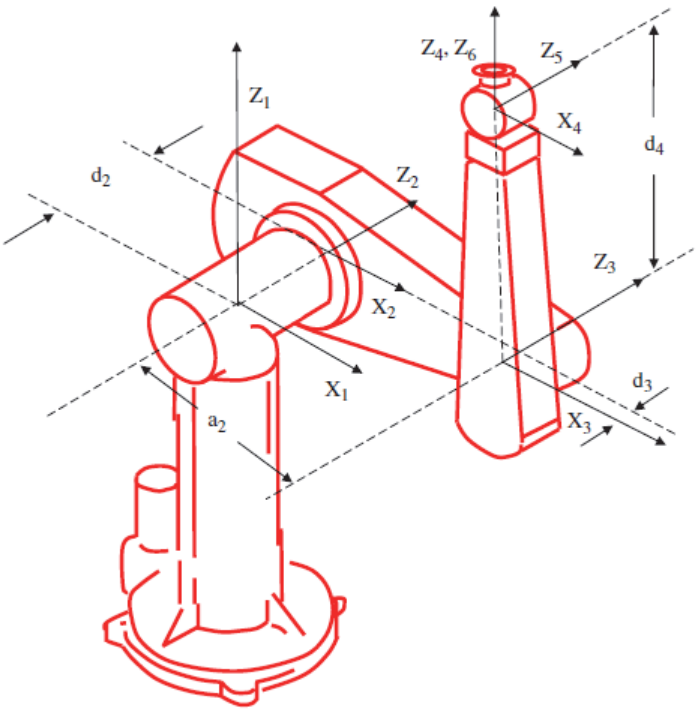


Fig. 2.3 PUMA 560 at home position [**22**].

Two types of singularities are handled and its algorithm is made. Jerky motions are removed with integrated damping control to the null space control and torque commands resulted from null space control, reduced by an amount proportional to joint velocities in type-I singularity. In type-II, the singular direction is shifted by null space that leads to the desired task or motion inside the singular region. These methods are adopted because of its simplicity and robustness which allows the end-effectors go through the singularities, which maximise the useful workspace for the manipulator. An effective, smooth and controllable motion for the end-effectors movement through singular configurations is achieved [**23**–**24**].

Robust control is designed for serial link mechanical manipulators having revolute or prismatic joints. By assuming properties for the manipulator system except uniform positive definiteness of inertia matrix the robust control is created [**25**]. Mainly two tasks are completed in this paper in which first is to explore the upper bound property of the inertia matrix for any number of links in serial manipulator and the other one is to develop robust control that guarantees the predefined performance in the presence of uncertainty. The properties of the inertia matrix and upper bound analysis are generic for any serial rigid link manipulator type having prismatic and revolute joints.

2.4 ROBOTICS D-H APPROACH

Concepts of Denavit-Hartenberg (D-H) based methods and Screw based methods are defined in this paper. A comparison of both is well presented by explaining both methods complete with examples [**26**]. The relationship between the links and the joints that composes a kinematic chain describes the kinematic modelling of a robot manipulator.

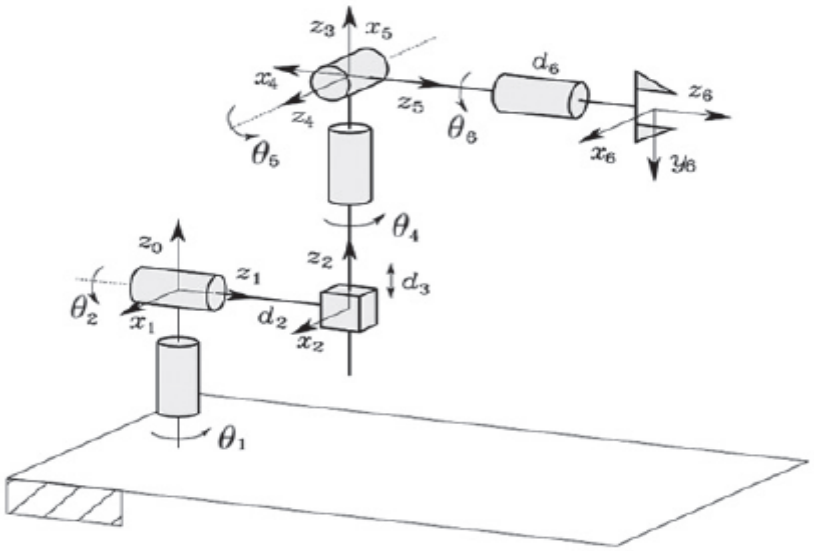


Fig. 2.4 D-H frames of Stanford Manipulator [**26**]

Minimal parameter representation of kinematic chain is used by D-H convention and it is the most popular method used. The D-H approach or convention is applied on this Stanford manipulator and its parameters are

Table 2.2 D-H parameters [**26**]

Link	a_i	α_i	d_i	θ_i
1	0	$-\pi/2$	0	θ_1
2	0	$\pi/2$	d_2	θ_2
3	0	0	d_3	0
4	0	$-\pi/2$	0	θ_4
5	0	$\pi/2$	0	θ_5
6	0	0	d_6	θ_6

A homogeneous matrix is used after finding out all the parameters for each link parameters and put those in the matrix ${}^{i-1}_{i}A$

$$ {}^{i-1}_{i}A = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1} $$

After putting all parameters for each link one by one then a resultant arm matrix is derived by multiplying all the matrices.

For static and kinematic analysis of rigid bodies and mechanisms the tool used is Screw theory. It is a non-minimal parameter representation and velocities are mainly find out by this theory. Both rotational and translation quantities both are represented geometric entities by it. It is composed with an axis by which both translation and rotation relation in a scalar pitch is defined. It associates purely geometric entities to physical meaning and by its use to express as twists and forces or torques as wrenches.

Table 2.3 Rodrigues parameters of a Stanford manipulator [**26**].

Joint	s_i	$s_{0,i}$	θ_i	t_i
1	(0,0,1)	(0,0,0)	θ_1	0
2	(1,0,0)	(0,0,0)	θ_2	0
3	(0,1,0)	(d_2,0,0)	0	d_3
4	(0,1,0)	(d_2,0,0)	θ_4	0
5	(0,0,1)	(d_2,0,0)	θ_5	0
6	(0,1,0)	(d_2,0,0)	θ_6	0

Several screw displacements are suffered by a rigid body, the resultant or equivalent screw displacement by multiplying number of homogeneous matrices as a number of joint parameters of each one.

$$R(\theta) = \begin{bmatrix} c_\theta + s_x^2(1-c_\theta) & s_y s_x(1-c_\theta) - s_z s_\theta & s_z s_x(1-c_\theta) - s_y s_\theta \\ s_y s_x(1-c_\theta) - s_z s_\theta & c_\theta + s_y^2(1-c_\theta) & s_y s_z(1-c_\theta) - s_x s_\theta \\ s_z s_x(1-c_\theta) - s_y s_\theta & s_y s_z(1-c_\theta) - s_x s_\theta & c_\theta + s_z^2(1-c_\theta) \end{bmatrix} \quad (2.2)$$

In comparison, of these two methods Denavit-Hartenberg and Screw, its obtained that screw theory has some advantages over D-H theory. Inverse kinematics are cannot be calculated by D-H method, although it is more popular than screw method because it's easy to implement, but the screw is a quiet complex method [**27**–**29**].

To handle singularities and also to overcome the multiple paths possibility in redundant robots, the approach here depends on computation of multiple numerical estimations for the inverse [**30**]. It selects the current best path for the desired configuration of the tool or end-effectors. C/C++ is used to implement the algorithm which can be easily expanded to its highest levels. The method is applied or experimented on four different robots at both singular and non-singular configurations up to 7 (seven) DOF redundant robot with demonstrating accuracy in real time performance. The robots used are PUMA 260, KUKA, PUMA 560 and SCARA Robot. The Jacobian matrix is defined on the basis of forward kinematics equations. The accuracy level is less than 1mm and in real time average 20.48ms.

For inverse kinematics equations a unique analytical solution is deduced, which helps in dynamics analysis, trajectory planning and control strategies. Here Shotcreting robot which is having 8 (eight) joints, including 3 (three) sets of coupling motions [**31**].

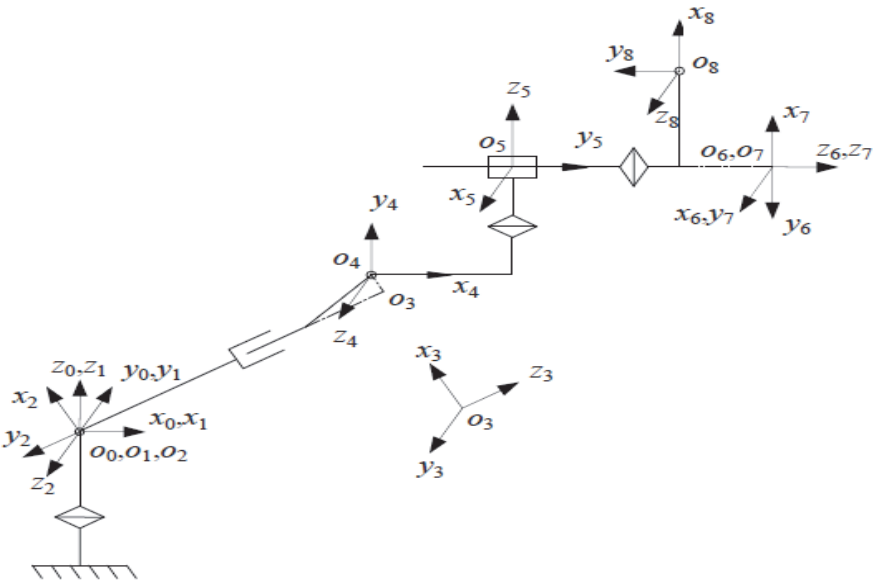


Fig 2.5 D-H frames of shotcreting robot [**31**].

Denavit-Hartenberg (D-H) convention is used to deduce the kinematic equations and for three sets to decouple the linked motions. The final transformation matrix after finding parameters and multiplying all the joint matrices is:

$$ {}^{0}_{8}T = \begin{bmatrix} -s\theta_8 & -c\theta_8 & 0 & d_6 + a_4 c\theta_1 + 4727 \\ -c\theta_7 c\theta_8 & c\alpha_7 s\theta_8 & s\theta_7 & -a_7 c\theta_7 + a_4 s\theta_1 + 4727 \tan\theta_2 \\ -s\theta_7 c\theta_8 & s\alpha_7 s\theta_8 & -c\alpha_7 & -a_7 s\theta_7 + d_5 + a_3 s\theta_2 - d_3 c\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3) $$

The first three columns are defined the posture of the spraying gun relative to the base frame and the fourth one denotes its position. So, by these values of joint variables for this shotcreting robot the transformation from joint space to operation space is completed. As here algebra method is adopted to solve the inverse kinematics for five independent joint equations [**32**–**34**].

2.5 BOND GRAPH MODELLING OF ROBOTIC MANIPULATORS

A new approach for symbolic derivation of Jacobian matrices is proposed relating to joint rates in Cartesian variable. Here by using the Bond Graph modelling of robotic manipulator is evaluated for the example [**35**]. This method is a recursive technique. For a link by relating the joint rates and angular velocities using Jacobian is derived from the rotation matrix. This approach is much simpler and computationally efficient and readily reacting to symbolic derivation.

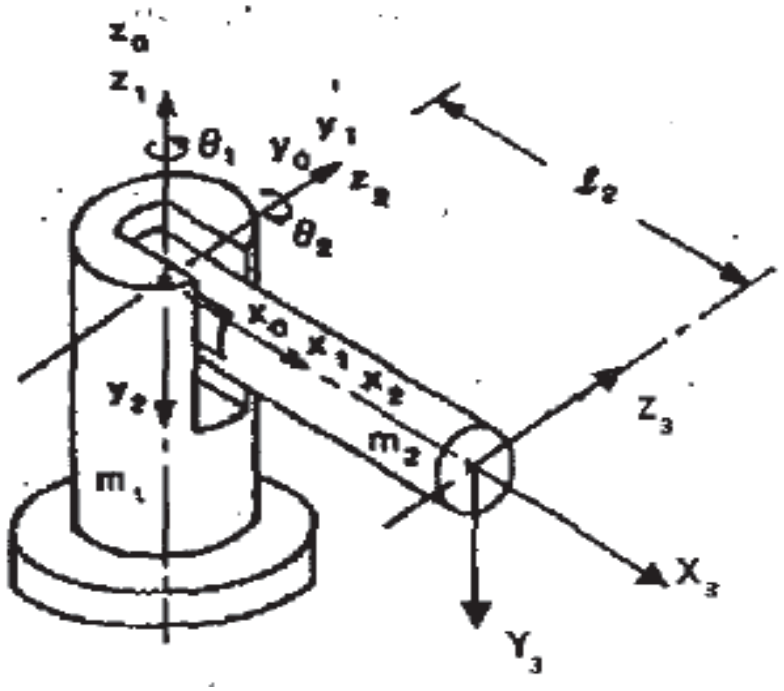


Fig. 2.6 Two link spatial manipulator [**36**]

The symbolic derivation of Jacobians is derived by showing an example of two link manipulator. Both the joints are rotational and parameters are calculated using the D-H method.

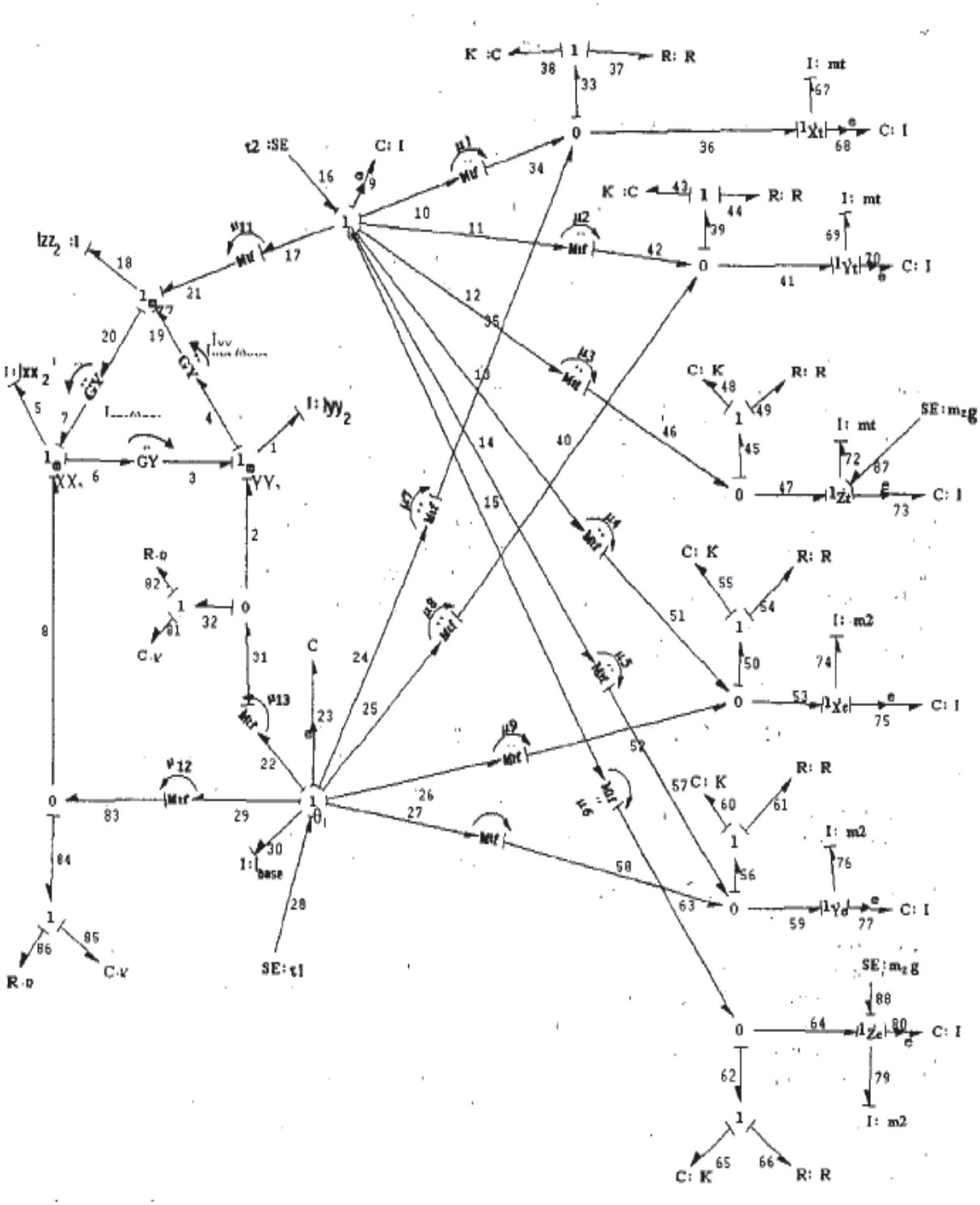


Fig. 2.7 Bond Graph of two link using Jacobian [**36**]

After all, this simulation is carried out using MATHEMATICA 4.0 for both rotational and translational velocities [**36**].

Using Extended Bond Graphs a multi body dynamic system is modelled. This formulation is used for deriving models having one link and two link flexible manipulator systems [**37**]. Simulation is considered for the rigid and deformable multi body systems which exhibits large rotations and translations.

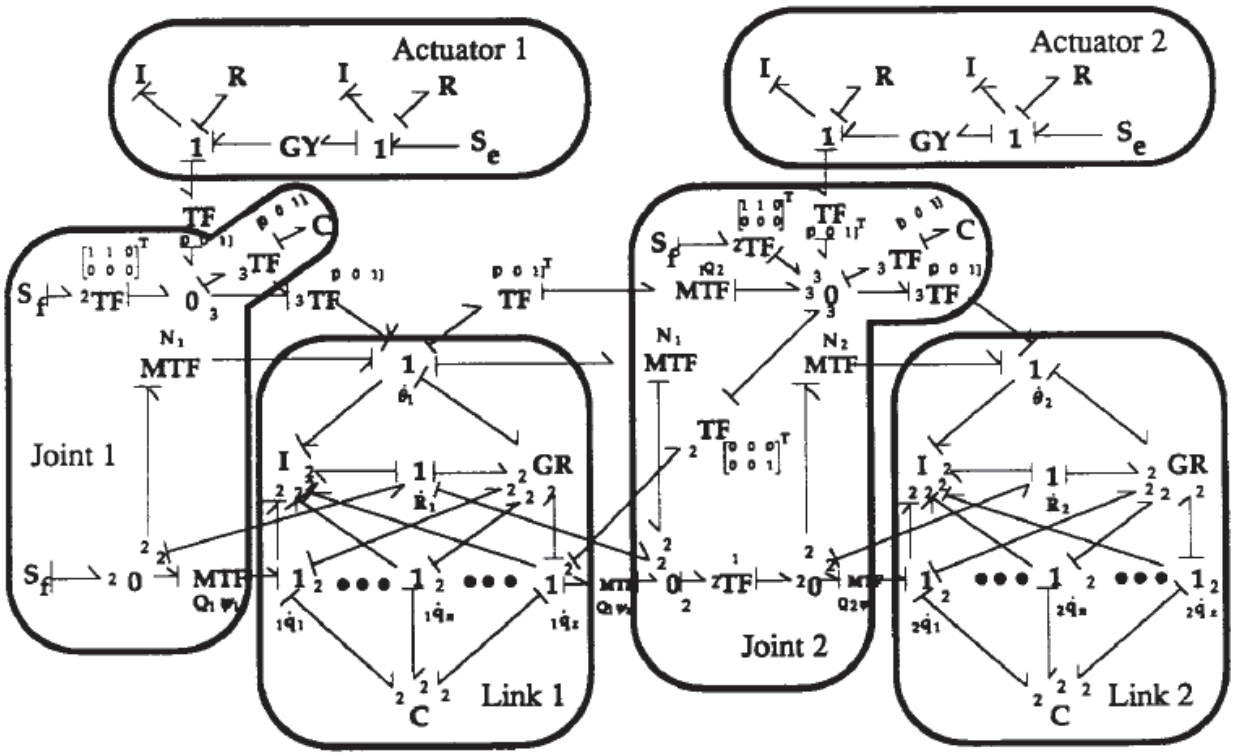


Fig. 2.8 Bond Graph of Two link manipulator [**37**]

In extended capacitance element the internal energy is made by Bond Graph network and by gyrator elements the total energy power of the system is made. For connecting the external efforts to the associated junctions, the modulated transformer is used in Bond Graph modelling [**38**].

2.6 OBJECTIVE OF PRESENT WORK

As my project is under CSIR-CSIO Chandigarh, they are making surgical system which performs surgery by having hands like manipulators on the patient's body which are controlled by doctor seating at different panel. So, for this serial manipulator is designing with all its working system like power transfer, manipulators limits, material *etcetera*. The objective of the present work is formulated and summary of these objectives is:

- To design manipulator for the considerations of functions, it performs, its limitations and motor housing.
- To make complete robotic manipulator's construction in Solid Works software with all the attachment of motors and how the motion transferred to different joints by the use of ropes or belts.
- To develop a forward model and inverse model using Bond Graph of robotic manipulator systems are developed.
- To solve the forward kinematics and inverse kinematics of the robotic manipulator by using the Denavit-Hartenberg (D-H) representation and Newton-Rapson method by using a Mat lab tool of this particular serial manipulator.
- To compare the results comes from both methods, *i.e.,* Bond Graph approach and D-H approach.

CHAPTER 3 MODELLING OF SERIAL MANIPULATOR

3.1 INTRODUCTION

This chapter introduces the precision surgery model with Denavit-Hartenberg Representation and Bond Graph Simulation Tool. Generally, we design the robotic manipulators practically and analyse that by making its prototype. Now, with the help of Denavit-Hartenberg approach we can calculate all its parameters, singularities, workspace, different angle require to reach the desired position *etcetera*. Both forward kinematics and inverse kinematics can find but in inverse there are more complexities to be faced, for that Newton-Rapson method, Jacobian method and some other methods are used to find, which makes it more complex and lengthy. So, due to all this Mat Lab tool is used here for solving complex equations.

Now, Bond Graph is a completely unique method to find forward kinematics and inverse kinematics in which only basic information has to provide in the SYMBOLS [**39**] tool by designing all its networks and relation between different parts of the manipulator including power generation, transmission. In Bond Graph all the types of systems can interact in a single network, which is not possible in other software's or tools such as electrical, hydraulic, mechanical *etcetera*.

3.2 BOND GRAPH MODELLING

In 1959, Prof. H. M. Paynter gave the revolutionary idea of portraying systems in terms of power bonds, connecting the elements of the physical system to the so called junction structures which were manifestations of the constraints. This power exchange portray of a system is called **Bond Graph**, which can be both powerful and information oriented. Later on, Bond Graph theory has been further developed by many researchers like Karnopp, Rosenberg, Thoma, Breedveld, *etc*. who have worked on extending this modelling technique to power hydraulics, mechatronics, general thermodynamic systems and recently to electronics and non-energetic systems like economics and queuing theory.

Bond Graph is a tool in which we can see performance of any system that can be any dynamic by giving relation with the direction of flow of energy or power exchange between different parts or smaller systems in our system. The lumped parameter elements of resistance, capacitance and inductance are interconnected in an energy conserving way by bonds and junctions resulting in a network structure. From the pictorial representation of the

bond graph, the derivation of system equations is so systematic that it can be algorithmized. The whole procedure of modelling and simulation of the system may be performed by some of the existing software Bond graphs is a kind of Signal flow graph or block diagrams but it more graphically represents its results.

It is a graphical tool for capturing the common energy structure of systems. It increases one's insight into systems behaviour. In the vector form, they give a concise description of complex systems. Moreover, the notations of causality provides a tool not only for formulation of system equations, but also for intuition based discussion of system behaviour, viz. controllability, observability, fault diagnosis, etc.

3.2.1 *Bond Graph Standard Elements*

In bond graphs, one needs to recognize only four groups of basic symbols, i.e., three basic one port passive elements, two basic active elements, two basic two port elements and two basic junctions [**40**]. The basic variables are effort (e), flow (f), the time integral of effort (P) and the time integral of flow (Q).

Table 3.1 Efforts and Flows in physical domain [**41**]

System	Effort (e)	Flow (f)
Mechanical	Force, F (N) Torque, τ (Nm)	Velocity, v (m/s) Angular velocity, ω (rad/s)
Electrical	Voltage, u (V)	Current, i (A)
Hydraulic	Pressure, p (P)	Vol. flow rate, $\dot{Q}$ (m^3/s)
Thermal	Temperature, T (K) Pressure, p (N/m^2)	Entropy change rate, $\dot{S}$ (J/K/s) Volume change rate, $\dot{V}$ (m^3/s)
Chemical	Chemical potential, μ (J/mole) Enthalpy, h (J/kg)	Mole flow, $\dot{N}$ (mole/s) Mass flow rate ($\dot{m}$)
Magnetic	Magneto-motive force, V (A)	Magnetic flux rate, $\dot{\Phi}$ (Wb/s)

3.2.2 *Basic 1-Port elements*

A 1-port element is addressed through a single power port, and at the port a single pair of effort and flow variables exists. Ports are classified as passive ports and active ports. The passive ports are idealized elements because they contain no sources of power. The inertia or inductor, compliance or capacitor, and resistor or dashpot are classified as passive elements.

- **R-Elements:**

The 1-port resistor is an element in which the effort and flow variables at the single port are related by a static function. Usually, resistors dissipate energy. This must be true for simple electrical resistors, mechanical dampers or dashpots, porous plugs in fluid lines, and other analogous passive elements.

$$\text{Effort} = (\text{a constant or a function}) \times (\text{Flow}),$$

$$\text{Flow} = (\text{a constant or a function}) \times (\text{Effort}).$$

- **C-Elements:**

Consider a 1-port device in which a static constitutive relation exists between an effort and a displacement. Such a device stores and gives up energy without loss. In bond graph terminology, an element that relates effort to the generalized displacement (or time integral of flow) is called a one port capacitor. In the physical terms, a capacitor is an idealization of devices like springs, torsion bars, electrical capacitors, gravity tanks, and accumulators, etc.

In general terms: $\text{Effort} = (\text{a constant or a function}) \times \int_{-\infty}^{t} (\text{flow}) dt$

- **I-Elements:**

A second energy storing 1-port arises if the momentum, *P*, is related by a static constitutive law to the flow, *f*. Such an element is called an inertial element in bond graph terminology. The inertial element is used to model inductance effects in electrical systems and mass or inertia effects in mechanical or fluid systems.

$$\text{Flow} = (\text{a constant or a function}) \times \int_{-\infty}^{t} (\text{effort}) dt \tag{3.3}$$

For mechanical system: Force = rate of change of momentum

$$P(t) = \int_{-\infty}^{t} e dt \tag{3.4}$$

- **Effort and Flow Sources :**

The active ports are those, which give a reaction to the source. For, example if we step on a rigid body, our feet reacts with a force or source. For this reason, sources are called active ports. Force is considered as an effort source and the surface of a rigid body gives a velocity source. They are represented as a half arrow pointing away from the source symbol.

3.2.3 *Basic 2-Port elements*

There are only two kinds of two port elements, namely ``**Transformer**'' and ``**Gyrator**''. The bond graph symbols for these elements are **TF** and **GY**, respectively. As the name suggests,

two bonds are attached to these elements.

- **The Transformer:**

The bond graphic transformer can represent an ideal electrical transformer, a mass less lever, etc. The transformer does not create, store or destroy energy. It conserves power and transmits the factors of power with proper scaling as defined by the transformer modulus.

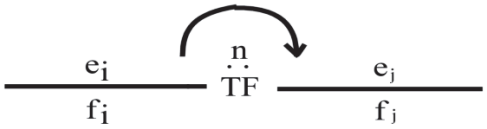


Fig. 3.1: Bond Graph for Transformer

It gives: $f_j = n f_i$ and $e_i = n e_j$

- **The Gyrator:**

A transformer relates flow-to-flow and effort-to-effort. Conversely, a gyrator establishes relationship between flow to effort and effort to flow, again keeping the power on the ports same. A Gyrator in Bond Graph is represented as shown in Fig. 3.2

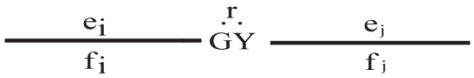


Fig. 3.2: Bond Graph for Gyrator

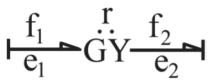
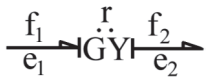
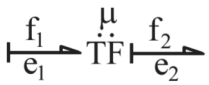
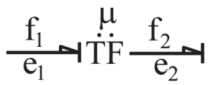
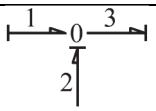
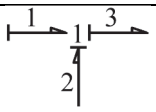
Here, 'r' is known as gyrator modulus. The definition of modulus is from a flow to the effort. In general form, flow and effort in gyrator are related as follows: $e_j = rf_i$ and $e_i = rf_j$

3.3.4 *The 3-Port junction elements*

The name 3-port used for junctions is a misnomer. In fact, junctions can connect two or more bonds. There are only two kinds of junctions, the **1** and the **0** junction. They conserve power and are reversible. They simply represent a system topology and hence the underlying layer of junctions and two-port elements in a complete model (also termed the **Junction Structure**) are power conserving.

1 junctions have equality of flows and the efforts sum up to zero with the same power orientation. They are also designated by the letter **S** in some older literature. Such a junction represents a common mass point in a mechanical system, a series connection (with same current flowing in all elements) in a electrical network and a hydraulic pipeline representing flow continuity, etc.

Table 3.2: Definition of Bond Graph Elements [**40**]

Type	Name	Symbol	Definition	
			Linear	Nonlinear
Storages	Inductor		$f=\frac{1}{m}\int_{-\infty}^{t} e\,\mathrm{d}t$	$f=\psi_{\mathrm{I}}\left(\int_{-\infty}^{t} e\,\mathrm{d}t\right)$
			$e=m\frac{\mathrm{d}f}{\mathrm{d}t}$	$e=\phi_{\mathrm{I}}\left(\frac{\mathrm{d}f}{\mathrm{d}t}\right)$
	Capacitance		$e=K\int_{-\infty}^{t} f\,\mathrm{d}t$	$e=\phi_{\mathrm{C}}\left(\int_{-\infty}^{t} f\,\mathrm{d}t\right)$
			$f=\frac{1}{K}\frac{\mathrm{d}e}{\mathrm{d}t}$	$f=\psi_{\mathrm{C}}\left(\frac{\mathrm{d}e}{\mathrm{d}t}\right)$
Dissipation	Resistance		$f=e/R$	$f=\psi_{\mathrm{R}}(e)$
			$e=Rf$	$e=\phi_{\mathrm{R}}(f)$
Transducers (ideal)	Gyrator I		$e_2=r\,f_1$ $e_1=r\,f_2$	$e_2=r(x)\,f_1$ $e_1=r(x)\,f_2$
	Gyrator II		$f_1=\frac{1}{r}e_2$ $f_2=\frac{1}{r}e_1$	$f_1=\frac{1}{r(x)}e_2$ $f_2=\frac{1}{r(x)}e_1$
	Transformer I		$f_2=\mu\,f_1$ $e_1=\mu\,e_2$	$f_2=\mu(x)\,f_1$ $e_1=\mu(x)\,e_2$
	Transformer II		$f_1=\frac{1}{\mu}f_2$ $e_2=\frac{1}{\mu}e_1$	$f_1=\frac{1}{\mu(x)}f_2$ $e_2=\frac{1}{\mu(x)}e_1$
Sources	Source of effort		$e=e(t)$	
	Source of flow		$f=f(t)$	
Junctions	Zero (0)		$e_1=e_2=e_3$ $f_1+f_2=f_3$	
	One (1)		$f_1=f_2=f_3$ $e_1+e_2=e_3$	

3.2.5 *Junction*

Power bond has two kinds of junctions *i.e.* 0 junctions and 1 junction

- In 0 junctions sum of flow are zero and effort is equal. This corresponds to a node in an electrical circuit or to a mechanical "stack" in which all the forces are equal.
- In a 1 junction, the sum of the efforts is zero and the flows are equal. This corresponds to an electrical loop or to a force balance at a mass in a mechanical system.

3.3.6 *Causality*

Bond graphs have a notion of causality and which side of bond determines the instantaneous effort and instantaneous flow. Causality defines the effect and causes relationship between two factor power. The inputs and the outputs are specified by the causal stroke which is shown by a small transverse line at one end of a bond. The effort and flow signals are directed towards the causal stroke and the un-stroked end of the bond, respectively. An example of causality as shown in figure that A receive flow (f) and give effort (e) output to B as causal stroke. On the other hand B receives the effort and gives flow to A.

(a) (b)

Fig. 3.3: (a) Exchange of Power (b) Causality in Bond Graph Representation

The proper causality for independent energy storage elements (I or C) which yield state variables is called integral causality, *i.e.* the cause is integrated to generate the effect. The state variables in a bond graph model are generalized momentums (p) and generalized charges (Q).

Causality is a symmetric relationship. When one side "causes" effort, the other side "causes" flow. Active components such as an ideal voltage or current source are also causal. In bond graph notation, causal stroke may be added to one stroke of the power bond to indicate that the opposite end is defining the effort. Consider torque motor driving a wheel, i.e. source of effort SE.

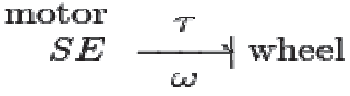


Fig. 3.4: Word bond graph motor to wheel

The side with causal stroke *i.e.* in the side of the wheel, it defines the flow of the bond. Only one end of a power bond can define the effort and so only one end of a bond can have a causal stroke. The two passive components with time-dependent behaviour, *I* and *C*, can only have one sort of causation: an *I* component determines flow; a *C* component defines effort. So from a junction *J*, the only legal configurations for I and *C* are

Fig.3.5: Causality of inertia and compliant element

A resistor has no time-dependent behaviour therefore apply a voltage and get the flow or apply a flow and get a voltage instantly or resistor can be end of the causal bond.

Fig.3.6: Causality of R element

Transformers are passive, neither dissipating nor storing energy and causality would be passed through it

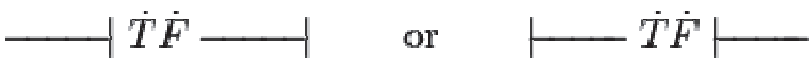


Fig. 3.7: Causality of transformer

A gyrator transforms the flow to effort and effort to flow, so if flow is caused on one side, effort is caused on the other side

Fig. 3.8: Causality of Gyrator

In a 0-junction, efforts are equal; in a 1-junction, flows are equal. Thus, with causal bonds, only one bond can cause the effort in a 0-junction and only one can cause the flow in a 1-junction. Thus, if the causality of one bond of a junction is known, the causality of the others is also known. That one bond is called the strong bond

Fig.3.9: Causality of junction element

- **Differential Causality**

Reverse of an integral causal orientation of storage elements is called differential causality or differential causal orientation or other hand differential causality arises for system models in which some storage elements are not dynamically independent. It does not mean that the parameters of these storage elements do not appear in the equations. Any storage element with differential causality does not yield state variables. A differential causalled C element and I element are shown in figure.

Fig.3.10: Differential Causality of C and I element

- **Activation**

Bond represented an exchange of power as well as information of flow and effort. Pure exchange of information is represented by the activation of the bond. A bond may either be flow activated or effort activated. In bond graph, some bonds can give only the information carriers. Therefore, such a bond where the power factor is masked are called activated bond. Signals may be represented in bond graph form as the so-called activated bonds. The power in an activated/signal bond is zero, which means a signal bond does not energetically interact with the system. For example, in a bond representing the velocity pick-up, the information of force must be masked and in a bond representing a force sensor, the information of the flow must be masked. A full arrow with causal stroke shows that some information is allowed to pass and some information is masked (refer to Table 3.3).

3.3.7 *Sensors and Actuators*

Additional pseudo-states can be added for measurement of any factor of power on a bond graph model by using pseudo-storage elements. A flow activated integrally causally C-element (DF *i.e.* detector of flow) would observe the flow (and consequently time integral of flow), whereas an effort activated integrally causally I-element (De *i.e.* detector of effort) would observe the effort (and consequently generalized momentum). Activated storage elements are perceived as the conceptual instrumentation on a model. They do not interfere with the dynamics of the system. The symbols for the effort and the flow detectors are shown in Table. 3. Detectors are usually weak bonds at the connected junctions. Therefore, the flow detector is usually connected to 1-junction whereas the effort detector is usually connected to 0-junction.

Table 3.3: Definition of Signal Bonds and Sensors

Name	Type	Symbol	Definition
Signal Bond	Effort		$e = e(t),\ f = 0$
	Flow		$f = f(t),\ e = 0$
Sensors (Detectors)	Effort	 De : e	$e = e(t),\ f = 0$
	Flow	 Df : f	$f = f(t),\ e = 0$

Activated bonds are used to implement modulated sources, and non-linear constitutive relations of two-port and one-port elements. These are called modulated sources, elements or two-ports and their bond graph notations are given by prefixing 'M' to their respective normal symbolic names. The modulating signal(s) may come from a real sensor (De or Df , if the modulation is due to control action) or through an activated bond emanating from a junction.

MS_e and MS_f are modulated sources whose outputs depend on one of more modulating signals, which are shown as inputs connected to the elements. Likewise, modulated two ports (MTF and MGY) have variable transformation ratio and the modulating signal is an additional signal input to the two-port besides its two power bonds. Modulated I, C and R elements are defined similarly.

3.2 SERIAL MANIPULATOR

Master slave robotic systems have been employed in surgical robotics for enhancing surgeons Dexterity as well as enabling motion scaling. The most famous surgical robotic system, the "da Vinci Surgical System," has its own master manipulator design, facilitating intuitive manipulation of the slave robotic arms. Many research groups have also tried to design their original master manipulators; however, to the best of the present knowledge, there is limited design strategy for surgical master manipulators. Master manipulator design affects system usability, the slave manipulators motion accuracy, and operator workload. Further, it is obvious that better master manipulator design facilitates the introduction of surgical robots to clinical cases.

Now, the base design on behalf of which the robotic manipulator is designed for the surgical system is shown in Fig. 3.11.

Fig. 3.11: Base Design

Fig. 3.12: End-effector of base design

In Fig. 3. The end part is shown, which shows the opening and closing function of the end-effectors. This part has no function in a robotics system of the manipulator, it is just to grip and leave the object to which surgeon is dealing with during operation. Determination of Position and Orientation of tooltip with respect to the base co-ordinate frame of the manipulator is known as forward kinematics. Position and orientation of tooltip are a function of joint variables. These joint variables may be of two types as follows:

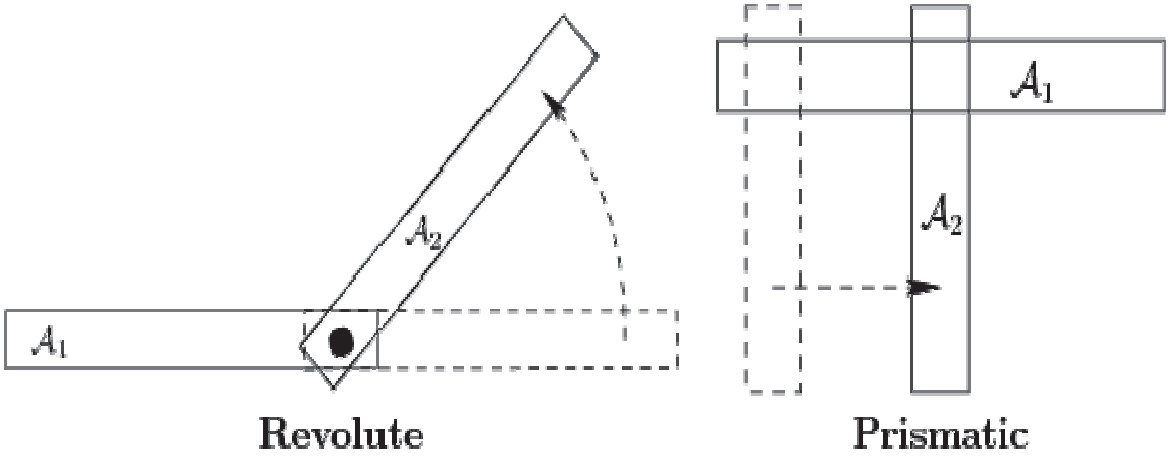


Fig. 3.13: Revolute and Prismatic Joint Schematic Representation [**47**].

1. Revolute joint (Θ)
2. Prismatic joint (L)

Revolute joint is motion of a link about an axis where's in prismatic joint motion of link is along an axis. The figure shows both joints.

In forward Kinematics, equation of motion is formed by means of *Homogeneous link co-ordinate transformations*. By using these transformations tooltip co-ordinate frames are mapped to a base co-ordinate frame. The final expression is a 4X4 homogeneous matrix representing orientation and position of the tool tip. In, final expression when values of joint variables for home position are substituted gives corresponding position and orientation of tool tip.

3.4 LINE DIAGRAM OF MANIPULATOR

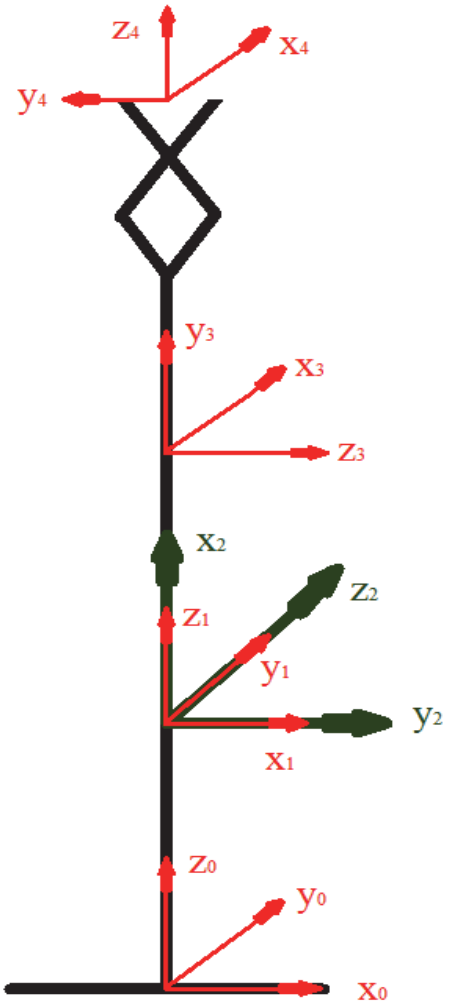


Fig. 3.14: Line Diagram of manipulator

In the Fig. 3.14, it shows the line diagram of the robotic manipulator on which Denavit-Hartenberg (D-H) approach and Bond Graph is applied and compared those results come out. In this figure five coordinate systems are shown from *0* to *4*. Except 0^{th} and 4^{th} coordinate system, all other coordinate systems represent the joints which rotate about its corresponding *z-axis* like z_1, z_2, z_3. The 0^{th}-coordinate system is base system which is fixed and all the joints are moving with respect of this system and the 4^{th} coordinate system is end tool or end-effectors system which represents the movement of tool only and this system has no effect on robotic system calculations for it three joints. It shows the opening and closing directions of end-effectors along y_4-*axis* and its end point approach along z_4-*axis* which is outwards. Joint parameters are specified relative position and orientation of two successive links, as shown in Fig. 3.15

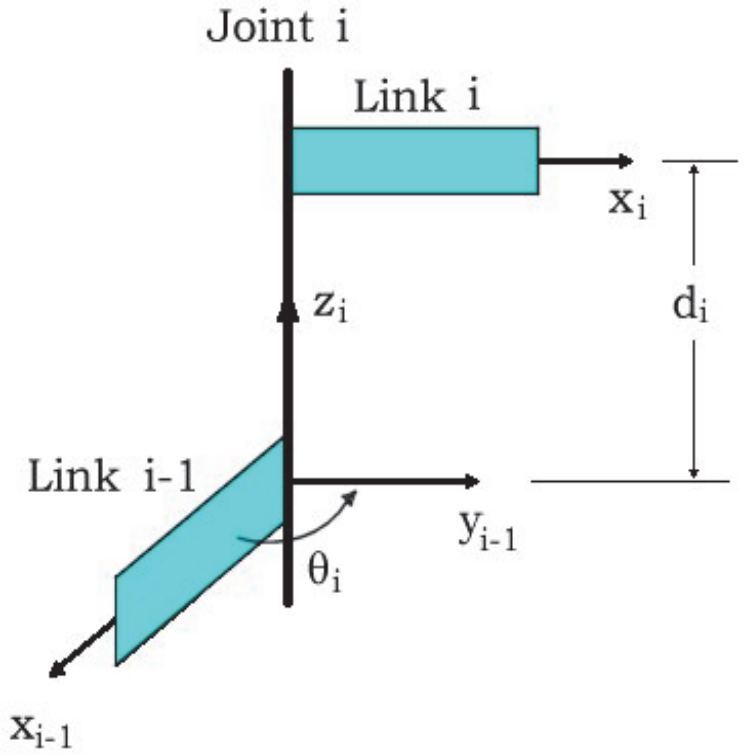


Fig. 3.15: Joint angle θ and joint distance *d*.

Here joint *i* connects link *i-1* to link *k*. The first joint parameter, θ_i is called the joint angle and it is the rotation about z_i needed to make axis x_{i-1} parallel with axis x_i. The second joint parameter, d_i is called the joint distance and it is the translation along z_i needed to make axis x_{i-1} intersect with axis x_i. It has always been the case that one of these parameters is fixed and the other is variable. Link parameters are species the relative position and orientation of the axis of two successive joints, as shown in Fig. 3.16.

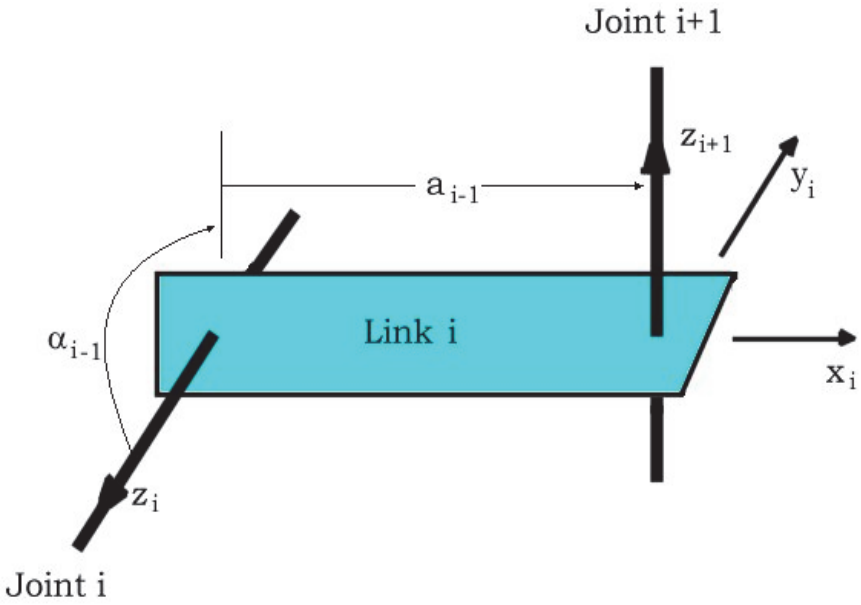


Fig. 3.16: Link length a_{i-1} and link twist angle α_{i-1}

Here, link *i* connects joint *i* to joint *i+1*. The parameters associated with link *i* are defined with respect to x_i, which is a common normal between the axis of joint *i* and the joint *i+1*. The first link parameter, a_{i-1} is called the link length and it is the translation along x_i needed to make axis z_i intersect with axis z_{i+1}. The second link parameter, α_{i-1} is called the link twist angle and it is the rotation about x_i needed to make axis z_i parallel to axis z_{i+1}. The link parameters are always constant and specified as part of the mechanical design.

Overall, we are calculating the relation between 0^{th} and 4^{th} coordinate system while passing through all motions or rotating joints. It's to find a final transformation matrix between these two coordinate systems by multiplying all relative matrices between each joint or coordinate system which are deduced by having its parameters for each system. Now, for calculating parameters, D-H representations followed in next topic after defining its basics that how to define different axis and process to calculate parameters and then putting joint variables one by one in a formulated matrix.

3.5 CONVENTIONAL D-H ALGORITHM

- **For Manipulator's Forward Kinematics**

Denavit-Hartenberg developed an algorithm to find out geometric parameters of a manipulator. It is a systematic notation for assigning right-handed orthonormal coordinate frames. These parameters include *joint variable* (θ_i) which is either of revolute or prismatic type as explained earlier, *joint distance* (d_i), *link length* (a_{i-1}) and *link twist angle* (α_{i-1}).

According to this algorithm, there are some rules that must be followed step by step in order to find out these parameters. These rules are as follows:

1. Draw neat and clean line diagram of the manipulator. In line diagram only axis of links are shown. After drawing line diagram, identify and draw with lines of infinite length, all the motion axis.
2. Mark these motion axis as z-axis.
3. Next, step is to assign x-axis, which is normal to both z_i and z_{i+1} at their point of intersection. In case if they don't intersect then x-axis will be drawn such that it is common normal to both away from z_i.
4. Next step is to assign y-axis which is assigned by means of *right hand thumb rule*.
5. After doing this next step is to determine these parameters which are as follows:
    - *Link twist angle* (α_{i-1}): It is the angle from z_i to z_{i+1} about x_i.
    - *Link length*(a_{i-1}): It is the distance from z_i to z_{i+1} along x_i.
    - *Joint distance* (d_i): It is the distance from x_{i-1} to x_i along z_i.
    - *Joint variable* (θ_i): It is the angle from x_{i-1} to x_i about z_i.

3.6 FORWARD KINEMATICS THROUGH D-H ALGORITHM

The Surgical robotic manipulator is a robot having three degrees of freedom and all rotational joints (*i.e.*, it is a 3R mechanism). All the rotations of this manipulator are perpendicular to each other and also at some distance with one another. The 3 Dimensional model is shown in Fig. 3.18.

Fig. 3.17: 3D model of manipulator [**45**]

The link coordinates frames are shown in Fig. 3.18.

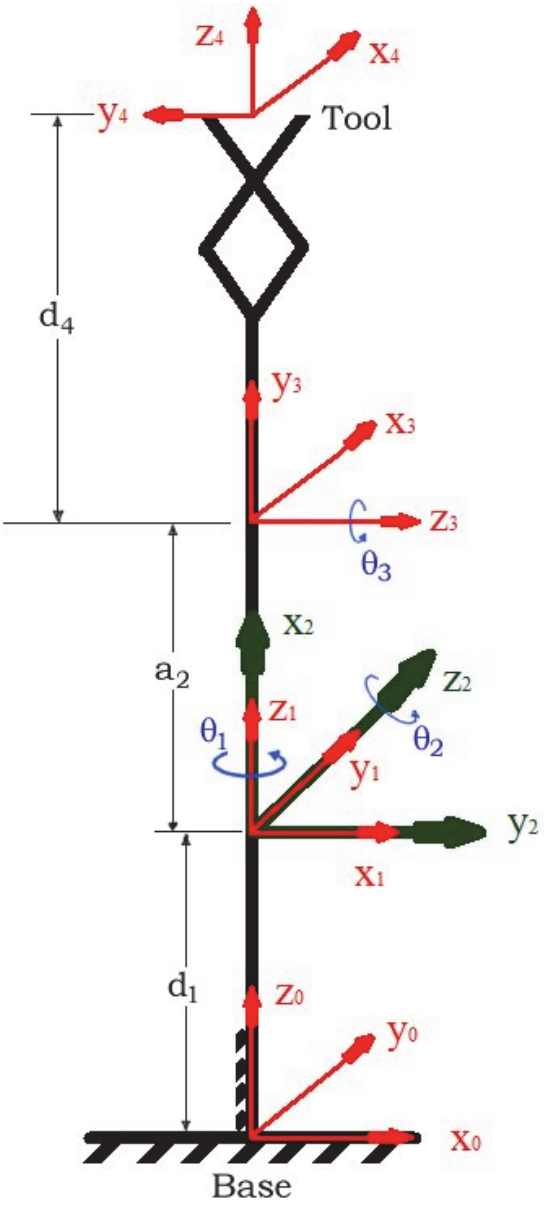


Fig. 3.18: Link coordinates of the manipulator.

Once a set of long coordinates is assigned using a D-H algorithm, then the transformation from coordinate frame *i+1* to coordinate frame *i* is to be done using a homogeneous coordinate transformation matrix. By multiplying all coordinate transformation matrices together, will deduce a composite homogeneous coordinate transformation matrix called as *arm matrix* which transforms or map tool coordinate into base coordinate. To find arm matrix there is a need to find joint parameters and link parameters by following step 5. After this one by one parameter is to put in a 4×4 matrix, which is formulated in the algorithm. In final we get an arm matrix as ${}^{Base}_{Tool}T$. Now, the parameters of all the rotational

joints are shown in Table 3.4. Here, there are four parameters because in this notation one extra base coordinate system is taken.

Table 3.4 Kinematic Parameters.

Link	a_{i-1}	d_i	α_{i-1}	θ_i	Home
1	0	d_1	0	θ_1	0
2	0	0	$-\pi/2$	θ_2	$-\pi/2$
3	a_2	0	$-\pi/2$	θ_3	$-\pi/2$
4	0	d_4	$-\pi/2$	θ_4	0

Now, the four link parameters is to put in the ${}^{i-1}_{i}T$ matrix one by one and then make its multiplication for arm matrix.

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1} $$

Here, ${}^{i-1}_{i}T$ denotes the transformation from the coordinate frame *i* to coordinate frame *i-1*. Generally, *T* denotes a homogeneous coordinate transformation, the superscript being the index of the source coordinate frame and the subscript being the index of the destination or reference, coordinate frame. The final expression is like:

$$ {}^{Base}_{Tool}T = {}^{0}_{1}T(\theta_1){}^{1}_{2}T(\theta_2){}^{2}_{3}T(\theta_3){}^{3}_{4}T(\theta_4) \tag{3.2} $$

Now, by putting all the parameters on links each in which *Home column* values should keep constant or can say variable terms should not be assigned in matrix till the arm matrix is deduced.

$$ {}^{0}_{1}T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3} $$

$$ {}^{1}_{2}T = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.4} $$

$$ {}^{2}_{3}T = \begin{bmatrix} c_3 & -s_3 & 0 & a_2 \\ 0 & 0 & 1 & 0 \\ -s_3 & -c_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.5} $$

$$ {}^{3}_{4}T = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ -s_4 & -c_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6} $$

The final arm matrix is

$$ {}^{0}_{4}T = $$

$$ \begin{bmatrix} c_1c_2c_3c_4 + s_1s_3c_4 + c_1s_2s_4 & -c_1c_2c_3s_4 - s_1s_3s_4 + c_1s_2c_4 & -c_1c_2s_3 + s_1c_3 & (-c_1c_2s_3 + s_1c_3)d_4 + c_1c_2a_2 \\ s_1c_2c_3c_4 - c_1s_3c_4 + s_1s_2s_4 & -s_1c_2c_3s_4 - c_1s_3s_4 + s_1s_2c_4 & -s_1c_2s_3 - c_1c_3 & (-s_1c_2s_3 - c_1c_3)d_4 + s_1c_2a_2 \\ -s_2c_3c_4 + c_2s_4 & s_2c_3s_4 + c_2c_4 & s_2s_3 & s_2s_3d_4 - s_2a_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.7} $$

The final expression for the position orientation of the tool relative to the base now depends on all of the kinematic parameters. It can make a partial check by evaluating the arm matrix at the soft home position, which from Table 3. is θ_1=[0, -π/2, -π/2, 0]^T. This yields

$$ {}^{0}_{4}T \; or \; {}^{Base}_{Tool}T \; = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & d_4 + a_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.8} $$

By this equation its observed or calculate that, the first row shows that axis y_4 of the tool frame has coordinates (0, -1, 0) in the base frame, means y_4 is in opposite direction with respect to x_0. Similarly, the second row shows that axis x_4 of the tool frame has coordinates (1, 0, 0) in the base frame, means x_4 is in the same direction with respect to y_0 and the third row shows that axis z_4 of the tool frame has coordinates (0, 0, 1) in the base frame, means z_4 is in the same direction with respect to z_0. Finally the position vector or fourth column indicates that the coordinates of the origin of the tool pitch frame L_4 relative to the base frame L_0 are (0, 0, d_4+ a_2+ d_1, 1). It means L_4 is located at a distance d_4+ a_2+ d_1 above the base. By putting d_1=300, a_2=10, and d_4=40 unit values in equation 3.8, the result is:

$$ {}^{0}_{4}T \; or \; {}^{Base}_{Tool}T \; = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 350 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.9} $$

For calculating and for confirmation of results, a program is made in Mat Lab as shown below:

Program in Mat Lab for Forward Kinematics after finding parameters.

```
%%SETTINGS
clc
clear all

% JOINT ANGLES
```

```
Q1 = sym('Q1');
Q2 = sym('Q2');
Q3 = sym('Q3');
Q4 = sym('Q4');

S1=sym('sin(Q1)');
C1=sym('cos(Q1)');

S2=sym('sin(Q2)');
C2=sym('cos(Q2)');

S3=sym('sin(Q3)');
C3=sym('cos(Q3)');

S4=sym('sin(Q4)');
C4=sym('cos(Q4)');
d1=sym('d1');
d4=sym('d4');
a2=sym('a2');
T01=[C1,-S1,0,0; S1,C1,0,0; 0,0,1,d1; 0,0,0,1];
T12=[C2,-S2,0,0; 0,0,1,0; -S2,-C2,0,0; 0,0,0,1];
T23=[C3,-S3,0,a2; 0,0,1,0; -S3,-C3,0,0; 0,0,0,1];
T34=[C4,-S4,0,0; 0,0,1,d4; -S4,-C4,0,0; 0,0,0,1];
T= T01*T12*T23*T34;
T
```

Solution or Output of the program is T, which represents ${}^{0}_{4}T$ *or* ${}^{Base}_{Tool}T$:

```
T =
1st row=
[cos(Q4)*(sin(Q1)*sin(Q3)+cos(Q1)*cos(Q2)*cos(Q3))+cos(Q1)*sin(Q2)*sin(Q4),
cos(Q1)*cos(Q4)*sin(Q2)-sin(Q4)*(sin(Q1)*sin(Q3)+cos(Q1)*cos(Q2)*cos(Q3)),
cos(Q3)*sin(Q1)-cos(Q1)*cos(Q2)*sin(Q3),
d4*(cos(Q3)*sin(Q1) - cos(Q1)*cos(Q2)*sin(Q3)) + a2*cos(Q1)*cos(Q2)] ;
2nd row=
[sin(Q1)*sin(Q2)*sin(Q4)-cos(Q4)*(cos(Q1)*sin(Q3)-cos(Q2)*cos(Q3)*sin(Q1)),
sin(Q4)*(cos(Q1)*sin(Q3)-cos(Q2)*cos(Q3)*sin(Q1))+cos(Q4)*sin(Q1)*sin(Q2),
-cos(Q1)*cos(Q3)-cos(Q2)*sin(Q1)*sin(Q3),
a2*cos(Q2)*sin(Q1) - d4*(cos(Q1)*cos(Q3) + cos(Q2)*sin(Q1)*sin(Q3))] ;
3rd row=
[cos(Q2)*sin(Q4) - cos(Q3)*cos(Q4)*sin(Q2),
cos(Q2)*cos(Q4) + cos(Q3)*sin(Q2)*sin(Q4),
sin(Q2)*sin(Q3),
d1 - a2*sin(Q2) + d4*sin(Q2)*sin(Q3)] ;
4th row=
[0,       0,       0,       1]
```

Now, by putting θ_i, a_{i-1} and d_i values as:

θ_1=0	θ_2=-π/2	θ_3=-π/2	θ_4=0	d_1=300	a_2=10	d_4=40

As the a_{i-1} and d_i values are given as per the manipulator.

The Output of ${}^{0}_{4}T$ *or* ${}^{Base}_{Tool}T$:

```
T= [  0,    -1,     0,     0;
      1,     0,     0,     0;
      0,     0,     1,   350;
      0,     0,     0,     1]
```

Thus, it is similar to the other one.

3.7 INVERSE KINEMATICS THROUGH D-H ALGORITHM

The solution to the inverse kinematics problem can be approached either iterative technique, numerically or analytically. Here, how to compute the set of joint angles which leads to the desired results by giving the desired position and orientation of the tool relative to the station. In this section the focus is the Inverse Kinematics of the manipulator.

The problem of solving the kinematic equations of a manipulator is a nonlinear one given the numerical value of ${}^{0}_{N}T$ *or* ${}^{Base}_{Tool}T$ is attempting to find values of θ_1, θ_2,....., θ_n. Consider the matrix given in *Equ. (3.7)*. In this case problem is, given ${}^{0}_{4}T$ as sixteen numeric values, solve for the three joint angles θ_1 through θ_3. By *Equ. (3.7),* the case of the arm with three degrees of freedom, twelve terms and three unknowns. However, among these the nine terms arising from the rotation, port, only three terms are independent and the three terms of the position vector part.

Numerous methods are available to find the zeroes of the *Equ. (3.7)*. However, the methods are general, *iterative.* The most common method is known as the *Newton-Rapson method*. In the iterative technique, to solve the kinematic equations:

$$ {}^{0}_{4}T(\theta_i) = 0 \tag{3.10}$$

for variables θ_i, it is to start with an initial guess

$$ \theta_i^* = \theta_i + \delta\theta_i \tag{3.11}$$

For the joint variables using the forward kinematics, it can determine the configuration of the end-effector frame for the guessed joint variable.

$$ {}^{0}_{n}T^*(\theta_i) = {}^{0}_{n}T(\theta_i^*) \tag{3.12}$$

The difference between the configuration calculated with the forward kinematics and the desired configuration represents an error, called residue, which must be minimised.

$$ {}^{0}_{n}\delta T(\theta_i) = {}^{0}_{n}T(\theta_i) - {}^{0}_{n}T^{*}(\theta_i) \tag{3.13} $$

The first Taylor expansion of the set of equations is:

$$ {}^{0}_{4}T(\theta_i) \ = {}^{0}_{4}T(\theta_i^{*} + \delta\theta_i) \tag{3.14} $$

$$ {}^{0}_{4}T(\theta_i) \ = {}^{0}_{n}T(\theta_i^{*}) \ + \frac{\partial T}{\partial \theta}\delta\theta_i + 0(\delta\theta_i^{2}) \tag{3.15} $$

Assuming $\delta\theta_i$<< I allow us to work with a set of linear equations

$$ {}^{0}_{4}T(\theta_i) \ = J * \delta\theta_i \tag{3.16} $$

where *J* is the Jacobian matrix of the set of terms in fourth column of *Equ. (3.7)*. To find Jacobian which is deduced by:

$$ J(\theta_i) = \left[\frac{\partial T_k}{\partial \theta_i}\right] \tag{3.17} $$

that implies

$$ \delta\theta_i = J^{-1} * \delta T \tag{3.18} $$

Therefore, the unknown variables θ_i are:

$$ \theta_i = \theta_i^{*} + J^{-1} * \delta T \tag{3.19} $$

Now, use the values obtained by *equation 3.19* as a new approximation to repeat the calculations and find newer values. Repeating the methods can be summarized in the following iterative equation to converge to the exact value of the variables.

$$ \theta_{i+1} = \theta_i + J^{-1}(\theta_i) * \delta T(\theta_i) \tag{3.20} $$

Now, to solve the inverse kinematics of thr manipulator and find the joint coordinates for a known position of the tip point, its define that

$$ \theta_i = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \tag{3.21} $$

$$ T_k = \begin{bmatrix} (-c_1c_2s_3 + s_1c_3)d_4 + c_1c_2a_2 \\ (-s_1c_2s_3 - c_1c_3)d_4 + s_1c_2a_2 \\ s_2s_3d_4 - s_2a_2 + d_1 \end{bmatrix} \tag{3.22} $$

Therefore, the Jacobian of the equations is:

$$ J(\theta_i) = \left[\frac{\partial T_k}{\partial \theta_i}\right] = \begin{bmatrix} \partial e_x/\partial\theta_1 & \partial e_x/\partial\theta_2 & \partial e_x/\partial\theta_3 \\ \partial e_y/\partial\theta_1 & \partial e_y/\partial\theta_2 & \partial e_y/\partial\theta_3 \\ \partial e_z/\partial\theta_1 & \partial e_z/\partial\theta_2 & \partial e_z/\partial\theta_3 \end{bmatrix} \tag{3.23} $$

$$J(\theta_i) = \begin{bmatrix} (s_1c_2s_3 + c_1c_3)d_4 + s_1c_2a_2 & (c_1s_2s_3)d_4 - c_1s_2a_2 & (-c_1c_2c_3 - s_1s_3)d_4 \\ (-c_1c_2s_3 + s_1c_3)d_4 + c_1c_2a_2 & (s_1s_2s_3)d_4 - s_1s_2a_2 & (-s_1c_2c_3 - c_1s_3)d_4 \\ 0 & (c_2s_3)d_4 - c_2a_2 & (s_2c_3)d_4 \end{bmatrix} \quad (3.24)$$

Now, the inverse of the Jacobian $J^{-1}(\theta_i)$ is to find and then put all the values in

$$\theta_{i+1} = \theta_i + J^{-1}(\theta_i) * (T_k - T_k^*) \quad (3.25)$$

Let's assume

θ_1=0	θ_2=-π/4	θ_3=-π/4	θ_4=0	d_1=300	a_2=10	d_4=40

By putting θ_i and d_i values in T_k matrix, the new matrix is T_k^* and T_k assumed as [20, 10, 320]^T.

All these calculations are much complex and cannot be solved manually, so Mat Lab tool is used to deduce these equations.

Program to find Jacobian

```
%%SETTINGS
clc
clear all

% JOINT ANGLES
Q1 = sym('Q1');
Q2 = sym('Q2');
Q3 = sym('Q3');
Q4 = sym('Q4');

S1=sym('sin(Q1)');
C1=sym('cos(Q1)');

S2=sym('sin(Q2)');
C2=sym('cos(Q2)');

S3=sym('sin(Q3)');
C3=sym('cos(Q3)');

S4=sym('sin(Q4)');
C4=sym('cos(Q4)');


d1=sym('d1');
d4=sym('d4');
a2=sym('a2');

T01=[C1,-S1,0,0; S1,C1,0,0; 0,0,1,d1; 0,0,0,1];
T12=[C2,-S2,0,0; 0,0,1,0; -S2,-C2,0,0; 0,0,0,1];
T23=[C3,-S3,0,a2; 0,0,1,0; -S3,-C3,0,0; 0,0,0,1];
T34=[C4,-S4,0,0; 0,0,1,d4; -S4,-C4,0,0; 0,0,0,1];

T= T01*T12*T23*T34;

DT141 = diff(T(1,4),Q1);
```

```
DT142 = diff(T(1,4),Q2);
DT143 = diff(T(1,4),Q3);

DT241 = diff(T(2,4),Q1);
DT242 = diff(T(2,4),Q2);
DT243 = diff(T(2,4),Q3);

DT341 = diff(T(3,4),Q1);
DT342 = diff(T(3,4),Q2);
DT343 = diff(T(3,4),Q3);

J = [DT141 DT142 DT143 ; DT241 DT242 DT243 ; DT341 DT342 DT343 ];
DE = [T(1,4); T(2,4); T(3,4)];
JINV = inv(J);

DQ = JINV * DE;
DQ
DE
J
```

Output of the program is introduce or taken in next program

Program for Newton-Rapson method

```
clc
clear all

Q1 = 0;
Q2 = -45;
Q3 = -45;
Q4 = 0;

d4=40;
d1=300;
a2=10;

lno=1;
otput=zeros(7, 7);

J0 = [ d4*(cosd(Q1)*cosd(Q3) + cosd(Q2)*sind(Q1)*sind(Q3))-
a2*cosd(Q2)*sind(Q1), d4*cosd(Q1)*sind(Q2)*sind(Q3) - a2*cosd(Q1)*sind(Q2),
-d4*(sind(Q1)*sind(Q3) + cosd(Q1)*cosd(Q2)*cosd(Q3));
d4*(cosd(Q3)*sind(Q1) - cosd(Q1)*cosd(Q2)*sind(Q3)) + a2*cosd(Q1)*cosd(Q2),
d4*sind(Q1)*sind(Q2)*sind(Q3) - a2*sind(Q1)*sind(Q2),d4*(cosd(Q1)*sind(Q3)
- cosd(Q2)*cosd(Q3)*sind(Q1));
 0, d4*cosd(Q2)*sind(Q3) - a2*cosd(Q2), d4*cosd(Q3)*sind(Q2)]; % 3 by 3


DE1 = [d4*(cosd(Q3)*sind(Q1) - cosd(Q1)*cosd(Q2)*sind(Q3)) +
a2*cosd(Q1)*cosd(Q2);
 a2*cosd(Q2)*sind(Q1) - d4*(cosd(Q1)*cosd(Q3) +
cosd(Q2)*sind(Q1)*sind(Q3));
                                  d1 - a2*sind(Q2) + d4*sind(Q2)*sind(Q3)];

JINV0=inv(J0);

DE0=[20; 10; 320];
DDE0 = DE0 - DE1;

DQ0 = [Q1; Q2; Q3]*(pi/180);
```

```
DQ1 = DQ0 + (JINV0 * DDE0);
k11= DQ1(1,1);
k21= DQ1(2,1);
k31= DQ1(3,1);
DQ1D=DQ1*(180/pi);

Q11=DQ1D(1,1);
Q21=DQ1D(2,1);
Q31=DQ1D(3,1);
otput(lno,:)=[lno,k11,k21,k31,Q11,Q21,Q31];

for i=1: 150
J1 = [ d4*(cosd(Q11)*cosd(Q31) + cosd(Q21)*sind(Q11)*sind(Q31)) -
a2*cosd(Q21)*sind(Q11), d4*cosd(Q1)*sind(Q21)*sind(Q31) -
a2*cosd(Q11)*sind(Q21), -d4*(sind(Q11)*sind(Q31) +
cosd(Q11)*cosd(Q21)*cosd(Q31));
d4*(cosd(Q31)*sind(Q11) - cosd(Q11)*cosd(Q21)*sind(Q31)) +
a2*cosd(Q11)*cosd(Q21), d4*sind(Q11)*sind(Q21)*sind(Q31) -
a2*sind(Q11)*sind(Q21),  d4*(cosd(Q11)*sind(Q31) -
cosd(Q21)*cosd(Q31)*sind(Q11));
                                    0,
d4*cosd(Q21)*sind(Q31) - a2*cosd(Q21),
d4*cosd(Q31)*sind(Q21)];
 JINV1=inv(J1);

DE1 = [d4*(cosd(Q31)*sind(Q11) - cosd(Q11)*cosd(Q21)*sind(Q31)) +
a2*cosd(Q11)*cosd(Q21);
 a2*cosd(Q21)*sind(Q11) - d4*(cosd(Q11)*cosd(Q31) +
cosd(Q21)*sind(Q11)*sind(Q31));
                                  d1 - a2*sind(Q21) +
d4*sind(Q21)*sind(Q31)];
DDE1=DE0-DE1;
DQ1 = DQ1 + (JINV1 * DDE1);

k11= DQ1(1,1);
k21= DQ1(2,1);
k31= DQ1(3,1);

DQ1D=DQ1*(180/pi);

Q11=DQ1D(1,1);
Q21=DQ1D(2,1);
Q31=DQ1D(3,1);
lno=lno+1;
otput(lno,:)=[lno,k11,k21,k31,Q11,Q21,Q31];
 if( (abs(int64(k31)) == abs(int64(k21))) &&
(abs(int64(k31))==abs(int64(k11))))
     break;
 end

end
 k11
 k21
 k31
 Q11
 Q21
 Q31
```

The Output result is taken from the iteration values when the all θ_i values become constant or can say won't change for the next iterations. That is the desired output, here, which it comes as

$\theta_1(in\ degree)$	θ_1=206.57	θ_2=-318.19	θ_3=-270	θ_4=0

The simulation is to be done for the confirmation the results, which is in next section of the chapter.

3.8 SIMULATION RESULTS OF D-H ALGORITHM

To confirm or validate this result, put all values of obtained θ_i and our manipulators d_i values in forward kinematics program. In the output of the program, the obtained matrix tells about all position and orientation coordinates with respect to the base frame.

$$ {}^{0}_{4}T \ or \ {}^{Base}_{Tool}T \ = \begin{bmatrix} -0.4473 & -0.5963 & 0.6666 & 19.9992 \\ 0.8944 & 0-0.2982 & 0.3334 & 10.0018 \\ 0 & 0.7454 & 0.6667 & 319.9999 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \quad (3.26) $$

Here, the last column is having same values as it is taken or assume values in the inverse kinematics program which are [20, 10, 320,]^{T}. It has ±0.0008 error only which is bearable for the time being. The rotation part tells the relation between tool coordinate system to the base coordinate system. In next section bond graph modelling is done for deducing forward and inverse kinematics.

3.9 BOND GRAPH MODELLING

In Bond Graph the only requirement is to convey some essential information of the system to the computer. This information needs to be conveyed in an unambiguous manner so that, every information that is conveyed to the computer can follow some algorithmic procedure to generate the required differential equations. So, the main idea in Bond Graph is to formulate some kind of language in which the essential characteristics of the system can be conveyed to the computer so that it can follow an algorithmic procedure which is already programmed into it in order to derive the differential equation. Now, the question arises that what are the essential characteristics that are to be conveyed to the computer.

In any system, there are elements such as capacitive, resistive and inertial elements that make up the whole system. Now, it is important to know the nature of interaction between these elements. They interact essentially by exchanging energy. For example, if there is a resistor in the circuit, it absorbs energy and then dissipates it into the environment. In a nutshell, it is the energy that is exchanged between different elements of the system residing in different energy domains. Further, it is not only the energy, but also the information that is exchanged between the elements of the system. Basically, energy per unit time is a power and this power is composed of two variables *i.e.*, effort and flow. It is a product of effort variable and flow variable. In case of mechanical system, the effort variables are force, torque and the flow variables are linear and angular velocity.

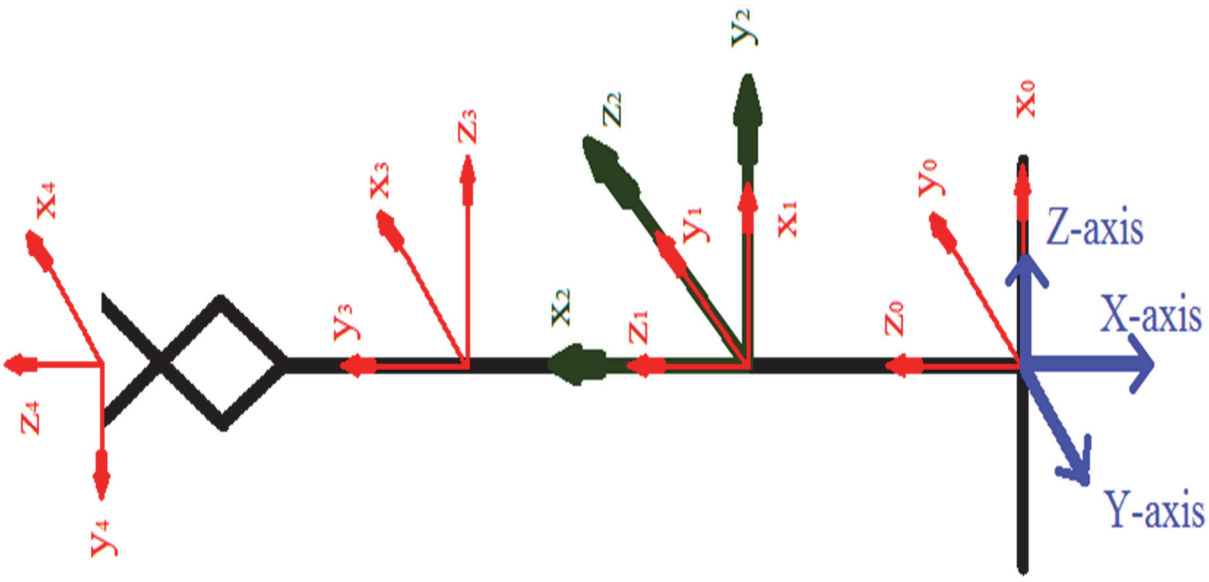


Fig. 3.19: Serial Manipulator with coordinate system

3.9.1 *Forward Kinematics*

Now, for the first rotation about *x-axis* is considered along Serial Manipulator. The bond graph model is shown in Fig. 3.20. In this model motor is also introduced to give rotation about *x-axis* only and the output is taken of *y* and *z* coordinates distance change linearly.

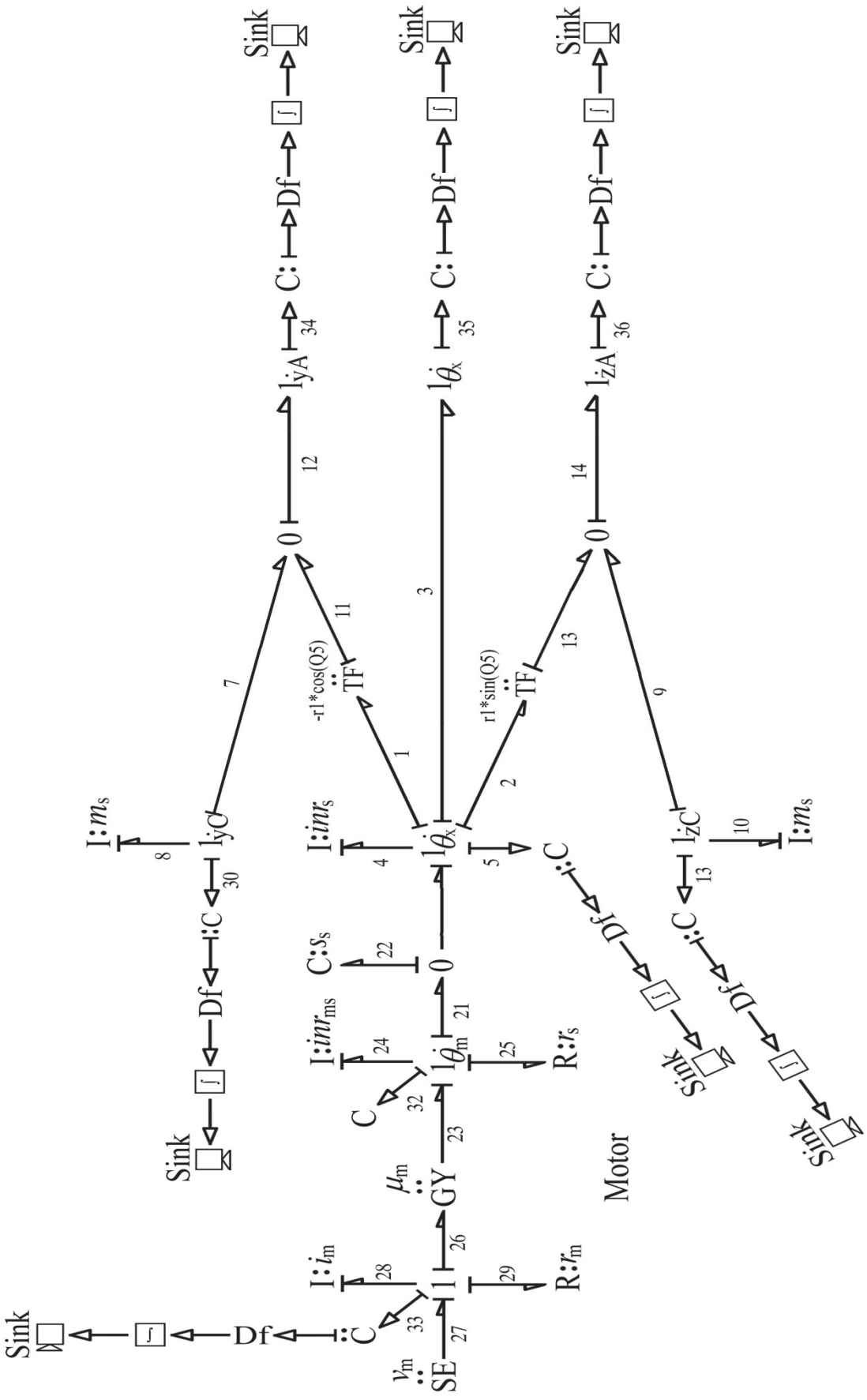


Fig. 3.20: Bond Graph Model for *x-axis* rotation

In this above bond graph model for *x-axis* rotation, to give angular rotation DC motor is used. The motor is made with the help of different elements of bond graph modelling. In beginning external source of input as effort is represented by SE. Through this voltage is given to the motor then effort goes to I-element and comes back as flow or current which then goes to GY via passing through R-element. Now, GY converts voltage into angular displacement and current into torque, the shaft of the motor gets rotated via a flywheel as I-element and resistance R-element and stiffness as C-element represented as $1_{\dot{\theta m}}$, the effort is applied on the axis of manipulator, which is represented in bond graph model as $1_{\dot{\theta x}}$ at point C as shown in above figure. After this through TF, which has functions as $r1 * \sin(Q5)$ this motion combines at 0-element with inertia of $1_{\dot{yC}}$ and results in $1_{\dot{yA}}$. This is the output of point A in y-direction. Similarly for z-direction the TF function is $r1 * \cos(Q5)$and for $1_{\dot{\theta A}}$ direct C-element is used as activated flow. The results of outputs are shown in the next section.

For the second rotation about *y-axis* which is considered outside from the paper or in the left direction from Serial Manipulator when seen from the right hand side. The bond graph model is shown in Fig. 3.21. In this model motor is also introduced to give rotation about *y-axis* only and the output is taken of *x* and *z* coordinates distance change linearly.

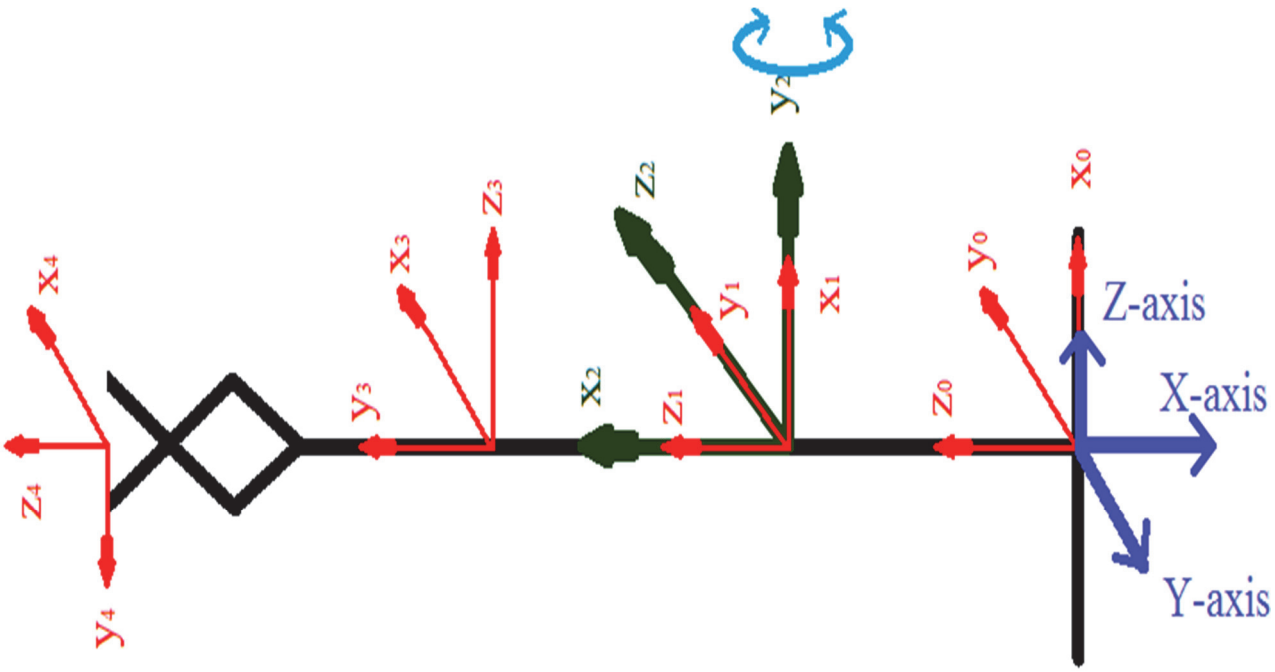


Fig. 3.21: About y-axis rotation in serial manipulator

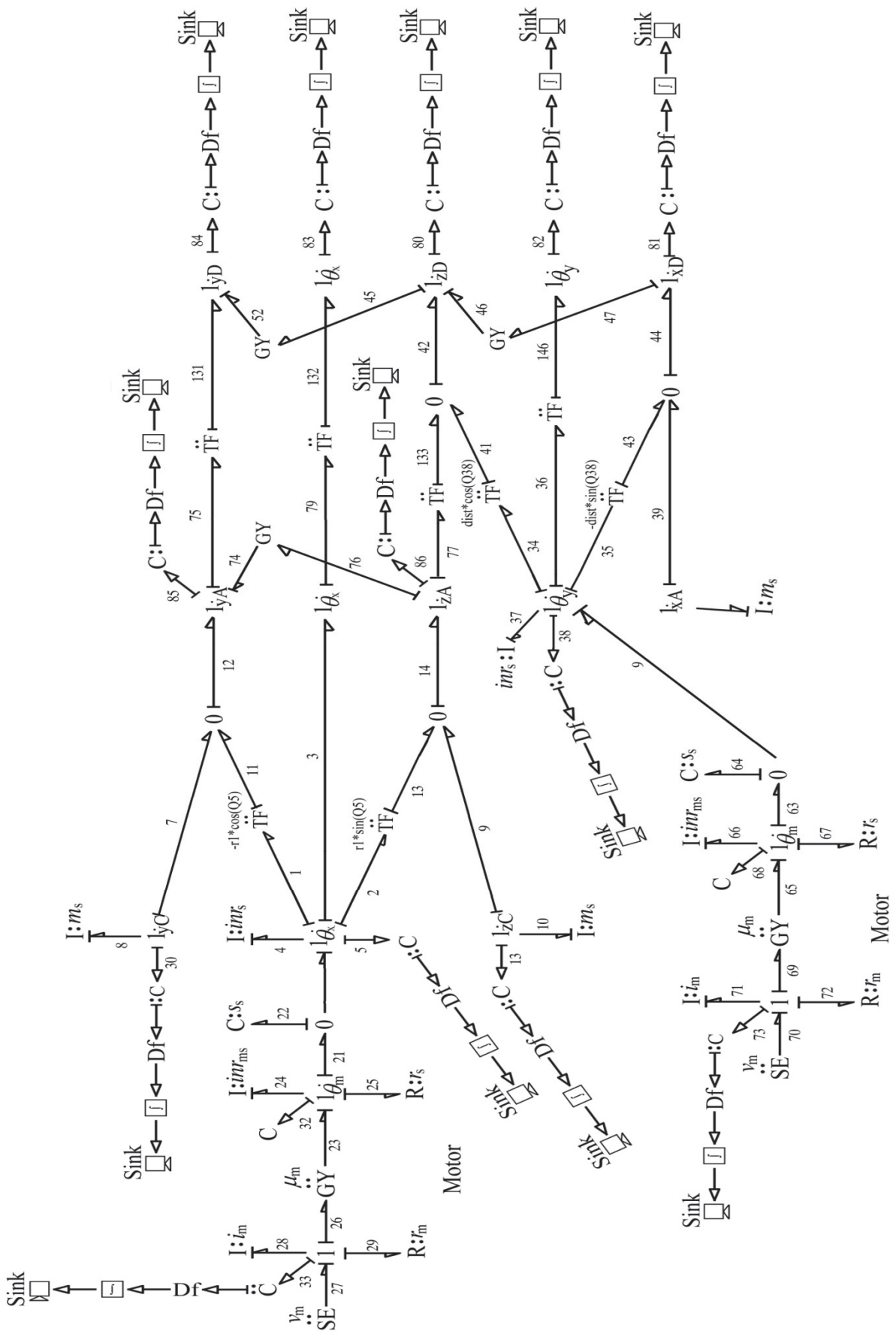


Fig. 3.22: Bond Graph for *y-axis* rotation

In the bond graph for *y-axis* rotation, another motor is introduced which has no connection with previous one but has same function as of motor for *x-axis* rotation. Now, here the motion or effort of the motor shaft is given to another element which is shown as $1_{\dot{\theta y}}$. After this through TF, which has functions as $dist * \cos(Q38)$ this motion combines at 0-element with inertia of $1_{\dot{xA}}$ and results in $1_{\dot{xD}}$, this is the output of point D in z-direction. Similarly for x-direction the function is used as $-dist * \sin(Q38)$ and for $1_{\dot{\theta D}}$ direct C-element is used as activated flow. Here three GY are used [**46**] for taking effects of different motions in x-direction or y-direction on one another but these are not actuated for this bond graph results. The results of outputs are shown in the next section.

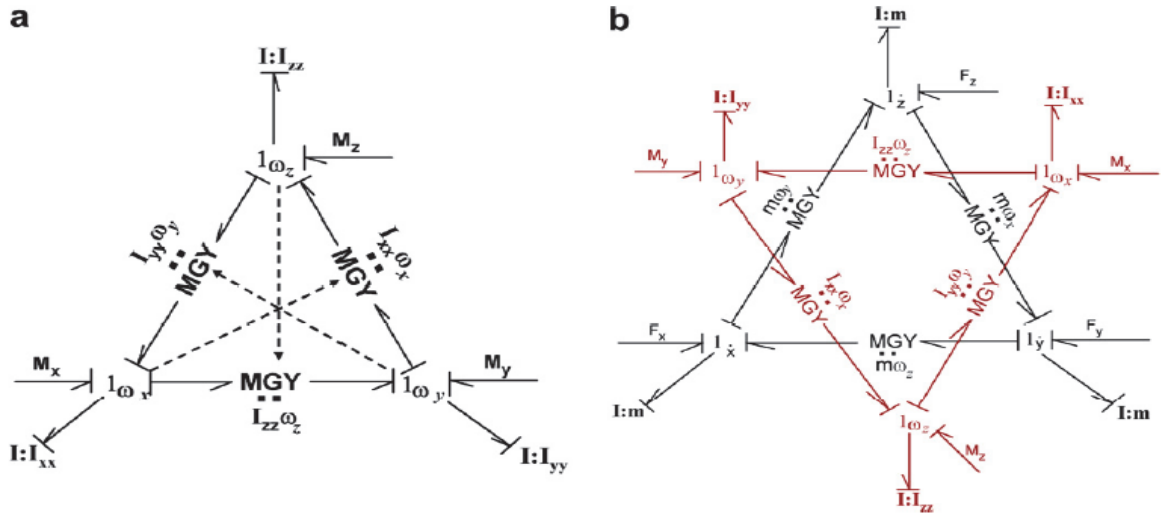


Fig. 3.23: Bond graph model of (a) Euler junction structure (b) Newton-Euler equations [**46**].

Consider a rigid body of mass m, moments of inertia about principal axes (x, y and z) with origin at the mass centre as Ixx, Iyy and Izz, angular velocities as seen from an inertial frame momentarily aligned along body fixed principal axes as xx, xy and xz, external moment components as Mx, My and Mz, external force components as Fx, Fy and Fz. Gyrator ring structure is used to model the Newton's equations [**46**]. The Newton–Euler equations are represented in the bond graph form by combining both to form a star-shaped junction structure given in Fig. 3.22 (b) It is the basic building block to construct the bond graph model of a multibody system.

For the third rotation about *z-axis* which is considered above from Serial Manipulator. The bond graph model is shown in Fig. 3.23. In this model motor is also introduced to give rotation about *z-axis* only and the output is taken of *x* and *y* coordinates distance change linearly.

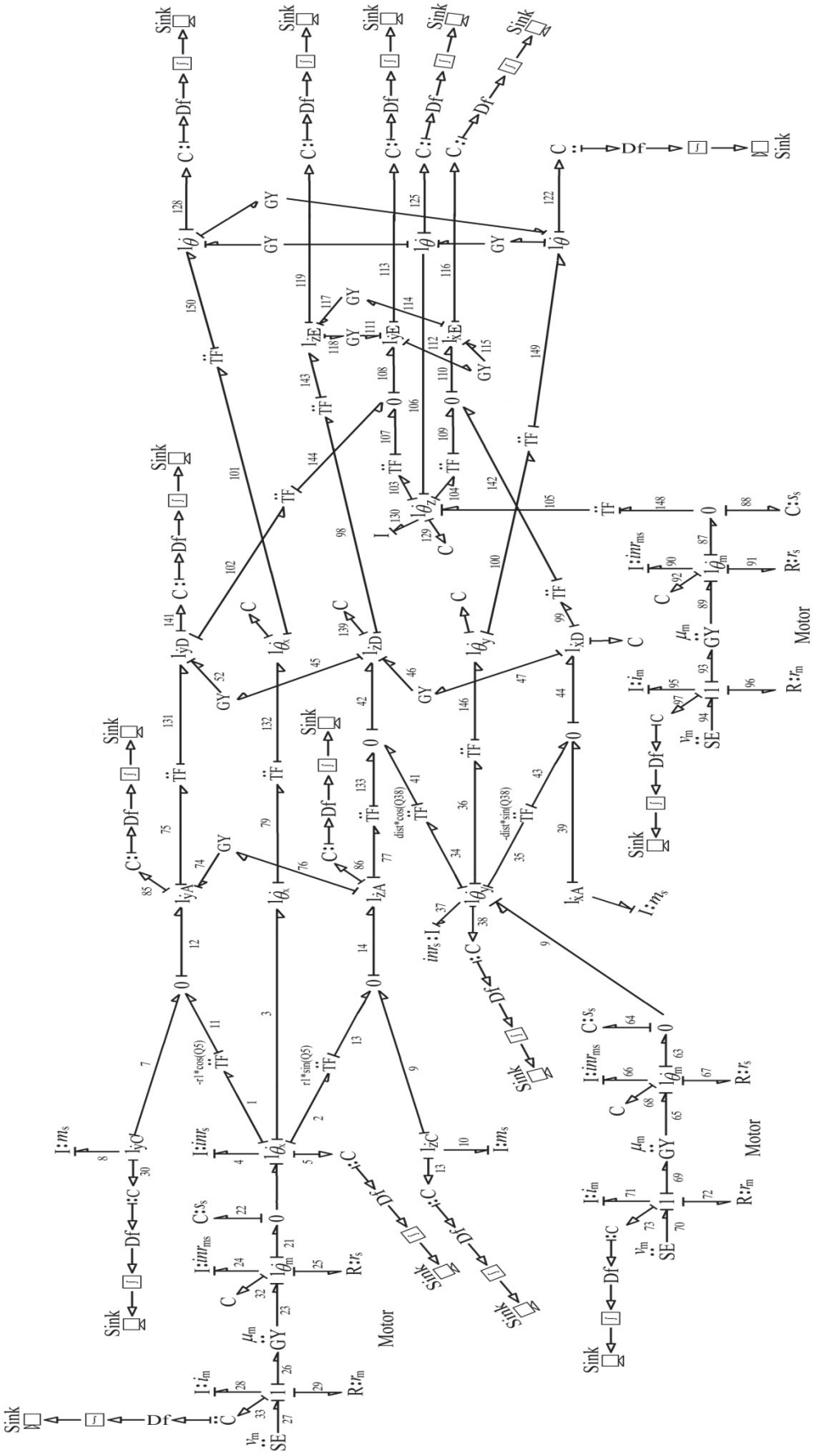


Fig. 3.24: Bond graph for rotation about z-coordinate.

In this robot all three rotations models are shown with the third rotation about *z-axis.* It has same motor functioning and the two TF are used for function as $-dist2 * \sin(Q129)$ which combines at 0-element with inertia of $1_{\dot{yD}}$ and results in $1_{\dot{yE}}$, this is the output of point E in y-direction. Similarly for x-direction the function is used $dist2 * \cos(Q129)$ which combines with $1_{\dot{xD}}$ and output taken as $1_{\dot{xE}}$ and for $1_{\dot{\theta D}}$ direct C-element is used as activated flow. Here three GY are used [**46**] for taking effects of different motions in x-direction or y-direction or z-direction on one another but these are not actuated for this bond graph results. The results of outputs are shown in the next section.

With the same bond graph model in Fig. 3.23, all the TF's and GY's are actuated for making all rotations in the same time with effect of inertia or motion on one another which gives the output in combining form.

3.9.2 *Inverse Kinematics*

The inverse kinematics problem is important because manipulation tasks are naturally formulated in terms of desired tool position and orientation. The inverse kinematics problem is more difficult than the direct kinematics problem because a systematic closed form solution applicable to robots or manipulators in general is not available. So, each robot or manipulator of similar class has to be treated separately. Now, the bond graph modelling of inverse kinematics, it is completely reversed each flow or effort according to the requirement.

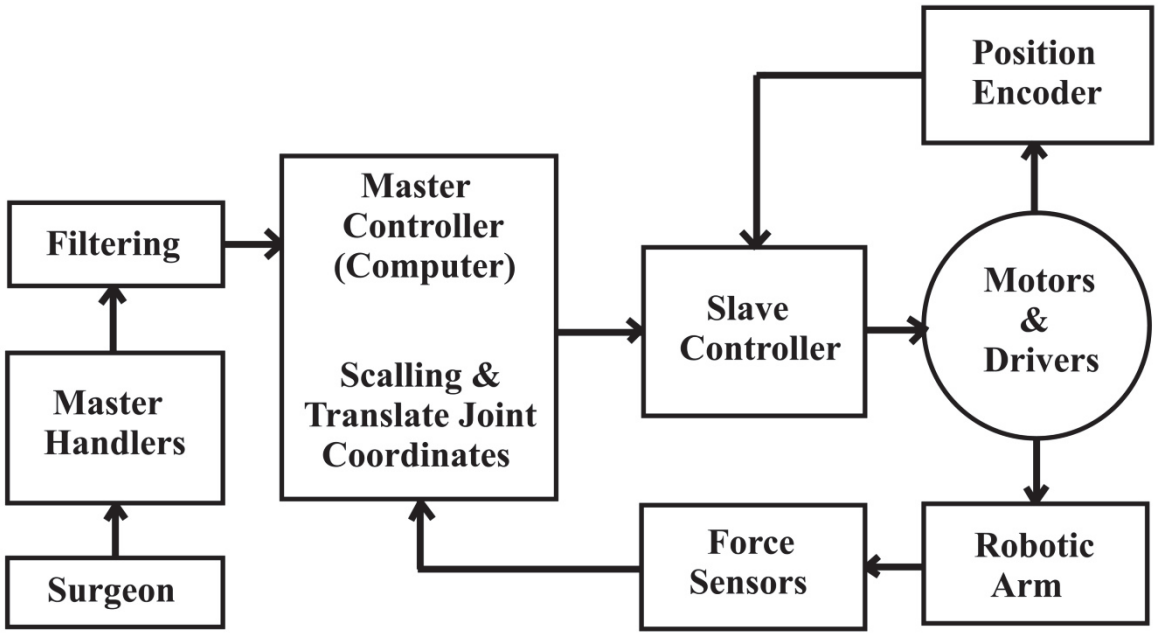


Fig. 3.25: Block Diagram of Robotic Arm System [CSIO].

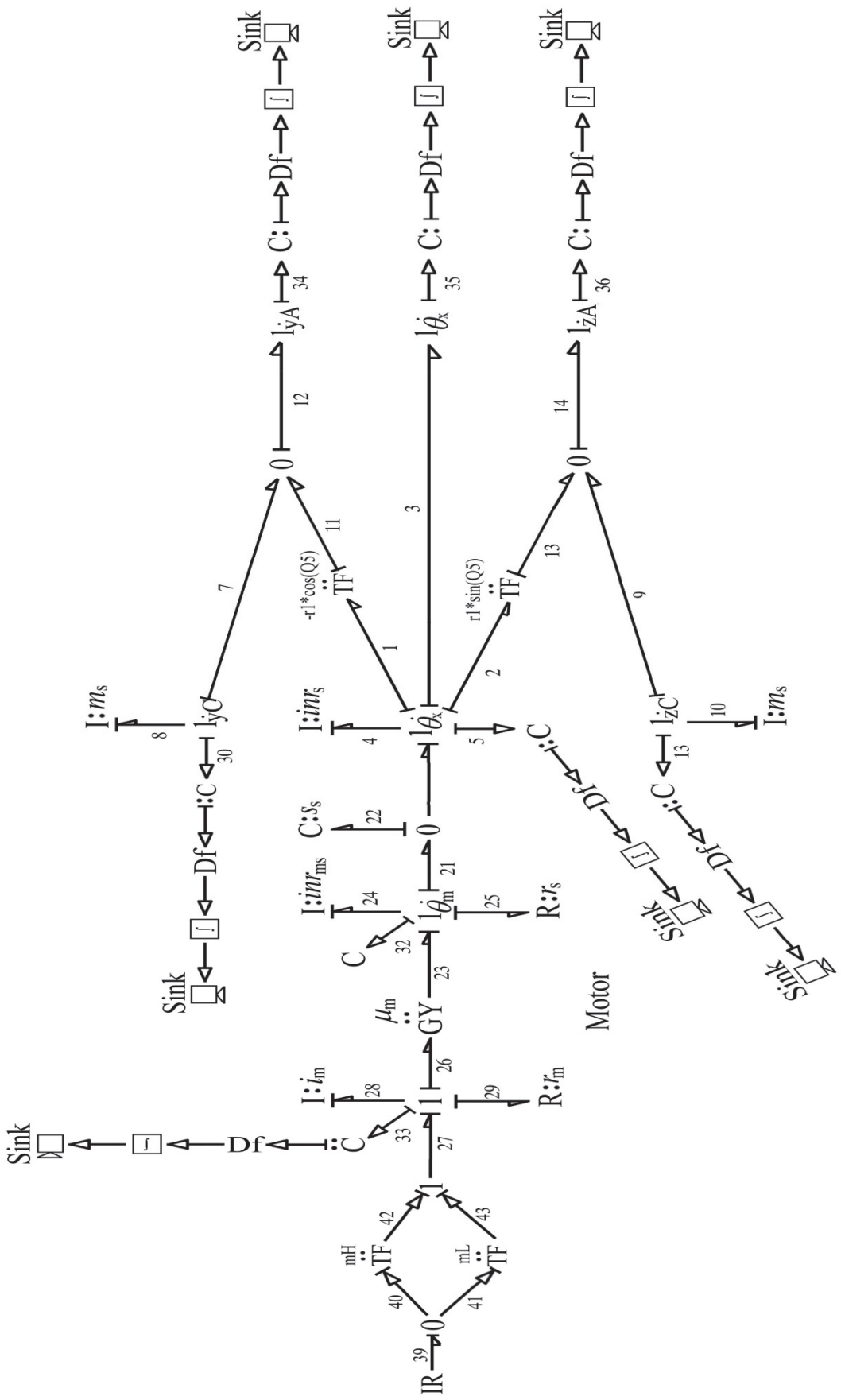


Fig. 3.26: Bond graph model combining both inverse and forward for *x-axis* rotation

The bond graph model is shown in above Fig. 3.24. The bond graph for *x- axis* rotation in forward model is same only its operational amplifier circuit. The input to this circuit is from IR which itself is a capsule containing inverse bond graph for this model. The two transformers (TF) used in here are multiplication factors for the circuit. The flow to flow TF oriented with higher modulus and the other effort to effort oriented with lower modulus. These muduli increases or decreases the effort or flow many times as it required in the model. IR is the capsule used for inverse kinematics means it has a complete inverse model of x-axis rotation in inverse terms. The IR capsule's bond graph is shown in the next figure with all its explanation.

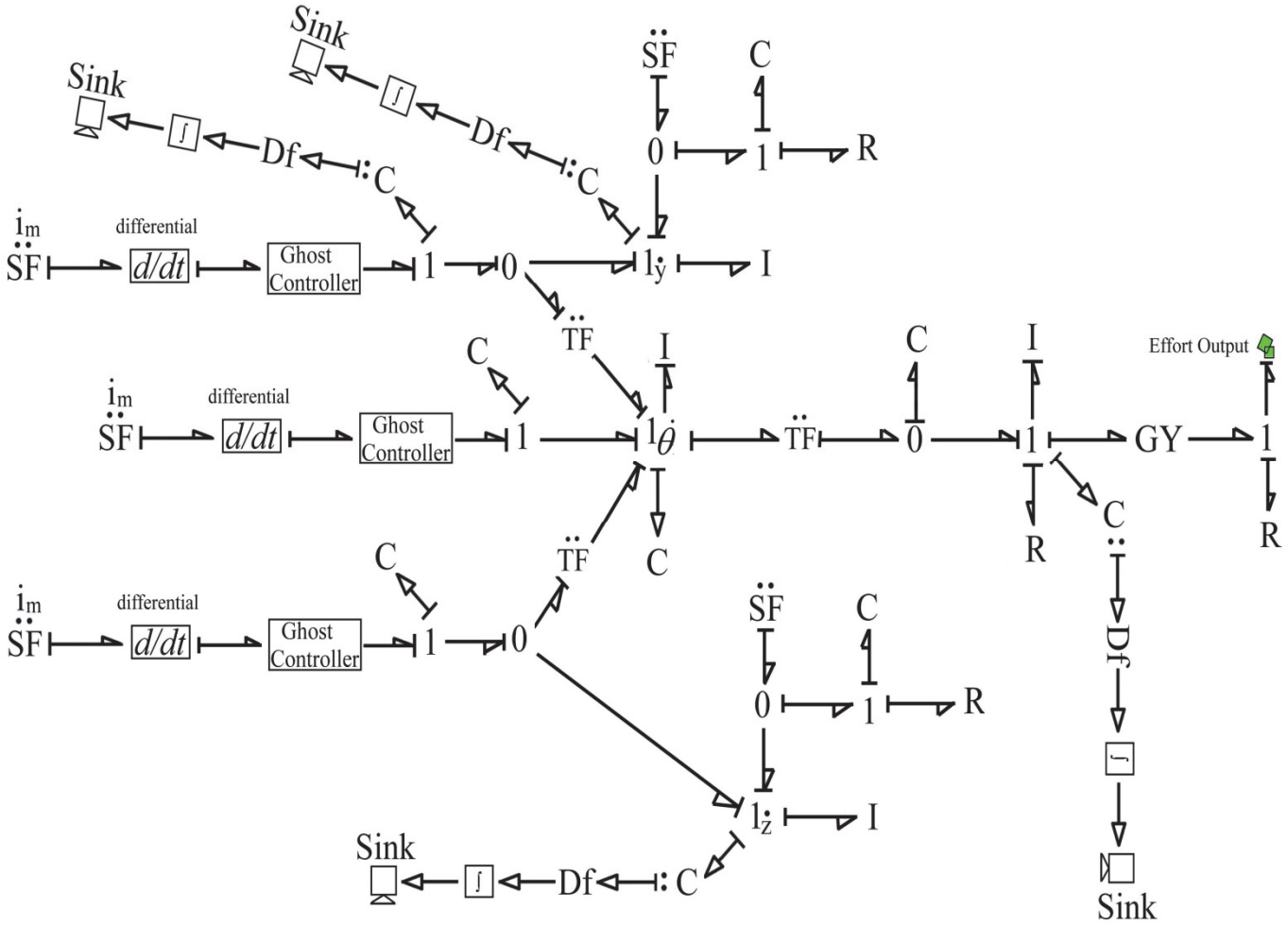


Fig. 3.27: IR capsule bond graph

The Fig. 3.26 is modelled bond graph for the capsule IR, shown in Fig. 3.25. This bond graph is the inverse model of x-axis rotation. Here, the input given to these three SF's is the output of the forward model having linear displacements of y and z coordinates and angular displacement of θ for x-axis. This scaling circuit is used and this is represented as

differential (d/dt) in the bond graph. Then the Ghost Controller (GC) is implemented to convert the flow into the effort, rest its bond graph is explained later. The signal is then passed to the motor via 1-element junction. Here, the motor is also working in reverse form which is acting as a generator. It means now it's taking input as flow and giving the output as effort which is then applied to forward model via operational amplifier circuit. The green icon in bond graph represents effort output from the inverse model as per the desired, for the x-axis rotation in the forward model. This output variations are as the input of SF's of inverse model readings given to it. The differential and ghost controller's bond graph is explained next.

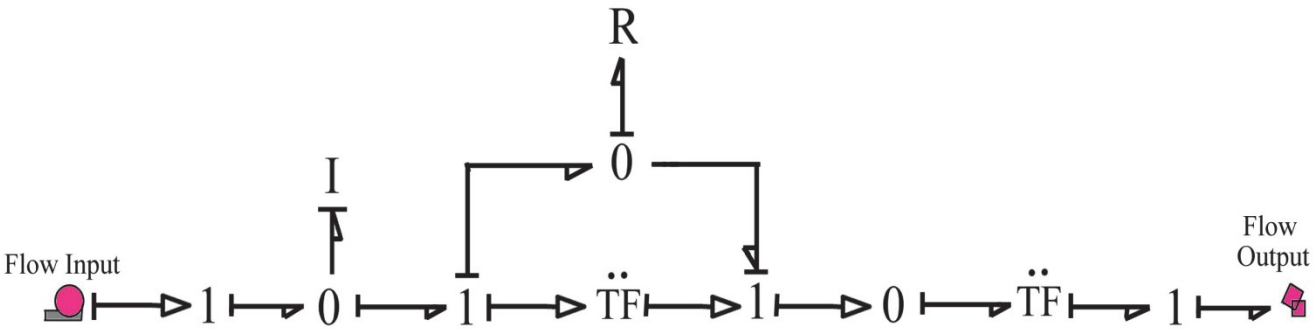


Fig. 3.28: Differential Bond Graph

The differential bond graph is shown in above Fig. 3.27. It is the scaling circuit or model in bond graph. The activated bond to 0-element is an observer of output flow. The implementation is loopy and to analyse it for gain through shortest route and not following another method which has many step to step method. The input is given by SF values given to it by previous readings and after differentiating its output went to the ghost controller explained below.

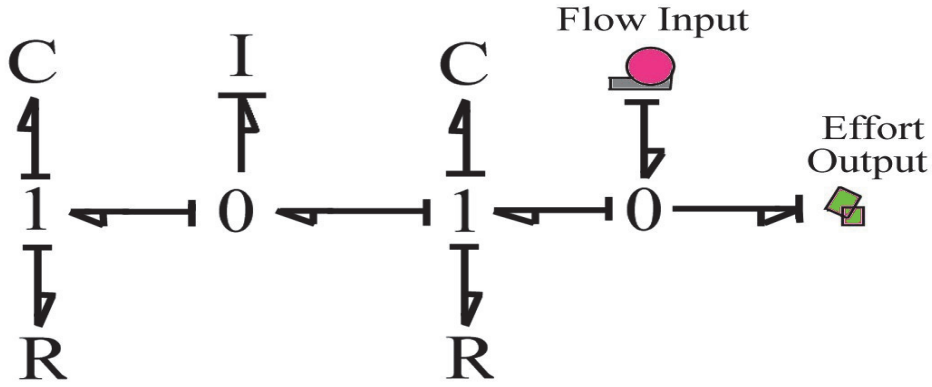


Fig. 3.29: Ghost Controller Bond Graph

The ghost controller used in IR capsule, it has containing bond graph is shown in above Fig. 3.27. Its function is to take flow input from the differential's output to convert it into effort output as per the input changes. The flow input passes through 1-element and via all inductance and resistance including capacitance changed to effort at 0-element junction which then taken as effort output stored as shown in figure by green icon which is the icon for effort output in bond graph modelling tool SYMBOLS.

3.10 PARAMETERS VALUES AND SIMULATION RESULTS

In this section all the results and graphs of bond graphs explained in the previous section are to be shown. Before that, all the parameter values are given to the bond graph as input required for its results or outputs. The parameter values are as follows:

Table 3.5 Parameter values

Parameters	Values
Radius, r1	0.01 m
Length, dist	0.05 m
Length, dist2	0.04 m
massA	0.5 kg
massC	0.5 kg
massD	0.5 kg
massE	0.5 kg
mu, μ	0.0001
Voltage (all), υ_m	12 Volts
Lambda, λ	7
Resistance (all)	0.05 ohm
Inductance (all)	0.001
Damping (all)	0.01
Clutch time, T1	10 sec.
Clutch time, T2	20 sec.

Now, the simulation results for each bond graph are to be shown. In the graphs Current, Torque and Speed versus Time are plotted for 60 seconds of time.

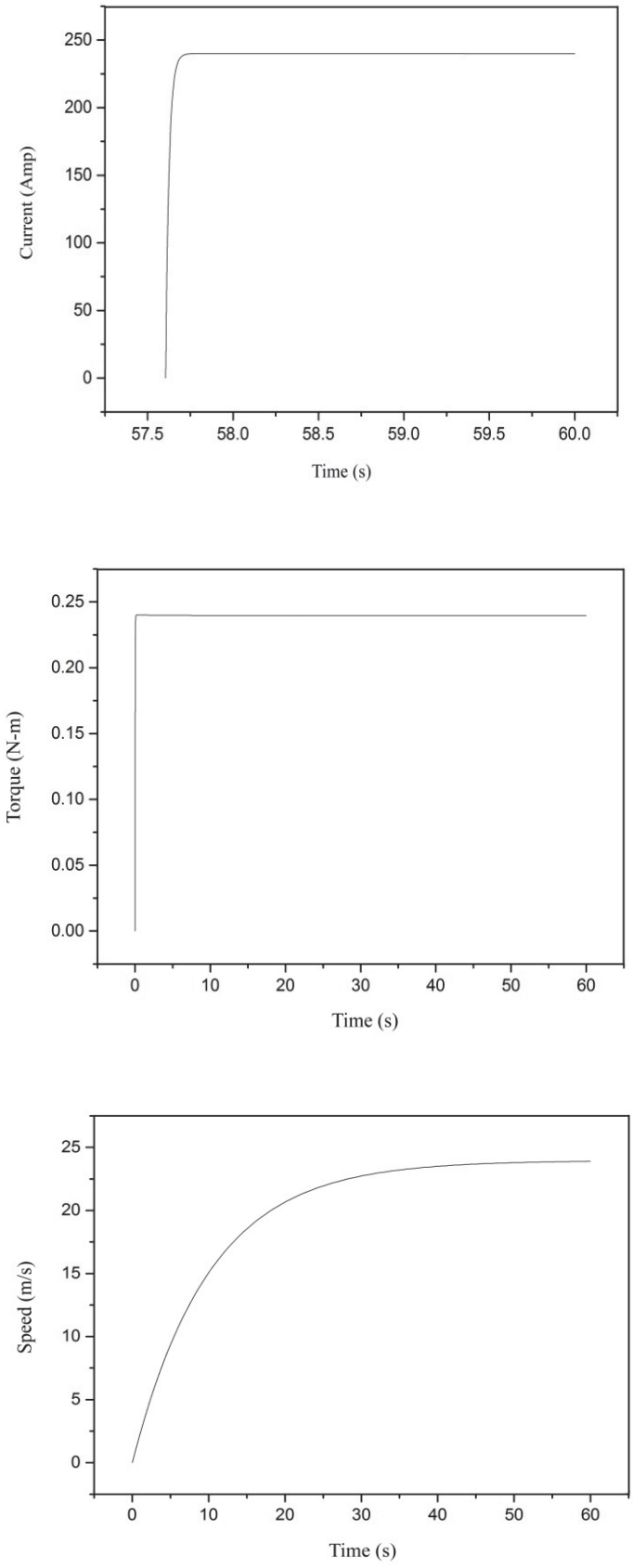


Fig. 3.30: Current, Torque and Speed *vs.* Time

The graphs presented in Fig. 3.29 are of current, torque and speed with respect to time of the motor shown in the bond graph in Fig. 3.20. In current v/s time, the current increases in last rapidly, for torque it shows that it reaches to its limit instantly and for speed, it increases slowly w.r.t. time which reaches its limit in last.

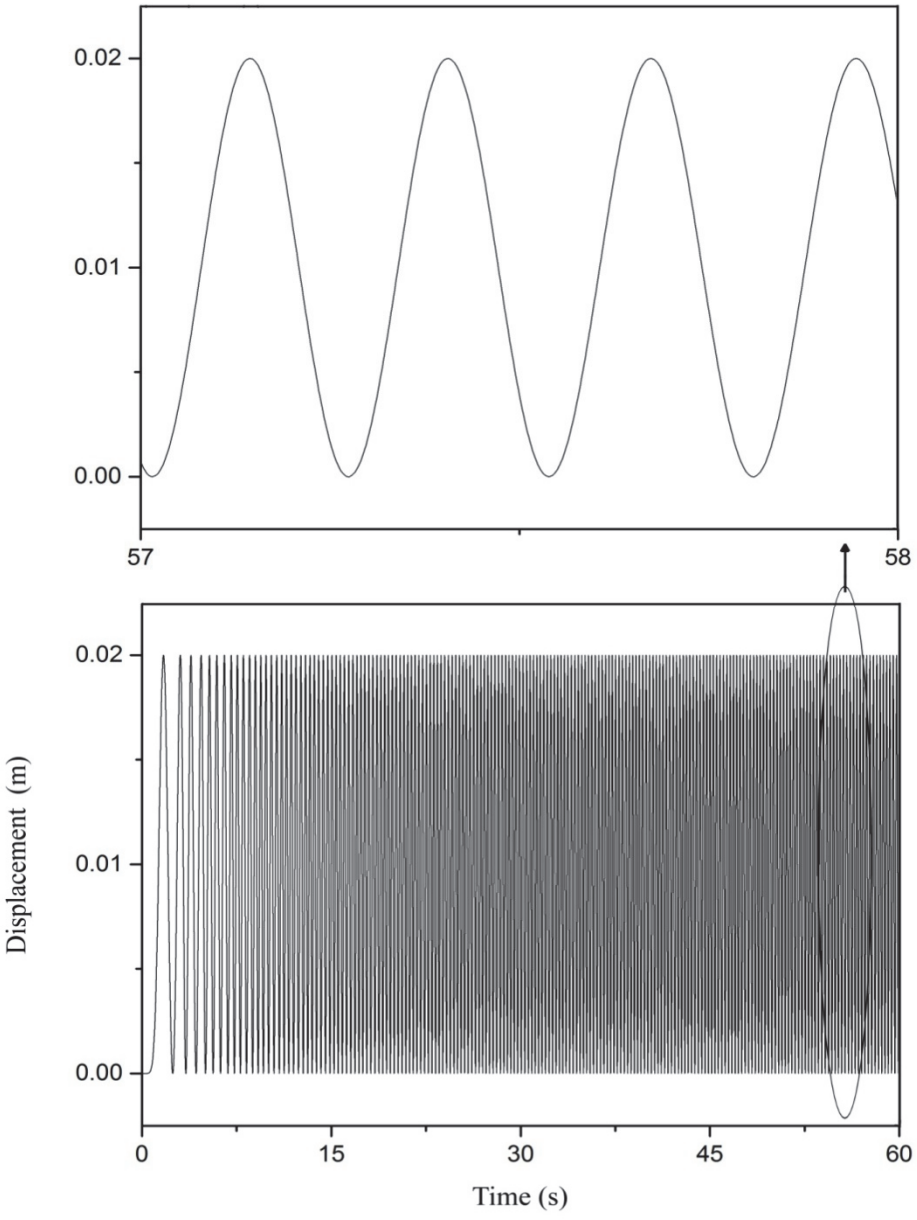


Fig. 3.31: Displacement in y-coordinate of point A v/s Time

Translation of A point as indicated in Fig. 3.19 in the y-coordinate is shown in above Fig. 3.30. The graph is for 60 seconds having sinusoidal nature and its small portion in shown in enlarged form which indicates that the A point is moving from zero or origin of coordinate system

to extreme length 0.02 m and rotates. It means the radius given as input is 0.01m, so it means A point is moving all around its radius which makes its diameter 0.02m.

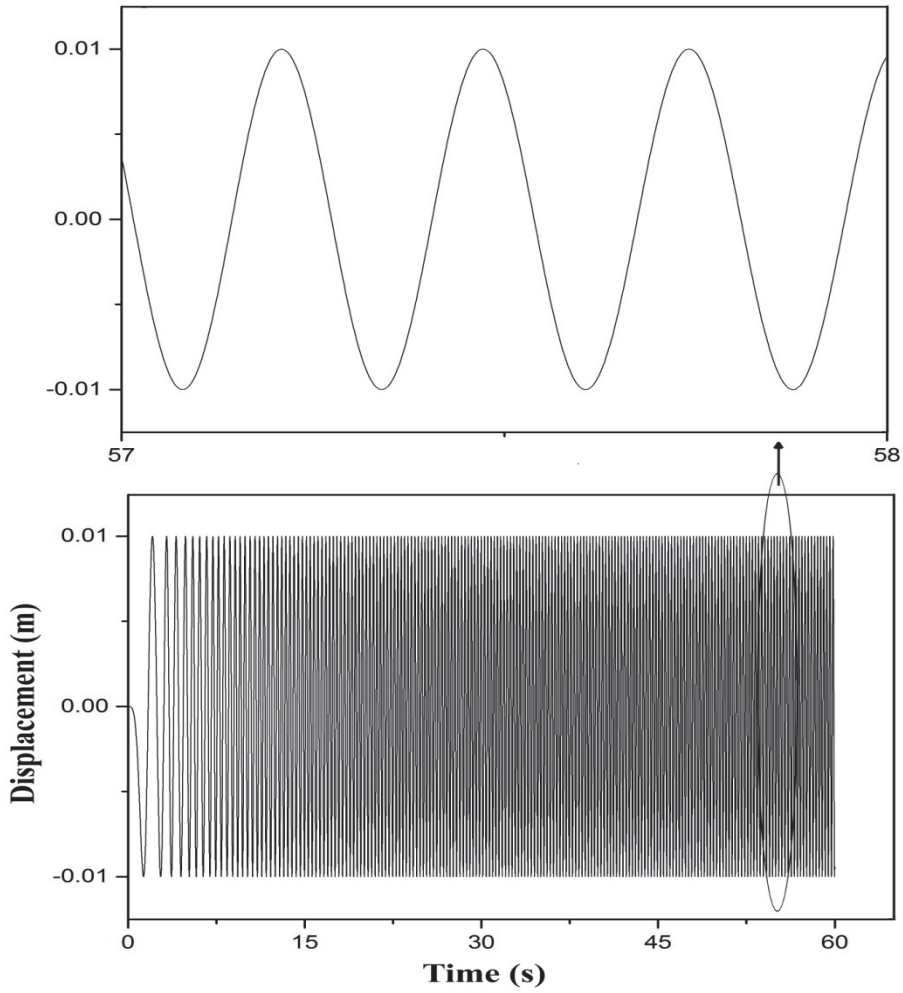


Fig. 3.32: Displacement in z-coordinate of point A *vs.* Time

Displacement of A point as shown in Fig. 3.19 in z-coordinate is shown in Fig. 3.31. The graph is for 60 seconds time having sinusoidal nature and its small portion in shown in enlarged form which indicates that the A point is moving from zero or origin of coordinate system to 0.01m extreme length and rotates in both sides of the coordinate. It means the radius given as input is 0.01m, so it means point A is moving all around its radius which makes that on one side, it is 0.01m and on the other side it is -0.01m.

Now for the rotation about all the axes, one by one. The bond graph is presented in Fig. 3.25 for combined simulation. The current, torque and speed are same as a previous motor graph. In the following graphs, point A displacement is shown in respective coordinates according to the axis of rotation.

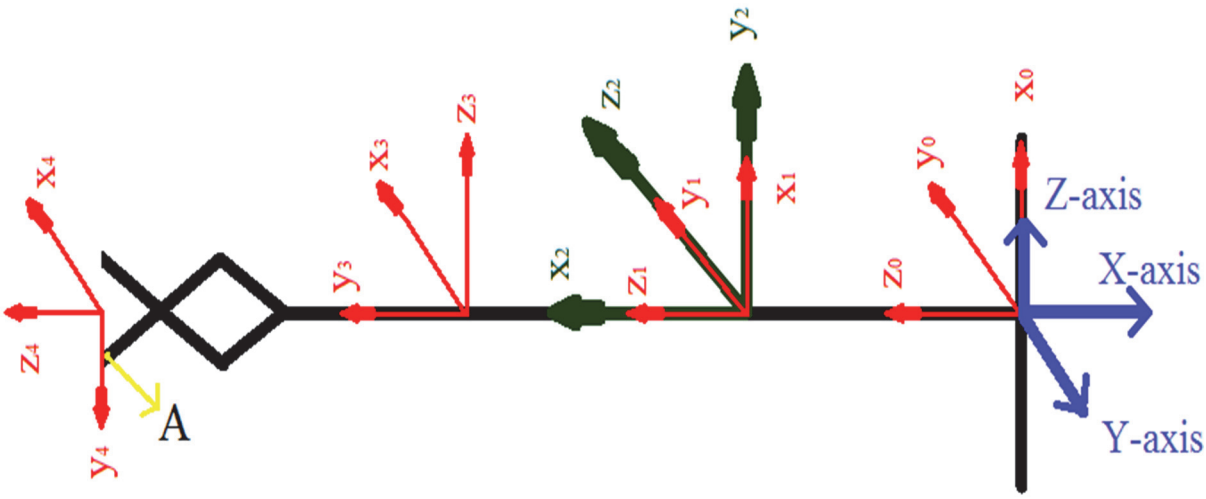


Fig. 3.33: Serial anipulator for end-effector point(about y-axis).

In the above figure point A is shown, which is to be traced or its movement is observed for different rotations. Now, for rotation about x-axis in combining bond graph is represented in following two graphs.

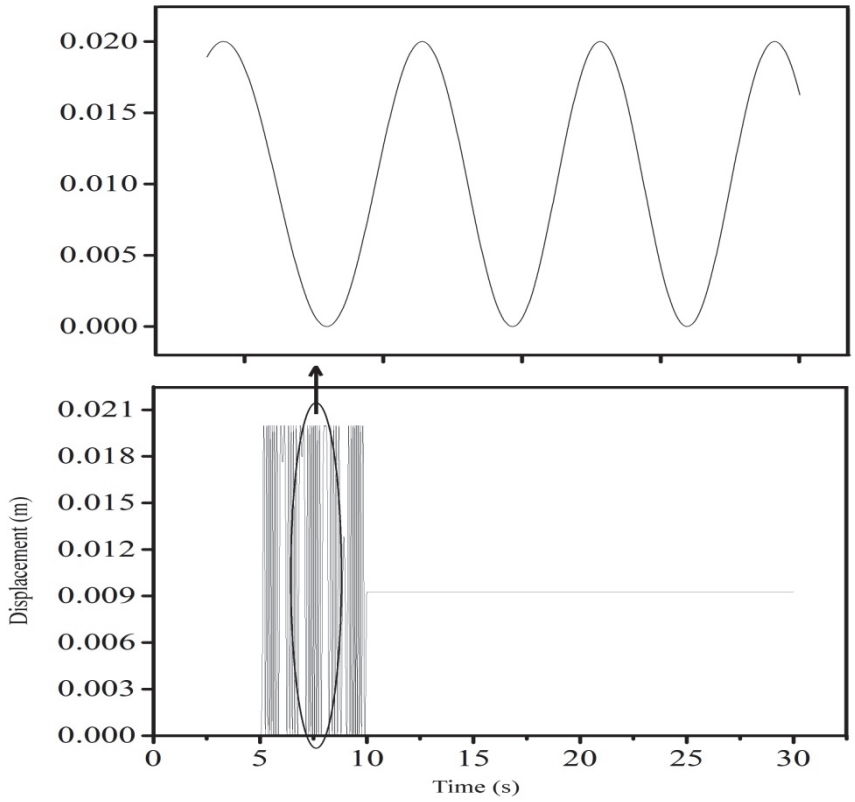


Fig. 3.34: Displacement in y-coordinate of A point v/s Time

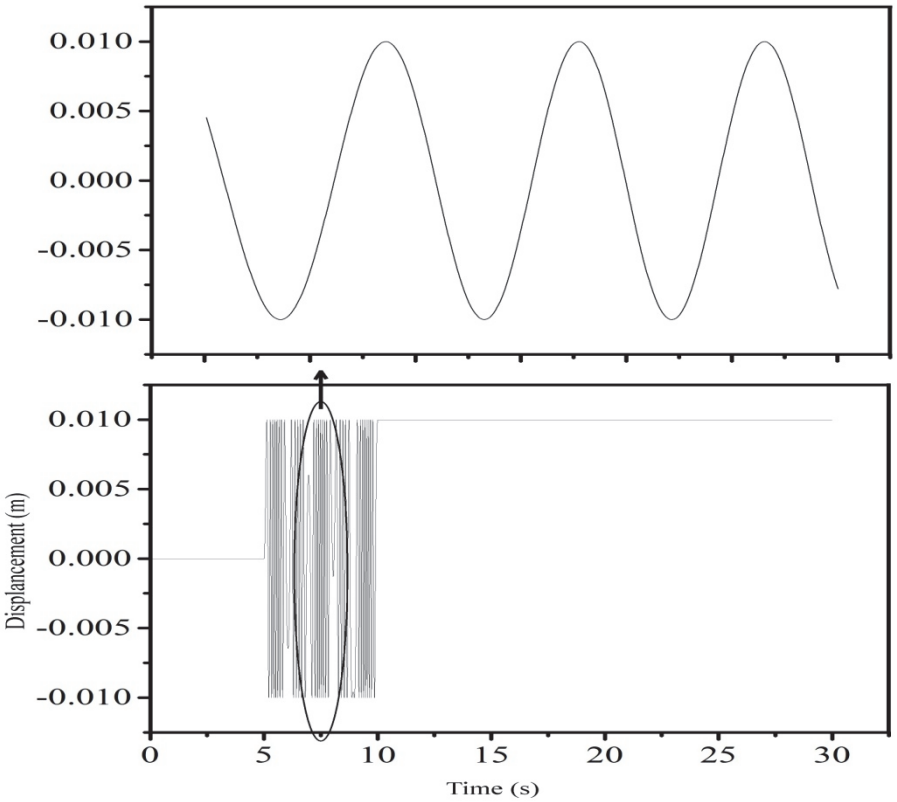


Fig. 3.35: Displacement in y-coordinate of point A *vs.* Time

Now, for rotation about y-axis, respective linear displacement graphs are shown.

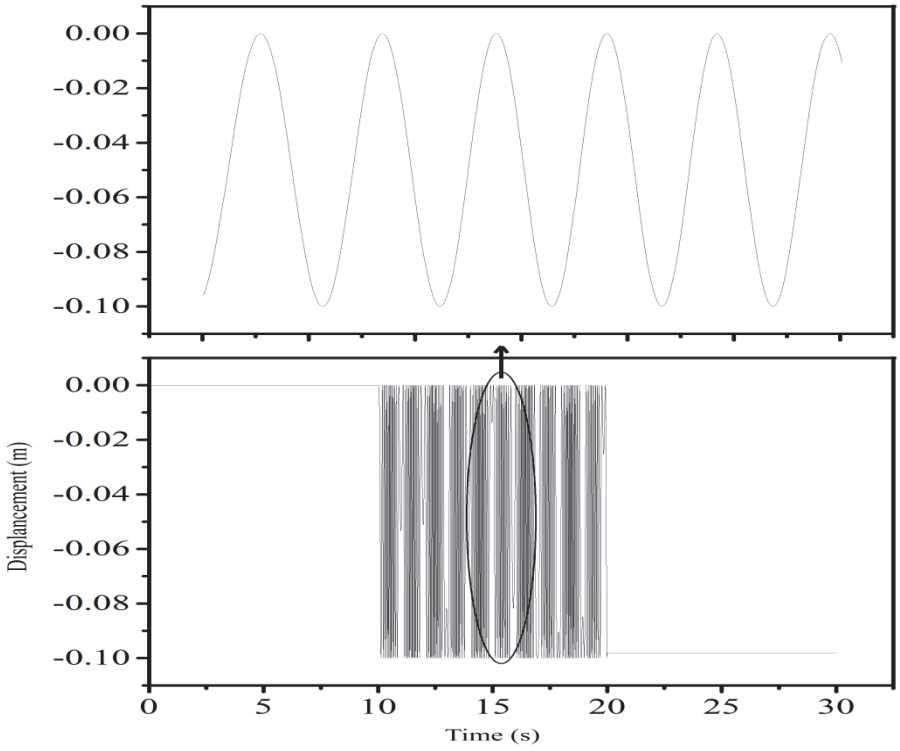


Fig. 3.35: Displacement in z-coordinate of A point v/s Time

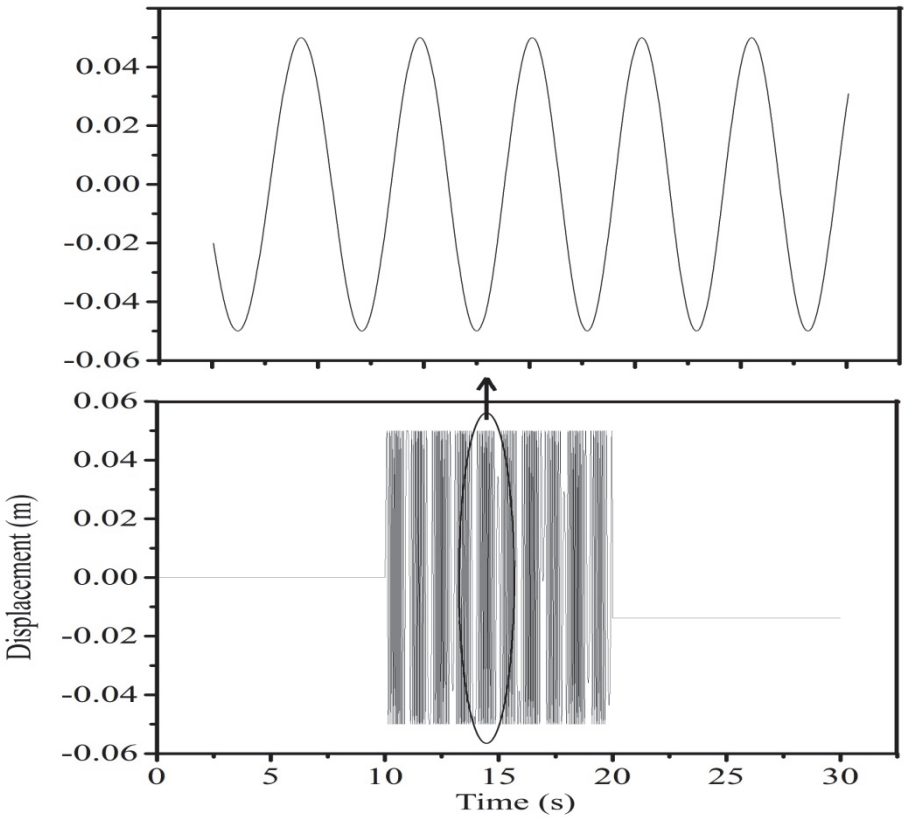


Fig. 3.37: Displacement in x-coordinate of point A v/s Time

Now, for rotation about z-axis, respective linear displacement graphs are shown.

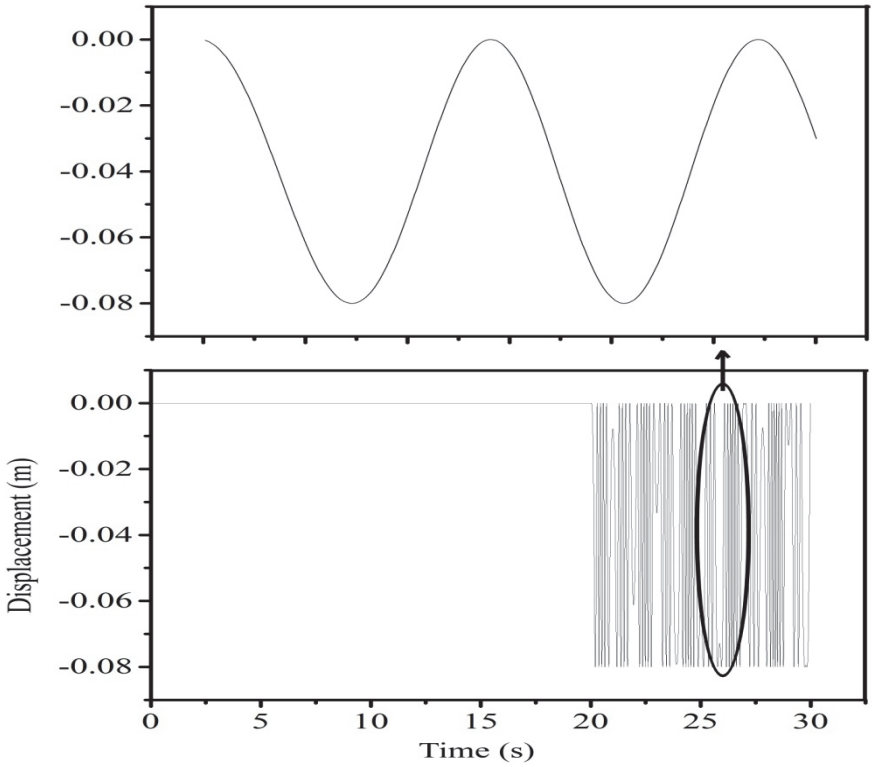


Fig. 3.38: Displacement in y-coordinate of A point v/s Time

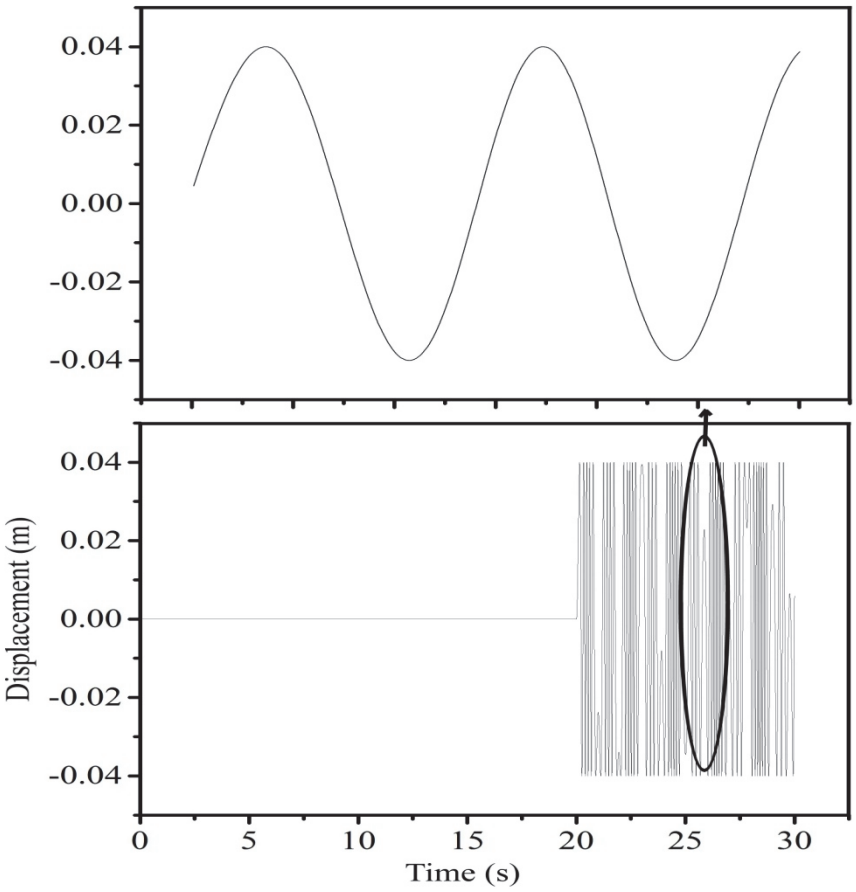


Fig. 3.39: Displacement in x-coordinate of A point v/s Time

In the above six graphs, it is shown that first rotation about x-axis is started after five seconds and ended after ten seconds. This is performed by motor power transmission control. Similarly, for the next ten seconds the second rotation about the y-axis and for the next ten seconds about z-axis is shown. The magnified graph is shown above all the main displacement graphs.

Now, for the same bond graph shown in Fig. 3.24, all the controlling of motors is omitted and every motor transfer power to its respective axis simultaneously. In this case different gyrators are also considered which gives relation or effect of any axis rotation on each other axes. It means that the effect of inertia of rotation about one axis is considered on other two axis of rotation. In this simulation, only the final detectors which are after the third rotation are considered. There's no need for medieval displacement outputs.

In the following graphs different angular rotations and linear displacement output graphs are presented. The angular displacement is only taken for 1.2 seconds because the axis are getting rotated up to measurable amount which is easily can read or observed and studied

for the simulation and comparison aspects. Here, from beginning motor is connected to the shafts, no acceleration time is taken. Thus, the graphs are as follows.

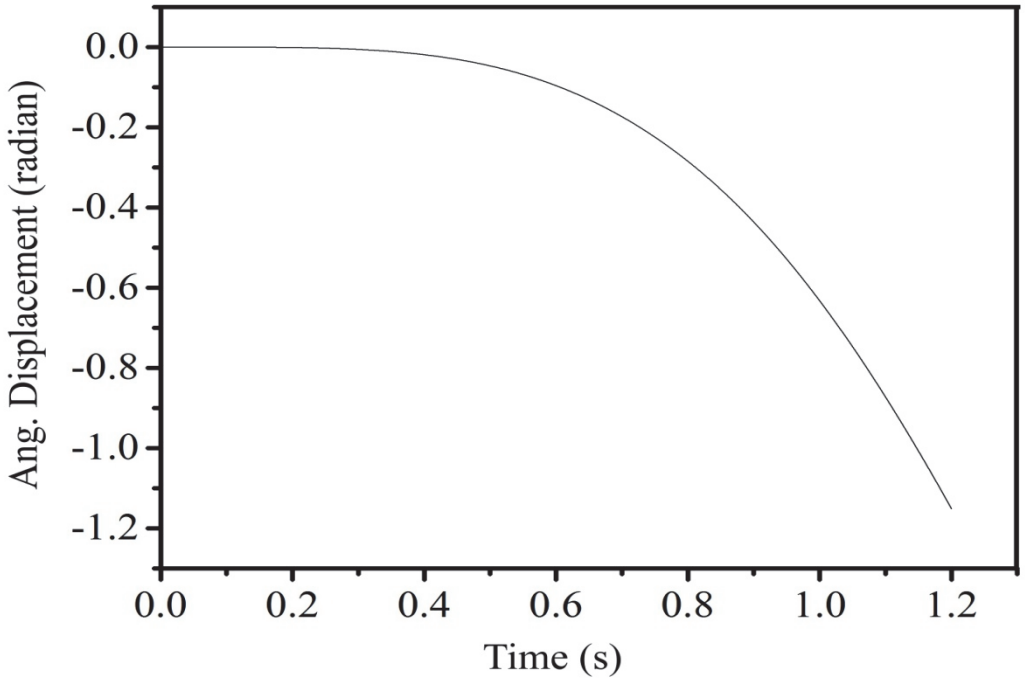


Fig. 3.40: Angular Displacement about x-axis w.r.t A v/s. Time

In the above graph, rotation about x-axis is shown. It is observed that in time 1.2 seconds, -1.15 radians is rotated.

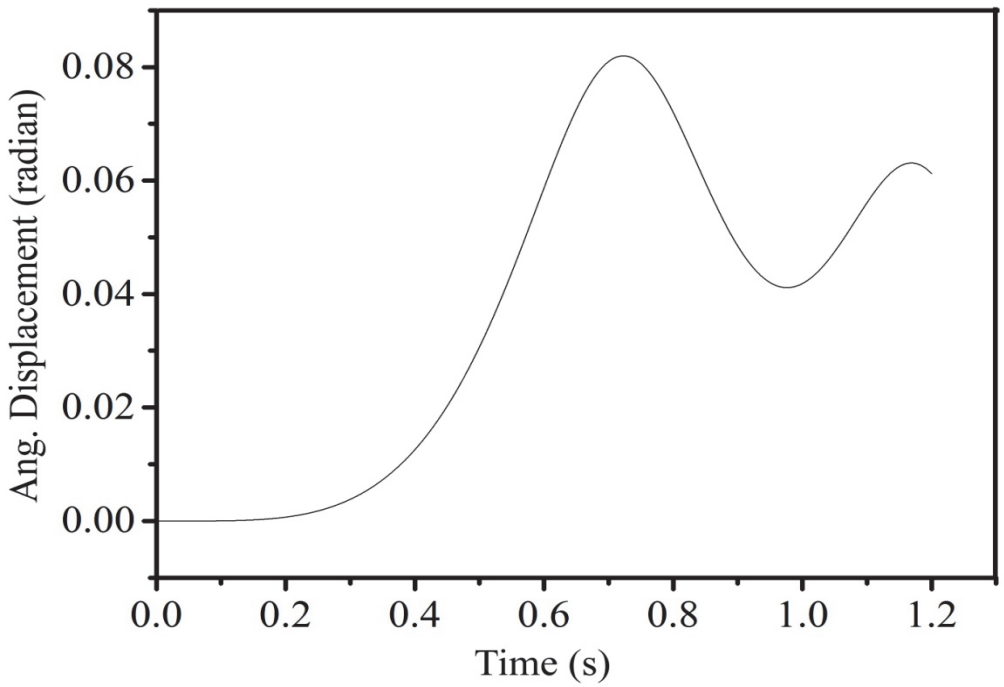


Fig. 3.40: Angular Displacement about y-axis w.r.t A v/s Time

In the above graph, rotation about x-axis is shown. It is observed that at time 1.2 seconds, -0.06 radians is rotated. The maximum rotation is 0.082 radians at time 0.72 seconds

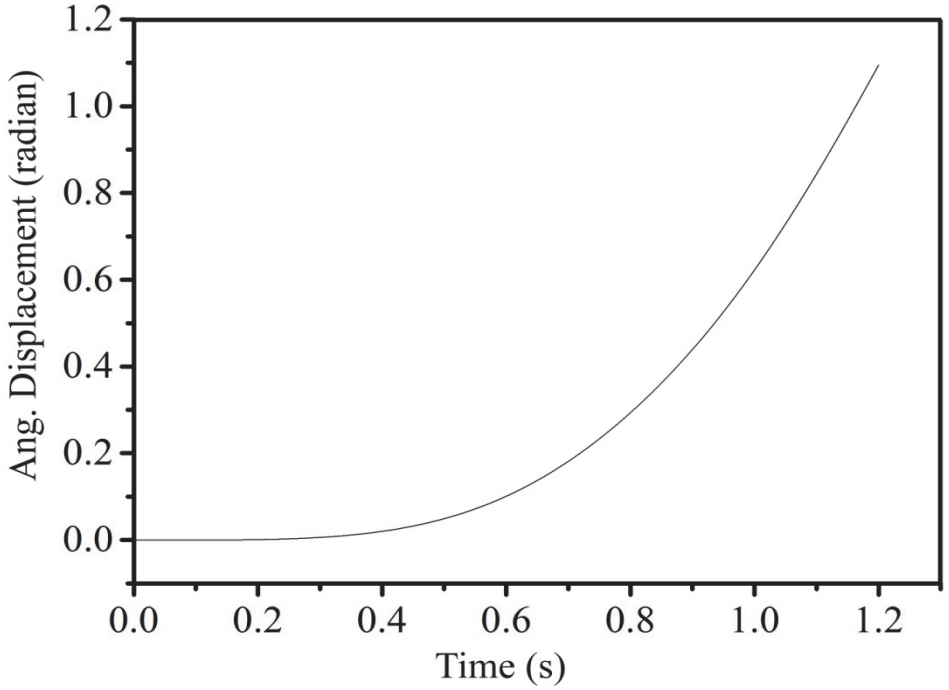


Fig. 3.42: Angular Displacement about z-axis w.r.t A v/s Time

In the above graph, rotation about x-axis is shown. It is observed that in time 1.2 seconds, 1.09 radians is rotated.

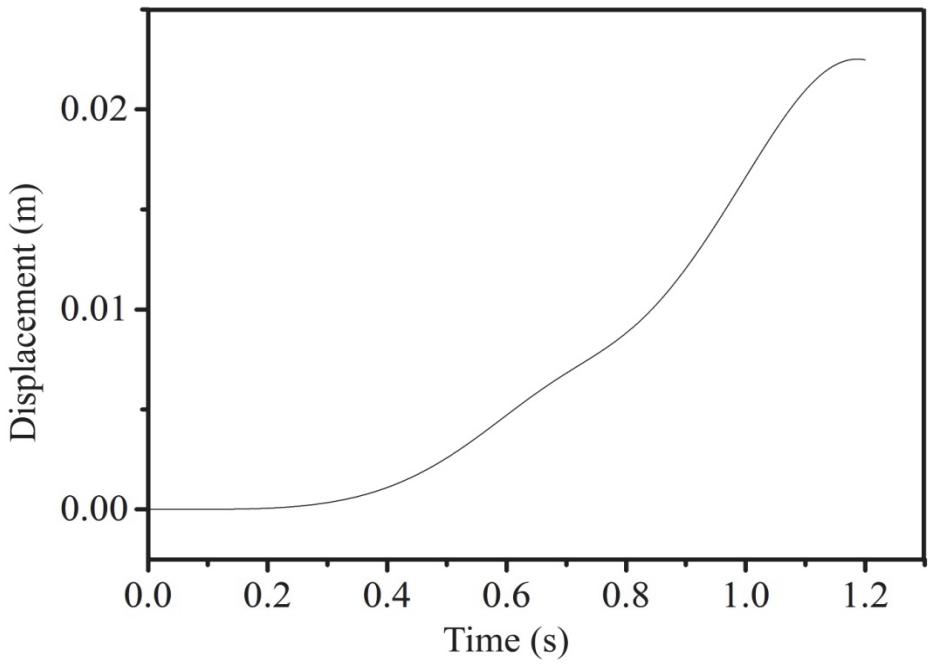


Fig. 3.43: Displacement in x-coordinate of A point v/s Time

In the graph, displacement in x-coordinate is expressed. It is observed that in time 1.2 seconds, 0.022 meters is displaced.

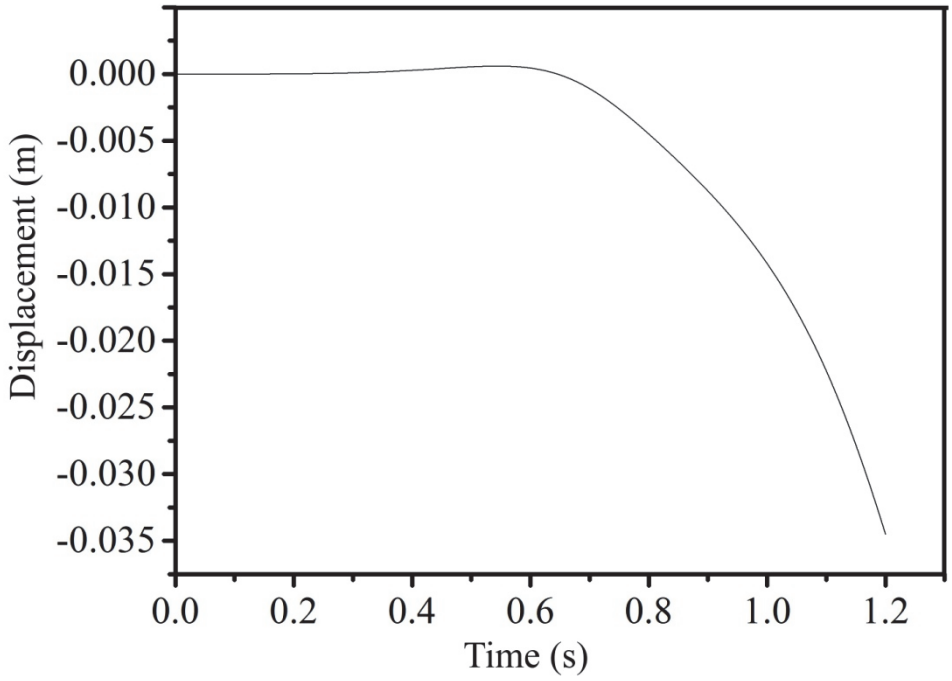


Fig. 3.44: Displacement in y-coordinate of A point v/s Time

In the above graph, displacement in y-coordinate is shown. It is observed that in time 1.2 seconds, -0.034 meters is displaced.

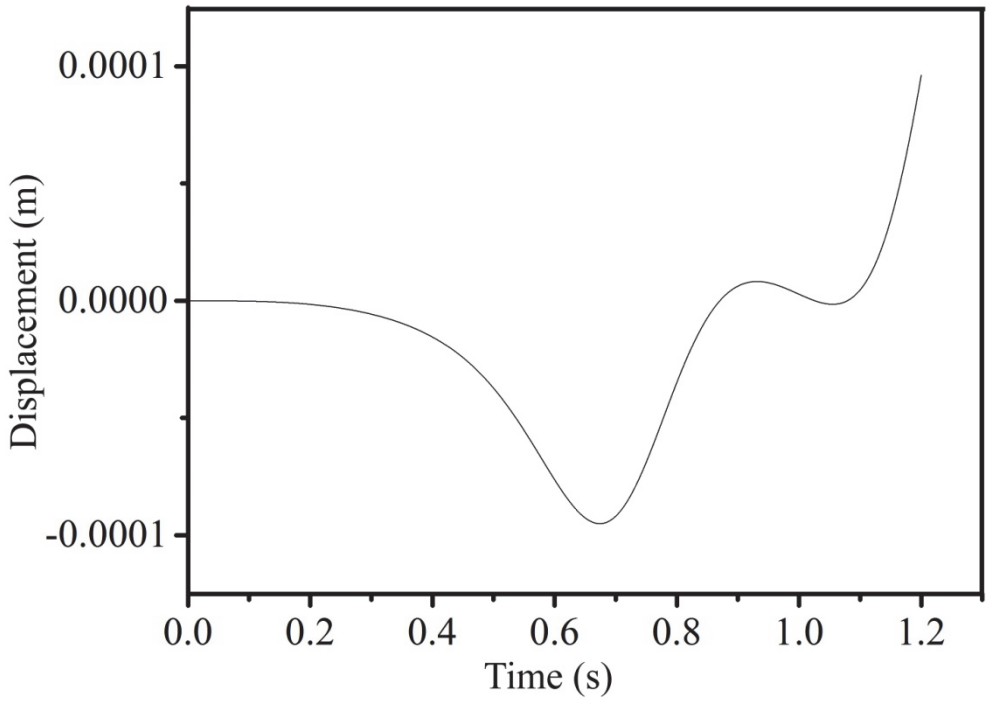


Fig. 3.45: Displacement in z-coordinate of A point v/s Time

In the above graph, displacement in z-coordinate is expressed. It is observed that in time 1.2 seconds, 0.000096 meters is displaced.

3.11 COMPARISON BETWEEN D-H ALGORITHM AND BOND GRAPH MODELLING IN FORWARD AND INVERSE KINEMATICS

The results of D-H algorithm and Bond graph approach same. It means that if we run bond graph for the particular angular displacements and the linear outputs is taken is same in the case of the D-H algorithm method by putting same angles of rotation.

CHAPTER 4 FABRICATION OF SERIAL MANIPULATOR

4.1 INTRODUCTION

Anything that a robot can use to interact with the environment is considered a manipulator. This generally takes the form of arm and other appendage, but also includes things such as giant ramps and flippers. The more complex design for manipulator is not always more honest or good; practical idea is more useful than a bright idea that goes down every other turn. Though, it would seem the first priority, a robot that can move itself and stay in one piece is more valuable than one that can wave a stick around.

4.2 DESIGN BASICS AND LIMITS

Important functionality points to consider:

- Properly adapted to the surgery and surgical procedures
- Not to compromise driving and structure of the robot
- Maintainable and feasible
- Remember about the weight, though, the manipulator is generally the heaviest element of a robot; an exceptionally heavy arm, for example, is much harder to move
- Use the proper motors for the motion transfer, overloading motors will cause the breakers to snap in the middle of moving an objective
- Manipulators should exemplify that simple is better to the fullest, it makes the mechanics easier
- Determinism, promotes the use of compliant mechanisms (no friction, no wear, low hysteresis, no play).
- Exact kinematic constraint design.
- Symmetry.
- Motion can be transferred from the motor to the axis of motion with belts, chains, and gears
- Should be able to move the manipulator easily without blowing up

Some common manipulator design elements are as follows:

- The mount
- The structure
- The frame
- The motors

4.3 MODELLING IN SOLID WORKS

Solid Works is a solid modelling CAD (computer-aided design) software that runs on Microsoft Windows and is produced by Dassault Systems Solid Works Corporation, a subsidiary of Dassault Systems, S. A. Solid Works is currently utilized by over 2 million engineers and designers at more than 165,000 companies worldwide. FY2011 revenue for Solid Works was 483 million dollars. Solid Works Corporation was established in December 1993 by Massachusetts Institute of Technology graduate Jon Hirsch tick. Solid Works is a Para solid modeller and utilizes a parametric feature-based approach to create models and assemblies.

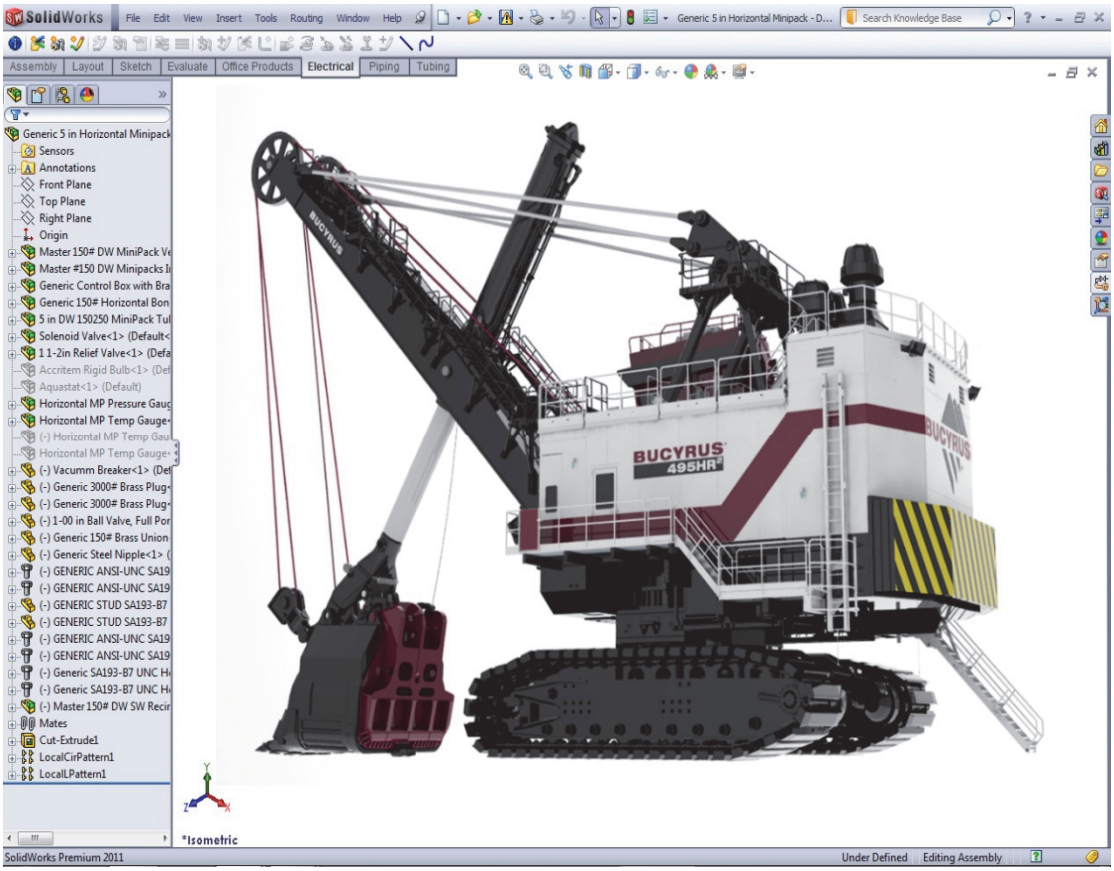


Fig. 4.1: Solid Works Window [**1**]

Parameters, refer to constraints whose values determine the shape or geometry of the model or assembly. Parameters can be either numeric parameters, such as line lengths or circle diameters, or geometric parameters, such as tangent, parallel, concentric, horizontal or vertical, etc. Numeric parameters can be associated with each other through the use of relations which allow them to capture design intent. *Design intent* is how the creator of the part wants it to respond to changes and updates. For example, it is required the hole at the top of a beverage can to stay at the top surface, regardless of the height or size of the can. Solid Works allows the user to specify that the hole is a feature on the top surface, and will then honour their design intent no matter what height they later assign to the can. *Features* refer to the building blocks of the part. They are the shapes and operations that construct the part. Shape-based features typically begin with a 2D or 3D sketch of shapes such as bosses, holes, slots *etc*. This shape is then extruded or cut to add or remove material from the part. Operation-based features are not sketch-based, and include features such as fillets, chamfers, shells, applying draft to the faces of a part *etc*.

Building a model in Solid Works usually starts with a 2D-sketch. The sketch consists of geometry such as points, lines, arcs, conics and splines. Dimensions are added to the sketch to define the size and location of the geometry. Relations are used to define attributes such as tangency, parallelism, perpendicularity and concentricity. The parametric nature of Solid Works means that the dimensions and relations drive the geometry, not the other way around. The dimensions in the sketch can be controlled independently, or by relationships to other parameters inside or outside of the sketch.

In an assembly, the analog to sketch relations are mates. Just as sketch relations define conditions such as tangency, parallelism, and concentricity with respect to sketch geometry, *assembly mates* define equivalent relations with respect to the individual parts or components, allowing the easy construction of assemblies. Finally, drawings can be created either from parts or assemblies. The views are automatically generated from the solid model, and notes, dimensions and tolerances can then be easily added to the drawing as needed. The drawing module includes most paper sizes and standards (ANSI, ISO, DIN, GOST, JIS, BSI and SAC). Solid Works 2013 opens/saves following file formats:

Table 4.1: Formats of files in Solid Works [**1**]

Solid Works Files (*.sldprt, *.sldasm, *.slddrw),	Part Files (*.prt, *.sldprt),
Assembly Files (*.asm, *.sldasm),	Drawing Files (*.drw, *.slddrw),
DXF (*.dxf), DWG (*.dwg),	Adobe Photoshop Files (*.psd),
Adobe Illustrator Files (*.ai),	Lib Feat Part (*.lfp, *sldlfp),
Template (*.prtdot, *.asmdot, *.drwdot),	Parasolid (*.x_t, *.x_b, *.smt_txt, *xmt_bin),
Stereolithographic STL (*.stl)	IGES (*.igs, *.iges),
STEP AP203/214 (*.step, *.stp),	ACIS (*.sat), VDAFS (*.vda),
VRML (*.wrl)	Catia Graphics (*.cgr),
ProEngineer Part (*.prt, *.prt.*, *.xpr),	ProEngineer Assembly (*.asm, *.asm.*, *.xas),
UGII (*.prt),	Autodesk Inventor Part (*.ipt),
Autodesk Assembly (*.iam),	Solid Edge Part (*.par, *.psm),
Solid Edge Assembly (*.asm),	CADKEY (*.prt, *.ckd),
Add-ins (*.dll),	IDF (*.emn, *.brd, *.bdf, *idb).

4.4 MODELLING OF MANIPULATOR AND ITS BASE (MOTOR HOUSING) IN SOLID WORKS

The model or design proposed for the surgical manipulator has three degrees of freedom. All are rotary joints perpendicular to each other, but at some distances to one another. At last it has an end-effector too, which serves as hold or drop any object to be taken. The actual part is not shown due to privacy (ongoing CSIR project). Thusly, the model in Solid Works is established beneath:

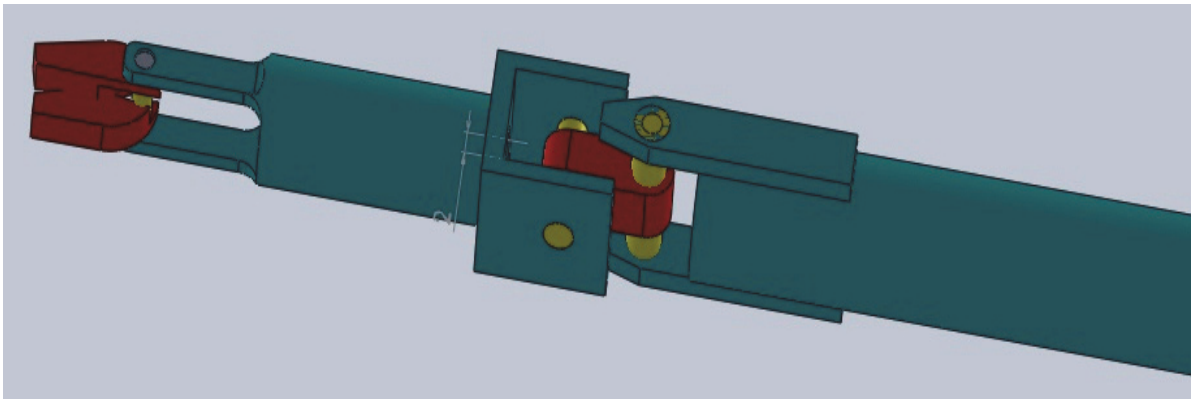


Fig. 4.2: Serial Manipulator in Solid Works

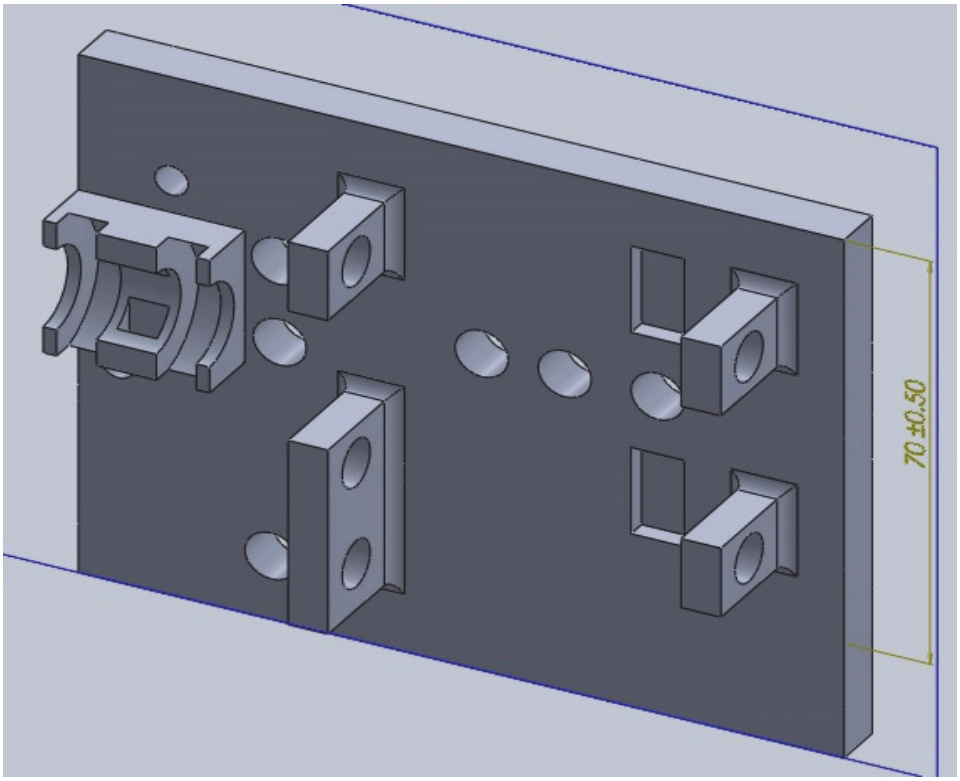


Fig. 4.3: Base Housing Design-I

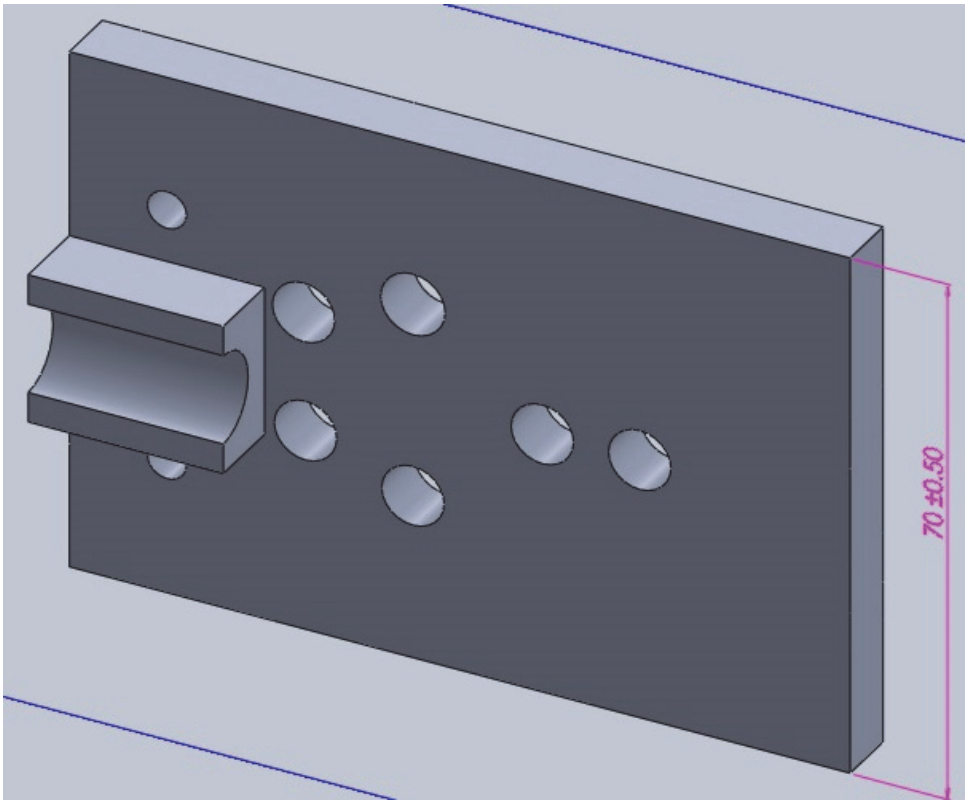


Fig. 4.4: Base Housing Design-II

The dimensions are also not shown on the model. Hither, as exhibited in Fig. 4.2, the end-effector is to be used inside the patient's body to hold any object or it is used to hold any tool for surgical operation. The correct conclusion is to be secured in the motor housing

explained later. The two of the rotations are easily recognised in the figure, which are having motion about the yellow shafts as shown in Fig. 4.2.

The two above motor housings as shown in Fig. 4.3 and Fig. 4.4 are the proposed or somehow similar to the design on which work is done. Some other parts are also used like where is the motor to be attached, how motion transfer from motor to different joints, what is the speed, which motor used *etcetera*. Those are not indicated here too, due to the same cause.

4.5 MATERIAL SELECTION

Material for the manipulator is aluminium and all its properties and specifications are explained why it is selected.

Aluminium

It is a chemical element in the boron group with symbol "*Al*" and atomic number 13. It is a silvery white, soft, ductile metal. Aluminium is the third most abundant element (after oxygen and silicon), and the most abundant metal in the Earth's crust. It makes up close to 8% by weight of the Earth's solid surface. The chief ore of aluminium is bauxite. Aluminium is remarkable for the metal's low density and for its ability to resist corrosion. Structural components made from aluminium and its alloys are vital to the aerospace industry and are important in other areas of transportation and structural materials. The most useful compounds of aluminium, at least on a weight basis, are the oxides and sulfates.

Physical Properties

Aluminium is a relatively soft, durable, lightweight, ductile and malleable metal with appearance ranging from silvery to dull gray, depending on the surface roughness. It is non magnetic and does not easily ignite. A fresh film of aluminium serves as a good reflector (approximately 92%) of visible light and an excellent reflector (as much as 98%) of medium and infrared radiation. The yield strength of pure aluminium is 7–11 MPa, while aluminium alloys have yield strengths ranging from 200 MPa to 600 MPa. It is well machined, cast, drawn and extruded. Aluminium atoms are arranged in a face-centred cubic (FCC) structure. Aluminium is a good thermal and electrical conductor, having 59% the conductivity of copper, both thermal and electrical, while having only 30% of copper's density.

Table 4.2: Properties value

Phase	Solid
Density	2.70 g-cm^3
M.P	933.47 K
B.P.	2743 K

Chemical Properties

Corrosion resistance can be excellent due to a thin surface layer of aluminium oxide that forms when the metal is exposed to air, effectively preventing further oxidation. The strongest aluminium alloys are less corrosion resistant due to galvanic reactions with alloyed copper. Aluminium is oxidized by water to produce hydrogen and heat:

$$2Al+3H_2O \rightarrow Al_2O_3 + 3H_2 \quad (4.1)$$

This conversion is of interest for the production of hydrogen. Challenges include circumventing the formed oxide layer, which inhibits the reaction and the expenses associated with the storage of energy by regeneration of the Al metal.

6061 Aluminium alloy

It is a precipitation hardening aluminium alloy, containing magnesium and silicon as its major alloying elements. It has good mechanical properties and exhibits good weldability. It is one of the most common alloys of aluminium for general purpose use.

Chemical composition

Table 4.3: Composition details for 6061 aluminium

Material	**Composition**
Rest Aluminium	95.85%–98.56%
Silicon range	0.4%- 0.8% by weight
Iron	up to 0.7%
Copper	0.15%–0.40%
Manganese	up to 0.15%
Magnesium	0.8%–1.2%
Chromium	0.04%–0.35%
Zinc	0.25%
Titanium	0.15%
Other elements,	0.15% total and up to 0.05% each

Mechanical Properties

The mechanical properties of 6061 depend greatly on the temper, or heat treatment, of the material. Young's Modulus is 10×10^6 psi (69 GPa) regardless of temper.

Uses

- Construction of aircraft structures, such as wings and fuselages, more commonly in homebuilt aircraft than commercial or military aircraft.
- Yacht construction, including small utility boats.
- Automotive parts, such as wheel spacers.
- The manufacture of aluminium cans for the packaging of foodstuffs and beverages.

Acrylonitrile butadiene styrene (ABS)

Its chemical formula is $(C_8H_8)_x\cdot\ (C_4H_6)_y\cdot(C_3H_3N)_z$) a common thermoplastic. Its glass transition temperature is approximately 105 °C (221 °F). ABS is amorphous and therefore has no true melting point. ABS is an opaque thermoplastic polymer material made from the monomers Acrylonitrile, 1,3-Butadiene and Styrene. Strong and durable even at low temperatures, it offers good resistance to heat and chemicals and is easy to process. ABS is an ideal material wherever superlative surface quality, colourfastness and lustre are required. ABS is a two phase polymer blend. A continuous phase of styrene-acrylonitrile copolymer (SAN) gives the materials rigidity, hardness and heat resistance. The toughness of ABS is the result of sub microscopically fine polybutadiene rubber particles uniformly distributed in the SAN matrix. ABS standard grades have been developed specifically to meet the requirements of major customers. ABS is readily modified both by the addition of additives and by variation of the ratio of the three monomers Acrylonitrile, Butadiene and Styrene: hence grades available include high and medium impact, high heat resistance, and electroplatable. Fibre reinforcement can be incorporated to increase stiffness and dimensional stability. ABS is readily blended or alloyed with other polymers further increasing the range of properties available. Fire retardancy may be obtained either by the inclusion of fire retardant additives or by blending with PVC. The natural material is an opaque ivory colour and is readily coloured with pigments or dyes. Transparent grades are also available.

Stratasys ABSplus-P430 is a true production-grade thermoplastic and is an ideal material for conceptual modelling, functional prototyping, manufacturing tools, and end-use-parts. The marriage of ABSplus-P430 with Fortus 3D Production Systems gives you the ability to create real parts directly from digital files that are stronger, smoother and with high feature detail.

Table 4.4: Properties of 6061 Aluminium

Flexible design	Excellent surface quality
Brilliant and deep colours	Attractive feel and touch
Dimensional stability	Chemical resistance
Impact resistance	High tensile strength & stiffness

Limitations

- Poor weather ability
- Poor solvent resistance
- High smoke generation when burned

4.6 RPT MACHINING PROCESS

Introduction

One of the easiest machines in the lab to use, this Fused Deposition Modelling (FDM) 3D Printer rapidly prototypes parts and even entire assemblies out of white ABS plastic. This is no hobby-grade desktop machine; the Stratasys Fortus 250mc will pleasantly amaze you with its fine resolution, good finish, and strong intra-layer bonds. Thanks to the support material, which provides a foundation for overhanging parts, designers don't need to worry about positive drafts, holes in sidewalls, and other manufacturing details.

How to Start

- Read the 3D Printing Guidelines and prepare your design using the preferred 3D Modelling Software (Solid Works, CREO, Rhino, Alias, Google Sketch Up Pro, etc).
- Using the preferred 3D Modelling Software, export your 3D Model as an STL file
- Determine the volume of material (in cubic inches) of your part.
- Upload the STL file using our online form.
- You will receive a confirmation email within 72 hours, to verify your submission was received. Then, you will receive another email when your part(s) are ready for pickup and payment. Expected turnaround time is 3-5 working days. Parts are generally run first-come first-served, however sometimes jobs are run out of order, for a variety of logistical reasons. Please be patient and wait for the email.

File Format

The 3D printer takes STL files, the most common 3D file format. When exporting STL files from your modelling software, if you get an option to select a resolution, there is no need to

go smaller than 0.001 inches. The resolution of finished parts on the Fortus 250mc is 0.0095 inches, so increasing the STL resolution much beyond this won't result in increased part resolution.

Material

The material used is a proprietary ABS plastic called ABS-Plus P430. Material properties such as Tensile Strength and Flexural Modulus are listed in the material datasheet. At this time the only available colour is white (ivory). Costs are based on the total number of cubic inches of modelling material your part will used.

Technical Details

Layer thickness is 0.013 inches. Parts are produced within an X-Y accuracy of ± 0.0095 inches, however accuracy is geometry dependent. Achievable accuracy specification derived from statistical data at 95% dimensional yield.

Fig. 4.5: RPT Machine (Fortus 250mc)

Materials

The Fortus 250mc builds parts in production-grade ABS *plus* thermoplastic, which offers nine colour choices. This lets your prototypes faithfully resemble finished products in look

and durability. The Fortus 250mc uses soluble support structures for easy, hands-free removal even for parts with complex geometry and twisted cavities.

Insight Software

Insight software prepares your CAD program's STL output for 3D manufacturing on a Fortus machine by automatically slicing and generating support structures and material extrusion paths. For maximum control, users can manually edit parameters that determine the look, strength and precision of parts as well as the speed and material use of the FDM process.

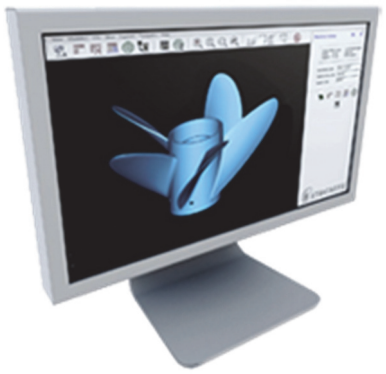


Fig. 4.6: Window of RPT software.

With Insight, you can:

- Optimize build orientation for maximum strength and smoothest surface finish
- Customize supports for fast, easy removal and best use of materials
- Program pauses into the build for any reason, such as to embed hardware or circuitry
- Manipulate tool paths for advanced control over part properties

Included with Insight is Control Centre. This sophisticated software application communicates with user workstations and Fortus systems, to manage jobs and monitor production status. Design, engineering and manufacturing teams can network and share 3D manufacturing capacity to maximize efficiency and throughput.

4.7 PROTOTYPE

The original or model on which something is based or formed. It is something that serves to illustrate the typical qualities of a class or model. Materials used in production may require manufacturing processes involving higher capital costs than what is practical for prototyping. Instead, engineers or prototyping specialists will attempt to substitute materials

with properties that simulate the intended final material. The processes are generally expensive and time consuming unique tooling is required to fabricate a custom design. Prototypes will often compromise by using more variable processes, repeatable or controlled methods; substandard, inefficient, or substandard technology sources; or insufficient testing for technology maturity.

The final design proposed for the surgical manipulator is fabricated for practical workout. It means in this prototype all the measured or methods results are tested, which are appreciable. Two of the rotations are taken place. Both are rotated at right angles. Here the end-effector is always open due to its internal mechanism. So, by this figure the practically movement of joints are proved as it is taken in the simulation process.

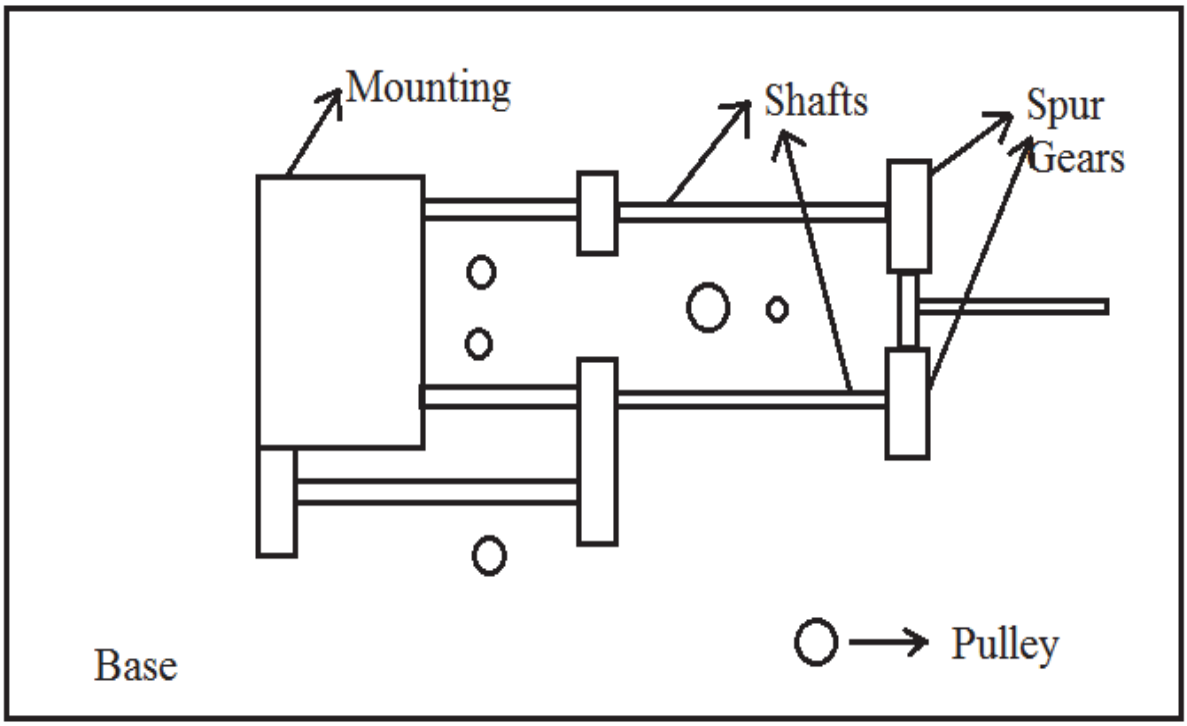


Fig. 4.7: Motor Housing (Prototype)

The main motor housing prototype is shown in Fig. 4.9. In this prototype, it is shown that how the motion is transferred from the pulley to the joints which are with the help rope type belt drives. The different pulleys are driven by different motors as per the required speed and accuracy. The control of motor speed and motion are made by a controller which has the program for doing so.

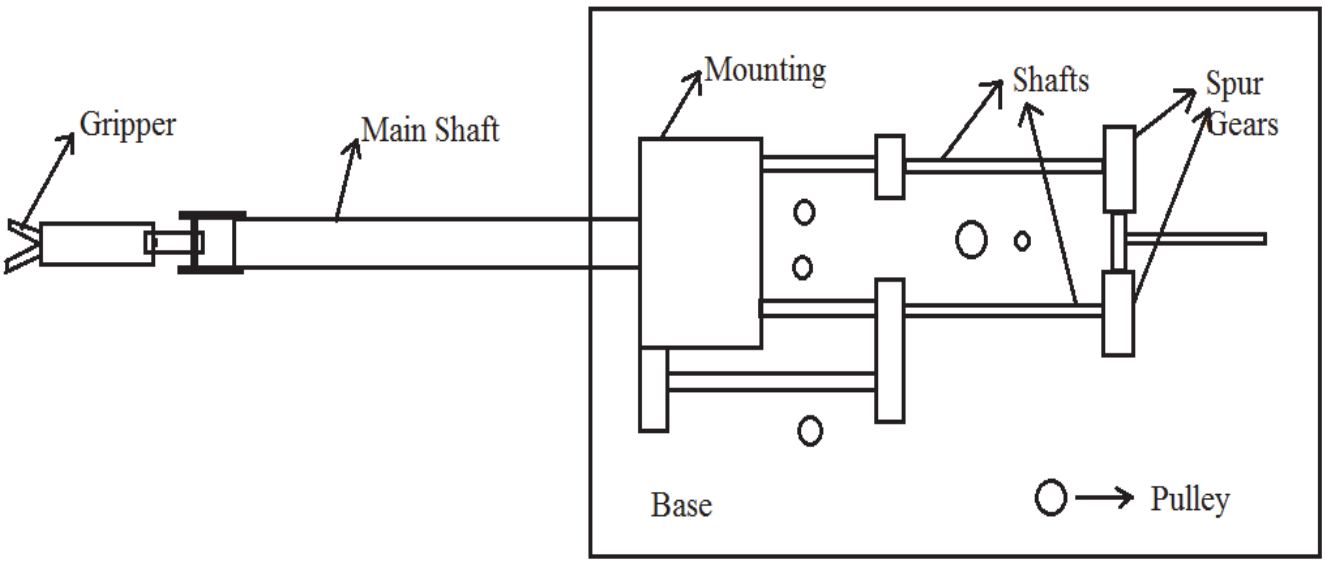


Fig. 4.8: Complete Manipulator (Prototype)

The complete assembly except the motors are shown in the above Fig. 4.10. In this figure the third rotation axis is also shown, which is about manipulator's main axis. It means that the end inserted in two bearings is to be rotated according to desired output. All the three axis of rotations are driven by different pulleys which has no relation to each other as output or input. Conclusions and scope of future work are discussed in next chapter of the thesis.

CHAPTER 5 CONCLUSION

5.1 CONCLUSION

The objective of this Thesis was to develop a robotic model for the serial manipulator using bond graph method and Denavit-Hartenberg method. Simulation results are compared with the experimental results to validate the model. The conclusions made from this Thesis work are given below:

- By using the Denavit-Hartenberg method complete forward kinematics are developed and with the help of Newton-Rapson method the inverse kinematics are generated. The solutions of forward and inverse kinematics are precise and exact. The final transformation matrix is completely dependent on the coordinate system of the particular joint and their parameters calculated from it. The link length and joint distance are responsible for the linear displacement and direction of end effectors points from its base coordinate system.
- Bond graph modelling is done for forward kinematics of serial manipulator. In this the effect of one joint rotation on another is considered, which has effects on the trajectory followed by the end-effector, but has no change in its end point to be reached by the end-effector. The mass of the serial manipulator after the joints towards end-effector has an effect on power to be transferred for that joint rotation.
- Both the Denavit-Hartenberg method and simulation results indicate that the trajectory of the end-effector is followed as per the rotation of angles given to the different joints of serial manipulator. The inverse is done by D-H algorithm which proves that the coordinate point is given by inverse equations gives the precise angles of all three joints.
- For the power transmission to the different joints of serial manipulator, the back hand housing for the motor and mechanism to transmit power is designed by using pulleys, gears and motor itself. It has a quiet complex design for giving power to three joints at different distance and which are also perpendicular to each other with one more mechanism for opening and closing the end-effector.
- For precise experiment work finishing in design and in its material of parts should be effective.

5.2 SCOPE OF FUTURE WORK

Grounded on this thesis, the observing fields of work are suggested for further probe:

- Bond graph model of inverse model for precise results of complete organization with forward model.
- Prepare bond graph model incorporating slippage of belt drive transmission of power and friction between the pulley and belt.
- Modification in the design of motor housing for power transmission to different joints by using bevel gears.
- When serial manipulator work in real environment, then it will not follow the exact trajectory given to it because of its weight, pressure and environmental condition. Thus, it can be omitted by giving time gap at some intervals to achieve the exact coordinate point.

REFRENCES

[1] http://www.roboticsbusinessreview.com

[2] Jazar, R.N. Theory of Applied Robotics: Kinematics, Dynamics and Control. *Springer New York Dordrecht Heidelberg, London*, 2006.

[3] www.howstuffworks.com

[4] Guthart, G.S., and Salisbury, J.K. The Intuitive Telesurgery System: Overview and Application. *IEEE International Conference on Robotics and Automation,* 2000, **1**, 618–621.

[5] Salisbury, J.K. The heart of microsurgery. *Mechanical Engineering Magazine, ASME International*, 1998, **210**, 47–51.

[6] Falk, V., McLoughlin, J., Guthart, G., Salisbury, J.K., Wather, T., Gummert, J., and Mohr, F. Dexterity enhancement in endoscopic surgery by a computer controlled mechanical wrist. *Minimum Invasive Therapy and Allied Technology*, 1999, **8**, 235–242.

[7] Liu, D., Cong, L., Liu, J., and Xu, D. The Research on the Control System of the Minimally Invasive Surgical Robot. *IEEE Conference on Robotics, Automation and Mechatronics*, 2008, 592–597.

[8] Sun, L.W., Meer, F.V., Bailly, Y., and Yeung, C.K. Design and Development of a Da Vinci Surgical System Simulator. *International Conference on Mechatronics and Automation*, 2007, 1050–1055.

[9] Sun, L.W., Meer, F.V., Schmid, J., Bailly, Y., Thakre, A.A., and Yeung C.K. Advanced Da Vinci Surgical Simulator for surgeon training and operation planning. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 2007, **3**, 245–251.

[10] Hubens, G., Coveliers, H., Balliu, L., Ruppert, M., and Vaneerdeweg, W. A Performance study comparing Manual and Robotically assisted Laparoscopic Surgery using Da Vinci System. *Surgical Endoscopy and other Interventional Techniques*, 2003, **17**, 1595–1599.

[11] Kovacs, L., Haidegger, T., and Rudas, I. Surgery from a Distance-Application of Intelligent Control for Telemedicine. *IEEE International Symposium on Applied Machine Intelligence and Informatics*, 2013, 125–129.

[12] Papachristos, C., Tsoulkas, V.N., and Pantelous, A.A. Advances in Medical Education on Surgical Techniques using Satellite Communications. *UKSim European Symposium on Computer Modelling and Simulation*, 2009, 361–366.

[13] Rosen, J., and Hannaford, B. Doctor at a distance. *IEEE Spectrum*, 2006, **43**, 34–39.

[14] Haidegger, T., Kovacs, L., Preitl, S., Precup, R.E., Benyo, B., and Benyo, Z. Controller Design Solutions for Long Distance Telesurgical Applications. *Journal of Artificial Intelligence*, 2011, **6**, 48–71.

[15] Oetomo, D., Daney, D., and Merlet, J.P. Design Strategy of Serial Manipulators With Certified Constraint Satisfaction. *IEEE Transactions on Robotics*, 2009, **25**, 1–11.

[16] Chen, Y.H., and Kuo, C.Y. Fundamental Properties of Rigid Serial Manipulators for Control Design. *American Control Conference*, 1999, **5**, 3003–3007.

[17] Fresonke, D.A., Hernandez, E., and Tesar, D. Deflection Prediction for Serial Manipulators. *International Conference on Robotics and Automation*, 1988, **1**, 482–487.

[18] Kelley, F.A., and Huston, R.L. Modelling of Flexibility Effects in Robot Arms. *Proceedings of the Joint Automatic Control Conference, Charlottesville,VA,* 1981.

[19] Wander, J.P., and Tesar, D. Software for Pipelined Computation of Manipulator Madelling Matrices. *IEEE Journal of Robotics and Automation*, 1987, **3**, 112–119.

[20] Fresonke, D., and Tesar, D. Deflection Prediction for Quasi-Static Serial Manipulators. *Program for Robotics Manufacturing and Logistics Manufacturing systems Engineering, University of Texas*, 1985.

[21] Fassi, I., Legnani, G., Tosi, D., and Omodei, A. Calibration of Serial Manipulators: Theory and Applications. *Industrial Robotics: Programming, Simulation and Applications*, 2006, **6**, 147–170.

[22] Oetomo, D., and Ang, M.H. Singularity robust algorithm in serial manipulators. *International Conference on Flexible Automation and Intelligent Manufacturing*, 2009, **25**, 122–134.

[23] Llyod, J. Removing the singularities of Serial Manipulators by Transforming the Workspace. IEEE *International Conference on Robotics and Automation*, 1998, **4**, 2935–2940.

[24] O'Neil, K., Chen, Y., and Seng, J. Removing Singularities of Resolved motion rate Control of Mechanisms including Self Motion. *IEEE Transactions on Robotics and Automation*, 1997, **13**, 741–751.

[25] Chen, Y.H., Leitmann, G., and Chen, J.S. Robust Control for Rigid Serial Manipulators: A General Setting. *American Control Conference*, 1998, **2**, 912–916.

[26] Rocha, C.R., Tonetto, C.P., and Dias, A. A Comparison between the Denavit-Hartenberg and the Screw-based methods used in Kinematic Modelling of Robot Manipulators. *Robotics and Computer Integrated Manufacturing*, 2011, **27**, 723–728.

[27] Denavit, J., and Hartenberg, R. A Kinematic notation for lower-pair Mechanisms based on Matrices. *ASME Journal of Applied Mechanics*, 1955, **22**, 215–221.

[28] Dai, J. An historical review of the theoretical development of rigid body displacements from Rodrigues Parameters to the Finite Twist. *Mechanism and Machine Theory*, 2006, **41**, 41–52.

[29] Campos, A., Guenther, R., and Martins, D. Differential Kinematics of Serial Manipulators using Virtual chains. *Journal of Brazilian Society of Mechanical Science and Engineering*, 2005, **27**, 345–356.

[30] Farzan, S., and DeSouza, G.N. From D-H to Inverse Kinematics: A Fast Numerical Solution for General Robotic Manipulators using Parallel Processing. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, 2507–2513.

[31] Xuewen, R., Yibin, L., and Rui, S. Kinematic Analysis of a Shotcreting Robot. *International Conference on Mechanic Automation and Control Engineering*, 2010, 2640–2643.

[32] Gonzalez, M.A., Angeles, J., and Ranjbaran, F. The Kinematic Synthesis of Serial Manipulators with a Prescribed Jacobian. *International Conference on Robotics and Automation*, **1**, 450–455.

[33] Santos, V.J., and Valero, F.J. Reported Anatomical Variability Naturally Leads to multimodal Distributions of Denavit-Hartenberg Parameters for the Human Thumb. *IEEE Trabsactions on Biomedical Engineering*, 2006, **53**, 155–163.

[34] Karlik, B., and Aydin, S. An improved approach to the Solution of Kinematics Problems for Robot Manipulators. *Engineering Applications of Artificial Intelligence*, 2000, **13**, 159–164.

[35] Vaz, A., Kansal, H., and Singla, A. Some aspects in the Bond Graph Modelling of Robotic Manipulators: Angular Velocities from Symbolic Manipulation of Rotation Matrices. *Conference on Convergent Technologies for the Asia-Pacific Region*, 2003, 1, 294–299.

[36] MATHEMATICA 4.0, Wolfram Research Inc. Champaign, Illinois, USA.

[37] Yen, C., and Masada, G.Y. Dynamic Analysis of Flexible Structures using Extended Bond Graphs, 1992, 2747–2753.

[38] Samanta, B., and Devasia, S. Modelling and Control of Flexible Manipulators using Distributed Actuators: A Bond Graph Approach, 99–104.

[39] Smbols User's Manual, High Tech Consultants. http://www.htcinfo.com.

[40] Mukherjee, A., Karmakar, R., and Samantaray, A.K. Bond Graph in Modelling, Simulation and Fault Identification, I.K. International Publishing House Pvt. Ltd. 2006.

[41] Borutzky, W. Bond Graph Modelling and Simulation of Mechatronic Systems: An Introduction into the Methodology. *European Conference on Modelling and Simulation*, 2006.

[42] Craig, J.J. Introduction to Robotics: Mechanics and Control, Pearson Education Inc. 1989.

[43] Schilling, R.J. Fundamentals of Robotics: Analysis and Control, Eastern Economy Edition. 1990.

[44] Virk, G.S., and Haidegger, T. Classification Guidelines for Personal Care Robots—Medical and non-medical applications. *IEEE IROS Workshop on Safety in Human- Robot Coexistence & Interaction, Vilamoura, Portugal*, 33–36, 2012.

CURRICULUM VITAE

Aashish Kumar did his graduation from Haryana College of Technology and Management in Bachelor of Mechanical Engineering (B. Tech), in the year 2011. After that he has worked as Lecturer in Institute of Technology and Sciences, District-Bhiwani, Haryana in the year 2011–2012. In the year 2012, he joined in the Master of Engineering (CAD/CAM Engineering) Programme at Thapar University, Patiala, Punjab. His ME thesis work is in the area of Robotic Surgical Manipulator as Experimental and Analytical Modelling and Control of Precision Surgery Manipulator. One conference paper is under preparation.