بسم الله الرحمن الرحيم

GRADUATION PROJECT

Upper-Limb Lifting-Load Assistive Exoskeleton

By

Mohammad M. Altamimi, Mechatronics Engineering
Mohammad H. Iqnabi, Mechatronics Engineering
Moath Y. Qdeimat, Applied Electronics

supervisor

Prof. Dr. Karim Tahboub

Submitted to the College of Engineering, College of Applied science In partial fullment of the requirements for the degree of Bachelor in Mechatronics Engineering, Applied Electronics

Palestine Polytechnic University

February 5, 2017

Contents

List of Figures

List of Tables

Part I

Theoretical Work

Design of Upper-Limbs Lifting-load Exoskeleton

Chapter 1

Introduction

1.1 Introduction

An exoskeleton (from Greek, (x) outer and (skeletos) skeleton) is the external skeleton that supports and empowers human body, A powered exoskeleton (also known as powered armor, exocrine, hard-suit or Exosuit) is a wearable mobile machine that is powered by a system of motors, pneumatics, levers or hydraulics that allow for limbs movement, increased strength and endurance [1].

Many people suffer from the difficulty of movement in the lower limb of the body for several reasons, like Cerebrovascular accident (CVA) an effect on the entire part of the body. Including the limbs upper or lower and prevent movement in the injured limb. The second, reason it is Muscular dystrophy, patient is not able to move any limbs, as normal. Other reason is presence of cut in the part of the nerve that transmits orders. Therefore, do not reach orders the muscles completely. As the number of infections by the Palestinian Ministry of Health (2000-2014) [2], the number of injured has reached 1229 cases. In the lower limbs is 34% from all injuries. In Hebron, there are approximately 30 cases monthly of CVA injuries. This forms the medical eld applications of Exoskeleton.

The Exoskeleton being more and more important in military field also, the soldiers carry heavy battle equipment, and that impedes moving and running easily, the Exoskeleton is being a trump card in battle, which capable the soldiers to be more functional on war when they carry heavy load.

In general, the Exoskeleton is used in assistive eld in daily activities such as walking, working, playing, going up stairs etc, and to make the human life more simple and functional.

To control the Exoskeleton movement, we depend on neuromuscular cycle. The brain sends the neurological signal to spinal cord which controls the human limbs then the spinal cord send orders to muscles to exion and extension. The muscle sensor sense this ex- ionextension movement and transmit it to microcontroller. The signal is processed and sent to the motors, Which works to move the limbs connected with the device to make its movement closer to the natural, and that helps the human to move and do everyday functions.

1.2 Recognition of the need

Before beginning any project, you should ask yourself, why you're going to make this project. If you answer this question correctly, you put your hand on the need, you may make a researches about your idea, reading article, visiting and asking specialized actors. To recognition of the need and define the problem. In this section we will define the need of the exoskeleton on each field separately.

To recognition of the need in medical filed, we showed a statistics in our society, and visited hospital and Rehabilitative centers and the result being as follow.

Distribution of handicapped persons registered by region of injury in west bank for ministry of health from 29/9/2000 to 31/12/2014 [2]:

1. Percentage Upper limb disability is 15.4% and the number of disability cases 543.
2. Percentage Lower limb disability is 34.9and the number of disability cases 1229. The total number of disability cases registered is 3524.

Al-Ahli Hospital, at The. 10/3/2016

(The Physical and Rehabilitative Medicine Department Dr. Samih Dweik)

- The percentage of disability in lower limb to another disability regions in Hebron 30-40 %.
- Every day 2-5 cases is Cerebrovascular Accident cases (at rate 30 cases in a month) .And the most CVA cases becomes have movement difficulty especially in lower limb.
- Muscular dystrophy percentage is so little approximately 1-2 cases in a year.
- Also Poliomyelitis percentage is so little (Approximately 1 case in a year).

Al- Ihsan Charitable Society, at The. 3/3/2016

We visited Al-Ahsan Charitable Society, and we talked about how the limbs work and receive the neurological signal .The doctor Kamal Al-Najie presented an idea about the project .its purpose is to help CVA patients, who have problem in moving Metatarsal .So that affected on walking of the patient .We asked about persons have problem in society, but Al-Ihsan Charitable Society is helping handicapped who have mental problem. We talked with Dr. Wafa al-Natsheh (Official Department Physical and Rehabilitative). We asked about explanation of the cycle transfer signal from brain to muscle and in opposite direction, and we ask about explanation if we can take muscle signal from any handicapped patient. Dr. Wafa al Natshah explain the cycle .that work by two type of nerve: 1-sensing nerve 2-motoring nerve. Nerves serve as the wires of the body that carry information to and from the brain. Motor nerves carry messages from the brain to muscles to make the body move. Sensory nerves carry messages to the brain from different parts of the body to signal pain, pressure, and temperature. So Dr. Kamal Al-Najie advise us to visit K. Abu Raya Rehabilitation Center in Ramallah and Bethlehem Arab Society for Rehabilitation.

K. Abu Raya Rehabilitation Center, at sat. 5/3/2016

When we went to center ,we found a lot of cases ,but most cases that treats with it is spinal cord injury cases .The problem in these cases is cutting for the spinal cord ,even the cutting is complete or incomplete .That causes disconnected for the signal from brain to muscle ,and that stopped muscles from moving. Rolla Humedan (Official Department Physical and Rehabilitative) and Nadia help us to know about the cases that treat with them, but we divide the cases for two types as depend on level of cutting spinal cord. The first type is $(1-3)$ cutting level in spinal cord in this case the cutting will effected on all muscle under region of cut ,and that effect on work of heart , Lung ... etc. And in this case the all limbs cannot move because disconnecting in signal for muscle. Maybe the cutting in this type is complete or incomplete, if the cutting incomplete not all of muscles are stop on working. The second type (4-5) cutting level in spinal cord .In this type the cutting happened in lower back region ,so the effect of cutting is happened in lower limb region. And also in this type we have completely cutting or incomplete cutting.

In complete cutting all muscles in leg is never work, so the patient cannot walk .Although in incomplete cutting some of muscles work, the patient cannot walk.

For recognition of the need in assistive and military fields we read articles, search on the internet, asked some industrial factories and Experienced person in this filed. And we showed that there are an important to exoskeleton in:

1. Reduction the loads carried by Factory workers and soldiers bodies.
2. Being more fast and functional on works and battle.
3. Protect them from spine and joint injured and diseases.

1.3 History

The earliest exoskeleton-like device was a set of walking, jumping and running assisted apparatus developed in 1890 by a Russian named Nicholas Yagn. As a unit, the apparatus used compressed gas bags to store energy that would assist with movements, although it was passive in operation and required human power. In 1917, US inventor Leslie C. Kelley developed what he called a pedomotor, which operated on steam power with artificial ligaments acting in parallel to the wearers movements. With the pedomotor, energy could be generated apart from the user.

The first true exoskeleton in the sense of being a mobile machine integrated with human movements was co-developed by General Electric and the United States military in the 1960s. The suit was named Hardiman, and made lifting 250 pounds (110 kg) feel like lifting 10 pounds ($4.5kg$). Powered by hydraulics and electricity, the suit allowed the wearer to amplify their strength by a factor of 25.

HAL is one of the most advanced exoskeleton suit is the first one in medical field with a long history since 1992 when the suit was released. Behind this project is a Japanese company called Cyberdyne, which recently released the fifth version of the costume with new and improved features. HAL 5 developed to use in assistive field to carry large load in 2004. Fig1.1 represent the important stages, which Exoskeleton go through it.

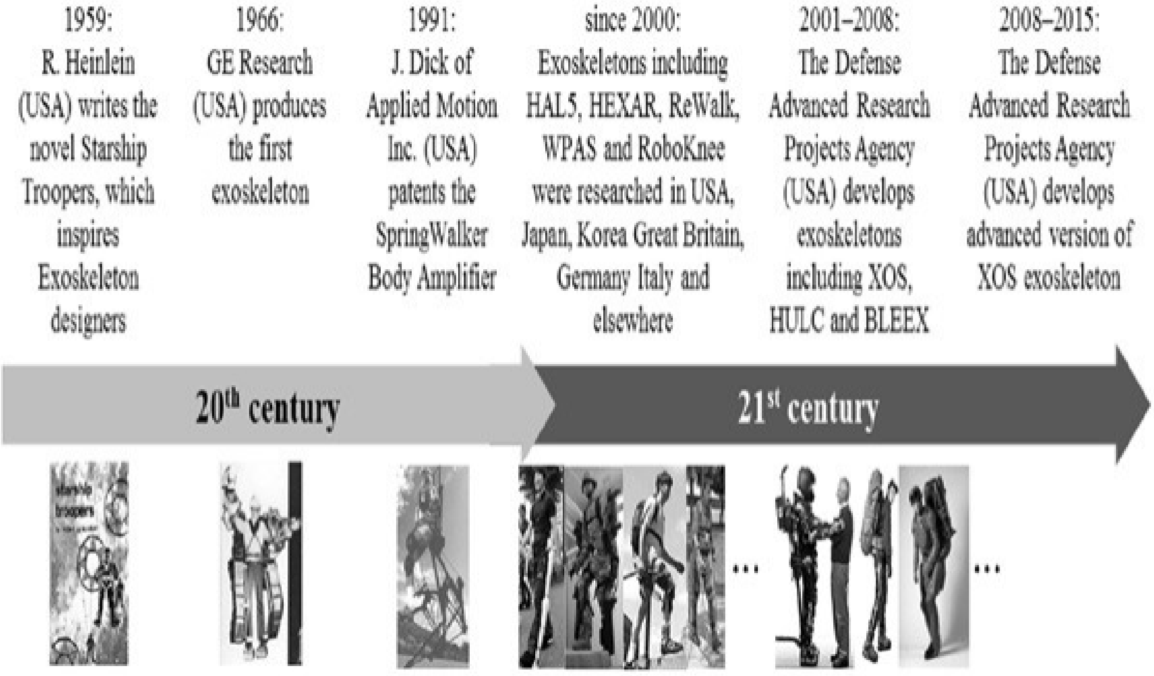


Figure 1.1: History [3]

1.4 Literature Review

In this section, we represent the most important papers that published in IEEE and mechatronics, IEEE/ASME Transaction, about this subject. And we will explain its in terms of designing, applying, function specification and equipment used in each in it.

1.4.1 An Assistive Control Approach for a Lower-Limb Exoskeleton

This study aims to make an exoskeleton robot, for Lower-extremity hemiparesis after stroke. The article authors made a study for stock patient and they found that 800,000 people in the US suffer a stroke or cerebrovascular accident (CVA). And approximately 200,000 annually are affected by lower-extremity hemiparesis. This article is concerned with actuator and control approach as we see.

Actuators: Four control actuators (brushless DC motors acting through speed reduction transmissions). Torques at each joint of approximately 20 Nm, and peak torques of approximately 80 Nm.

Control: Three types of behaviors, gravity compensation, feed-forward movement assistance during swing, and knee joint stability reinforcement during stance form of emulated spring-damper elements [4].

1.4.2 Design and Control of Robotic Exoskeleton with Balance Stabilizer

This study aims to help the stroke and spinal cord injury patients to regain the ability of walking. And its concerned on how to make balance waking with exoskeleton, which can generate dynamically stable and tunable gait pattern. And they present some functional specification as control, actuator and mechanisms.

Actuator: there are two DC motors at each leg powering up the joints by using maxon flat motor.

Control: gait pattern generation algorithm then inverse kinematics ZMP.

Mechanisms: harmonic drive mechanism (ratio: 100:1). Legs: one degree of freedom at the hip (active); one degree of freedom at the knee (active) stance phase knee flexion, knee mechanisms, foot mechanisms [5].

1.4.3 Upper-Limb powered exoskeleton design

This article works on Exoskeleton for rehabilitation medicine and virtual reality simulation, with 7-DoF by hand, they makes kinematics and dynamic analysis, mechanical human machine interfaces, safety requirement, and present functional specification for their Exoskeleton.

Sensor: EMG sensor to take orders from muscles, potentiometers and encoders for feedback, limit switches and emergency for safety.

Actuator: high power-to-weight ratio, torque, high power-to-weight motor (1.0Nm, 4.2Nm/kg). Motors are rare earth (RE), brushed motors (Maxon Motor, Switzerland). Vary with joint from shoulder to wrist.

Mechanisms: Cable drive system, two stage pulley reduction, semi-circular bearing design to limit movement, joint idler pulley [6].

1.4.4 Control of a Robotic Hip Exoskeleton

This study aims to assist or replace movement functionality and to achieve controllability and acceptability for an active pelvis orthosis (APO), for normal human, and it use this function specification.

Sensors: EMG sensor and Shoes pressure-sensitive.

Actuators: Torque up to 35 N.m for hip joint, Series elastic actuator (sea) and Servo DC Motor.

Mechanisms: two rotating linkages connecting it with the thighs; the frame carries two actuation. And it works for human weight 70kg with velocity 4km/h [7].

1.4.5 Light arm exoskeleton for rehabilitation

This study aims to make an Exoskeleton for rehabilitation stroke long-term disability, based on EMG signal, they work on dynamic and kinematics analysis for upper-limb, they several experiments to three arm joints shoulder elbow and wrist, in dynamic and static mode, Kinematic and EMG results demonstrated significant back-drive-ability and transparency of the exoskeleton as the subjects could perform the three daily life motor tasks in a quite physiological way [8].

1.4.6 Design and Actuator Selection of an Extremities Exoskeleton

This study works on wheelchair users experience difficulties when moving on rough surfaces and climbing staircases which limit their social life, for who lost their walking ability due to disease, trauma, or aging, they define the degree of freedom and range of motion for each joint, and generate mechanical design for each joint and for whole Exoskeleton. This article present some functional specification as we see.

Sensors: 40 sensors some of which are embedded in the shoe pads, the EMG Sensors, force sensors are placed at foot soles and Limit switches are embedded in the joints in order to prevent excessive joint rotations.

Actuators: powered by unique pneumatic actuators legs driven by hydraulic actuators. The hip and knee joints are powered by electrical actuators such as pneumatic, hydraulic, electrical, piezoelectric, electro active polymers. Control: d fuzzy logic control, ATHM 800 data acquisition card is used in transferring data.

Mechanisms: Waist design Upper and lower leg design Foot,hip and knee design Waist design Upper and lower leg design Foot, hip and knee design.

Battery: two 23.5 V 10 Ah Li-Po battery packs [9].

1.4.7 Berkeley Lower Extremity Exoskeleton (BLEEX)

This study made to solve the problem of Incapable of transporting heavy materials over rough terrain or up staircases. In military application. They works in control approach, degree of freedom, range of motion for each joint and mechanical and electrical design. They present some functional specification as we see.

Sensors: EMG and Force sensor.

Actuators: Unique Pneumatic actuators .Linear actuators, Hydraulic actuator.

Control: novel

control scheme, high-powered compact power supplies, special communication protocol and electronics, and a design architecture to decrease the complexity and power consumption.

Mechanisms: 3dof hip, 1dof knee, 3dof ankle, Shank design and Tight design.

Power: Power consumption 1140W Position on lower torso.

The device can carry 75kg and walk with velocity 1.3m/s [10].

1.4.8 Titan arm

This project makes by three student Form University of Pennsylvania, which device makes for patient rehabilitation suffer from stroke, the device link made from titanium, it can carry load up to 70Ib, when you feel the load is 30Ib. the project functional specification as we see.

Sensor: EMG sensor.

Actuator: pneumatic system.

Control: controlled by Arduino [11].

1.4.9 Mind-walked Exoskeleton

This study aims to empower paraplegia to walk and stand. The article authors found that spinal cord injury incidence of SCI lies between 10.4 and 83 per million inhabitants per year worldwide. This article depend on patient mind to control movement. They made mechanical design and control approach for exoskeleton. The function specification as follow.

Sensor: 17-bit resolution Encoder for feedback.

Actuators: series elastic actuator, which can deliver 100 Nm torque and 1 kW power. Ball screw and an out runner BLDC motor.

Control: force-controllable actuators are desired. These allow different control implementations, e.g. force control, bio-inspired control, and impedance control step-width adaptation (SWA) algorithm is presented in this paper, in an attempt to improve gait stability.

Mechanisms: Ball and socket for hip antagonist spring pairs with its neutral position at 0 degree ankle angle. Spring has a double-spiral disc shape and is made of a single piece of highgrade titanium [12].

1.4.10 Hydraulically Powered Upper Limb Exoskeleton

An exoskeleton made for stroke patient developed to be used in identifying the reflex properties of the arm in stroke survivors. Information on joint reflexes helps in designing optimal patient specific therapy programs. The device is dynamically transparent by combining a 8 lightweight

skeleton with high power to weight ratio actuators. Its weight is reduced five times as gravitational forces are lowered using a model-based gravity compensation algorithm. The impedance controller ensures tracking of a cycloidal joint angle reference. A cycloid with an amplitude of 1.3 rd and a maximum velocity of 6.5 rd/s.
Actuators: The torque-controlled motors have a maximum torque bandwidth of 97 Hz which is required for fast torque perturbations and smooth zero impedance control.
Mechanisms: It is supported by a passive weight balancing mechanism to compensate for the weight of the exoskeleton and the human arm. Various self-aligning mechanisms allow the human joint axes to align with the axes of the exoskeleton which ensure safety and short don/doff times [13].

In next Chapter we put a last ten studies in Tables as a matrices, we make a comparison between it in these field (problem(Need), Group beneficiary which we discuses in this chapter, and Algorithm(controller), in table 2.1 , (sensor, actuator and mechanisms), the weight of Exoskeleton and the weight can lifted by exoskeleton , power and speed. All of these functional specification and conceptual design parameter shown in Table 2.2 and 2.3 .

1.5 Upper-Limp Assistive Exoskeleton

As we saw previously, there are many types of exoskeleton, with vary application, vary body region, vary degree of freedom. In our project we decide to work on exoskeleton, which Assist and control upper body limbs (i.e. hand region), with four degree of freedom, two degree for each hand, in heavy load assistive application.

When we carry certain load, on hand we have specify muscles most of load force reac- tion reect on them, these muscles are biceps, triceps and forearm muscle, if we assist this ceps we have carry most of the load. And to avoid the reaction force to go through human body, we will debate this force to earth by mechanism connect the exoskeleton to earth.

1.6 Project Objective

our project objective is to design, build and control an external skelton (Exoskeleton), to assise indestrial factory worker to lifting some load rach to 50kg.

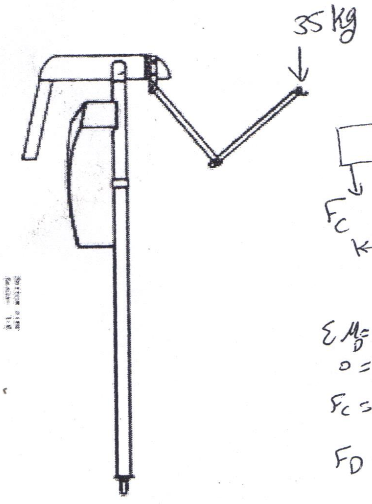
1.7 Project Requierment

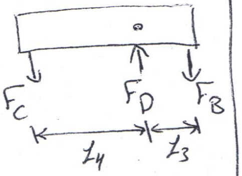
1. Enable person lifting load up to 50kg.
2. Ergonomic design.
3. Doesn't constrain human motion.
4. Earthing reaction force.
5. Enable tendency.
6. HMI being easy and fast, and inference of human intention.
7. The power is, 1 phase 220 volt AC.
8. Design according safety requirement.

1.8 Exoskeleton Mechanical Design alternative

After detected the conceptual design, we start finding suitable design which done by brain storming to generate many ideas and record all idea as a morphological chart which give a simple solution to our problem. We make analysis for forces and reactions of exoskeleton design as following pages.

Dates – 26/4/2016



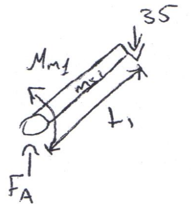


$$\Sigma M_D = 0$$

$$0 = -F_C * L_4 + F_B * L_3$$

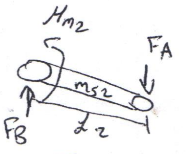
$$F_C = \frac{F_B * L_3}{L_4}$$

$$F_D = F_B + F_C$$



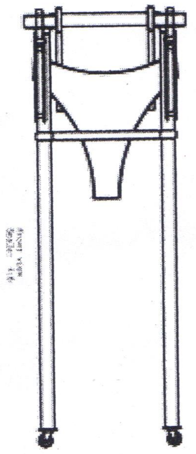
$$F_A = 35 + m_{s1}$$

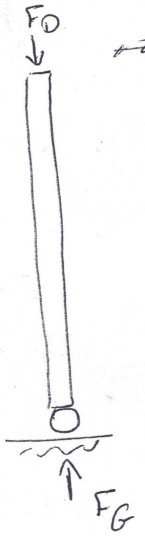
$$M_{m1} = 35 * L_1 + m_s * L/2$$



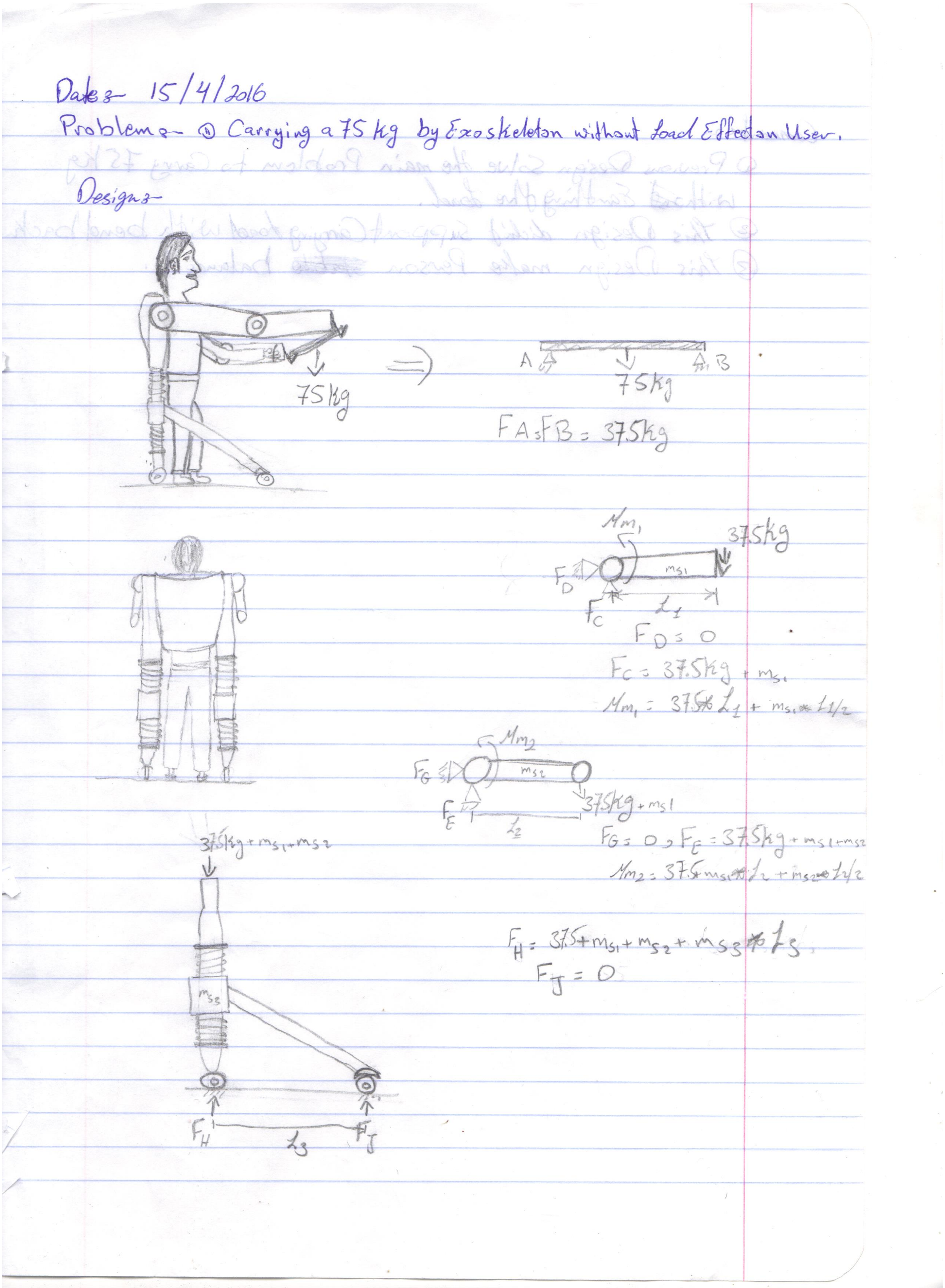
$$F_B = F_A + m_{s2}$$

$$M_{m2} = F_A * L_2 + m_{s2} * L_2/2$$





$$F_G = F_D$$



Date:- 15/4/2016

Problem:- ① Carrying a 75 kg by Exoskeleton without Load Effect on User.

Design:-

$F_A = F_B = 37.5\,kg$

$F_D = 0$

$F_C = 37.5\,kg + m_{s1}$

$M_{m_1} = 37.5 * L_1 + m_{s1} * L_1/2$

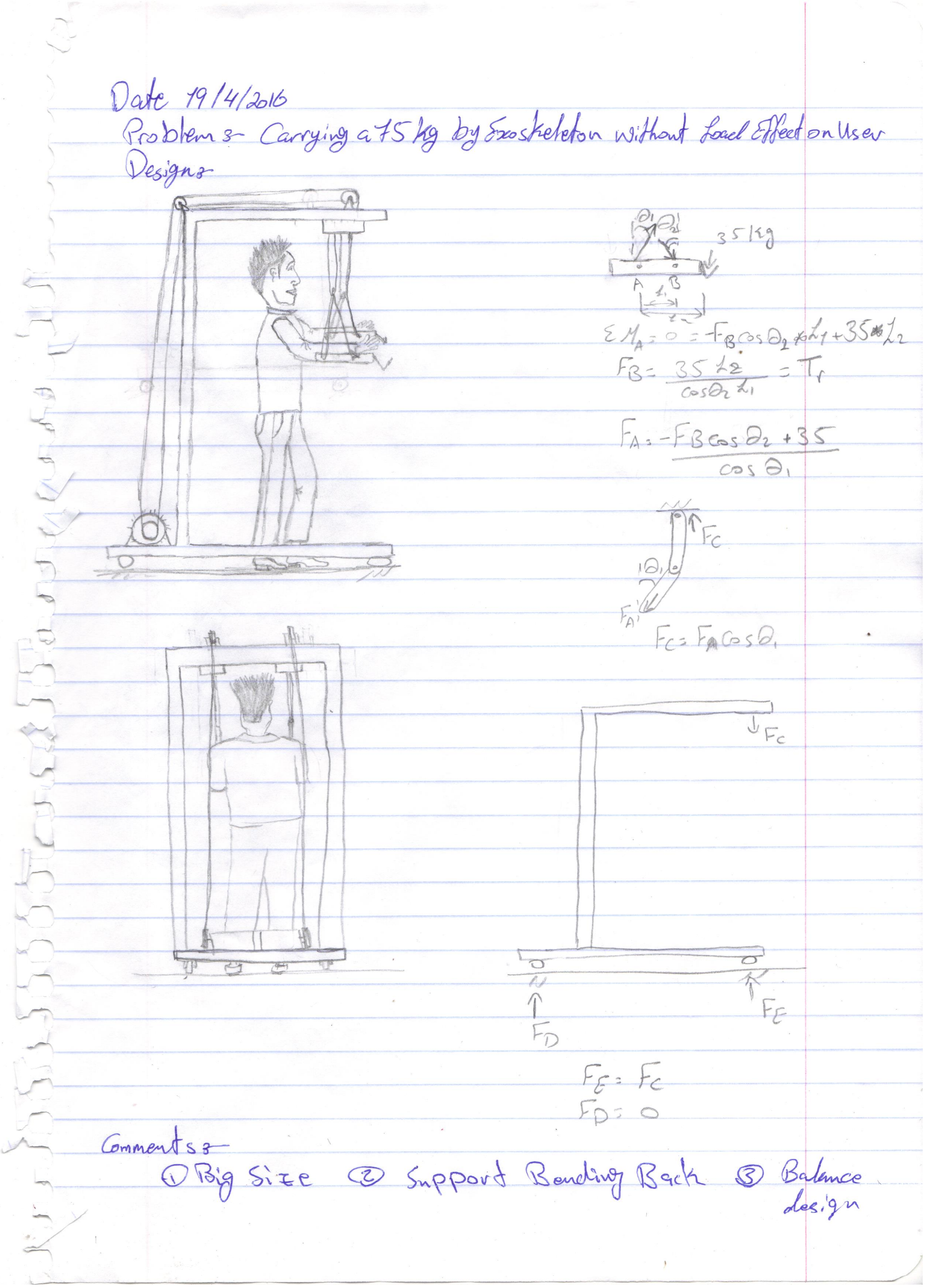
$F_G = 0, F_E = 37.5\,kg + m_{s1} + m_{s2}$

$M_{m_2} = 37.5\,m_{s1} * L_2 + m_{s2} * L_2/2$

$F_H = 37.5 + m_{s1} + m_{s2} + m_{s3} * L_3$

$F_J = 0$

Figure 1.2: Design No.2



Date 19/4/2016

Problem 3- Carrying a 75 kg by Exoskeleton without Load Effect on User

Designs:

$\Sigma M_A = 0 = -F_B \cos\theta_2 * l_1 + 35 * l_2$

$F_B = \frac{35 l_2}{\cos\theta_2 l_1} = T_r$

$F_A = \frac{-F_B \cos\theta_2 + 35}{\cos\theta_1}$

$F_C = F_A \cos\theta_1$

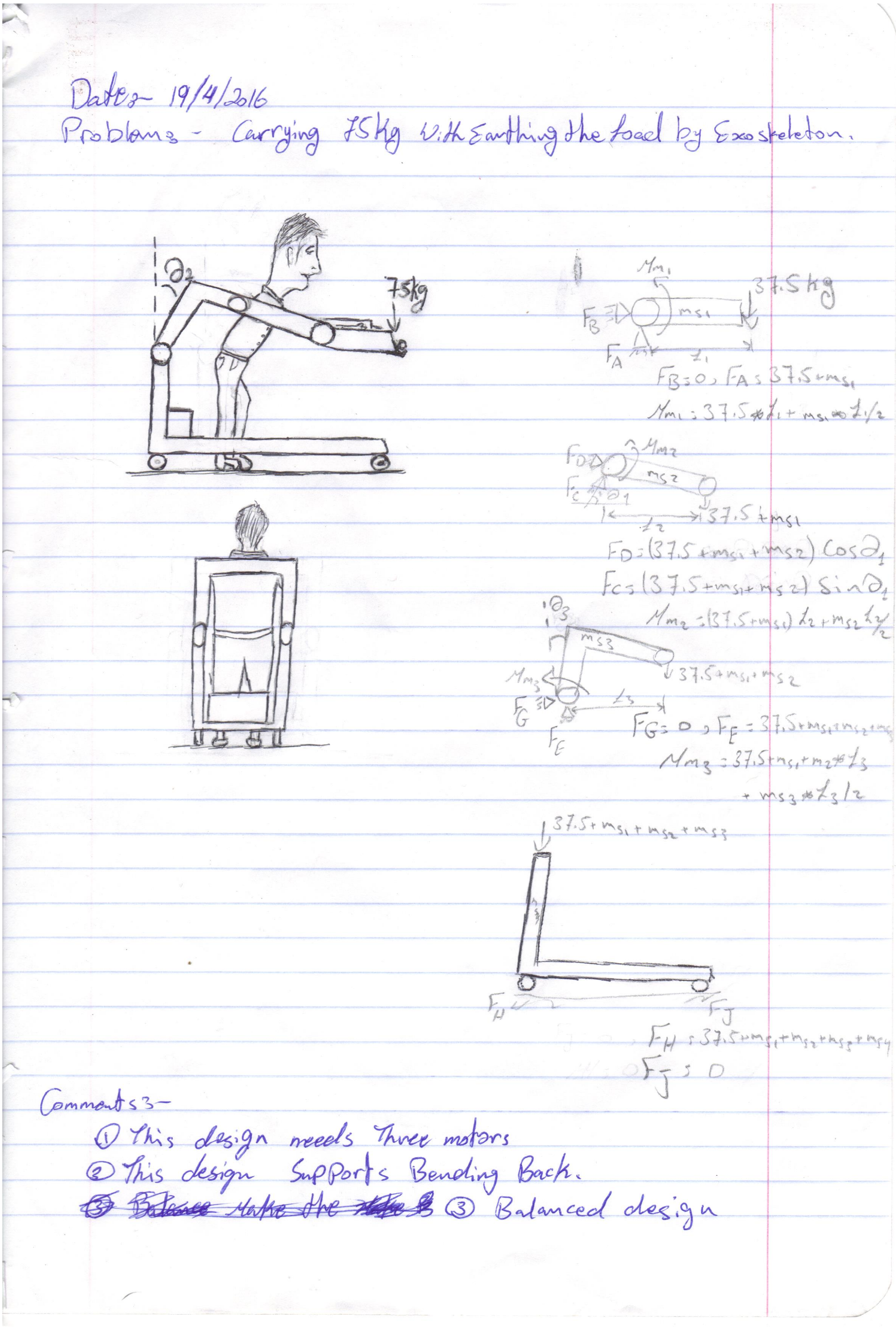
$F_E = F_C$

$F_D = 0$

Comments:

① Big Size ② Support Bending Back ③ Balance design

Figure 1.3: Design No.3



Date:- 19/4/2016

Problems - Carrying 75Kg With Earthing the load by Exoskeleton.

$F_B = 0$, $F_A = 37.5 + m_{s1}$

$M_{m1} = 37.5 * l_1 + m_{s1} * l_1/2$

$F_D = (37.5 + m_{s1} + m_{s2}) \cos\theta_1$

$F_C = (37.5 + m_{s1} + m_{s2}) \sin\theta_1$

$M_{m2} = (37.5 + m_{s1}) l_2 + m_{s2} l_2/2$

$F_G = 0$, $F_E = 37.5 + m_{s1} + m_{s2} + m$

$M_{m3} = 37.5 + m_{s1} + m_2 * l_3 + m_{s3} * l_3/2$

$F_H = 37.5 + m_{s1} + m_{s2} + m_{s3} + m_{s4}$

$F_J = 0$

Comments:-

① This design needs Three motors

② This design Supports Bending Back.

~~③ Balance Make the [illegible]~~ ③ Balanced design

Figure 1.4: Final Design No.4

1.9 Designs Evaluation

The evaluation is depends on achieving the requirements by (perfect) design.

1.9.1 First Design

This design can achieve the solution of some requirements. First, it can lift 50 Kg. Second, speed and movement possible joint is compatible with human movement. Third, it is comfortable, but it has a problem in the additional weight that increase the load, so it make resis- tance on moving when pulling the wearable Exoskeleton. Fourth, earthing meaning transfer the force by exoskeleton frame far away about human skeleton and dissipate the reaction in earth. Fifth, this design does not achieve the back bending. Finally, this design is not selected, because it is not efficient to achieve all requirements design.

1.9.2 Second Design

At first, can lifting 50 Kg. Second, speed and movement possible joint is incompatible with all human movement. Third, is a comfortable and balance design. Fourth, earthing meaning transfer the force by frame far away about human skeleton and dissipate the reaction in earth. Fifth, this design does limited achieve the back bending. Finally, this design is not selected because it resist smoothly moving of the joints and didn’t give enough support for hand.

1.9.3 Third Design

First, can lifting 50 Kg. Second, speed and movement possible joint is compatible with human movement. Third, is a comfortable and balance design. Fourth, earthing meaning transfer the force by frame far away about human skeleton and dissipate the reaction in earth. Fifth, this design does not achieve the back bending. Finally, this design is not selected, because it does not achieve back bending.

1.9.4 Fourth Design

First, can lifting 50 Kg. Second, speed and movement possible joint is compatible with human movement. Third, is a comfortable and balance design. Fourth, earthing meaning transfer the force by frame far away about human skeleton and dissipate the reaction in earth. Fifth, this design does achieve the back bending. Finally, this design is selected, because it achieve all requirements. In next chapter will explain this conceptual design.

1.10 Project Outline

Depending on Mechatronics project design by Shetty, we separate the project to these three level, modeling/simulation, prototyping and deployment, the first one is separated to recognition of the need, conceptual design, function specification, sensors and actuator, mathematical, control system design and optimizing design, the second level separate to hardware in the loop simulation and optimizing designs, then third level contain, deployment of embedded software and life cycle optimization, as we see in Fig1.3, and all of these block will be complete or part of chapter as follow.

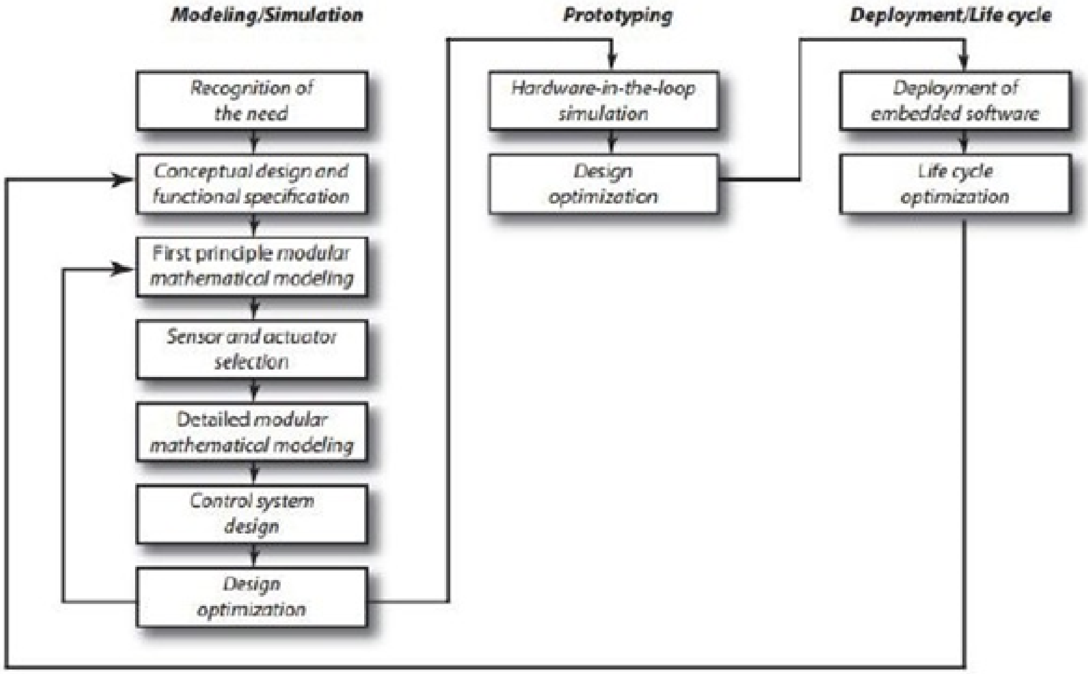


Figure 1.5: Mechatronics Project Design by Shetty

Chapter 2 presents the conceptual design and function specication of the project. in- cluding schematic diagram for our device, intelligent system and comparison between ten studies about conceptual design and function specication in (problem(Need) Group bene- ciary, sensors, actuator, Algorithm (controller), mechanism, weight, power and speed).

Chapter 3 is devoted first principles modular mathematical modeling, including several mechanical design of the project, and the chapter include first dynamics analysis and kinematics modeling.

Chapter 4 presents the component selection for our project. These component are sensors, actuators, microcontroller and mechanisms. With budget of the project.

Chapter 2

Conceptual Design and Functional Specifications

2.1 Introduction

In this chapter, we will identify conceptual design in section 2.2 and functional specification in section 2.3 that will be used in the project. As any Robot, the Exoskeleton consist of some sensors, actuators, controller and mechanisms. That works together in functionality interference to achieve main purpose that we need.

The sensor is an object whose purpose is to detect events or changes in its environment and surrounding, and then provide a corresponding output. The sensor output is used as in- put or feedback signal. Then this transmitted to embedded microcontroller. The microcontroller is a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. It is controlling in actuator action by comparing its value with the sensor values. Then microcontroller send signals to control actuators movement until reach desired response. These chain is discussed in schematic diagram in section 2.4.

Section 2.5 is talked about intelligence system and synergistic integration in mechanical, electrical and control part in our project. The algorithm of our exoskeleton that worked on it in designing this project.

2.2 Conceptual Design

After we define the need and requirement, and generate some solution by brain storming and morphological chart. Now we want to specify design concept that meet project requirement and functional specification chose candidate design, to move forward to functional specifications in next section.

Basically for any exoskeleton design we have a mechanical frame, that cover all passive and active joint, this frame must be light weight and in the same time strong enough to carry require load in safe mode, which require accurate design and correct material selection such as Aluminum. To carry any load we note that most of these load carried by biceps, triceps forearm ceps, so we need active joint in shoulder and elbow joint in each hand also we need some mechanisms on each joint that not constrain human motion, in the same time we need fast actuator and processor to control motion as ergonomic as we can.

The frame also must expand to reach the ground to dissipate the reaction force to earth (earth), in the earth also must expand to reach below load center of gravity of cretin load to cancel bending moment that may cause fail, as we see in Figure 2.1. In the same figure we have another joint in-addition of elbow and shoulder joint, this joint concept is to enable tendency of exoskeleton.

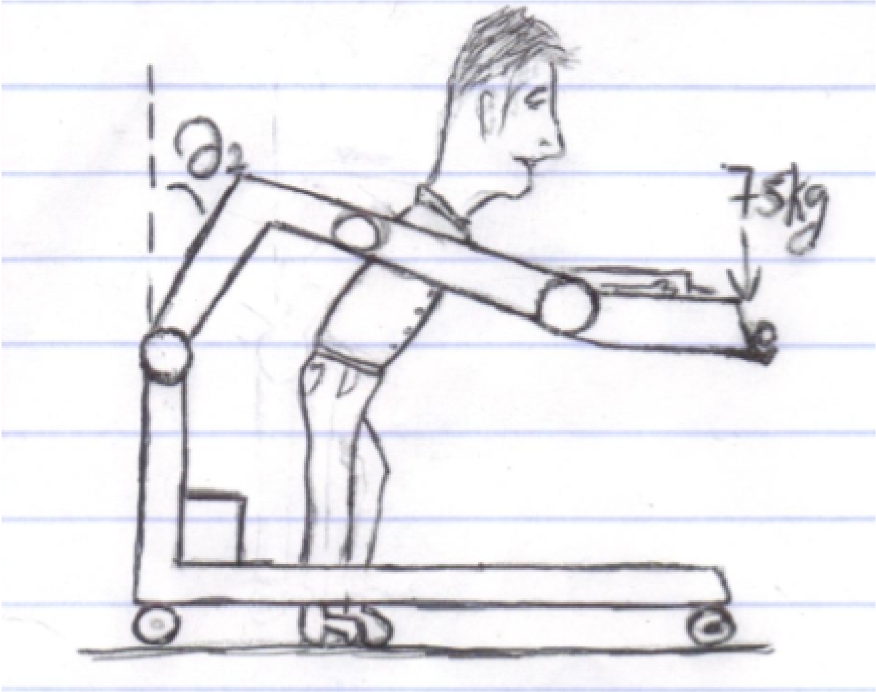


Figure 2.1: Inetial design hand drawing

For Human Machine Interaction (HMI), our design concept is to make inference of human intention, and this can be implemented by putting sensors on each joint that sense the human intention to move and control it by software to yield compliant design. The conceptual design of power delivered to the system we need just one wire 1-Phase 220 V to feed our system. And for safety we have three levels of safety mechanical,software and electrical level of safety.

2.3 Functional Specifications

It explains how everything works together in greater detail and compliant with requirements, and exploring the feasibility of a product, to achieve the most valuable design, And this will explain the functional specifications on three stages.

the functional specification is as following points.

1. Using the aluminum in frame building, because Aluminum is a relatively soft, durable, lightweight, ductile, and malleable metal and can achieve stress and strain for 50 kg load.
2. For joints is used to achieve ergonomic design and lifting the load by control in motors on elbow and shoulder.
3. An aluminum arm and forearm. the arm with length 23 cm and forearm length is 37 cm on both hands.
4. Four high Performance AC servomotor able to carry 50 Kg on both hands, 25 kg on each one, which mean 155 Nm on shoulder and 92.5 Nm elbow joint. With velocity equal 3 rpm.
5. Passive joint used to allow of user moving without constraint.
6. Two force sensors to take the movement signs, one for upward movement and other one for down word.
7. Eight Limit switches, two on each joint stopping mechanisms and algorithm are used to make the design more safety.
8. high Frequency controller, at least $10K\ Hz$ to achieve control requirements.
9. The power feeding the system must be 220 V.
10. Four wheals can carry the exoskeleton weight 35 Kg, and move with wearer.

Figure 2.2 below, The most valuable design, yield by the cross of the Requirement,which we talked about it in last chapter. conceptual design and functional specifications.And the validity, which we will talk about it in nal chapter.

Figure 2.2: The most Valuable Design [18]

As we said before in last chapter we made a matrices table in these field (problem(Need), Group beneficiary which we discuses in this chapter, and Algorithm(controller), in table 2.1 , (sensor, actuator and mechanisms), the weight of Exoskeleton and the weight can lifted by exoskeleton , power and speed. All of these functional specification and conceptual design parameter shown in Table 2.2 and 2.3 .

Article Name	problem(Need)	Group Beneciary	Algorithm(controller)
Adam Zoss [10]	Incapable of transporting heavy materials over rough terrain or up staircases.	Military	Novel control scheme, special communication protocol, and a design architecture decrease the complexity and power consumption
Wasnzo [15]	To enhance lower body strength To improve the efciency of the exoskeleton robot to carry a heavy load.	Aged, feeble, or disabled person	The main controller includes a FPGA programed to serve as communication and ADC for sub-controllers
Grazi [7]	Assist or replace movement. Achieve controllability and acceptability.	For normal people to reduce the activation peaks of the muscles.	Figure2 in article.
Lei [5]	To help the stroke and spinal cord injury pa- tients to regain the abil- ity of walking.	The stroke and spinal cord injury muscle weakness,paraplegia.	Figure 4. in article Then inverse kinematics (ZMP).
Spencer [4]	Lower-extremity hemi- paresis after stroke.	800,000 people in the US suf- fer a stroke or cerebrovascu- lar accident (CVA). Approx- imately 200,000 annually are affected by lower-extremity hemiparesis.	Three types of behaviors: gravity compensation, feedforward movement assistance during swing, and knee joint stability reinforcement during stance form of emulated spring-damper elements
Shiqian [12]	Empower paraplegics to stand and walk.	Spinal cord injury incidence of SCI lies between 10.4 and 83 per million inhabitants per year worldwide.	force controllable actuators. These allow different control implementations, force control, bio-inspired control, algorithm is presented in this paper, in an attempt to improve stability

Article Name	problem(Need)	Group Beneficiary	Algorithm(controller)
Umit [9]	Wheelchair users experience difficulties when moving on rough surfaces and climbing staircases which limit their social life.	Who lost their walking ability due to disease, trauma, or aging	D fuzzy logic control fig 9. in article, ATHM 800 data ac- quisition card is used in trans- ferring data
Ferris [17]	Medical field.	Ncomplete spinal cord injury undergoing motor retraining with an ankle exoskeleton.	Controlling air pressure linear behavior proportional to the displacement of the plunger the pressure in the pneumatic muscle was controlled in a bangbang mode based on the signal from a foot-switch Proportional myoelectric control Fig.2 in article, Proportional myoelectric control of antagonistic artificial muscles
Parry [6]	Ones in rehabilitation medicine and virtual reality simulation.	Disabled and healthy populations, For patients of stroke and cervical spine injury .	Various control algorithms were therapy, (human proposed, Assistive amplifier), Physiodevice Haptic device, Master device, (position,forceimpedance). To trigger motion in the exoskeleton
Kong [18]	The control is of exoskeletons is challenging because exoskeletons have to interact with another controller in the human body	Humans in battle (Military).	Proportional derivative (PD) controller or a lead-lag compensator) not used, fictitious gain (FG) used in Figure 1,2,3 in article, appropriate joint torque estimator

Table 2.1: Functional specification(Need, Group beneficiary, control) comparison between ten studies

Article Name	Sensor	Actuator	Mechanisms
Adam Zoss [10]	Table 1 in article contain sensors and Force. sensor	Unique Pneumatic actuators. Linear actuators. Hydraulic actuator.	3DoF hip Figure 2 in article, 1DoF knee,Figure 15 in arti- cle, 3Dof ankle Figure 3 in ar- ticle, Shank design Figure 23 in article, Tight mechanisms Figure 24 in article.
Wasnzo [15]	Torque sensor. Encoder,AHRS, Insole sensor. 6axis FT sensor. Figure 7 in article.	BLDC motors	has seven degrees of freedom per leg. Knee, Torase and Ankle modal
Grazi [7]	EMG sensor. Shoes pressure-sensitive.	Torque up to 35 N.m for hip joint. Series elastic actua- tor(sea). Servo DC Motor.	Figure2 in article.
Lei [5]	——————.	Two DC Motors at each leg powering up the joints by us- ing Maxon at Motor .	Harmonic drive mechanism (ratio: 100:1). 1DoF at the hip(active). 1DoF at the knee (active). gure 11 in arti- cle. Stance phase knee ex- ion, knee mechanisms, foot mechanisms.
Spencer [4]	EMG sensor.	Brush-less DC Motors, act- ing through speed reduction transmissions). Torques at each joint approximately 20 Nm .	——————.
Shiqian [12]	Encoder 17bit resolution for feed-bake	A series elastic actuator, which can deliver 100 Nm torque. Table 3 in article, BLDC motor Figure 2,3 inarticle	Table 2 in article, Ball an socket for hip joint. spring pairs with neutral position at 0 ankle angle. spring has a double-spiral disc shape and is made of a single piece of high-grade titanium.

Article Name	Sensor	Actuator	Mechanisms
Umit [9]	40 sensors some of them are embedded in the shoe pads.EMG Sensor. Sensors are placed at foot soles. Limit switches are embedded in the joints in order to prevent excessive joint rotations.	Powered by unique pneumatic actuators. legs driven by hydraulic actuators. The hip and knee joints are powered by electrical actuators. actuators such as pneumatic, hydraulic, electrical, piezoelectric, electro-active polymers.	Waist design Upper and lower leg design Foot, hip and knee design.
Ferris [17]	EMG Sensors	pneumatic muscles Actuators	Hydraulic cylinder Mechanisms
Parry [6]	EMG sensor to take orders from muscles, potentiometers and encoders for feedback, limit switches and emergency for safety.	High power-to-weight motor (1.0Nm,4.2Nm/kg). Mo- tors are rare earth (RE), brushed motors (Maxon Mo- tor, Switzerland). Vary with joint from shoulder to wrist	Cable drive system, two stage pulley reduction, semicircular bearing design to limit movement, joint idler pulley
Kong [18]	EMGSensorusedmuscle ber expansion sensors not used muscle hardness sensors not used incremental encoder56 500 pulses per revolution.	rotary series elastic actuator .	Figure 10 in article.

Table 2.2: Functional specification(Sensors, Actuators, Mechanisms) comparison between ten studies ten studies

Article Name	Weight	Power	Speed
Adam Zoss [10]	Up to 75 kg can carry.	High-powered compact power(1140W),on lower torso .	1.3m/s.
Wasnzo [15]	80kgcandcarry+ 30kg Exo wieght.	Battery on Back bone rear.	1.67 m/s.
Grazi [7]	70kg person weight.	———————	1.33 m/s.
Lei [5]	———————.	battery on back .	———————.
Spencer [4]	———————.	power supply	———————.
Shiqian [12]	30kgExo. weight	———————.	0.8m/s
Umit [9]	———————.	Two 23.5 V 10 Ah Li-Po battery packs.	———————.
Ferris [17]	———————.	Battery on back.	———————.
Parry [6]	Human arm wight 7.5kg.	Power supply.	———————.
Kong [18]	———————.	———————.	———————.

Table 2.3: Functional specication(weight, power, speed) comparison between ten studies

2.4 Schematic Diagram

The Exoskeleton conceptual design is a synergistic integration of mechanical parts, with sensors, actuators and micro controller which are interfacing between humane and machine. The first, mechanical exoskeleton is a wearable machine that allows for user freely movement without resist or difficulty in joints, and carry the load with earthing the reaction force. Second, sensors is used to know the user desired input value and actuator response as feedback and send there for microcontroller. After that, microcontroller is process the sensor signals and control it. The result of microcontroller is send to actuator to move as desired and the actuator give the power need to lifting load and its connected to sensor to get the feedback signal. In Figure 2.1 we show the exoskeleton schematic diagram, separated as input sensor, microcontroller, actuator coupling by feedback sensor, the green wire mean input mode, and the red one mean output, the parallel line refare to coupler.

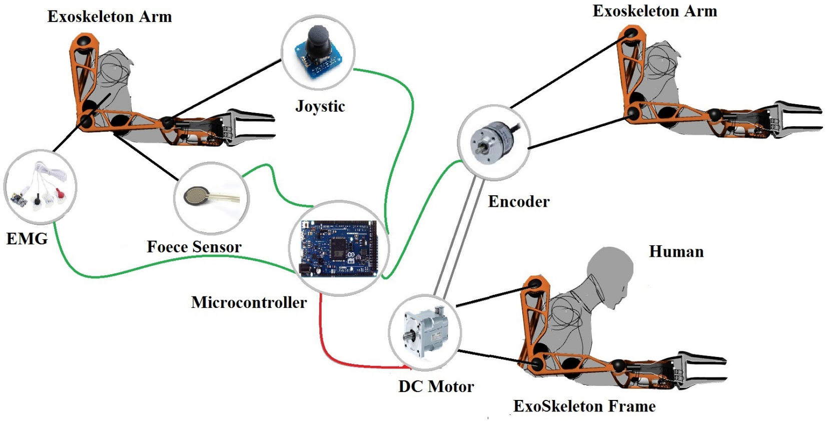


Figure 2.3: Exoskeleton Schematic Diagram

2.5 Intelligent System

Intelligence has been defined in many different ways including ones capacity for logic, abstract thought, understanding, self-awareness, communication, learning, emotional knowledge, memory, planning, creativity and problem solving. It can be more generally described as the ability to perceive information, and retain it as knowledge to be applied towards adaptive behaviors within an environment. project intelligent system using control algorithm as following:

- The main point of intelligent system in this project is to control movement through human intention.

- At first the exoskeleton measure the load wight through force sensor. Then determine if the exoskeleton can carry this load or not.
- If the exoskeleton can carry this load. The exoskeleton sense the human intention through Strain Gage sensor.
- To satisfy the condition of compliant system in exoskeleton. We have make zero torque joint through Torque sensor.
- All of the above done by Arduino microcontroller, which control the exoskeleton motor movement in each joint. The motor movement by four motor, two in elbow joint in both hands, and another two motor in shoulder joint, when the elbow motor move two degree the shoulder motor move one degree.
- To control position by feedbake system is going through Encoders xed in each joint. The schematic diagram explain this procedure.

Chapter 3

First Principle Modular Mathematical Modeling

3.1 Introduction

In first chapter, we generate several designs for exoskeleton and made an morphological chart for these designs, we make an mechanical analysis and mathematical modeling for them, to make sure that the exoskeleton can lift the require load without undesired effect and deflection for load.In section 3.2 we define range of motion for each joints, according to range of motion we draw the position, speed, acceleration and jerk for each joint. Then we generate them equation prelude to determine maximum torque and maximum power in next chapters.

Section 3.3 talked about the mobility and the number of freely join can exoskeleton make.

3.2 Joints Range of Motion (RoM)

It's very important to define range of motion for joint, to verify the position, speed, acceleration and jerk profile. And to take your sense to safety configuration especially in critical point, which may bond cut happen. Another important for range of motion it estimate the actuator location in sense to make sure that it will not cross or resist the motion of another joint.

In this section we will verify the range of motion for shoulder and elbow joint generally, in all application and condition, but we will define the range of motion for Upper-Limb load lifting Exoskeleton, and health and safety configuration that we will take it in our device in each joint.

3.2.1 Shoulder joint

The shoulder and the shoulder girdle make up one of the most complex joint groups of the human body. The hemispherical head of the humerus (upper arm) forms a ball-and-socket- type synovial joint with the glenoid cavity of the scapula. This arrangement allows three degrees of freedom its Flexionextension, Abductionadduction and Rotation. In our project we will concern also on Flexionextension, due to most of load weight lifting by this axis.

Flexionextension. The range of motion of the shoulder in flexion from the anatomical position is between 130 and 180, while in extension (also called hyperextension) the range is between 30 and 80. There are several muscles involved in stabilizing the joint during movement. Shoulder flexion is produced by combined action of the anterior part of the deltoid, the pectoralis major and the coracobrachialis. The posterior part of the deltoid also provides arm extension, the shoulder profile as we see in Figure 3.1.

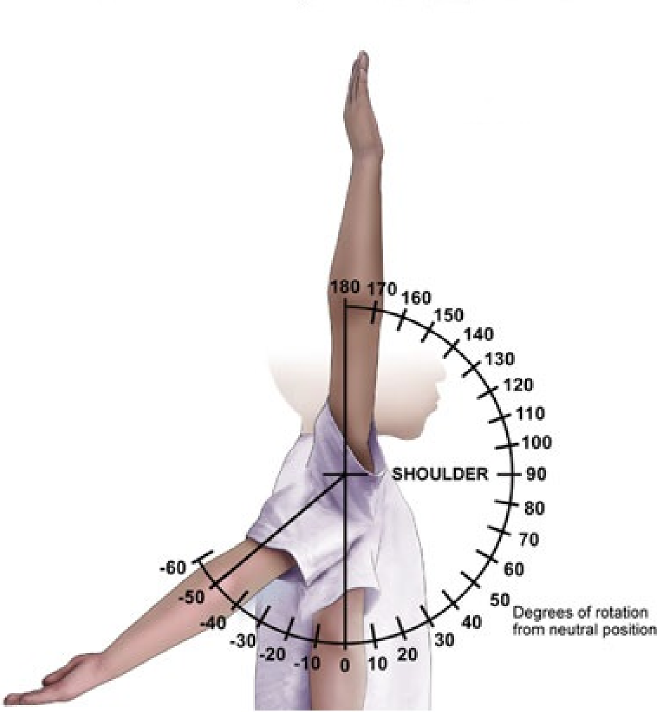
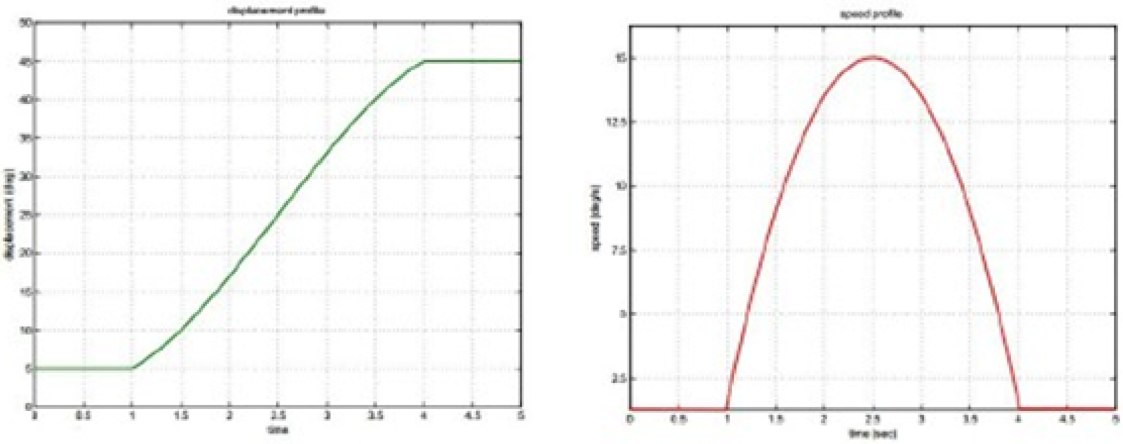


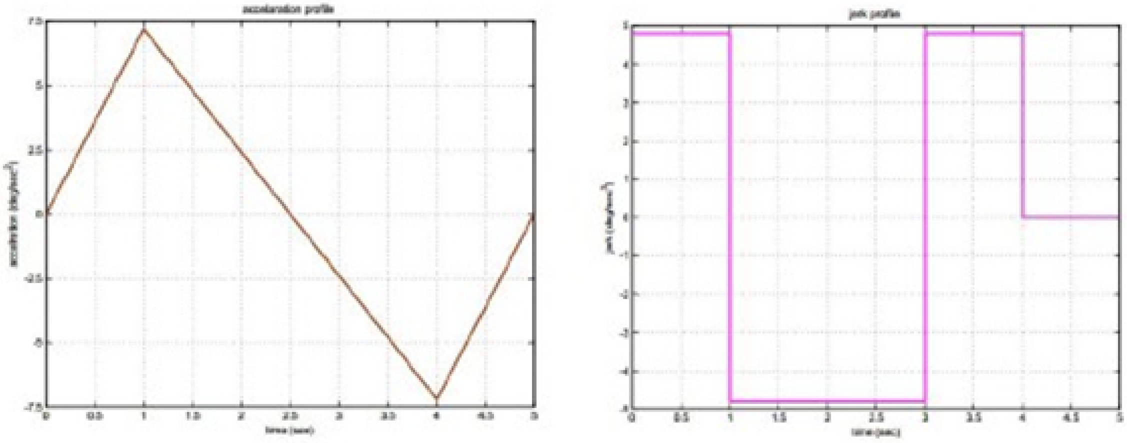
Figure 3.1: Shoulder Joint Range of Motion

In this project we need to lifting loads from 40 degrees from 50 to 90, in 3 second.



(a) position profile

(b) velocity profile



(c) acceleration profile

(d) jerk profile

Figure 3.2: : Position, Velocity, Acceleration and Jerk profile

According to these data we plot position, velocity, acceleration and jerk profile and generate their equations as follow.

$$\theta = 0.81t^3 + 6.1t^2 + 0.26 \tag{3.1}$$

$$\dot{\theta} = 2.43t^2 + 12.2t \tag{3.2}$$

$$\ddot{\theta} = 4.9t + 12.2 \tag{3.3}$$

$$\dddot{\theta} = 4.9 \tag{3.4}$$

3.2.2 Elbow joint

The elbow joint is the one that links the upper arm and lower arm. In generally elbow joint has two degrees of freedom, its Flexion-Extension and Pronation-supination. In our project to lifting a load we need just Flexionextension axis, but we have to not cross and resist another axis Pronationsupination.

Flexionextension. Elbow flexion is the movement whereby the forearm approaches the upper arm. The opposite movement is extension. In this case, the elbow functions as a hinge joint between the distal end of the humerus and the proximal ends of the ulna and radius.

The range of flexionextension motion varies between full extension, 0 and active maximal flexion, 140146. The elbow can be passively fixed up to 160. However, the angle range in day-to-day activities varies between 30 and 130. An important consideration in the design of exoskeletons is the orientation of the axis of rotation of the elbow. The range of motion of elbow as we see in Figure 3.3.

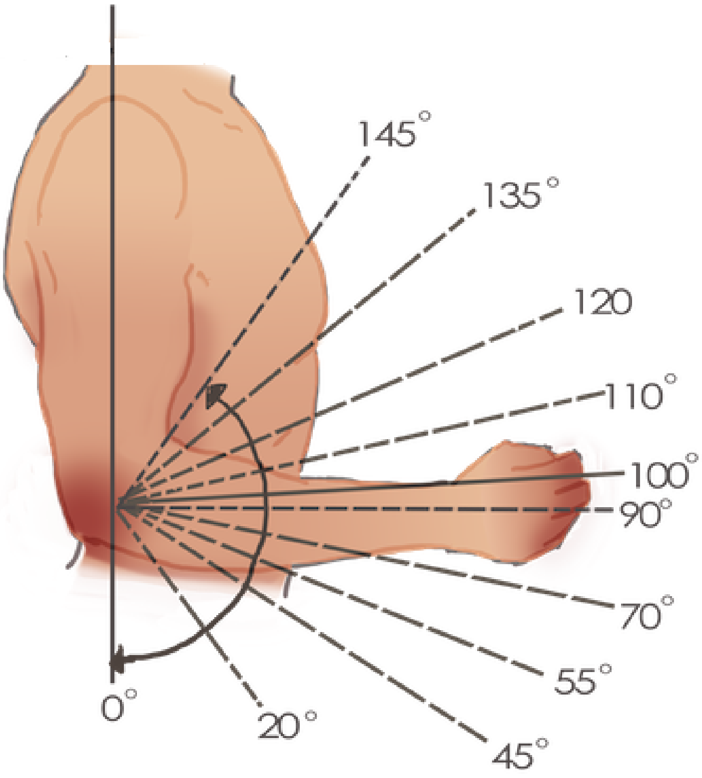
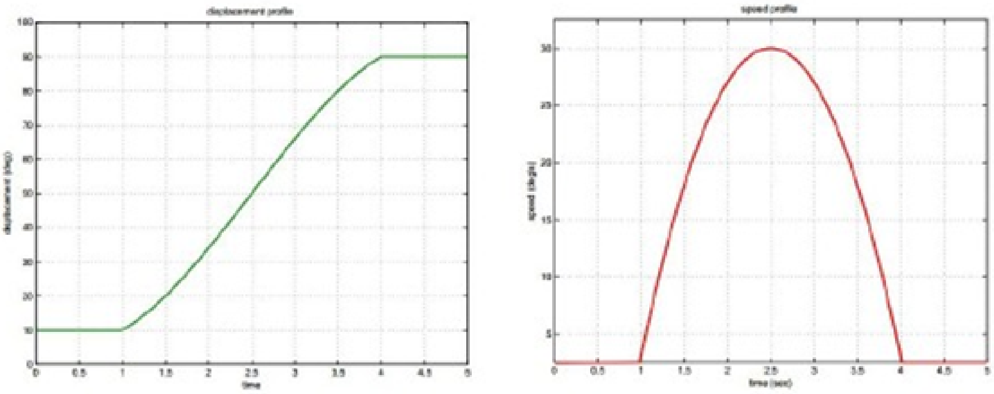


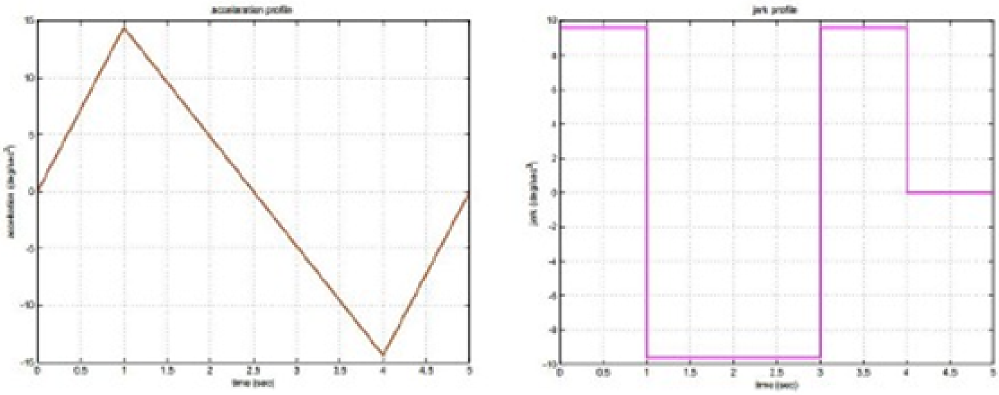
Figure 3.3: Elbow Joint Range of Motion

In our device, which concerned on lifting load we identify range of motion for elbow joint from 0 to 90, but for safety configuration we modify the range of motion to be from 10 to 90 and we need to move this interval just in 3 second. According to this range of motion we plot a position, velocity, acceleration and jerk profile as we see in figure 3.4 . And from matlab we generate the position, velocity, acceleration and jerk equation as follow.



(a) position profile

(b) velocity profile



(c) acceleration profile

(d) jerk profile

Figure 3.4: Elbow Joint Range of Motion

$$\theta = 1.62t^3 + 12.2t^2 + 0.52 \tag{3.5}$$

$$\dot{\theta} = 4.86t^2 + 24.4t \tag{3.6}$$

$$\ddot{\theta} = -9.8t + 24.4 \tag{3.7}$$

$$\dddot{\theta} = 9.6 \tag{3.8}$$

3.3 Degrees of Freedom and Mobility

3.3.1 Degrees of freedom

Degree of freedom (DoF)of a system is a number of independent parameters that are needed to uniquely define position of a rigid body links or systems of bodies at any instant in time. Each link has *six* degree of freedom in space, and *three* in plane,. And when links connected by joints number of DoF will removed. Figure 3.5 explain the number that will remove when links connected by joint.

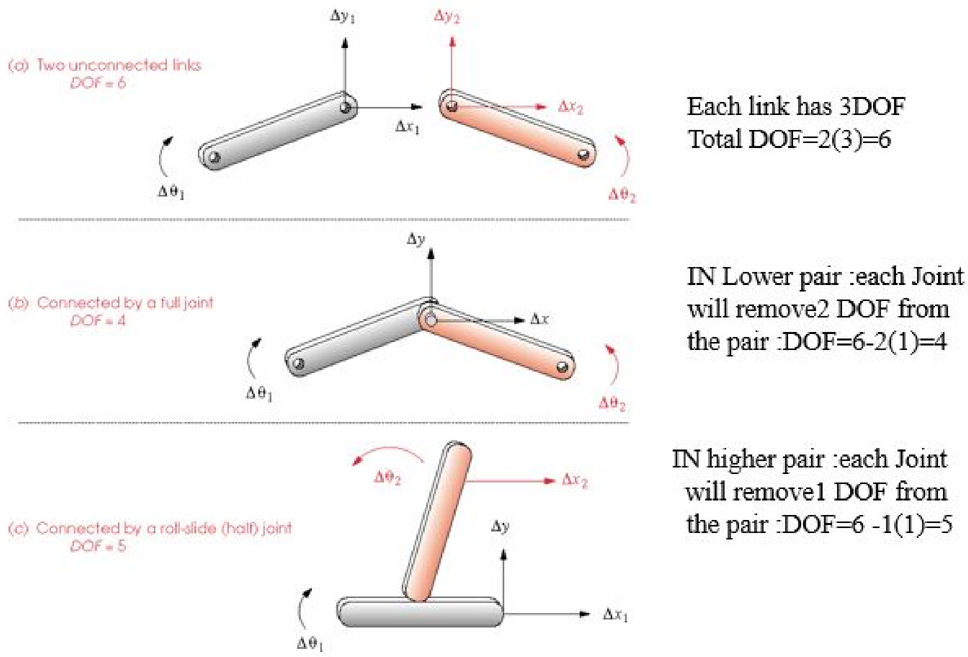


Figure 3.5: Elbow Joint Range of Motion

3.3.2 Mobility

Mobility is the number of motions, which can be controlled independently to bring the device to a particular position configuration.

- $\mathbf{M < 0}$, system has redundant constrains, its over constrained and it is called a statically indeterminate structure , the forces in every link cannot be determined.
- **M=0**, motions are impossible and the system firms as structure.
- **M=1**, mechanisms can driven by a single input for motion.
- **M=2**, mechanism require two independent input for motion, to produce constrain define of motion, and so on.

To determine the Mobility for a planner mechanism, Kutzbach Gr*ü*blers criteria was used.

$$M = 3(n-1) - 2J_2 - J_1 \tag{3.9}$$

Where

M : is Mobility.

n : is number of links.

J_1 : is number of joint that remove 1 Dof.

J_2 : is number of joint that remove 2 Dof.

Figure 3.6, showing the number links in red color, and the number of joints in green color.

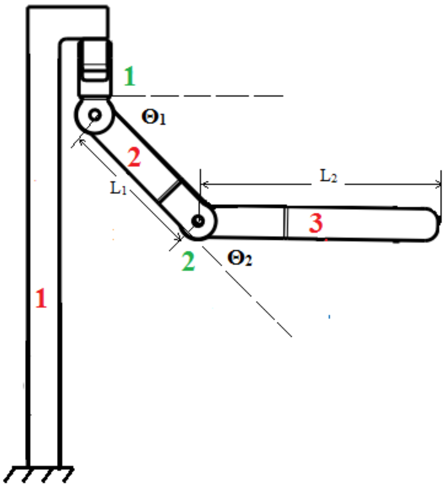


Figure 3.6: Elbow Joint Range of Motion

By applying this numbers on equation 3.5, Kutzbach Gr*ü*blers criteria, the Mobility will be 2, which mean three independent input are require to move the Exoskeleton. one of them are passive, and other two are active.

Chapter 4

Sensor and Actuator Selection

4.1 Introduction

This chapter reviews the sensor, actuators and controller selection, and The technologies and alternative of each one. The interface between human and Exoskeleton and how the signal exchanges in order to drive an actuators, provide feedback for human motor control.

The following section present sensors, actuators, controller and mechanisms we should use in our project, and its alternative, to make a comparison between each and take the best solution that yield the desired requirements.

When we talk about sensor, for Exoskeleton application, we have two type of sensor feedforward sensors such as EMG, Strain Gage and joystick sensors that input signals to microcontroller such as Arduino. And another type of sensor is feedback sensor such as En- coder, Potentiometer, Limit Switch, Force and Torque sensor that give controller feedback signal, to control position , Force and Torque. Section 4.2 discuss this key element in details.

Actuators required at the level of the human joints or limbs, which takes respond to signals from controller after processing. The actuator must match with position accuracy, force, torque, speed and size requirement. The review of actuator selection in Section 4.3 focuses on principles, requirement, alternative and limitations.

To transmit power as functional as we can, we should use some mechanisms to yield the desire movement. In section 4.4 we will discuss these issues in detail. In section 4.5 we consider the element that we will use in our project and its cost.

4.2 Sensors

As we said previously we need sensor to transmit order, knowing the machine location by feedback sensor, and for some human machine interfaces, here we will present some sensor, which we will use in project

4.2.1 Force Sensor

Force sensor: A force-sensing resistor is polymer, which changes resistance in a predictable manner following application of force to its surface. It is normally supplied as a polymer sheet, which has had the sensing film applied by screen-printing. The sensing film consists of both electrically conducting and non-conducting particles suspended in matrix. The par- ticle sizes are of the order of fraction of microns, and are formulated to reduce the tempera- ture dependence, improve mechanical properties and increase surface durability. Applying a force to the surface of the sensing film causes particles to touch the conducting electrodes, changing the resistance of the film.

Force sensors can measure force between almost any two surfaces and are durable enough to stand up to most environments. Our sensors are available off-the-shelf for prototyping or can be customized to meet the specific needs of your product design and application requirements. So that we may know whether the machine holds a mass or not. value.

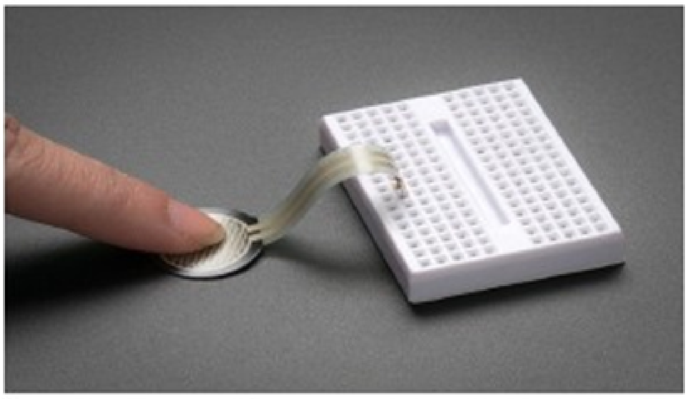


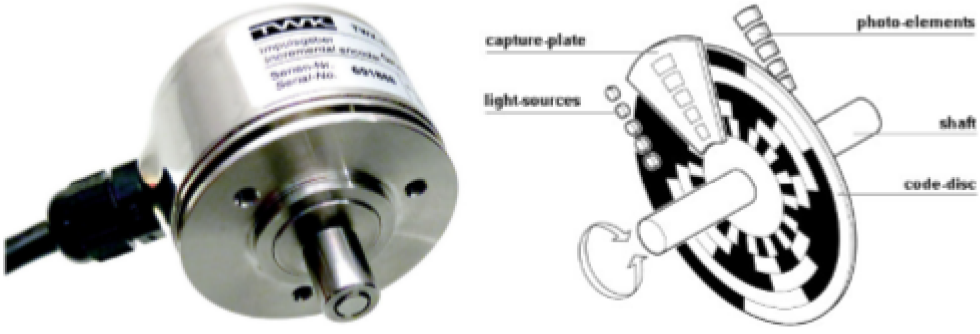
Figure 4.1: Force sensor

4.2.2 Rotary Encoder

A rotary encoder, also called a shaft encoder, is an electro-mechanical device that converts the angular position or motion of a shaft or axle to an analog or digital code.

There are two main types: absolute and incremental (relative). The output of absolute encoders indicates the current position of the shaft, making them angle transducers. The output of incremental encoders provides information about the motion of the shaft, which is typically further processed elsewhere into information such as speed, distance and position.

This sensor is used to measure the feedback from motor in order to know the value of the movement of the motor and to observe the movement of the motor and find out the angle at which it is moving the motor to ensure the correct action. And keep safe the work of the engine and true.



(a) Rotary Encoder

(b) schismatic diagram

Figure 4.2: Rotary Encoder sensor

4.2.3 Limit Switch

Limit switch: In order to increase the safety limit switch used so that the electric sensor, if the associated motor joint approach to the end of the movement needed point gives this sensor signal to prevent the movement of motor in the same direction.

Figure 4.3: Limmit Switch

4.3 Controller

Arduino is open-source hardware, so you can easy-to-use programming (C++). We needs high-speed transfer of information and response motor so it is best controller: Arduino Due is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM core microcontroller. It has 54 digital input/output pins (of which 12 can be used as PWM outputs), 12 analog inputs, 4 UARTs (hardware serial ports), a 84 MHz clock, an USB OTG capable connection, 2 DAC (digital to analog), 2 TWI, a power jack, an SPI header, a JTAG header, a reset button and an erase button.

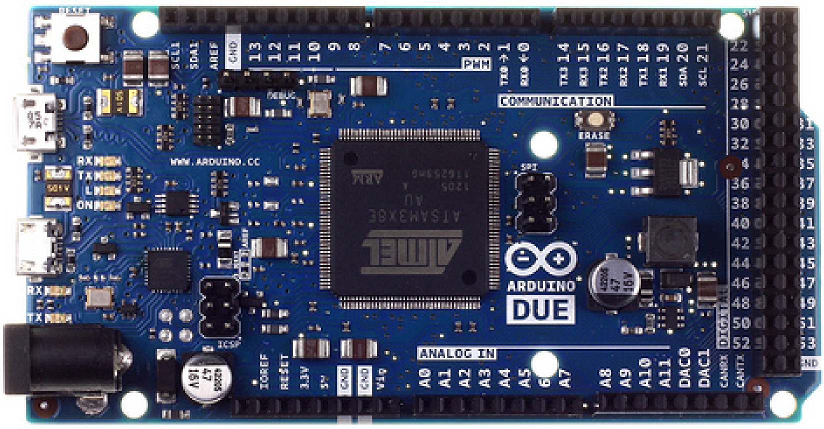


Figure 4.4: Arduino

4.4 Actuators

In Exoskeleton the most important component is the actuators, we need to lift high weight equal 50 Kg, and by 0.6 $m$the torque will equal 300 Nm in both hands, so in one hand the torque will be half of this torque on shoulder joint and quarter of this number in elbow joint.

This type needed motor with very high torque that is not true, so we need gear like 1:100 and motor high speed. This kind expensive compared with others, but advantages servomotor is accuracy in micrometer. In the other type, you can use the hydraulic pump so that gives high torque and do not require high voltage. However, the disadvantages it needs to pump may increase the weight of the device or that use of the device in limited area movement because it is connected with a xed pump if the torque is needed high pump, and not the free movement such as motors, And you can use linear motor.

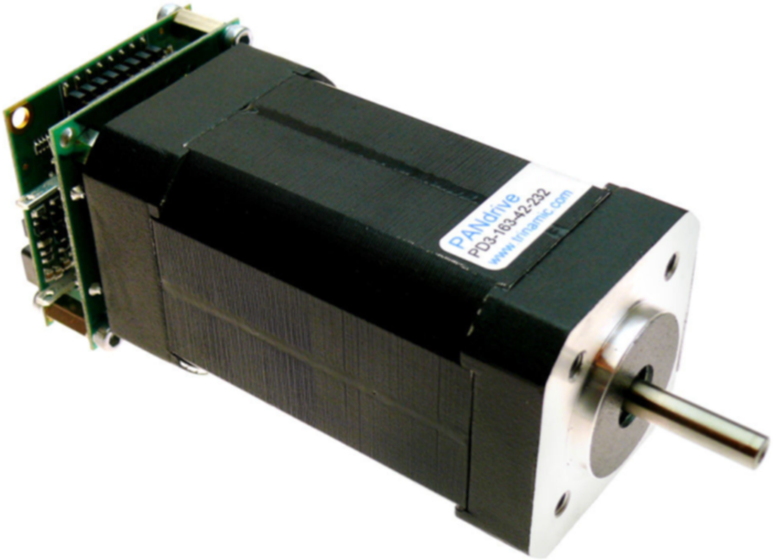


Figure 4.5: Actuators

4.5 Mechanisms

In this project we will work on two main joint, shoulder and elbow, because most of load is concentrated on them. The mechanisms related with these joint as follow.

4.5.1 Shoulder Joint

The glenoid cavity is shallow and contains the glenoid labrum which deepens it and aids in stability. With 120 degrees of unassisted fexion, the shoulder joint is the most mobile joint in the body. Possible Movements of shoulder

1. Abduction = moving away from the body, push outside.
2. Adduction = movement towards the body.
3. Flexion = the body parts bend.
4. Extension = the body extends, stretches.
5. Circumduction = turning around.
6. Rotation = rotation around an ax.

.

To achieve these movements we need a mechanisms work in 3 DOF

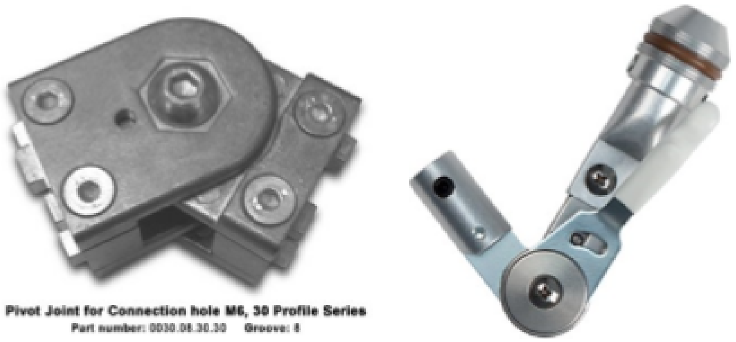
1. Rotational joints in transversal plane.
2. Rotational joints in sagittal plane.
3. Rotational joints in vertical plane (coronal plane).
4. Safety mechanism for shoulder.

Figure 4.6: shoulder joint

4.5.2 Elbow Joint

The function of the elbow joint is to extend and flex the arm grasp and reach for objects. The range of movement in the elbow is from 0 degrees of elbow extension to 150 of elbow flexion. Muscles contributing to function are all flexion (biceps brachii, brachialis, and brachioradialis) and extensor muscles (triceps and anconeus). In humans, the main task of the elbow is to properly place the hand in space by shortening and lengthening the upper limb. While the superior radioulnar joint shares joint capsule with the elbow joint, it plays no functional role at the elbow. With the elbow extended, the long axis of the humerus and that of the ulna coincide.

At the same time, the articular surfaces on both bones are located in front of those axes and deviate from them at an angle of 45. Additionally, the forearm muscles that originate at the elbow are grouped at the sides of the joint in order not to interfere with its movement. The wide angle of exion at the elbow made possible by this arrangement almost 180 allows the bones to be brought almost in parallel to each other. The mechanism used for elbow is rotational joints in vertical plane. And safety mechanism.



(a) Elbwo Joint (b) Elbwo Joint safty

Figure 4.7: Elbow joint

4.6 Exoskeleton Budget

When you going to make any project, you have to take attention that your project will cost a certain amount of many. If you plane to make a deployment for your project, you must take attention more and more, to minimize the cost as possible as you can, to make a beautiful Profit margin. The Components and prices of the project as we see in Table 4.1.

No.	Componant	Price
4	Ac Servomotors	$2000
1	Arduino due	$150
4	Gear Box	$400
1	Power Supply	$50
4	Force Sensor	$50
——	mechanical parts and mechanisms	$500
——	Weirs and electronic parts (resistor, capacitor,..)	$200
——	Machining	$1000
	Total	$4250

Table 4.1: Budget

Chapter 5

Mathematical Modeling

5.1 Introduction

To understand the complexity of robots and their applications, that require knowledge of Electrical engineering, Mechanical engineering, Systems and Industrial Engineering, Computer science, Economics, and Mathematics. The integration of these field causes complete analysis and control of robots.

In this chapter we will consider the forward kinematics for the Exoskeleton end-effector of two hands. The problem of kinematics is to describe the motion of the Exoskeleton hands without consideration of the forces and torques causing the motion. The kinematic description is therefore a geometric one. We first consider the problem of forward kinematics, which is to determine the position and orientation of the end-effector given the values for the joint variables of the robot all of this will be in section 7.2. Then we will describe the mathematical model. Mathematical model describes the motion with forces consecration and torques, and we will generate the exoskeleton control matrices. All of this will be in section 7.3.

5.2 Kinematics

Kinematics is a part of classical mechanics, which describes the motion of points, bodies (objects), and systems of bodies without consideration of the masses of those objects,the forces and torques causing the motion. Kinematics is often referred to ”geometry of motion” and as such may be seen as a branch of mathematics. Kinematics begins with a description of the geometry of the system and the initial conditions of known values of the position, velocity and or acceleration of various points that are a part of the system, then from geometrical arguments it can determine the position, the velocity and the acceleration of any part of the system. The study of the influence of forces acting on masses falls within the purview of kinetics. For further details, see analytical dynamics.

As we know the Kinematics separate to two part forward kinematics and inverse kinematics. We rst consider the forward kinematics, which determine the position and orientation of the end-eector given the values for the joint variables. The inverse kinematics to determine the values of the joint variables given the end-erector position and orientation.

- Forward Kinematics:the position(x,y)of end-effector as a function of joint angles(θ_1, θ_2).
- Inverse Kinematics:angles($\theta_1\, theta_2$)as a functions of(x,y).

There is a commonly used convention transformation for selecting frames in robotic applications is the Denavit-Hartenberg, or DH convention. each homogeneous transformation is represented as a product of four basic transformation.

$$A_i = Rot_z, \theta_i Trans_z, d_i Trans_x, a_i Rot_x, \theta_i \tag{5.1}$$

$$A_i = \begin{bmatrix} cos\theta_i & -sin\theta_i cos\alpha_i & sin\theta_i sine\alpha_i & a_i cos\theta_i \\ sin\theta_i & cos\theta_i cos\alpha_i & -cos\theta_i sin\alpha_i & a_i sin\theta_i \\ 0 & sin\alpha_i & cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.2}$$

In this project the problem is find the position of end-effector not to find the orientation angle form the position so we will constrain on forward kinematics and DH convention in our work, which passes through these steps.

5.2.1 Forward Kinematics

- **Exoskeleton Frame Assignment**

  the first step in forward kinematics is to detect the frame assignment for your device. the first step is to chose the base of Exoskeleton and other joint($j_1, j_2, ...j_n$), then chose the z-axis for each joint($z_0 - z_i$), and relative to z-axis we find (x, y). As we said before we have two joint and 2DoF in our Exoskeleton. Elbow and shoulder, the first both of them are active, so we will control the active joint and make the forward kinematics for them. In figure 5.1 we show only active joint, and other DH coordinate α, a, θ.

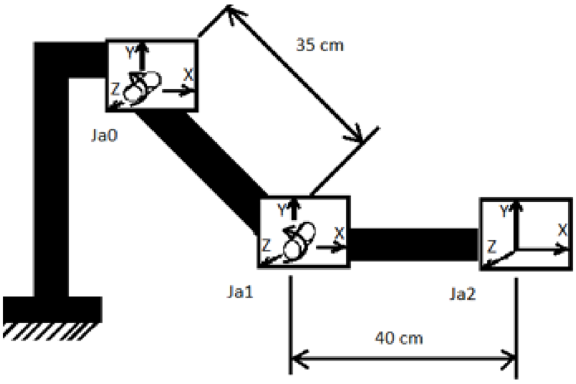


Figure 5.1: Exoskeleton active joint frame assignment

- **HD Convention Parameters**

Consider the two-link planar arm of Figure 5.2, which somehow its problem like an upper-limb 2DoF Exoskeleton. The joint *axis* z_0 and z_1 are normal to the page. We establish the base frame o_0 x_0 y_0 z_0 as shown. The origin is chosen at the point of intersection of the *axis* z_0 with the page and the direction of the *axis* x_0 is completely arbitrary. Once the base frame is established, theo_1 x_1 y_1 z_1 frame is fixed as shown by the DH convention, where the origin o_1 has been located at the intersection ofz_1 and the page. The final frame o_2 x_2 y_2 z_2 is fixed by choosing the origin o_2 at the end of link 2 as shown.

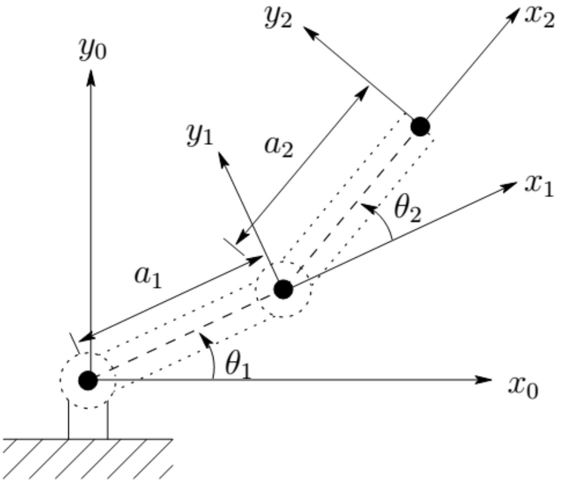


Figure 5.2: Exoskeleton active joint frame assignment

The HD-convention parameters are shown in table 5.1.

Joint Name	Joint No.	a_i	α_i	d_i	θ_i
Shoulder joint	0	a_1	0	0	θ_1
Elbow joint	2	a_2	0	0	θ_2

Table 5.1: Parameters

- **Transformation matrices**

The transformation A-matrices are determined as follow

$$A_1 = \begin{bmatrix} cos\theta_1 & -sin\theta_1 & 0 & a_1 cos\theta_1 \\ sin\theta_1 & cos\theta_1 & 0 & a_1 sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.3}$$

$$A_2 = \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 & a_2 cos\theta_2 \\ sin\theta_2 & cos\theta_2 & 0 & a_2 sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.4}$$

The T-matrices are thus given by

$$T_0^2 = A_1.A_2 \tag{5.5}$$

$$T_0^2 = \begin{bmatrix} (cos(\theta_1+\theta_2) & -sin(\theta_1+\theta_2) & 0 & a_1 cos\theta_1 + a_2 cos(\theta_1+\theta_2) \\ sin(\theta_1+\theta_2) & cos(\theta_1+\theta_2) & 0 & a_1 sin\theta_1 + a_2 sin(\theta_1+\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.6}$$

Notice that the first two entries of the last column of T_0^2 are the x and y components of the origin o_2 in the base frame; that is.

$$x = a_1 cos\theta_1 + a_2 cos(\theta_1+\theta_2) \tag{5.7}$$

$$y = a_1 sin\theta_1 + a_2 sin(\theta_1+\theta_2) \tag{5.8}$$

Are the coordinates of the end-effecter in the base frame. The rotational part ofT_0^2 gives the orientation of the frame $o_2\, x_2\, y_2\, z_2$ relative to the base frame.

5.2.2 Inverse Kinematics

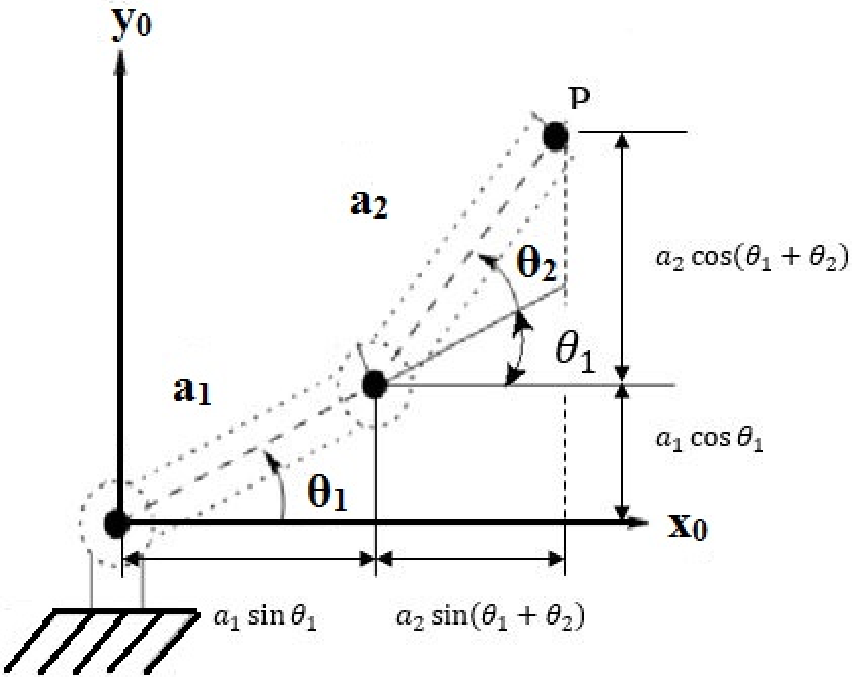


Figure 5.3: Exoskeleton active joint frame assignment

$$P_x = a_1 cos\theta_1 + a_2 cos(\theta_1 + \theta_2) \tag{5.9}$$

$$P_y = a_2 sin\theta_1 + a_2 sin(\theta_1 + \theta_2) \tag{5.10}$$

The solution of 2 can be computed from summation of squaring both equations 5.12 and 5.13.

$$P_x^2 = a_1^2 cos^2\theta_1 + a_2^2 cos^2(\theta_1 + \theta_2) + 2a_1 a_2 cos\theta_1 cos(\theta_1 + \theta_2) \tag{5.11}$$

$$P_y^2 = a_1^2 sin^2\theta_1 + a_2^2 sin^2(\theta_1 + \theta_2) + 2a_1 a_2 sin\theta_1 sin(\theta_1 + \theta_2) \tag{5.12}$$

$$P_x^2 + P_y^2 = a_1^2 + a_2^2 + 2a_1 a_2 cos\theta_2 \tag{5.13}$$

And so

$$cos\theta_2 = \frac{P_x^2 + P_y^2 + a_1^2 + a_2^2}{2a_1 a_2} \tag{5.14}$$

Since, $cos^2\theta_i + sin^2\theta_i = 1(i = 1,2,3,), sin\theta_2$ is obtained as.

$$sin\theta_2 = \pm\sqrt{1 - \left(\frac{P_x^2 + P_y^2 + a_1^2 + a_2^2}{2a_1a_2}\right)^2} \tag{5.15}$$

Finally, two possible solutions for θ_2 can be written as.

$$\theta_2 = \arctan 2\left(\pm\sqrt{1 - \left(\frac{P_x^2 + P_y^2 + a_1^2 + a_2^2}{2a_1a_2}\right)^2}, \quad \frac{P_x^2 + P_y^2 + a_1^2 + a_2^2}{2a_1a_2}\right) \tag{5.16}$$

Now to fine θ_1

$$cos\theta_1 P_x = a_1 cos^2\theta_1 + a_2 cos^2\theta_1 cos\theta_2 - a_2 cos\theta_1 sin\theta_1 sin\theta_2 \tag{5.17}$$

$$sin\theta_1 P_y = a_1 sin^2\theta_1 + a_2 sin^2\theta_1 cos\theta_2 + a_2 sin\theta_1 cos\theta_1 sin\theta_2 \tag{5.18}$$

$$cos\theta_1 P_x + sin\theta_1 P_y = a_1 + a_2 cos\theta_2 \tag{5.19}$$

$$-sin\theta_1 P_x + cos\theta_1 P_y = a_2 sin\theta_2 \tag{5.20}$$

$$cos\theta_1 P_x^2 + sin\theta_1 P_x P_y = P_x(a_1 + a_2 cos\theta_2) \tag{5.21}$$

$$-sin\theta_1 P_x P_y + cos\theta_1 P_y^2 = P_y a_2 sin\theta_2 \tag{5.22}$$

$$cos\theta_1(P_x^2 + P_y^2) = P_x(a_1 + a_2 cos\theta_2) + P_y a_2 sin\theta_2 \tag{5.23}$$

$$cos\theta_1 = \frac{P_x(a_1 + a_2 cos\theta_2) + P_y a_2 sin\theta_2}{P_x^2 + P_y^2} \tag{5.24}$$

$$sin\theta_1 = \pm\sqrt{1 - \left(\frac{P_x(a_1 + a_2 cos\theta_2) + P_y a_2 sin\theta_2}{P_x^2 + P_y^2}\right)^2} \tag{5.25}$$

As a result, two possible solutions for 1 can be written.

$$\theta_1 = \arctan 2\left(\pm\sqrt{1 - \left(\frac{P_x(a_1 + a_2 cos\theta_2 + P_y a_2 sin\theta_2}{P_x^2 + P_y^2}\right)^2}, \quad \frac{P_x(a_1 + a_2 cos\theta_2 + P_y a_2 sin\theta_2}{P_x^2 + P_y^2}\right) \tag{5.26}$$

Although the planar manipulator has a very simple structure, as can be seen, its inverse kinematics solution based on geometric approach is very cumber some.

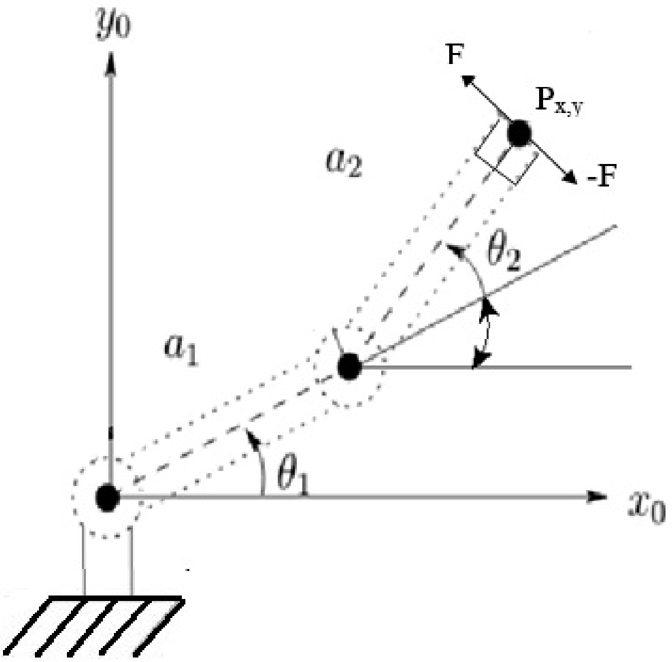


Figure 5.4: Exoskeleton Frame Assignment

From δF we can find the change in position value of $P_{x,y}$. So we use inverse kinematics to get θ_1 & θ_2. And use inverse kinematics to avoid joint move over limit movement and go to singularities state .

5.3 Trajectory planning for Exoskeleton

A path on which timing law is specified, for example velocities and accelerations in its each point is called trajectory planning is use to get approximate response to desired path. For exoskeleton arm used interpolation Linear functions with parabolic blends because it the simplicity method and it give smooth path.

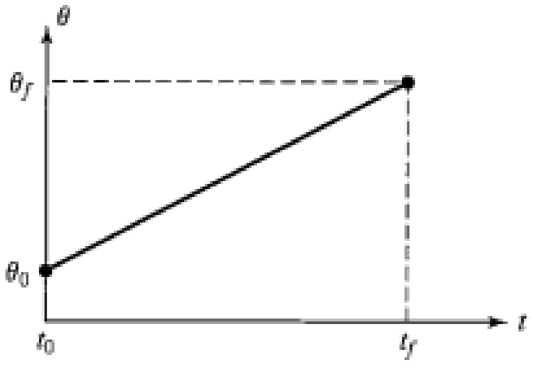


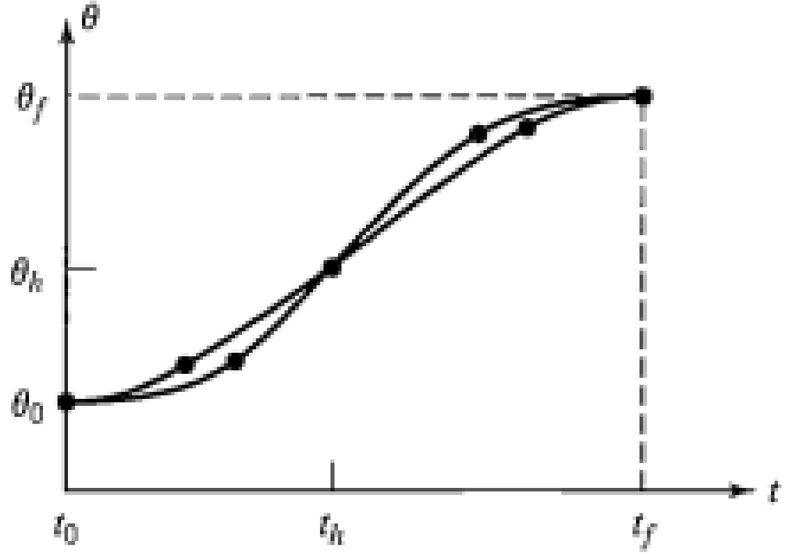
Figure 5.5: Exoskeleton path

To construct this single segment, we will assume that the parabolic blends both have the same duration; therefore, the same constant acceleration is used during both blends. As indicated in last figure, there are many solutions to the problem, but note that the answer is always symmetric about the halfway point in time, t_h, and about the halfway point θ_h. in position, the velocity at the end of the blend region must equal the velocity of the linear section, and so we have this equation.

$$\ddot{\theta}\; t_b = \frac{\theta_h \;-\; theta_b}{t_h - t_b} \tag{5.27}$$

where θ_b is the value of θ at the end of the blend region, and is the acceleration acting during the blend region. The value of θ_b is given by.

$$\theta_b = \theta_0 + \frac{1}{2}\ddot{\theta} t_b^2 \tag{5.28}$$



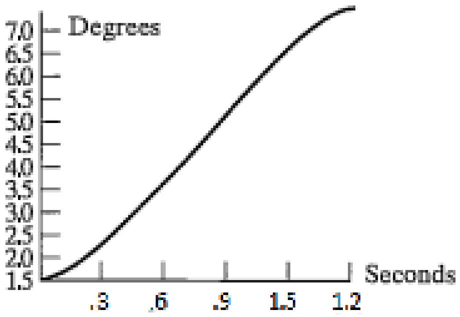
Now by combining equation (5.27) and (5.28) and $t = 2_t h$, we get.

$$\ddot{\theta} t_b^2 - \ddot{\theta} t t_b + (\theta - \theta 0) = 0 \tag{5.29}$$

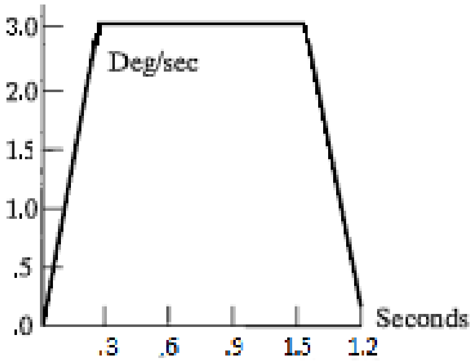
where t is the desired duration of the motion. Given any θ_f, θ_0, and t, we can follow any of the paths given by the choices of $\ddot{\theta}$ and t_b that satisfy in equation(5.29). Usually, an acceleration, $\ddot{\theta}$, is chosen, and equation (5.29) is solved for the corresponding t_b. The acceleration chosen must be sufficiently high, or a solution will not exist. Solving equation(5.29) for in terms of the acceleration and other known parameters, we obtain.

$$t_b = \frac{t}{2} - \frac{\sqrt{\ddot{\theta}^2 t^2 - 4\ddot{\theta}(\theta_f - \theta_0)}}{2\ddot{\theta}} \tag{5.30}$$

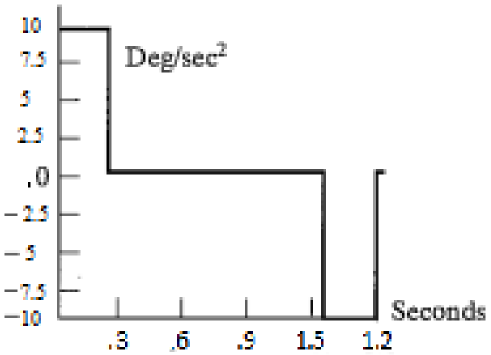
In Exoskeleton robot arm used high θ because the speed so slow 3 deg/sec and the value of delta theta from force sensor is 1.7 degree at max. press every 1 *sec* so the effect of trajectory planning is small and the distance not exceed 7 degree as shown in next figures .



(a) Position



(b) Velocity



(c) Acceleration

5.4 Dynamics

Design of control systems based on a Dynamics mathematical model of the system is to control that system. In traditional control the model is usually presented as a transfer function, a transfer function usually used for linear, time invariant system, and it a very usful tool to design control system. But its obvious that the transfer function has a limitations to model just *SISO-systems*. So to model systems like *MIMO-system* we have to sue state space model.

As we know Dynamics mathematical model through state space model passes through several steps. At firs making free body diagram and find deferential equation (DE) and generate the state space matrices. The second step is linearizing the state space model if that require. Then design the controller and simulate it. And the second step if the simulation work, apply the controller on

Exoskeleton system. And we will go in these steps in details as follow.

5.4.1 Free body diagram and equations of motion

1. Free body Diagram

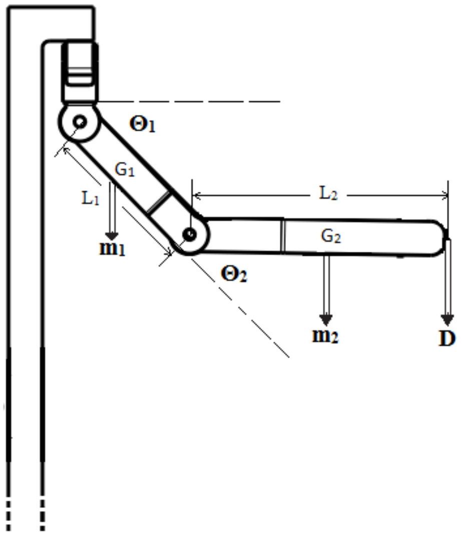


Figure 5.7: Exoskeleton Frame Assignment

2. Equation of motion. To formulate the equation of motion for exoskeleton arm, the Lagrange method can be used.The Lagrange method is defined as the difference between the kinetic and potential energies

$$L = K - P.$$

K and P are the total kinetic and potential energies of the exoskeleton system.

- Kinetic Energy

$$K = \frac{1}{2}\, m_1 v_{G1}^2 \;+\; \frac{1}{2}\, J_1 \dot{\theta}_1^2 \;+\; \frac{1}{2}\, m_2 v_{G2}^2 \;+\; \frac{1}{2}\, J_2(\dot{\theta}_1 + \dot{\theta}_2)^2 \tag{5.31}$$

The coordination of the two centers of gravity of two segment of the arm are.

$$(x_{G1}\,,\, y_{G1}) \;=\; \frac{l_1}{2} cos\theta_1 \,,\, \frac{l_1}{2} sin\theta_1 \tag{5.32}$$

$$(x_{G2}\,,\, y_{G2}) = \; l_1 cos\theta_1 \;+\; \frac{l_2}{2} cos(\theta_1 +\; \theta_2) \,,\, l_1 sin\theta_1 \;+\; \frac{l_2}{2} sin(\theta_1 +\; \theta_2) \tag{5.33}$$

And the velocities will be the differentiation of the positions. After that the the total kinematic energy will be.

$$K = \Big(\frac{1}{2}\,m_1\frac{l_1^2}{4} + \frac{1}{2}\,J_1 + \frac{1}{2}\,m_2 l_1^2\Big)\,\dot{\theta}_1^2 \;+\; \Big(\frac{1}{2}\,m_2\frac{l_2^2}{4} \;+\; \frac{1}{2}\,J_2\Big)\;(\dot{\theta}_1 + \dot{\theta}_2)^2$$

$$+\frac{1}{2}m_2 l_1 l_2 cos(\theta_2)\dot{\theta}_1(\dot{\theta}_1 + \dot{\theta}_2) \tag{5.34}$$

- Potential Energy

$$P = m_1 g\frac{l_1}{2}sin\theta_1 + m_2 g(l_1 sin\theta_1 + \frac{l_2}{2}sin(\theta_1 + \theta_2) \tag{5.35}$$

The torque in shoulder and elbow motor are define in Lagrange as.

$$\frac{d}{dt}(\frac{\partial L}{\partial\dot{\theta}_1}) - \frac{\partial L}{\partial\theta_1} = \tau_1$$

$$\frac{d}{dt}(\frac{\partial L}{\partial\dot{\theta}_2}) - \frac{\partial L}{\partial\theta_1} = \tau_2 \tag{5.36}$$

After some straightforward calculation of the previous equation, its arrives to the following equations of motion.

$$M_1\ddot{\theta}_1 + M_2\ddot{\theta}_2 + K_1 + G_1 + D_1 = \tau_1$$

$$M_3\ddot{\theta}_1 + M_4\ddot{\theta}_2 + K_2 + G_2 + D_2 = \tau_2 \tag{5.37}$$

Where

$M_1 = (\frac{1}{4}m1 + m2)l_1 + \frac{1}{4}m2l_2 + J_1 + J_2 + m_2 l_1 l_2 cos(\theta 2)$

$M_2 = \frac{1}{4}m2l_2 + J_2 + \frac{1}{2}m2l_1 * l_2 * cos(\theta_2))$

$M_3 = M_2$

$M_4 = \frac{1}{4}m2l_2 + J_2)$

The centrifugal and coriolis force are.
$K_1 = -\frac{1}{2} m_2 l_2 l_1 sin(\theta_2).\dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2)$
$K_2 = \frac{1}{2} m_2 l_2 l_1 sin(\theta_2).\dot{\theta}_1^2$
The terms due to gravity are.
$G_1 = m_1 g \frac{l_1}{2} cos(\theta_1) + (m_2 g(l_1 * cos(\theta 1) + \frac{l_2}{2} * cos(\theta_1 + \theta_2)$
$G_2 = m_2 g(\frac{l_2}{2} * cos(\theta_1 + \theta_2)$

The terms of disturbance load are.
$D_1 = Dg(l_1 cos(\theta_1) + l_2 cos(\theta_1 + \theta_2)$
$D_2 = Dgl_2 * cos(\theta_1 + \theta_2)$

5.4.2 State space variable

As we know we have to convert the second order differential equation in last section, to several first order differential equation in this form.

$$\dot{x} = Ax + Bu + Dd \tag{5.38}$$

where D is disturbance matrix

$$y = Cx + Du \tag{5.39}$$

Now we have to chose a sufficient number of state space variable to describe the system. We have two second order differential equation, so when we go to convert these two equation to several first order equations, we need four state space variable, its θ_1, θ_2, $\dot{\theta}_1$ and $\dot{\theta}_2$. According to the equation 5.14, and With $H = M^{-}1$ the non-linear state equation as following.

$$x_1 = \theta_1 \qquad \dot{x}_1 = \dot{\theta}_1 = x_2 \tag{5.40}$$

$$x_3 = \dot{\theta}_1 \qquad \dot{x}_3 = \ddot{\theta}_1 = -H_1(K_1 + G_1)(\theta_1) - H_2(K_2 + G_2)(\theta_2) + H_1\tau_1 + H_2\tau_2 + D_1 \tag{5.41}$$

$$x_3 = \theta_2 \qquad \dot{x}_2 = \dot{\theta}_2 = x_4 \tag{5.42}$$

$$x_4 = \dot{\theta}_2 \qquad \dot{x}_4 = \ddot{\theta}_2 = -H_3(K_1 + G_1)(\theta_1) - H_4(K_2 + G_2)(\theta_2) + H_3\tau_1 + H_4\tau_2 + D_2. \tag{5.43}$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -H_1(K_1+G_1) & 0 & -H_2(K_2+G_2) & 0 \\ 0 & 0 & 0 & 1 \\ -H_3(K_1+G_1) & 0 & -H_4(K_2+G_2) & 0 \end{bmatrix} x \quad + \quad \begin{bmatrix} 0 & 0 \\ H_1 & H_2 \\ 0 & 0 \\ H_3 & H_4 \end{bmatrix} u \quad + \quad \begin{bmatrix} 0 \\ D_1 \\ 0 \\ D_2 \end{bmatrix} d \tag{5.44}$$

5.4.3 Linearization

As we see the state space model is non-linear, we have to linearize them, by make a partial differentiation on the require states. The most danger angles of lifting loads by exoskeleton is angle $[\,0,\,0\,]$, because the load will be perpendicular on joints. In the scale of figure 3.1 and 3.3 in chapter three the angles were presented as 45 for shoulder and 0 foe elbow joint. So we made the partial differentiation on them.

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{\partial}{\partial \theta_1}(H_1+G_1) & 0 & -\frac{\partial}{\partial \theta_2}(H_2+G_2) & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{\partial}{\partial \theta_1}(H_3+G_1) & 0 & -\frac{\partial}{\partial \theta_2}(H_4+G_2) & 0 \end{bmatrix} x \quad + \quad \begin{bmatrix} 0 & 0 \\ H_1 & H_2 \\ 0 & 0 \\ H_3 & H_4 \end{bmatrix} u \tag{5.45}$$

The coriolis and centrifugal forces are all quadratic in the angular velocity, so that for any linearization around a stationary point,that mean $\theta_1 0[= \theta_2 0 = 0$.

Parameters :

$m_1 = 0.573 kg,\ l_1 = 23.5 cm,\ J_1 = 0.004 kg.m^2,$

$m_2 = 0.441 kg,\ l_2 = 37.5 cm,\ J_2 = 0.006 kg.m^2.$

Now the final state space is become.

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 5.8 & 0 & -0.79 & 0 \\ 0 & 0 & 0 & 1 \\ -8.7 & 0 & 1.2 & 0 \end{bmatrix} x \quad + \quad \begin{bmatrix} 0 & 0 \\ 0.31 & -0.69 \\ 0 & 0 \\ -0.69 & 1.78 \end{bmatrix} u \tag{5.46}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x \tag{5.47}$$

After we implement these equation we process them in MatLab to test the controllability and observability for the Exoskeleton system, after we generate the controllability and observability matrices we find that the rank of this matrices is fore(Rc=Ro=0). So our System is controllable and observable.

5.5 Control System Design

After we defined the Exoskeleton System, and made a mathematical modeling for it, and estimate the state space modeling and implement its matrices. The next stage its to design a suitable control for this system, which must satisfy the project requirement. These requirement state.

1. Gravity Compensator.
2. Compliancy.
3. Force control.
4. Force Loop.

Before start with design the control system we should to know that any mechatronics system must include at least four components its.

1. Planet (System)
2. The actuators or drivers
3. The sensors which measure system input and output
4. The controller which takes input from system sensors, process it and gives output to the system actuators.

Figure 5.4 shoes this four components in block diagram and connect them with loop.

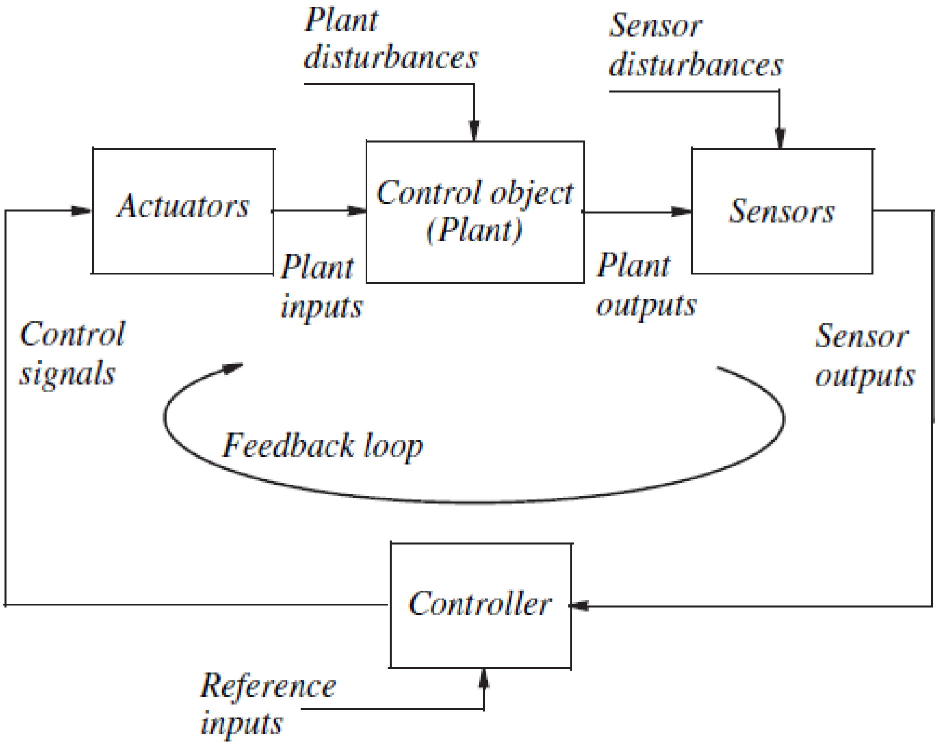


Figure 5.8: Block Diagram of a Control System

In exoskeleton system the actuators is high performance AC servomotors with drivers and gear box, able to lift 50 Kg, the planet is exoskeleton body, the planet disturbances is the weight of exoskeleton and the weight of the load, the sensors are encoders in feedback, force sensors in forward, torque sensors for Compliancy, limit switches sensors for safety, and the controller is Arduino due with *MatLab* library.

5.5.1 Concept of Control

A control system is a dynamic system which is designed to operate in a prescribed manner without external interference, in spite of unavoidable effects (disturbances) which impede its proper operation. The high-level control concepts based on position torque control. The force sensor take force signal from user to give the change value of position. It is the command for position control that give control torque command for driver that make torque control. The relation between the kinematic and kinetic values is typically given by a second order system, that is both approaches simulate a virtual system in each joint with dynamic behavior in equation 5.22.

$$J_v\ddot{\phi} + D_v\dot{\phi} + C_v\phi = \tau_h \tag{5.48}$$

The angle ϕ_d denotes the desired elbow joint position, The impedance parameters $J_v, D_v and C_v$ are the desired moment of inertia, damping and stiffness. The resulting net torque τ_h denotes the total desired torque commanded by the user.

To achieve this concept of control and the requirement of exoskeleton control, two method of control must be made, position control and torque control. The details as follow.

1. **Position Control**

   To control the end=effector position of exoskeleton, you should made a gravity compensation and that done by state feed back control, and to make the steady state error equal zero when exoskeleton end-effector moving the tracker should be done.

   feedback compensation can yield faster responses and can be used in cases where noise problems preclude the use of cascade compensation. Also, feedback compensation may not require additional amplification, since the signal passing through the compensator originates at the high-level output of the forward path and is delivered to a low level input in the forward path. So based on equation 5.22 its get transfer function for system in equation 5.23.

   $$Y(s) = \frac{L(\phi_d)}{L(\tau_h)} = \frac{1}{J_v s^2 + D_v s + C_v} \tag{5.49}$$

   with s denoting the Laplace variable and the Laplace transformation. As no static force is desired, stiffness is set to zero, that is $C_v = 0$. It is desirable to set the dynamic parameters for inertia and damping as small as possible in order to achieve high dynamic movements, so suppose$D_v = 0$ and J as we discus before.

   According to this data, which estimated before in this chapter we have matrix A,B and C, and made sure that the system is controllable because controllability matrix is full rank. After that find the A_e, B_e matrices and place poles to find the gains $K_p, K_i and K_d$, of tracker and state feedback control. The block diagram as shown in figure 5.5.

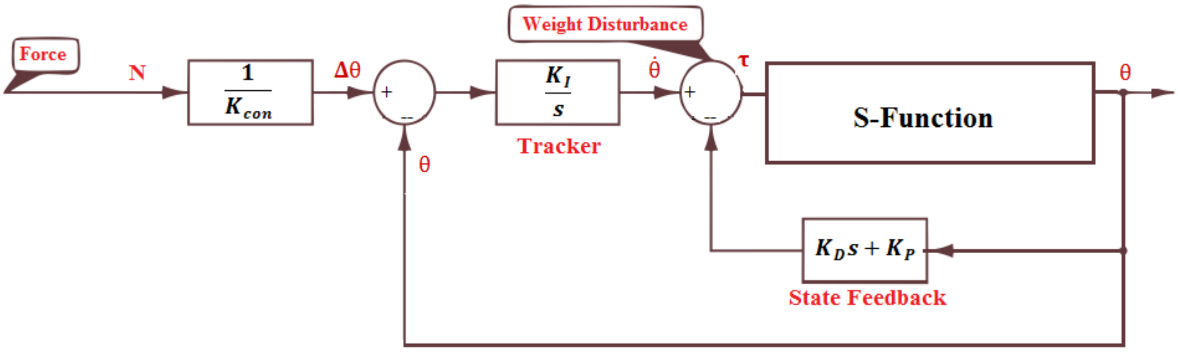


Figure 5.9: Block Diagram of a Control System

2. **Torque Control**

   Because of the high torque lift by exoskeleton, which possible to reach 500 N, the position control is insufficient to control it. Moreover the position control, torque control have to made. In this case we chose the motors and the drivers have the internal torque control. Figure 5.6 showing the torque control block diagram, and for more information and details we will add an appendix in the in of this bock.

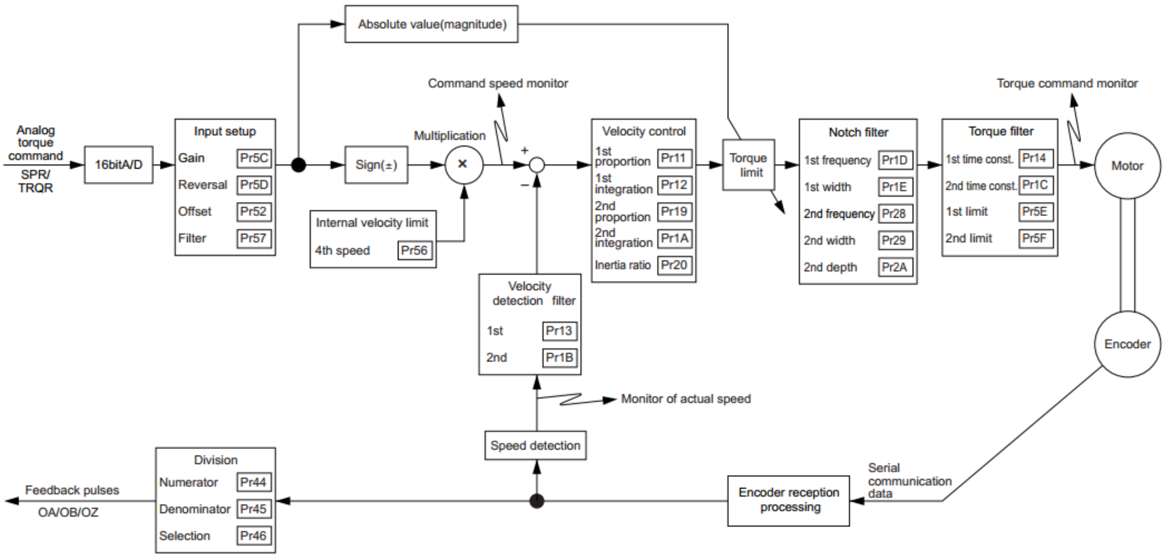


Figure 5.10: Torque control block diagram and parameters

5.5.2 Simulink and result by *MatLab*

After the control system and the control concept have been defined, the next step is to build the *MatLab* simulink for these control, to test the control concept, to showing the control response, and to verify that these control system is suitable for these planet, and able to solve the exoskeleton problem the *MatLab* simulink and its result as we see in following figure.

- **simulink by *MatLab***

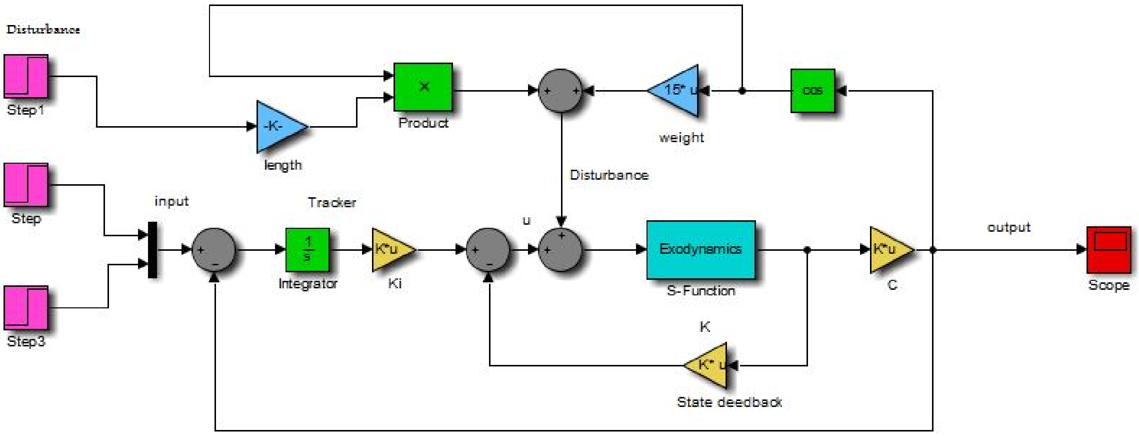


Figure 5.11: Simulation *MatLab* simulink

- **Response without disturbance**

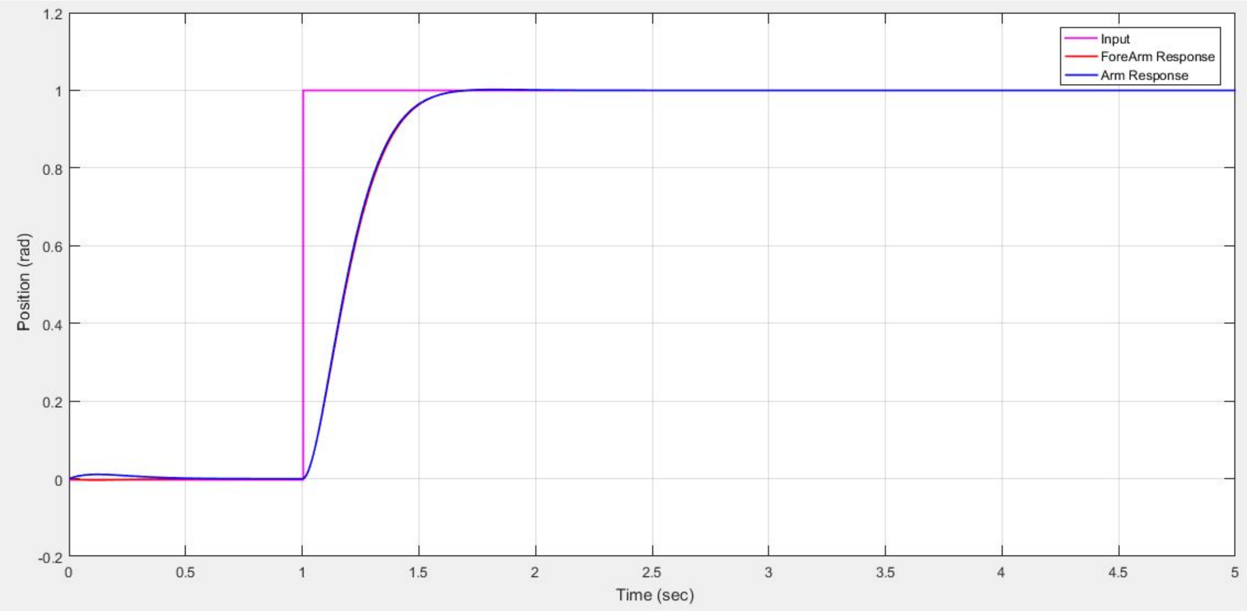


Figure 5.12: Simulation Response

- **Response with disturbance**

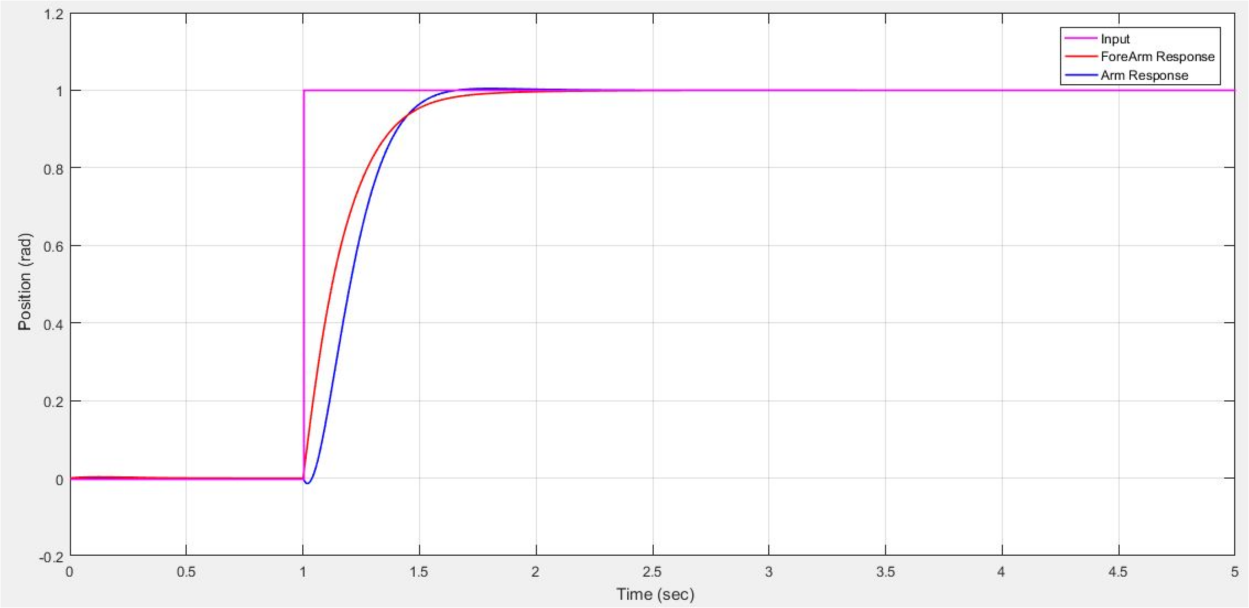


Figure 5.13: Simulation Response

As we see in last figures the Exoskeleton moves without load simply without overshoot and with fair velocity, but when we put the 50;Kgs on it, we see how the response bit fall down and that is a nature for caring any load, and the exoskeleton goes as require.

- **Total Torque Requier to lift** 50 Kg

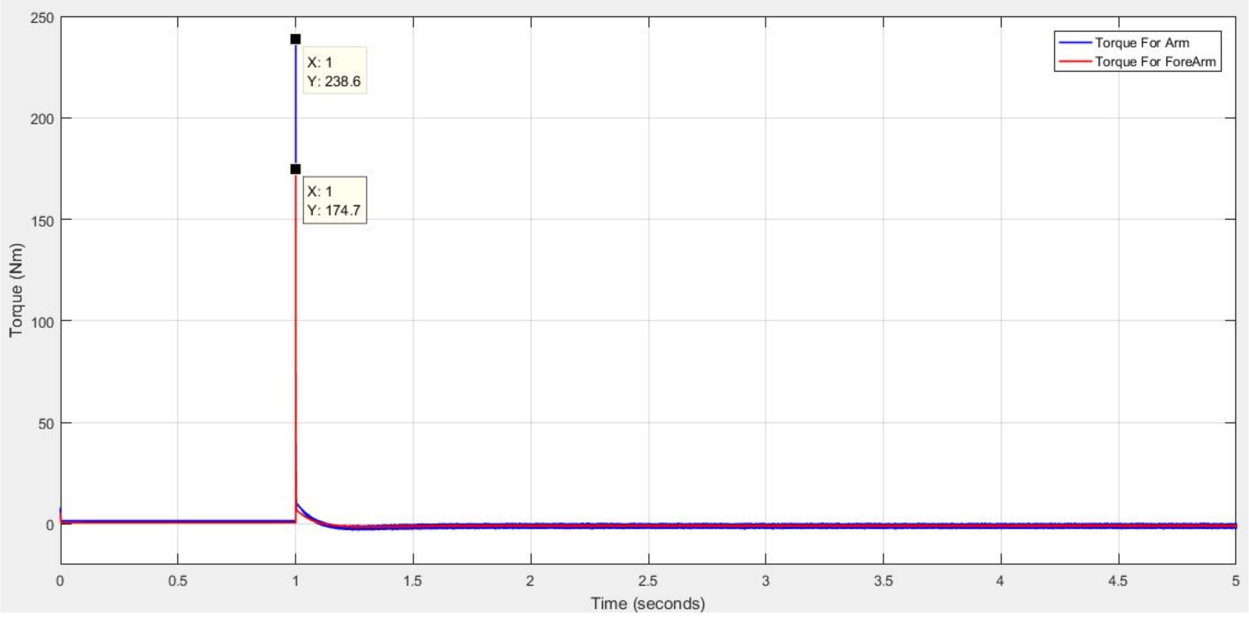


Figure 5.14: Simulation Response

We can also from simulink shows the requires torques for motors,as we see in last figure the require torque for elbow joint is more than 174 Nm, and for shoulder joint is 238 Nm.

5.6 Human in the loop

Despite the advances in autonomous robotics and automation, some tasks still require human intervention or guidance to mediate uncertainties in the environment or to manage the complexities of a task that autonomous robots are not yet equipped to handle. Like, many safety-critical systems are interactive, in other word they interact with a human being, and the human operator's role is central to the correct working of the system.

The symbiotic relationship between humans and robots transcends the boundaries of simple physical interaction. It involves smart sensors, actuators, algorithms and control strategies capable of gathering and decoding complex human expressions or physiological phenomena. Once this process is complete, robots use the information to adapt, learn and optimize their functions, or even to transmit back a response resulting from a cognitive process occurring within the robot.

A cognitive process is a sequence of tasks including reasoning, planning, and finally the execution of a previously identified problem or goal. Originally this concept was restricted to living creatures, but now it can also be applied to smart robots that accomplish the above-mentioned sequence of tasks. The humanrobot cognitive interface is the link between human and robot, in which the information regarding these processes is acquired and transmitted bidirectionally.

Where are we need adding human interaction to system ?, and Why?

- Biological reasons. cognitive humanrobot interface systems seek to take advantage of the natural control mechanisms fully optimized in humans. Moreover, a lot of information is lost in the translation of biologically executed tasks into discrete events, for example a natural movements or gestures into buttons or joysticks.
- Practical reasons. Delays are introduced when natural cognitive processes are encoded into an imposed sequence of tasks. In addition, a training phase is needed to teach the user to generate these non expected commands or to map a cognitive process into a new set of outputs. Both factors, the delays and the mapping, can induce fatigue in the user, both at a musculoskeletal level and at a mental level. These factors can be obviated if the natural outputs of a cognitive process are used for cognitive humanrobot interface.
- Rehabilitation. One of the main applications of wearable robotics is rehabilitation. Interacting directly with the phenomena involved in the cognitive process is a means to excite them

and assess the evolution of the rehabilitation therapy.

Figure 5.9 shows three levels of interaction are. One of them related to reasoning and planning, other one related to muscle activity, and one related to the wearers motion.

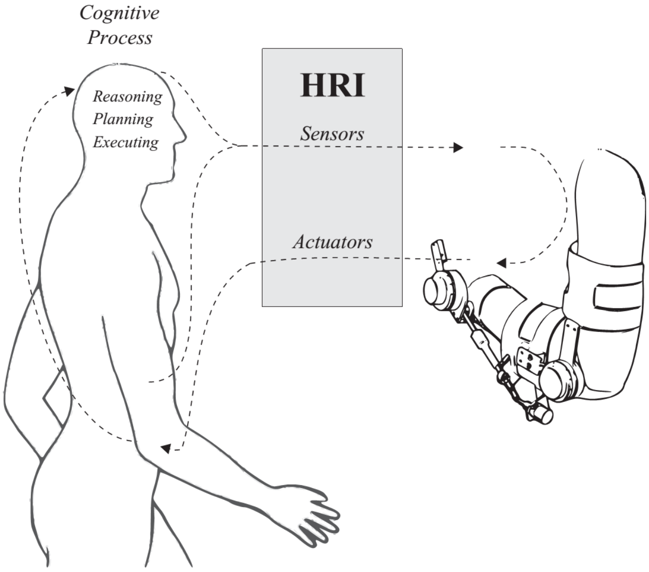


Figure 5.15: *MatLab* simulink

The human operator is an integral part of Exoskeleton control loop. Therefore, understanding and maximizing the collaboration between the control system and the human operator is essential. Adopting a systematic design approach is crucial for reasons of safety and optimum system performance.

The Human interaction in lifting load upper limb Exoskeleton is done by force and torque estimated by sensor. The control commands from force sensor is transformed to position command for control system. So the human make integral control on position command for the control that decreasing from percentage of error . Figure 5.10 is showing the Block Diagram for Human-in-the Loop Control of a Dynamic System. The human command a position (vk) the control command a torque(uk) and the system give a position(x).

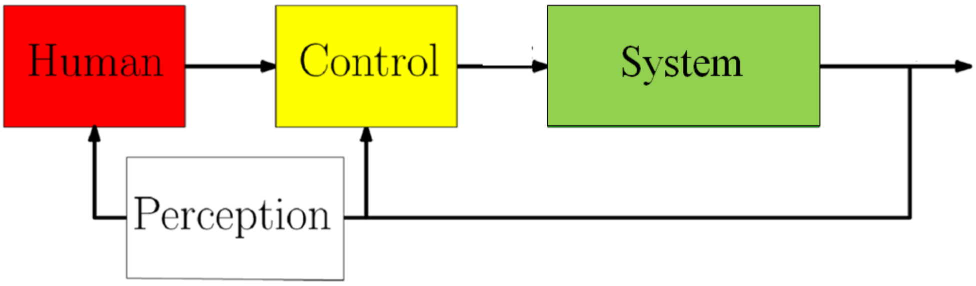


Figure 5.16: Human in the loop block diagram

Chapter 6

Exoskeleton Mechanical Design

6.1 Introduction

When you are going to design, you are either going to formulate a plan for the satisfaction of a specified need, or to solve a specific problem. If the plan results in the creation of something having a physical reality, then the product must be functional, safe, reliable, competitive, usable, manufactural, and marketable.

Mechanical engineering design is associated to formulate a planes or solve problem related to the mechanical part, such fluid flow, heat transfer, friction, energy transport, material selection, thermomechanical treatments, statistical descriptions, and so on. And it is concentrating mostly on loading stress strain analysis.

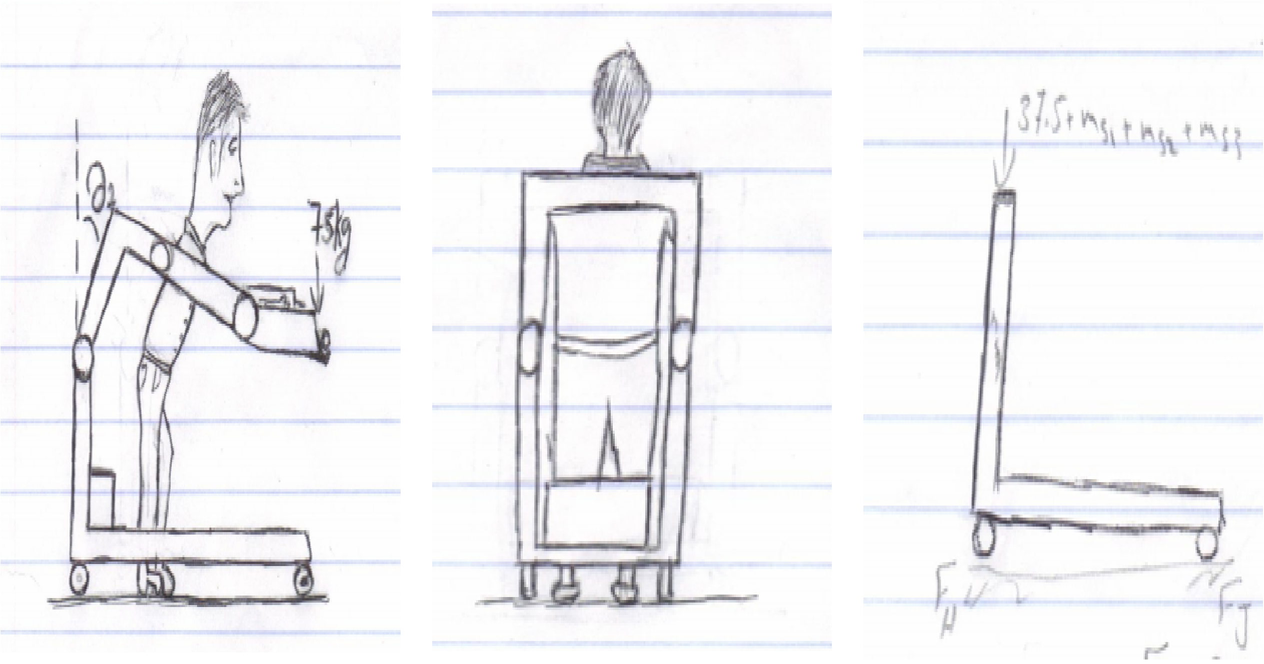
As we said before in chapter one, mechanics and mechatronics design going through several steps, start with identification the need, Definition of problem, Synthesis, Analysis and optimization, Evaluation and Presentation wirh acknowledging the many feedbacks and iterations. as we see in figure 6.1.

6.2 Design Requirement and Free Hand Drawing

The first step in designing any project its to define the project requirement. In other words to define the specifications which your project suppose to do. then you have to make many hand drawing designs and chose one of them, which satisfy your requirement. And its became as follow.

1. Enable person lifting load up to 50kg.
2. Doesnt constrain human motion.
3. Ergonomic design.
4. Earthing reaction force.
5. Enable tendency

And the Hand Drawing design in figure 6.2.



(a) Side view (b) Behinde view (c) Reaction

Figure 6.1: Hand Drawing Design

6.3 Design using *CATIA*

The word CATIA is an acronym of Computer Aided Three-dimensional Interactive Application, and its one of the best software program used for CAD,CAE and CAM, and its able to make part design for every part of any machine or project, then makes assembly for them, then to make kinematics for them, and it able to estimate stress and strain and load analysis for your machine or project.

As we see in figure 6.2 our design include movable base to enable movement with wearer, a passive back joint connected directly with base, then when we go above we will see the shoulder joint and this an active joint moved and controlled it by motor and Arduino, finally we have elbow joint also moved and controlled it by motor and Arduino. And we will show this joint in details in next sections.

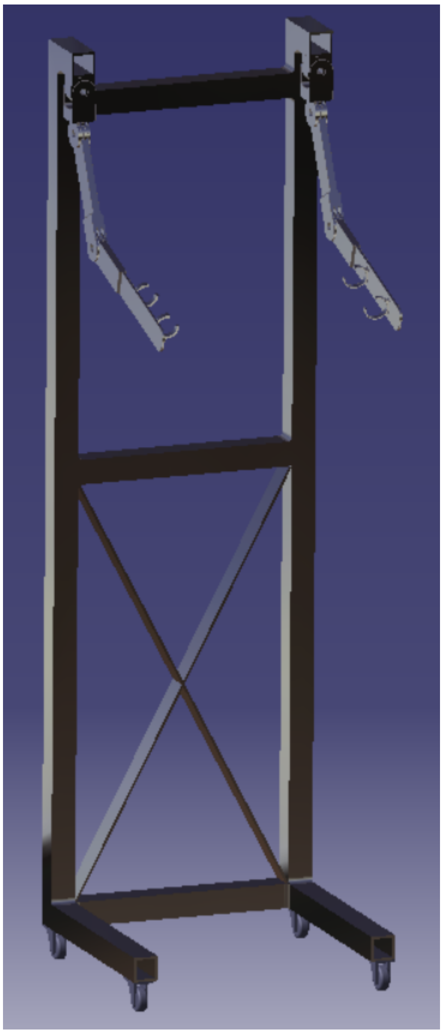


Figure 6.2: Exoskeleton CATIA Design

6.4 Forearm and Elbow joint Design and Analysis

Elbow joint is joint that links the upper arm and lower arm. And it contain two degree of freedom one of them is flexion-extension, its the degree which move the arm vertically on $y - axis$ and its an action degree. The second degree called Pronationsupination, its the one that move the arm horizontally in $x - axis$ and its a passive degree, but we have to didn't constrain is. The Design and analysis was made for right hand, and as the symmetry left hand has the same design and analysis.

This section discuss several points. At first drawing and designing the Forearm part and elbow of exoskeleton, then present inertia calculation, and at the end this section shows the loading and structure analysis when the exoskeleton carry 50 Kg.

6.4.1 Drawing and Designing

In figure 6.3 we see exoskeleton Forearm segment on *CATIA* design which begin from elbow to wrist joint. and its connect with two hangs to fixed the human arm on exoskeleton body, the length of Forearm segment is 37.5 cm, the width is 5 cm and the thickness is 1.5 cm.

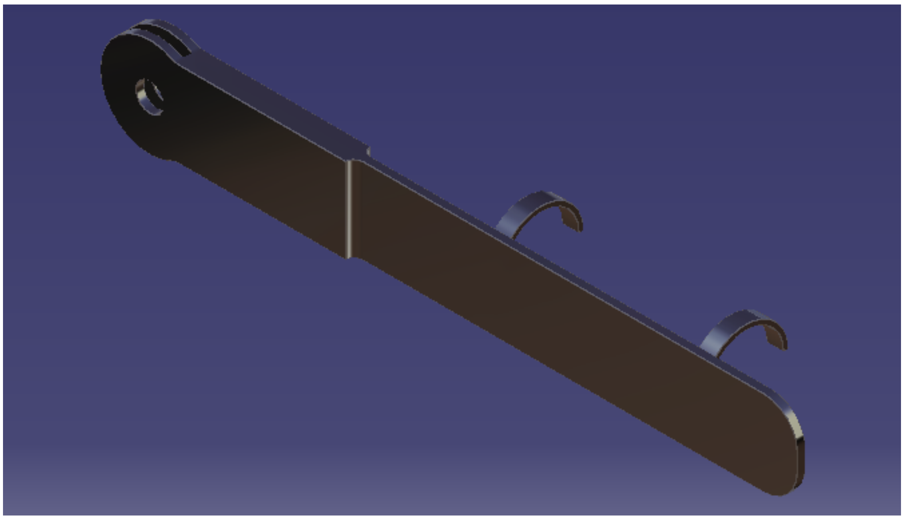
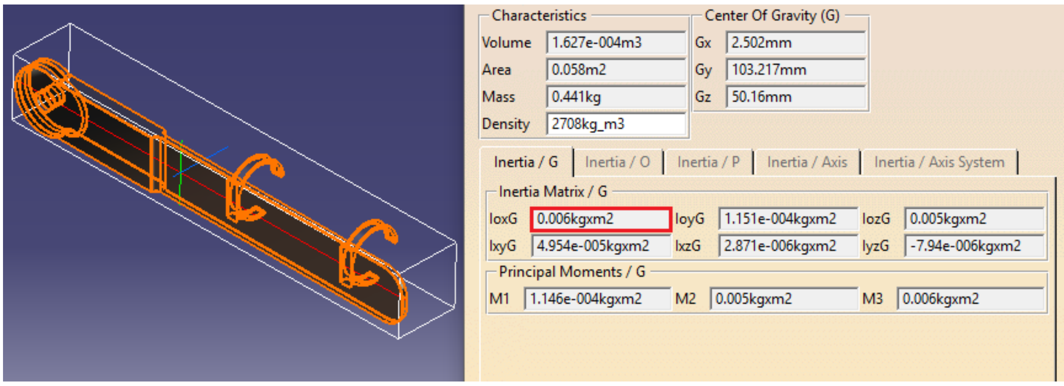


Figure 6.3: Forearm CATIA Design

6.4.2 Inertia Calculation

After drawing and Designing the forearm, the next step finding the inertia of this forearm, the inertia J is an important term in control system, this term uses in generating the differential equation by this equation $T = J\ddot{\theta}$, and the over all transfer function $1/js^2$. The forearm inertia was found also using *CATIA*. The forearm inertia as we see in figure 6.4 equal 0.006 $Kg\,m^2$. also we some of specification of forearm design.



Volume 1.627e-004m3
Area 0.058m2
Mass 0.441kg
Density 2708kg_m3

Gx 2.502mm
Gy 103.217mm
Gz 50.16mm

IoxG 0.006kgxm2
IoyG 1.151e-004kgxm2
IozG 0.005kgxm2
IxyG 4.954e-005kgxm2
IxzG 2.871e-006kgxm2
IyzG -7.94e-006kgxm2

M1 1.146e-004kgxm2
M2 0.005kgxm2
M3 0.006kgxm2

Figure 6.4: Inertia Calculation

6.4.3 structure and load analysis

Now in the final designing step for forearm, the stress and deflection analysis was made to define the maximum stress and maximum deflection, which act when the maximum possible load placed on exoskeleton, the maximum load is $50\,Kg$ which equal approximately $500\,N$ on both hands, but just $250\,N$ go through each hand, so the analysis made with $250\,N$ occurred on forearm. The stress and deflection analysis are as follow.

- **Stress Analysis**

  Stress is a physical quantity that expresses the internal forces that neighboring particles of a continuous material exert on each other, also using *CATIA*, figure 6.5 show these internal forces, and we see that the load Concentrated on $37.5cm$ Out of elbow joint, and the stress Concentrated on elbow joint.

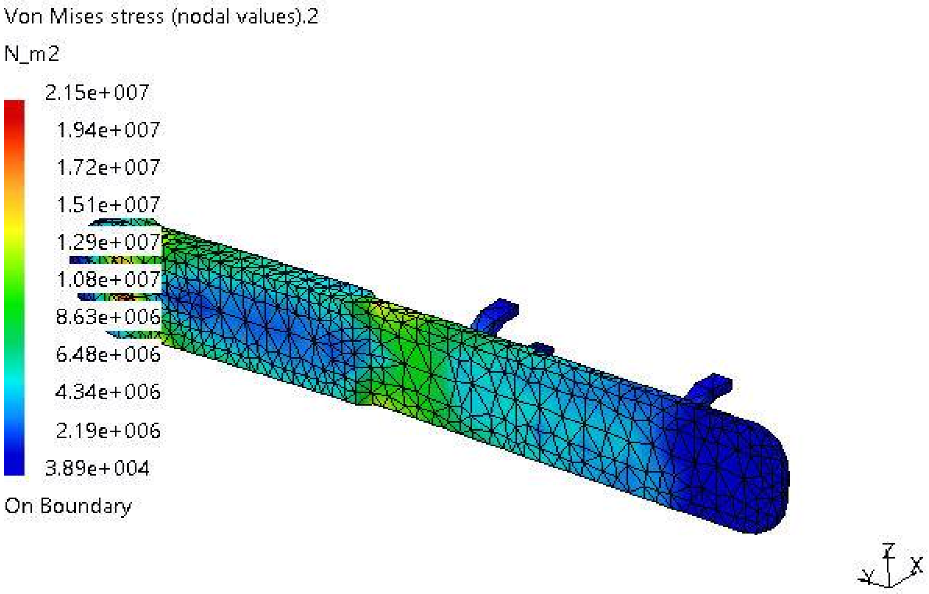


Figure 6.5: Stress Analysis

  As we seeing in figure 6.5 the maximum stress is$21.5\,MPa$, so we can use Aluminum or Aluminum alloy, which has an uletmat strength Ranging from $(95\,MPa\,to\,441\,MPa)$. So we can use this design without any hazard.

- **Deflection Analysis**

  The degree to which a structural element is displaced under a load. It may refer to an angle or a distance this quantity called *Deflection*, as we see in figure 6.6, the distribution of deflections on the forearm the maximum deflection is out of elbow joint with 37.5 *cm*.

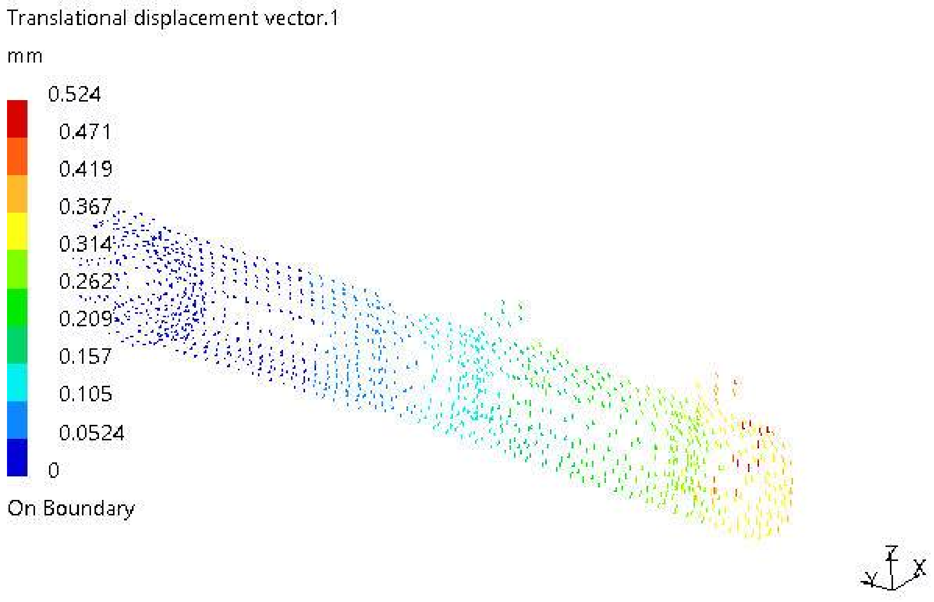


Figure 6.6: Deflection Analysis

As we seeing in figure 6.6 the maximum defection is 0.525 *mm* and its an acceptable deflection foe this design.

6.5 Arm segment and Shoulder joint design and analysis

Arm segment is a part of human, which links the shoulder joint with elbow joint, also in this part the human has the bi and ceps. Shoulder joint on of the most complex joint of human body, its locate in the head of the upper arm, this joint form like a ball-and-socket-type, and as we know this form allows three degrees of freedom.

- Translation on $x - axis$ (horizontal movement), which name is Abductionadduction.
- Translation on $y - axis$ (vertical movement), which name is Flexionextension
- Rotation

In this exoskeleton design, just one degree of freedom is an action one, its the translation in $y - axis$ degree, and we have not to constrain other tow degrees of freedom.
This section discuss several points. At first drawing and designing the arm part and shoulder joint of exoskeleton, then present inertia calculation, and at the end of this section shows the loading and structure analysis when the exoskeleton carry $50\,Kg$.

6.5.1 Drawing and Designing

Figure 6.7 is showing exoskeleton arm segment on *CATIA* design, which begin from shoulder to elbow joint. Also we see the shoulder joint in the top of figure and the elbow joint in the bottom, the length of arm segment from the center to center of shoulder and elbow joint is 23.5 *cm*, the width is 5 *cm*, the thickness is 1.5 *cm* to be 2.5 *cm* in elbow joint.

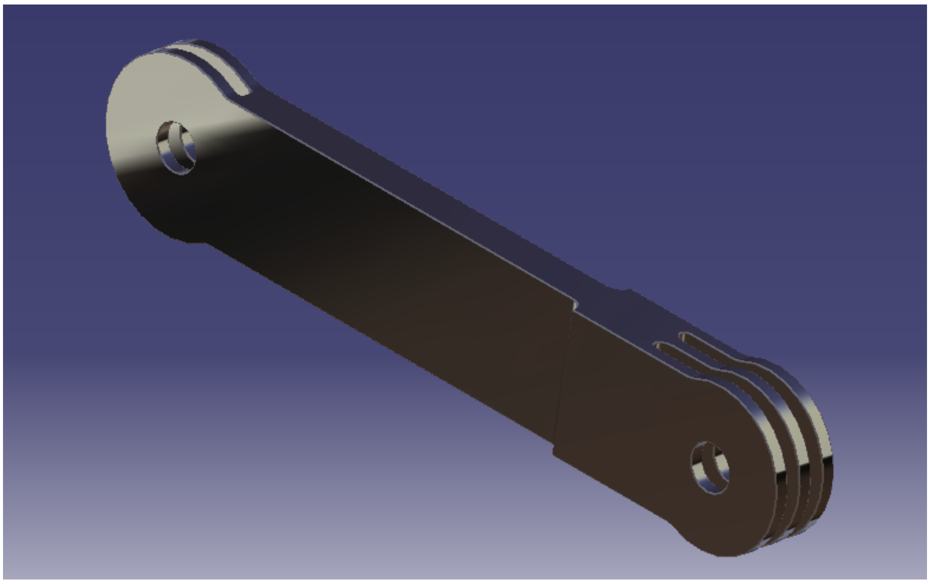


Figure 6.7: Arm segment

Figure 6.8 showing the shoulder joint mechanism, this mechanism location is in the head of the arm, and link the arm with human back. Also its enable vertical and horizontal movement.

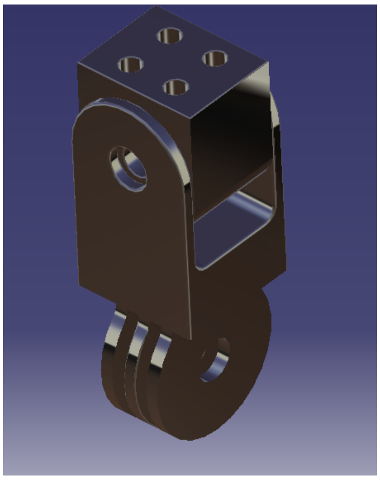
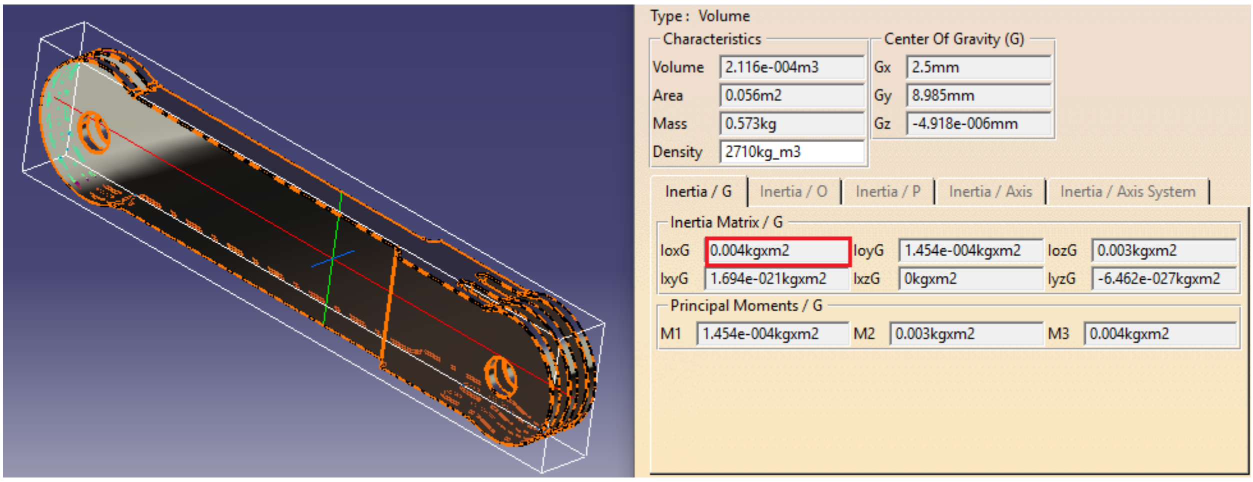


Figure 6.8: shoulder joint Mechanism

6.5.2 Inertia Calculation

After drawing and Designing the arm segment and shoulder joint, the next step is estimating the inertia of this arm, the inertia J as we said before is an important term in control system. But in this part the shoulder motor will carry both of arm and forearm segments, The forearm inertia as found in last section using *CATIA* was equal $0.006\,Kg\,m^2$. and the arm inertia as figure 6.9 shows is equal $0.004\,Kg\,m^2$. So the net inertia equal the forearm inertia plus arm inertia and its become $0.01\,Kg\,m^2$. figure 6.10 also present some of specification of arm design.



Type : Volume

Characteristics

Volume 2.116e-004m3

Area 0.056m2

Mass 0.573kg

Density 2710kg_m3

Center Of Gravity (G)

Gx 2.5mm

Gy 8.985mm

Gz -4.918e-006mm

Inertia Matrix / G

IoxG 0.004kgxm2

IoyG 1.454e-004kgxm2

IozG 0.003kgxm2

IxyG 1.694e-021kgxm2

IxzG 0kgxm2

IyzG -6.462e-027kgxm2

Principal Moments / G

M1 1.454e-004kgxm2

M2 0.003kgxm2

M3 0.004kgxm2

Figure 6.9: Arm Inertia Calculation

6.5.3 structure and Load analysis

Now in the final designing step for arm, the stress and deflection analysis was made to define the maximum stress and maximum deflection of arm segment, which act when the maximum possible load placed on exoskeleton, the maximum load is $50\,Kg$ which equal approximately $500\,N$ on both hands, but just $250\,N$ go through each hand, so the analysis made with $250\,N$ occurred on arm. Also we have the mass of forearm and arm link which equal $1\,Kg$, and the mass of second motor which actuate the forearm and this equal $4\,Kg$, The stress and deflection analysis are as follow.

- **Stress Analysis**

  Figure 6.10 shows the stress analysis, and we see that the load Concentrated on $23.5cm$ Out of shoulder joint, and the maximum stress has been in shoulder joint.

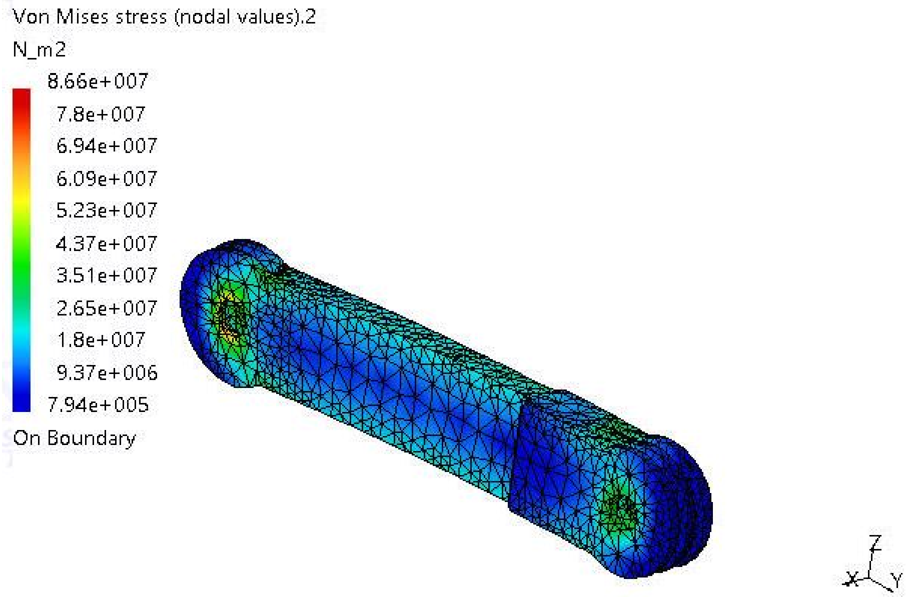


Figure 6.10: Stress Analysis

Figure 6.10 shows the maximum stress is 86.10 *MPa*, so we can Aluminum alloy, which has an uletmat strength is up to 441 *MPa*. So we can use this design without any hazard.

- **Deflection Analysis**

  The degree to which a structural element is displaced under a load. It may refer to an angle or a distance this quantity called *Deflection*, as we see in figure 6.11, the distribution of deflections on the forearm the maximum deflection is out of shoulder joint with 23.5 *cm*.

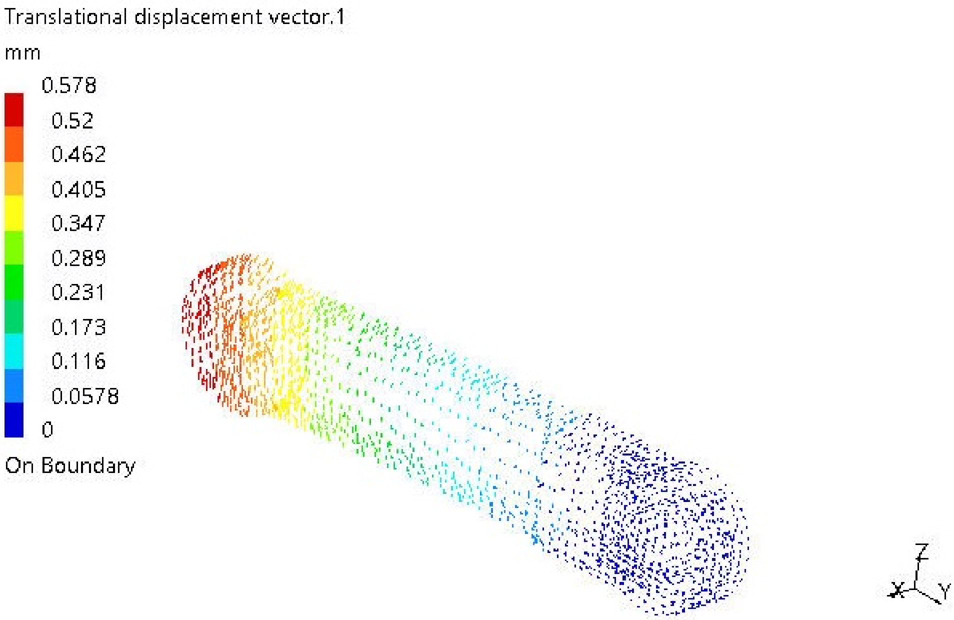


Figure 6.11: Deflection Analysis

As we seeing in figure 6.11 the maximum defection is 0.598 *mm* and its an acceptable deflection foe this design.

6.6 Back joint design and analysis

Back joint was established to enable tendency of human back, we know that when you going to carry a certain weight you have to make a tendency with certain angle, the same thing you have to do when you put down this weight. So in exoskeleton design its important to make this joint and make analysis for it.

Back joint is a joint which connect the exoskeleton base with back segment, back segment is a link to connect the back joint with shoulder joint. the back joint design with structure and load analysis as following.

6.6.1 Drawing and Designing

Figure 6.12 is showing back segment and joint design by *CATIA*, the length of back segment from center to center of back and shoulder joint is 48.5 *cm*, the width is 5 *cm*, the thickness is 5.5 *cm*.

Figure 6.12: back segment and joint

6.6.2 structure and load analysis

Stress and deflection analysis was made to define the maximum stress and deflection on back, t when the maximum load is act, moreover the weight arm and forearm and Their motor.

- **Stress Analysis**

  Figure 6.11 shows stress analysis, the load Concentrate on 48.5*cm* Out of base joint. joint.

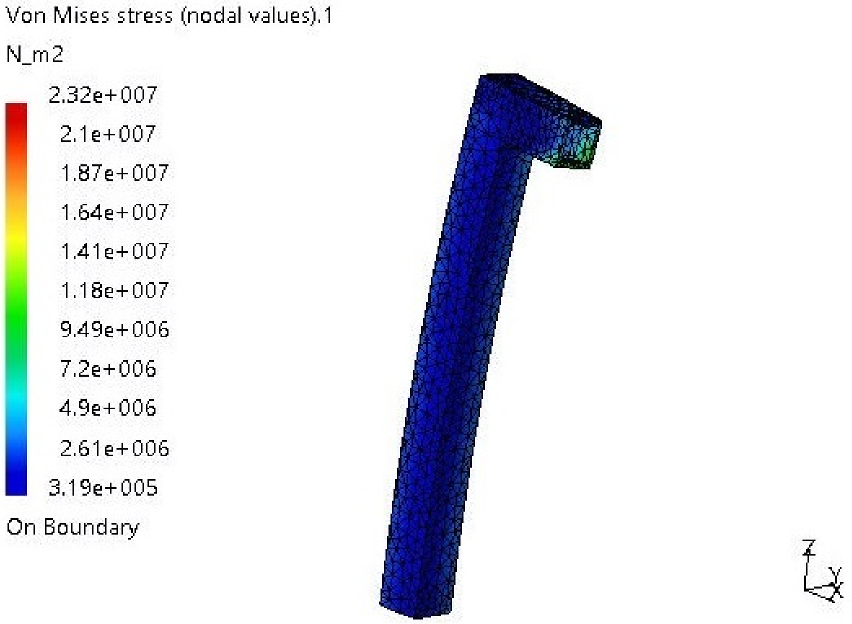


Figure 6.13: Stress Analysis

As we seeing in figure 6.14 the maximum stress is 23.2 *MPa*, so we can Aluminum alloy, which has an ultimate strength is up to 441 *MPa*. So we can use this design without any hazard.

- **Deflection Analysis** Figure 6.15 shows the deflection analysis for back segment, and we see that the maximum deflection Concentrated on 48.5*cm* Out of base joint.

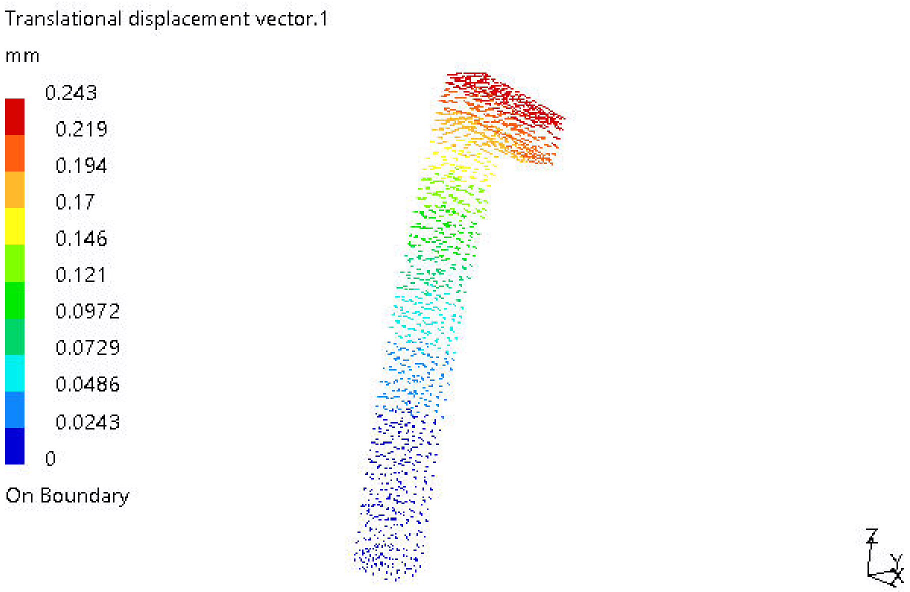


Figure 6.14: Deflection Analysis

As we seeing in figure 6.7 the maximum defection is 0.243 *mm* and its an acceptable deflection foe this design.

6.7 Conclusion

From part one of this project we conclude this points.

- Designing and building any project needs identifying the strategy of your work.
- Before building any project, you have to make a design for it, and make sure that your design is robust design.
- Mathematical model is a essential tool for your project to analyze your work, and test your system.
- Stress and strain analysis are important tools to material selection, and make sure that your exoskeleton is enable to lift the require weight.

Part II

Experimental work for One Degree of Freedom Prototype for Elbow joint

Chapter 7

1 DoF Prototype for Elbow joint test

7.1 Introduction and Conceptual design

Before begin building any project, you should make a small prototype for your project, to test and evaluate your idea, without building the whole project, and carry many risks, problem and spent more many on idea doesn't work.

The idea of this prototype is to test the validity of using force sensor as an input for Exoskeleton controller instead of EMG sensor. Also this prototype must fulfill these requirement of control.

1. Gravity Compensator.
2. Compliancy.
3. Force.
4. Force Loop.

In our project we have four joint with 2 Dof, in this chapter we will explain how and why we made this prototype in elbow joint and one DoF, the mechanical, electrical and control parts of this prototype, the kinematics, mathematical, and mechanical design analysis of this prototype, and the control loop how and why we made it, and the programs that we use it to control and design this proto type.

7.2 Schematic design

figure 7.1 shows the prototype schematic diagram. The input signs are entering the system by force sensors, then the microcntroller take this signs to process it and convert the force signs to $\delta\theta$, after that that the microcontroller send torque command to the motor driver, and thent the motor driver sind the orders to the motor to move to the desired position with suitable torque, the last step is feedback signs, which the encoder takes it to microcontroller to define the current position.

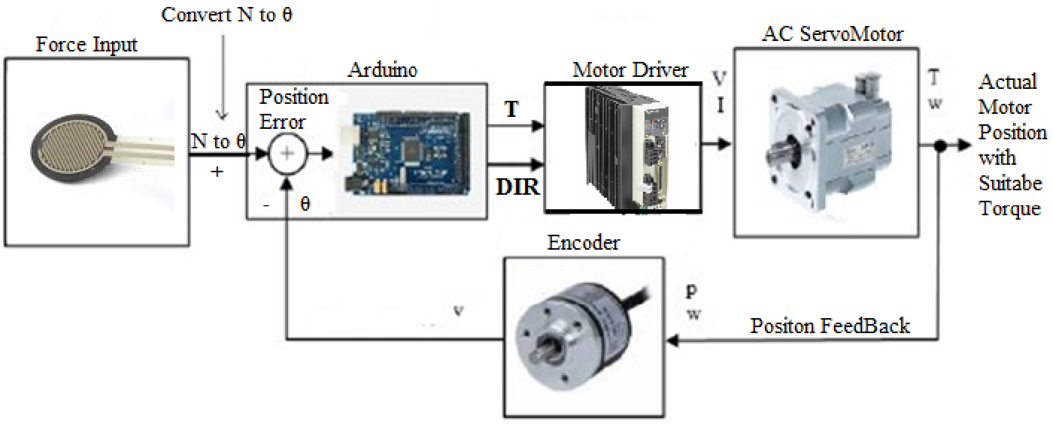


Figure 7.1: Prototype Schematic Diagram

7.3 Mechanical Design analysis

When you are going to design, you are either going to formulate a plan for the satisfaction of a specified need, or to solve a specific problem. If the plan results in the creation of something having a physical reality, then the product must be functional, safe, reliable, competitive, usable, manufactural, and marketable.

Mechanical engineering design is associated to formulate a planes or solve problem related to the mechanical part, such fluid flow, heat transfer, friction, energy transport, material selection, thermomechanical treatments, statistical descriptions, and so on. And it is concentrating mostly on loading stress strain analysis.

In this section we use computer software such as CATIA to make to draw the prototype and make a mechanical analysis on it, the mechanical drawing on CATIA. here we insert two prototype designs, first one is a reqtangular wood, and the last one is circular iron rod.

7.3.1 Design using *CATIA*

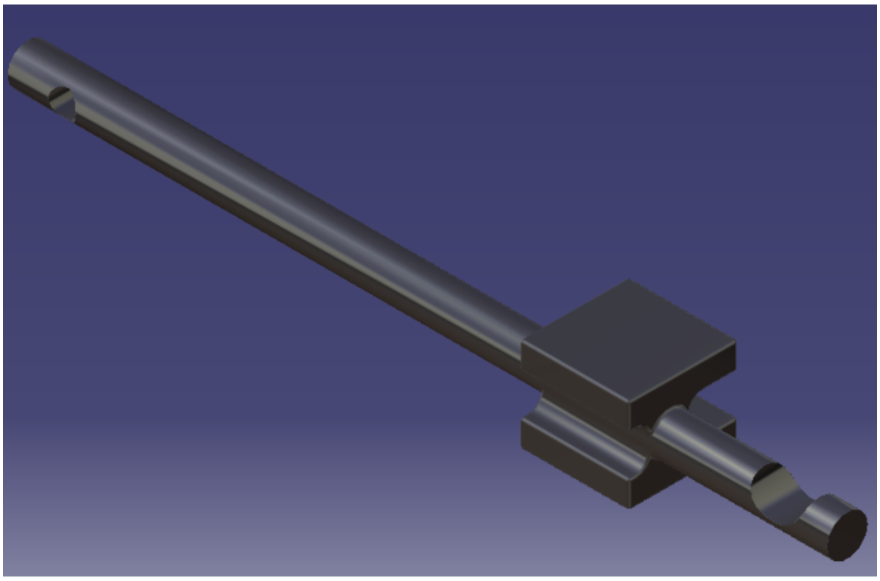
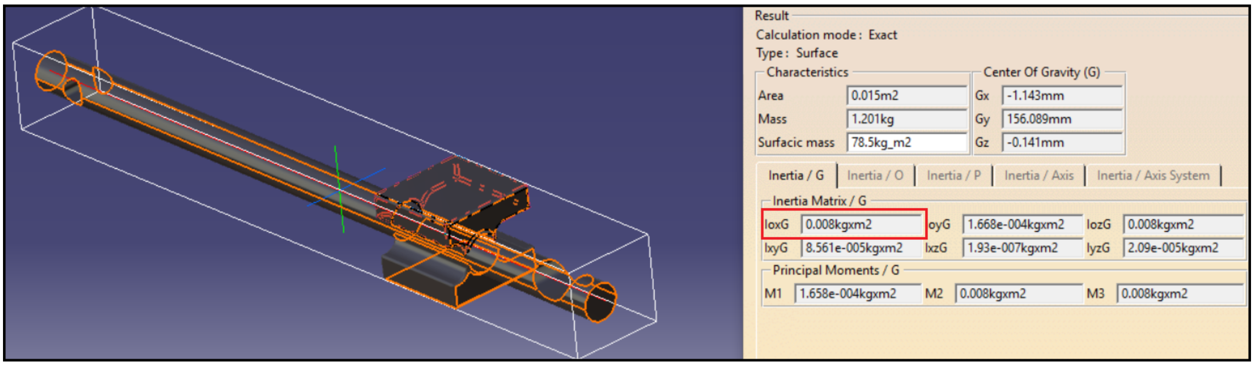


Figure 7.2: Prototype CATIA Design

After draw and build the prototype, we fined the inertia of the system in order to use it in next steps of controlling the system.



Result
Calculation mode : Exact
Type : Surface

Characteristics
Area 0.015m2
Mass 1.201kg
Surfacic mass 78.5kg_m2

Center Of Gravity (G)
Gx -1.143mm
Gy 156.089mm
Gz -0.141mm

Inertia Matrix / G
IoxG 0.008kgxm2
IoyG 1.668e-004kgxm2
IozG 0.008kgxm2
IxyG 8.561e-005kgxm2
IxzG 1.93e-007kgxm2
IyzG 2.09e-005kgxm2

Principal Moments / G
M1 1.658e-004kgxm2
M2 0.008kgxm2
M3 0.008kgxm2

Figure 7.3: Determine the inertia with CATIA

Also we made a loading, stress and strain analysis, on Catia when we applied 100N load as we see in next figures 7.4, 7.5.

- **Stress Analysis**

  Stress is a physical quantity that expresses the internal forces that neighboring particles of a continuous material exert on each other using CATIA we show these internal forces, and we see that its constrain on 30cm Out of motor.

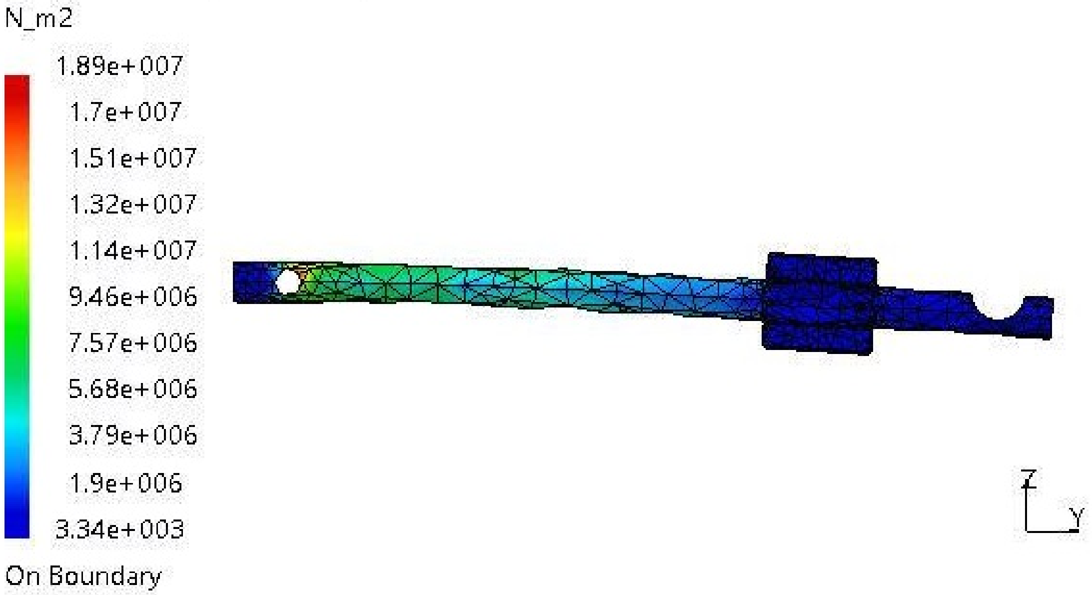


Figure 7.4: Stress Analysis in CATIA

The Ultimate strength for iron is 250 *MPa*, thats mean we are in safe side, since the maximum stress on wood rod is 18.9 *MPa*.

- **Deflection Analysis**

  The degree to which a structural element is displaced under a load. It may refer to an angle or a distance this quantity called *Deflection*, as we see in figure 7.4, the distribution of deflections on the wood rod the maximum deflection is out of motor with 30*cm*.

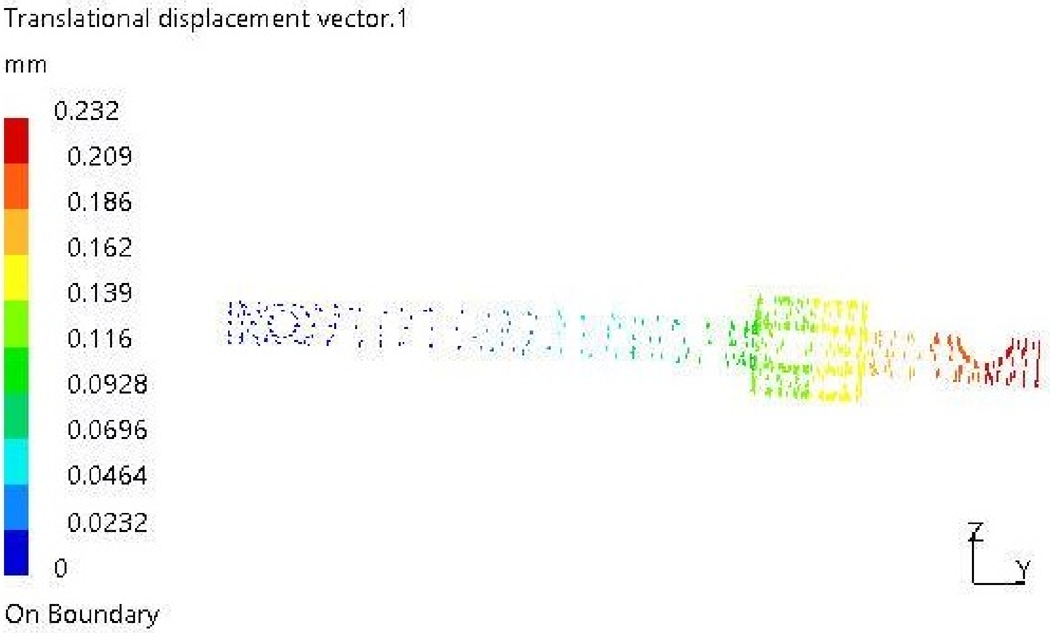


Figure 7.5: Deflection Analysis using CATIA

The maximum deflection is $0.2mm$, and for forces less than $50N$ the deflection will be less than $0.2mm$.

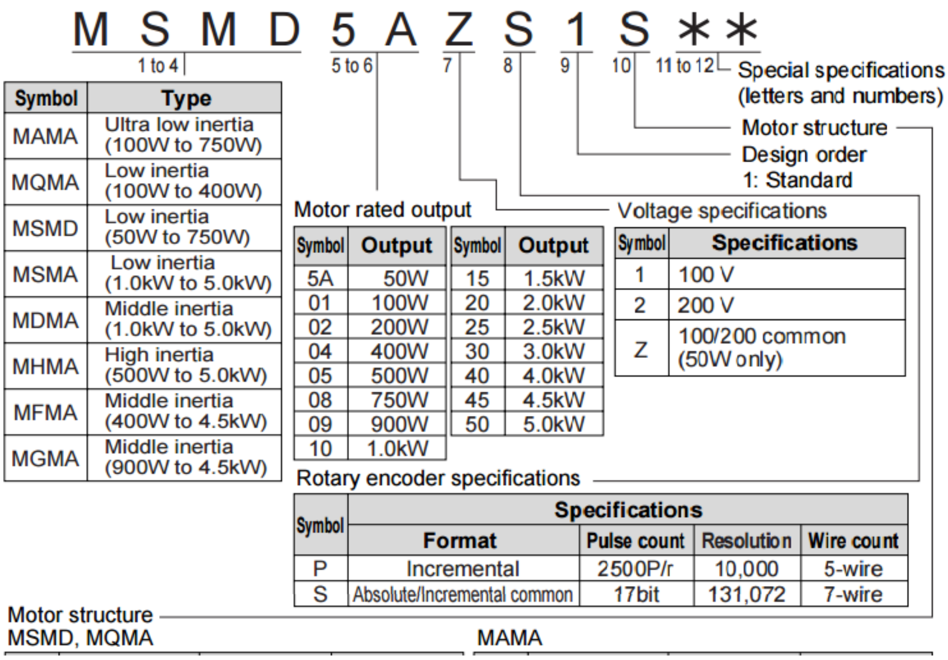
7.4 Prototype Components

In mechatronics project, all mechatronics project, you must enclose three main components, which electrical and electronics, mechanical and control components. In other word sensors, actuators and controller. We will present them as following.

7.4.1 Motor

in our prototype we actuated the system by motor, and we chose AC servo motors, produced by Panasonic company, with Model No. MSMD5AZS1S. The motor, content of Model No. in figure 7.6, 7.7 and name plate table in table 7.1.

Figure 7.6: Panasonic AC Servomotor



Symbol	Type
MAMA	Ultra low inertia (100W to 750W)
MQMA	Low inertia (100W to 400W)
MSMD	Low inertia (50W to 750W)
MSMA	Low inertia (1.0kW to 5.0kW)
MDMA	Middle inertia (1.0kW to 5.0kW)
MHMA	High inertia (500W to 5.0kW)
MFMA	Middle inertia (400W to 4.5kW)
MGMA	Middle inertia (900W to 4.5kW)

Motor rated output

Symbol	Output	Symbol	Output
5A	50W	15	1.5kW
01	100W	20	2.0kW
02	200W	25	2.5kW
04	400W	30	3.0kW
05	500W	40	4.0kW
08	750W	45	4.5kW
09	900W	50	5.0kW
10	1.0kW		

Voltage specifications

Symbol	Specifications
1	100 V
2	200 V
Z	100/200 common (50W only)

Rotary encoder specifications

Symbol	Specifications			
	Format	Pulse count	Resolution	Wire count
P	Incremental	2500P/r	10,000	5-wire
S	Absolute/Incremental common	17bit	131,072	7-wire

Figure 7.7: Motor Model No. Analysis

Panasonic AC servomotor Model No. MSMD5AZS1S			
INPUT	$3\,\phi$ AC	$69\,V$	$1.3\,A$
RATED OUT.	$0.2\,KW$	CONT. TORQUE	$0.64\,Nm$
MAX. CONT. TORQUE	$1.3\,Nm$	PEAQ TORQUE	$3.8\,Nm$
RATED FRQ.	$200\,Hz$	RATING	$S1$
RATED REV.	$3000\,rpm$	CONNECTION	STAR (*)

Table 7.1: Name Plat Table for motor

7.4.2 Mechanisms

We design a prototype mechanism to be interfaced to the elbow joint, and to fit with requirements, which we talked about it in introduction of this chapter. And this Mechanisms function is to conduct the force and torque to the Wight, in order to lift its. The mechanism design is in figure 7.2.

Also because of the continuous torque is insufficient as we see in name plate table in table 7.1, we add gear box with gear ratio 1:20, to enlarge torque to be more than $20Nm$. the gear box and over all Mechanisms shown in figures 7.8 and 7.9.

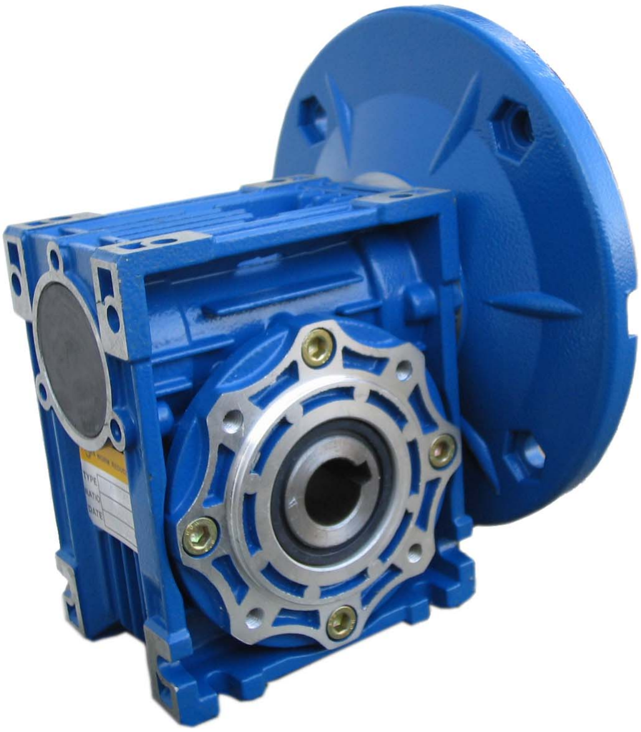


Figure 7.8: Gear Box 1:50

Figure 7.9: Over All Mechanisms

7.4.3 Sensors

as we said before we use some sensors in order to measure these parameter.

1. **Angle**

Tom DeMarco said " you can't control what you cant measure.", according to this, if you want to control the position of prototype end-effector we have to measure these position, based on that we use an 2500 P/rev incremental encoder to do this job, and we see the specification of this encoder at Model No. specification in figure 7.7.

The output of this incremental encoders provides information about the motion of the rotated motor shaft, which is typically further processed elsewhere into information such as speed, distance and position. But in this case just the position of the prototype end-effector needed.

At first we try to take the encoder pulses to the *MatLab* simulink, then process it and take the required angle from encoder, but unfortunately this simulink worked only on simulated pulses, but when we connect it with encoder it did not work as required the *MatLab* simulink for the encoder shown in figure 7.10.

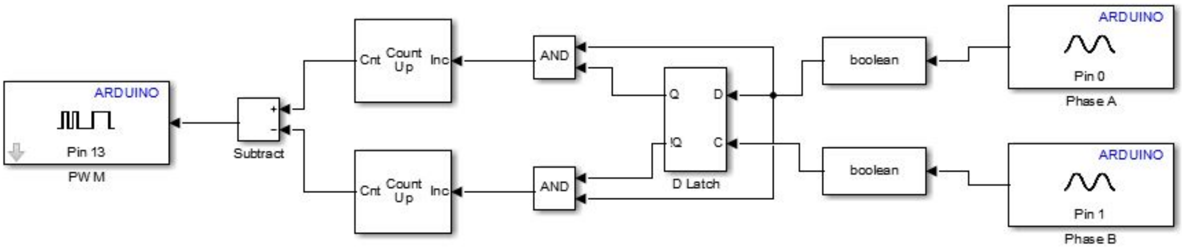


Figure 7.10: Encoder on *MatLab* simulink

After that we think to use Arduino code to get the angle reading from encoder, and that being easy by using the Arduino encoder library $< Encoder.h >$, but in this situation we have to use two controller one for *MatLab* simulink, and other one for talking pulse, process, count and then send it to the first one by serial communication. but when you know that the encoder count $10,000\ puls\ by\ revelution$ and after adding the $1:50$ gear box, the Arduino musr send $500,000\ pulse$ in one revelation, and this can;t verfy by using one serial $8-bit$, $255\ pulse$, nor two serial $16-bit$, $65,536\ pulse$, but by use three serial $24-bit$, $16,777,216\ pulse$. The number of pulse must divide to three packages each one in $8-bit$ form, then add them by shift in *MatLab*, the Arduino code is in appendix and the *MatLab* in figure 7.11.

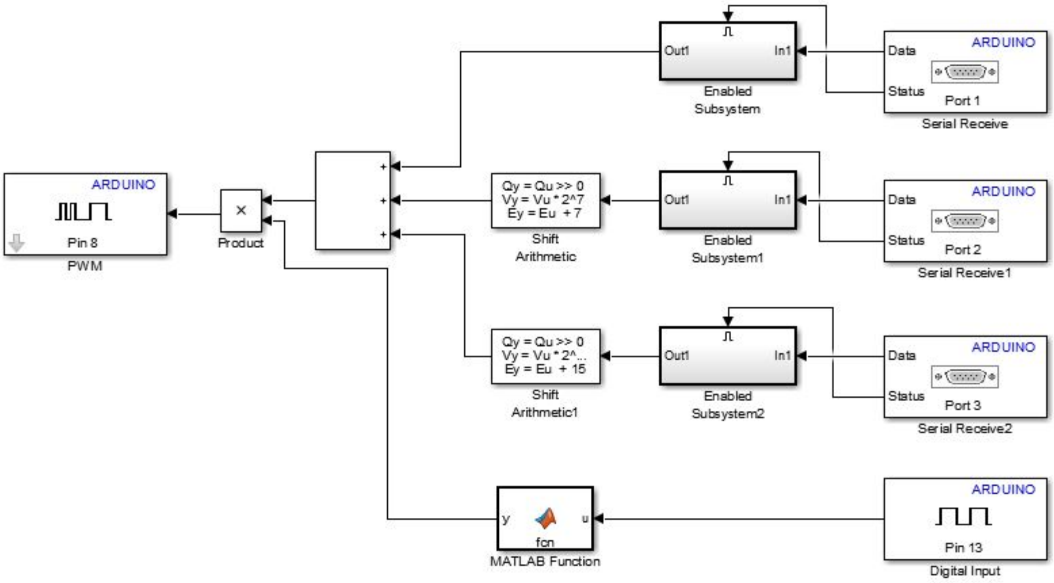


Figure 7.11: Encoder on *MatLab* simulink serial method

AS we see in figure we have three serial ports one, two and three. The first port is add directly, the second port was added after making $8-bit$ shift, and the third one was added after making $16-bit$ shift, and as we see the encoder sign is sent alone in digital pin 13, the MatLab Function is out 1 if the digital pin is 0, and out -1 is the digital pin is 1, then the sign multiply to the sum. After that we can take the position measurement correctly.

2. **Force**

Force sensor is used to determine the force located above and under the prototype arm, and send the force signals to the microcontroller, which convert it to $\delta\theta$,then send analog signal to motor to give suitable torque to the system to lift the weight the mass.

In this project, we used two-force sensor upper and lower the prototype arm to control the movement of it up or down according to the desired angle for arm. the signals of this two force sensors have been taken and processed by the first Arduino, and you can find the code in appendix, the as the encoder, the desired angle $(\delta\theta)$ has been sent by serial communication to the second one, which connected with *Matlab* simulink. the force sensors connection with Arduino as we see in figure 7.12.

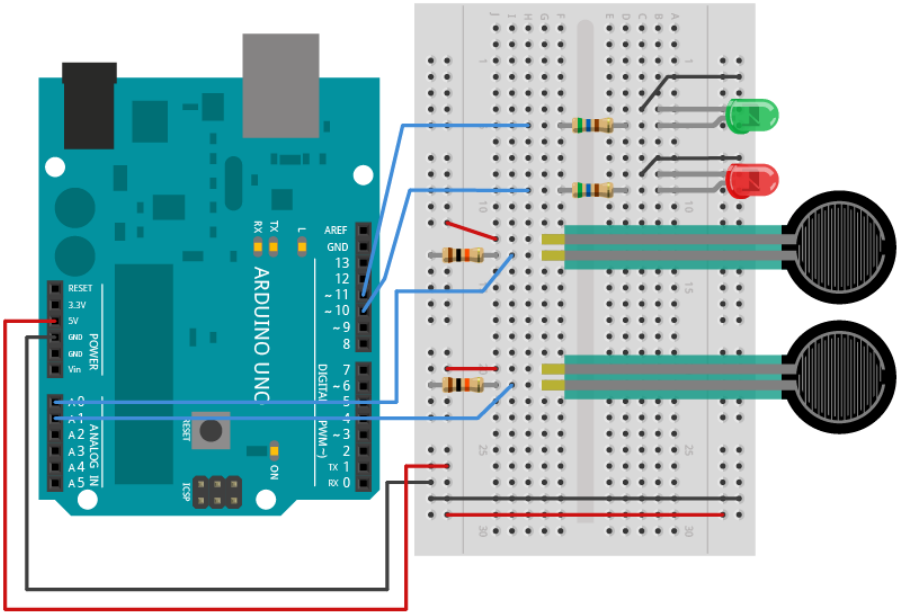


Figure 7.12: two force sensor connection

For Exoskeleton, four sensor Enough to control the Exoskeleton movement for upper limbs,by putting two sensor on each hand. And we ensure some of the force sensors specification in these following point.

- **Force sensor test**

  we test this force by adding some known weight on it and showing by oscilloscope the the voltage reading of this weight, we take this reading and plot it as we see in figure 7.13.

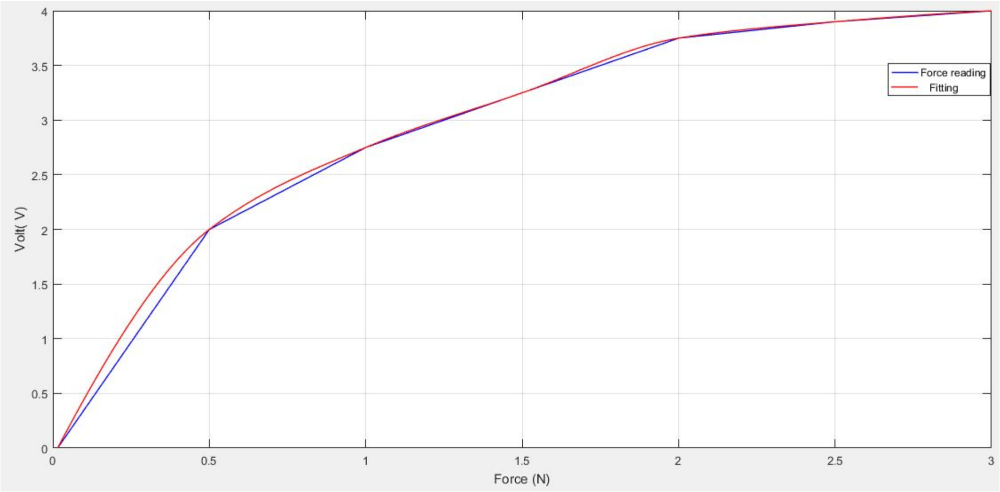


Figure 7.13: Force sensor test

- **Equation**

  According to reading of the force and voltage, the reading sent to *MatlaB*, and then the cubic equation generated as we see in equation 7.1.

$$V = 0.41\,N^3 - 2.7\,N^2 + 5.9\,N - 0.63 \tag{7.1}$$

- **Range of force sensors**

  The range of overall signals that the force sent is from $0.15\,N\ \ to\ \ 10\,N$. But the signal goes to saturation after 3 . So the effective range for its is from $0.15\,N\ \ to\ \ 3\,N$.

- **Dead zone**

  Is the range of values of the input (quantity being measured) for which the instrument gives no reading (it is the threshold when the input starts from a zero value). In this force sensors the dead zone is $0.15\,N$.

- **Sensitivity** A sensor's sensitivity indicates how much the sensor's output changes when the input quantity being measured changes. So in force sensor the sensitivity is a ratio between the outputs (volt) and inputs (Force). This concept for linear relations, but for non-linear relation the sensitivity is be as equation 7.2.

$$S = K\ +\ \frac{NI}{I}, \qquad K = \frac{\delta Output}{\delta Input}, \qquad \frac{NI}{I} = Max.\ Nonlinearity \tag{7.2}$$

  according to the plot, $K = 1.22$, and maximum non linearity will equal 0.2, the overall sensitivity$S = 1.42$.

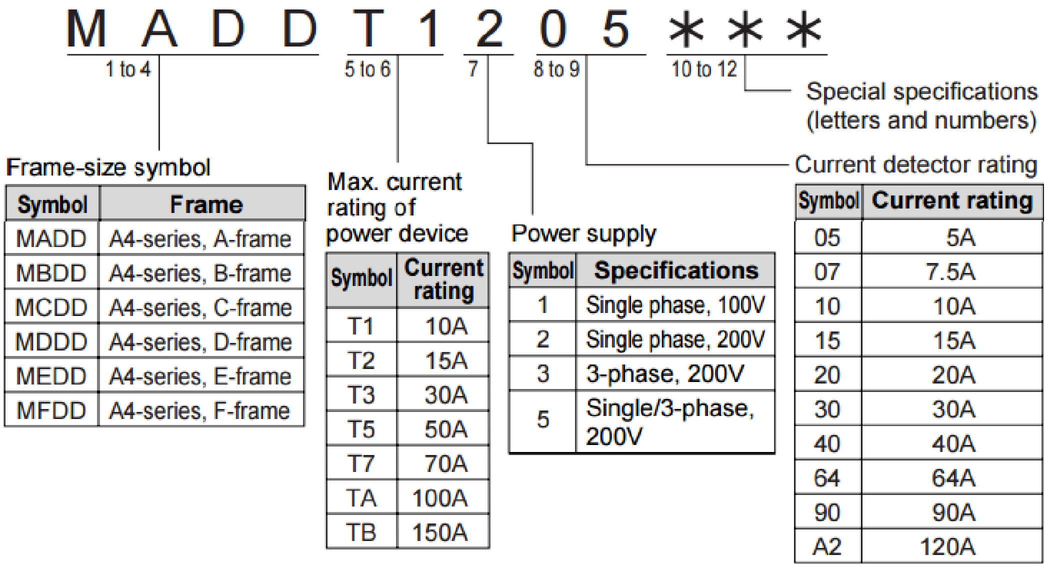
- **Resolution** The resolution of an instrument is the smallest change in the quantity being measured that will produce an observable change in the reading of the instrument. According to the test, the resolution will equal $R = 0.01$.

7.4.4 Driver

A motor driver is a device or group of devices that serves to govern in some predetermined manner the performance of a motor. A motor driver might include a manual or automatic means for starting and stopping the motor, selecting forward or reverse rotation, selecting and regulating the speed, regulating or limiting the torque, and protecting against overloads and faults and so on.

In our prototype we use a Panasonic driver with Model number MADDT1205. Three phase motor, to control and converting signals. The motor drive and its Model number details shown in figure 7.14,7.15 and name plate table in table 7.2.

Figure 7.14: Panasonic AC Servo Motors Drive



Frame-size symbol

Symbol	Frame
MADD	A4-series, A-frame
MBDD	A4-series, B-frame
MCDD	A4-series, C-frame
MDDD	A4-series, D-frame
MEDD	A4-series, E-frame
MFDD	A4-series, F-frame

Max. current rating of power device

Symbol	Current rating
T1	10A
T2	15A
T3	30A
T5	50A
T7	70A
TA	100A
TB	150A

Power supply

Symbol	Specifications
1	Single phase, 100V
2	Single phase, 200V
3	3-phase, 200V
5	Single/3-phase, 200V

Current detector rating

Symbol	Current rating
05	5A
07	7.5A
10	10A
15	15A
20	20A
30	30A
40	40A
64	64A
90	90A
A2	120A

Figure 7.15: Model Number Details for motor derive

Panasonic AC servomotor Drive Model No. MSMD5AZS1S		
	INPUT	Output
Voltage	$200-240\,V$	$69\,V$
Phase	$1\,\phi$	$3\,\phi$
F.L.C.	$1.3\,A$	$1.2\,A$
FREQ.	$50/60\,Hz$	$0-333\,Hz$
POWER		$100\,W$

Table 7.2: Name Plat Table for motor driver

This driver has multifunction, and it can deal with 50 inputs and outputs, and hundreds of parameters in this prototype we just need to use 8 input and output for torque control. Table 7.3 shows the inputs and outputs number and the function of them. And for more details the product manual for torque control is in appendix.

7	$+12\,V$	25	Encoder common
14	Torque command	29	Servo On
15	Torque Common command	41	Driver common
21	Encoder channel A	48	Encoder channel B

Table 7.3: Driver inputs and outputs

7.4.5 Circuits

We use some circuits in this prototype, in order to make matching between motor drive and Arduino controller, we know that the maximum output and input for and from Arduino is 5 volt, and the motor driver receive signal from -10 to 10 volts, we use the following circuit to make this matching. Also we mention the circuits, driver, Arduino and sensors connections for this prototype in figure 7.18.

1. **Input signal 0-5 volt, Output 0-10 volt**

   We use Operational Amplifier TL084, Transistors, some Resistors and Capacitor to create this circuit, which shown in figure 7.16.

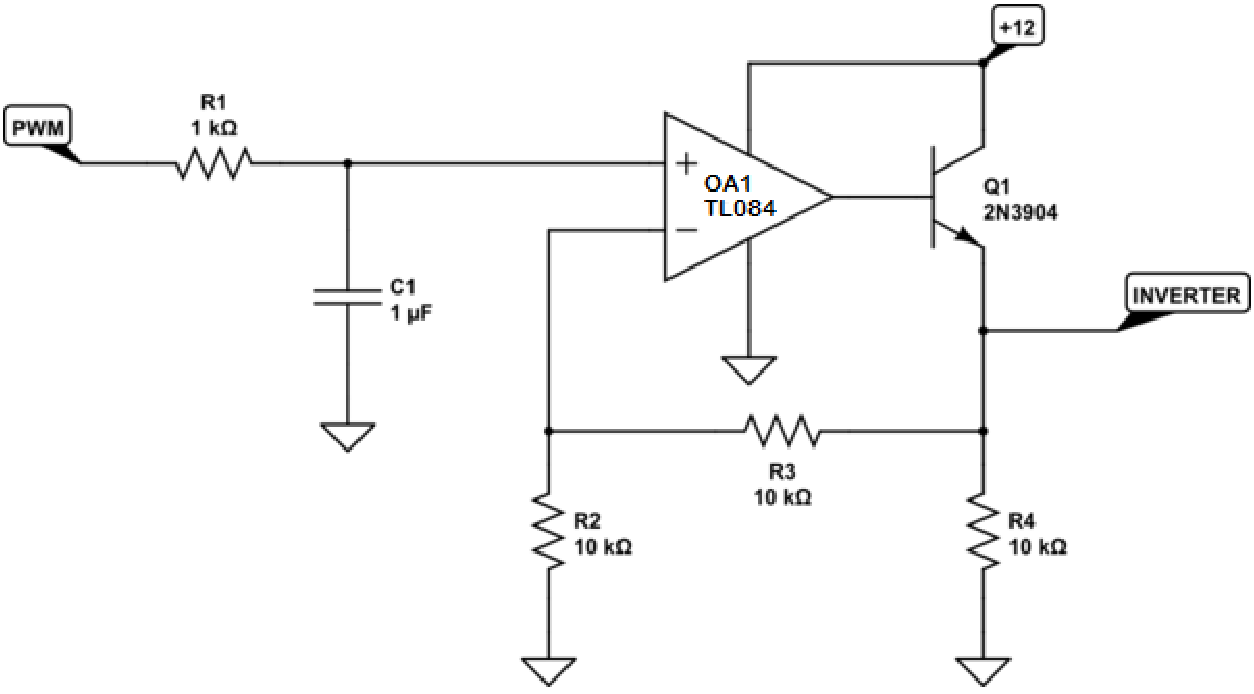


Figure 7.16: $0-5V$ to $0-10V$ conversion Circuit

2. **Input signal 0-5 volt, Output -10 to 10 volt**

   We use Operational Amplifier TL084, some Resistors and Capacitor to create this circuit, which shown in figure 7.17.

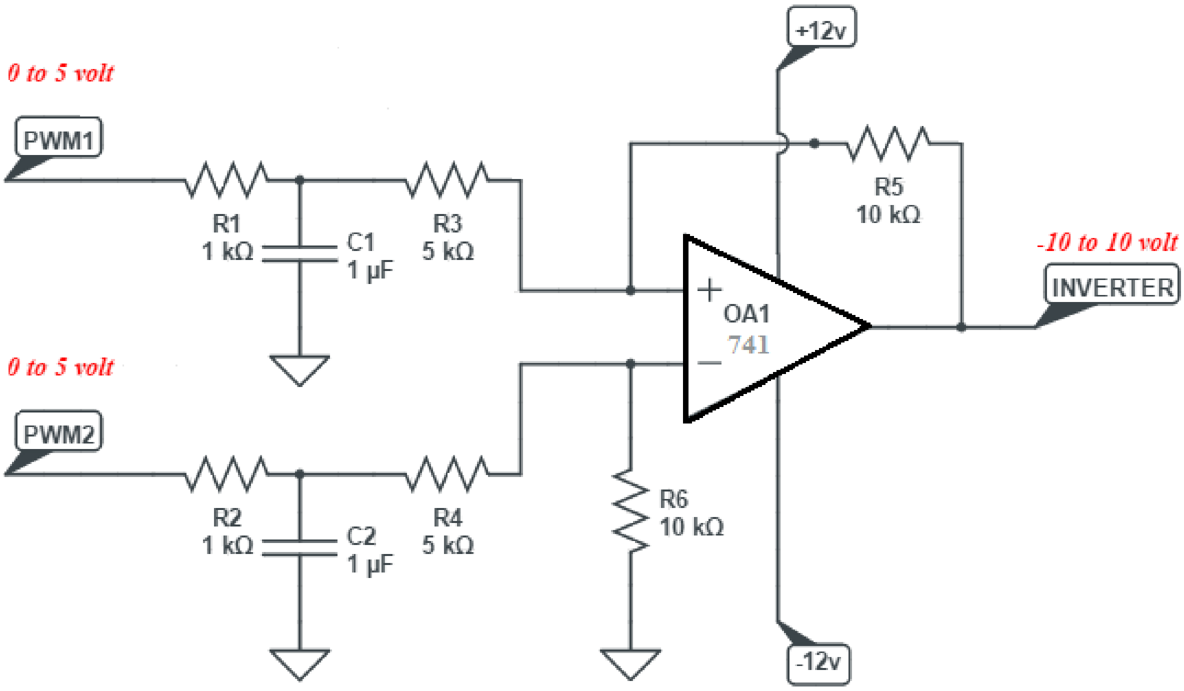


Figure 7.17: $0-5\,V$ to $-10\,to\,10\,V$ conversion Circuit

3. **Over all circuits and connections**

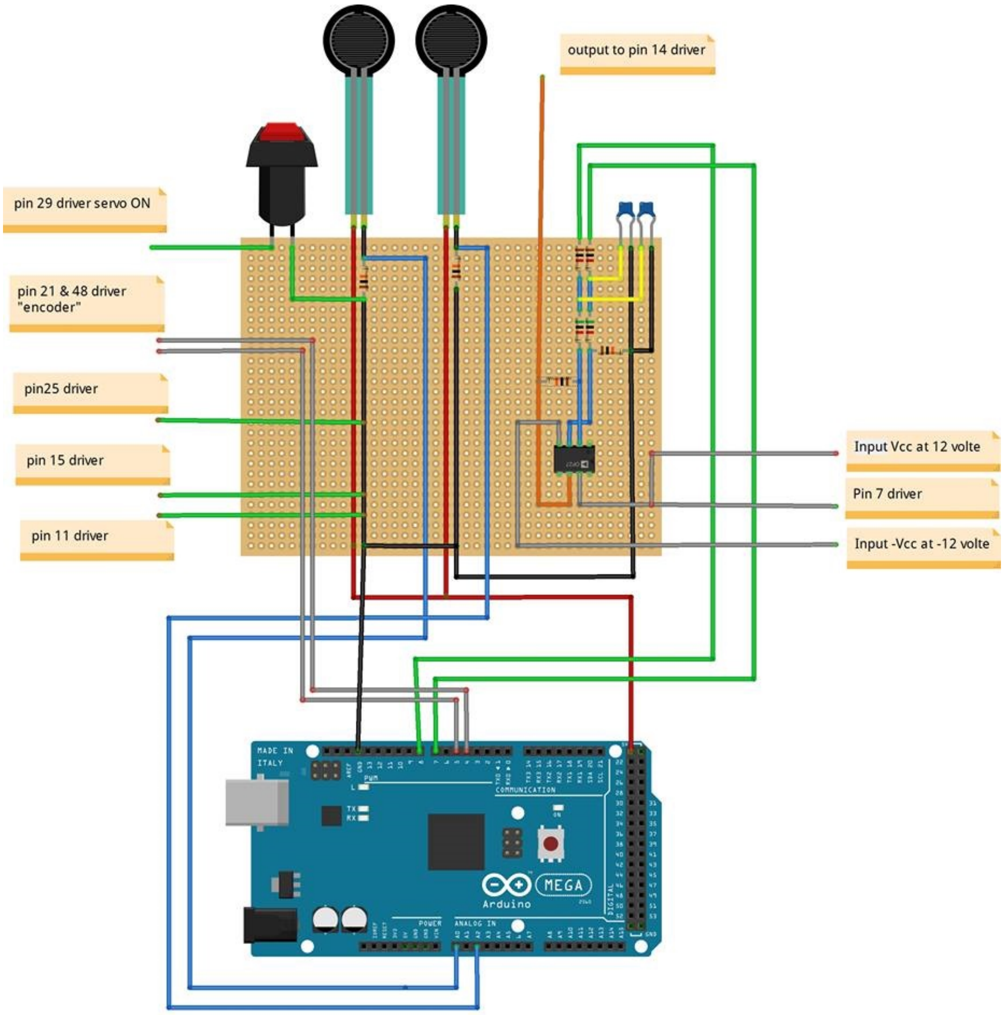


Figure 7.18: Over all circuits and connections

The operational amplifier *OP AMP* 741 circuit is expanded in figure 7.17, and the force sensors circuit is in figure 7.12 and the resistance value set as color system with 10 K.

7.4.6 Controller

In order to obtain high quality, performance and high frequency, which founded in Arduino Mega, We used tow controller due to the large memory which occupied by *MatLab* simulink and wrong sensor read. so we chose one Arduino to read sensor data, and the other one its to install *MatLab* simulink control on it.

So we read sensor data from first one and send data to second controller to control our system. The figure 7.19 shows Arduino Mega and some of specification of it.

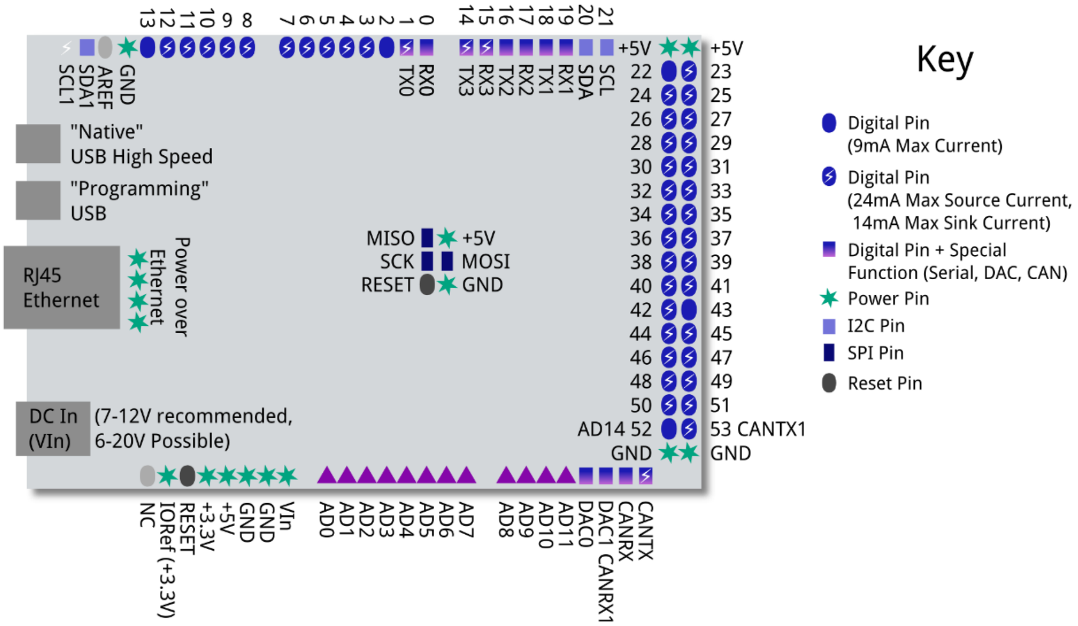


Figure 7.19: Arduino Mega and some of its specification

7.5 Prototype Control

The control in these section is made to verify the project requirement, which we talked about in introduction section. At first we use *LabVIEW* program to make a test for torque and velocity control. Then we made the mathematical modelling, found the state space matrices and gains. Then we made a control block diagram. Then we made a suitable controller to satisfy control requirements. Then we use MatLab to make a Simulink controller, all of these program showed below.

7.5.1 Control Pre Tests

Before start with implement the controller block diagram even on *LabVIEW* or *MatLab* simulink, based on Arduino, we made some tests. These tests were made to make sure that the torque and velocity control are work correctly, also to be familiar with the servomotor inputs and outputs, how related with each other and whats is the suitable signs to make this control.

At first we run the motor in trial mode. then we run it in velocity control mode and torque control mode and control these variable through potentiometers to control the value of torque and speed, which vary from $(-10\,to\,10\,V)$.

- **In Torque control** every $3\,V$ gives $100\,\%$ of rated torque which equal $0.4\,V$ so every $1\,V$ gives $0.13\,Nm$, so when we input $10\,V$ the driver will give me the maximum continuous torque $1.3\,Nm$. And the minus sign its for direction.

- **In Velocity control** every $1\,V$ gives $300\,rpm$, so when we input $10\,V$ the driver will give me the rated velocity, which equal $3000\,rpm$. And the minus sign its for direction.

Then we made a *LabVIEW* program based on Arduino micro-controller to make the potentiometers task. The program font panel and block diagram shown in figure 7.20 and 7.21. also we used the circuits in section 7.3.5 .

1. Font Panel

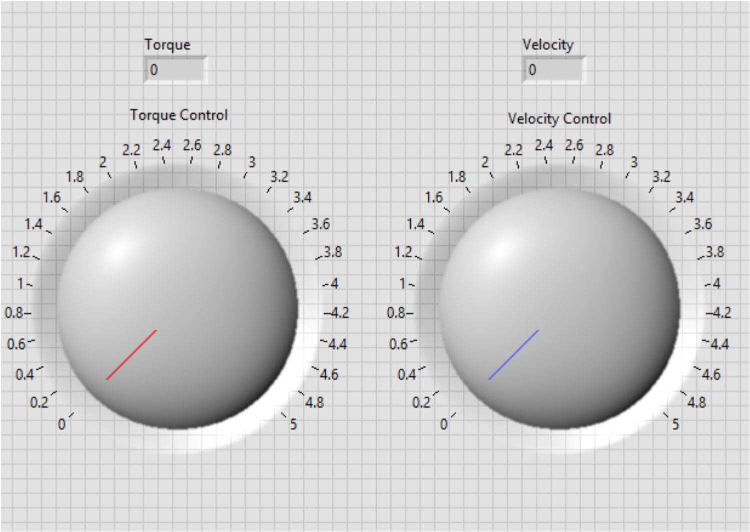


Figure 7.20: *LabVIEW* Font panel for Test

2. Block Diagram

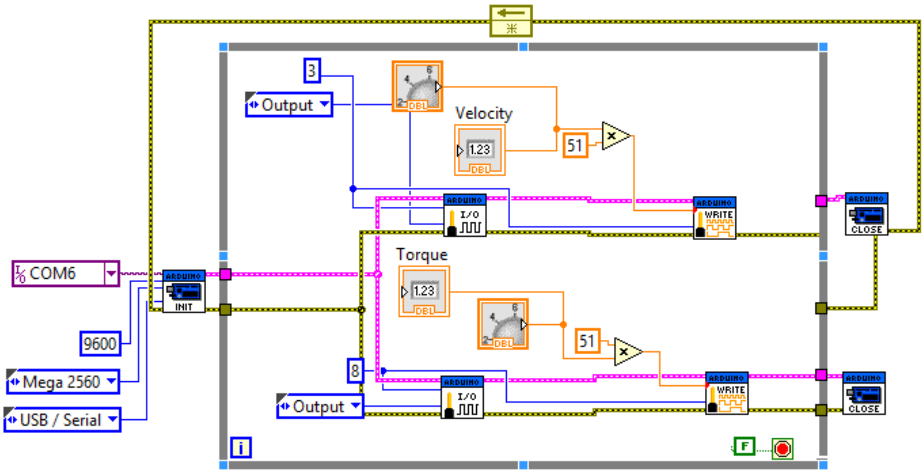


Figure 7.21: *LabVIEW* Block Diagram for Test

7.5.2 Mathematical model and state space matrices

mathematical model through state space model passes through several steps. At firs making free body diagram and find deferential equation (DE) and generate the state space matrices. The second step is linearizing the state space model if that require. Then design the controller and simulate it.

1. **Free body diagram**

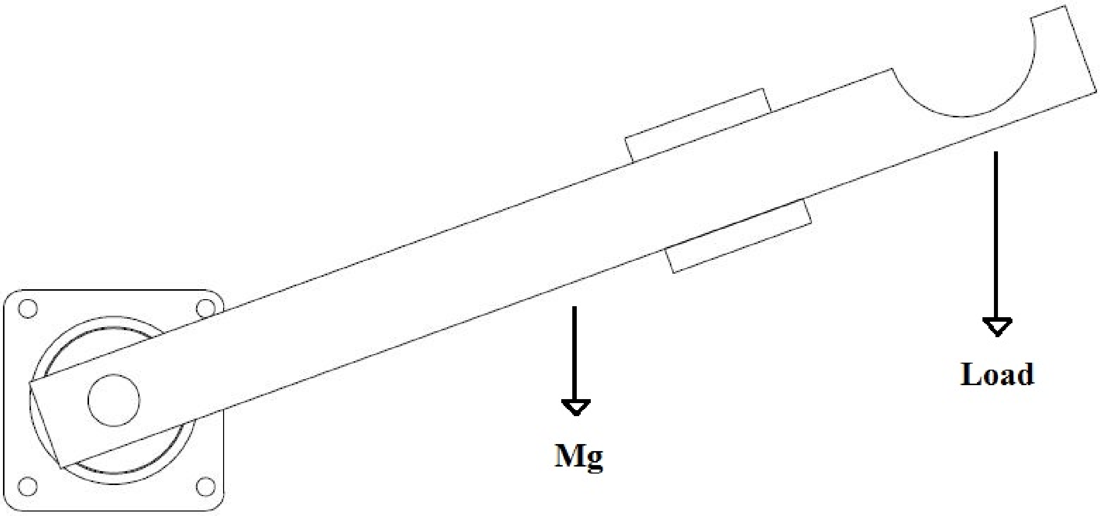


Figure 7.22: Prototype free body diagram

2. **Differential equation**

$$T = J\ddot{\theta}$$

DE :

$$u_2 - mg\frac{L}{2}cos\theta - DgLcos\theta = 0 \tag{7.3}$$

Parameters :

$M = 0.3\,Kg,\ L = 30cm,\ J = 0.002kg.m^2$

And because of the mass of the prototype rod is too small than the mass of disturbance, we can ignore it. and the differential equation will be like this.

$$u - D = J\ddot{\theta} \tag{7.4}$$

3. **state space variable**

$$x_1 = \theta_1 \qquad \dot{x_1} = \dot{\theta}_1 = x_2 \tag{7.5}$$

$$x_2 = \theta_2 \qquad \dot{x_2} = \dot{\theta}_2 = \frac{1}{J}\,u \;-\; \frac{1}{J}\,D \tag{7.6}$$

4. **state space matrices**

$$\dot{x} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} x \;+\; \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} u \;+\; \begin{bmatrix} 0 \\ -\frac{1}{J}\,Dglcos\theta \end{bmatrix} d \tag{7.7}$$

By substituting variable (J_1, J_2)

$$\dot{x} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} x \;+\; \begin{bmatrix} 0 \\ 333 \end{bmatrix} u \;+\; \begin{bmatrix} 0 \\ -333\,Dglcos\theta \end{bmatrix} d \tag{7.8}$$

$$y = \begin{bmatrix} 1\,0 \end{bmatrix} x \tag{7.9}$$

After the state space matrices have been done, the next step is to fill the *MatLab* m-file, and ensure the variable on it,to generate the state feedbake and the tracker gains. Then to make a block diagram and the simulink. the m-file will be in appendix.

7.5.3 Control Block Diagram

To satisfy the control requirement, we made a block diagram to describe the system inputs and outputs and the control process, which made to fulfillment this requirement.

The first step is to read the force sensor sense in N which ranger from $1\,Nto10\,N$, which interned the system by human hand, then we convert this force quantity to angle in degree, in other side the Encoder sense and determine the current angle of the system also in unit of degree. Then the sum point subtract the desired angle, which the motor have to reach it, from the current angle, and enter it to controller.

In controller we have integration element (tracker) in forward, also we have a derivation added to gain in reverse, also we have the disturbance which occurred by the weight and we have to cancel it by the controller, all of them join the sum point and subtract each other, to enter the motor as a torque command, then the motor mechanically convert the torque to angle which feedback the current angle to the system again. The block diagram controller as shown in figure 7.23.

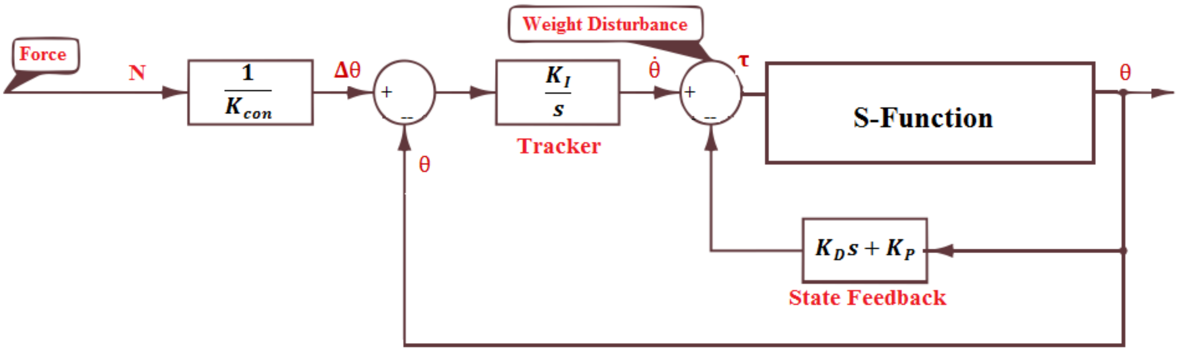


Figure 7.23: Control Block Diagram

7.5.4 Control using *MatLab*

MatLab is multi-paradigm numerical computing environment and fourth-generation programming language. MatLab allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs. Also *MatLab* supports Arduino hardware, language and function, after download and install Arduino support packages for *MatLab*.

The first step of building a controller for any system, is to test this controller before you practice it on you plant, according to this theorem, we made a *MatLab* simulink in simulation mode and showed the response if that be as desired or not, then in next step we test the simulink on exoskeleton prototype when the input is step and ramp, when we show that the response of all of these as required, we test the system by actual input from force sensors, these tests is ordered respectively as follow.

1. **simulation test**

   All part of this test is simulation, the input, output, feedback,control also the disturbance are simulated.

- **Simulink**

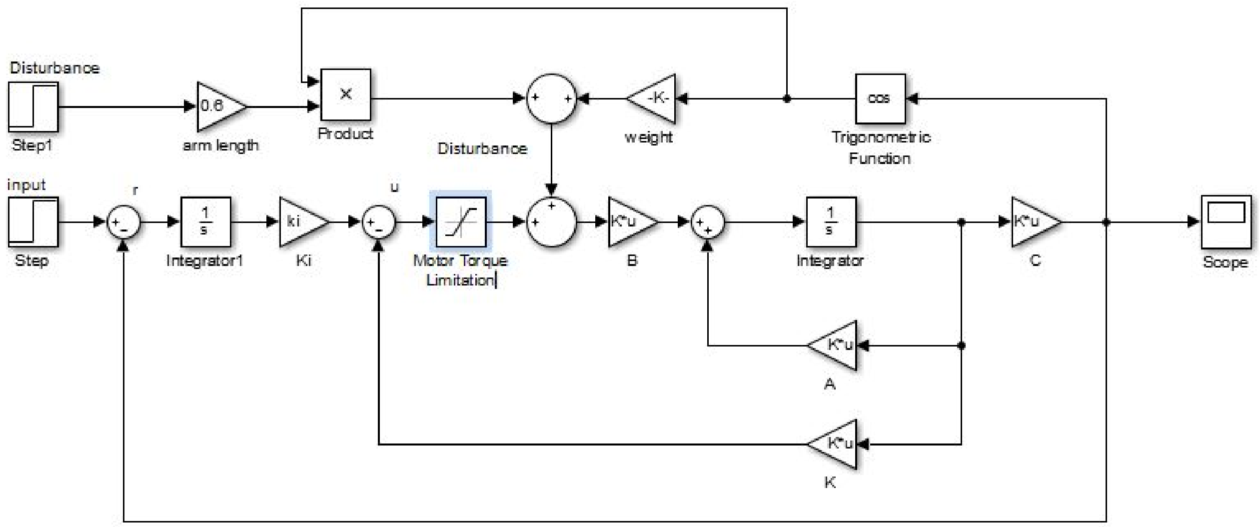


Figure 7.24: Simulation *MatLab* simulink

- **Response without disturbance**

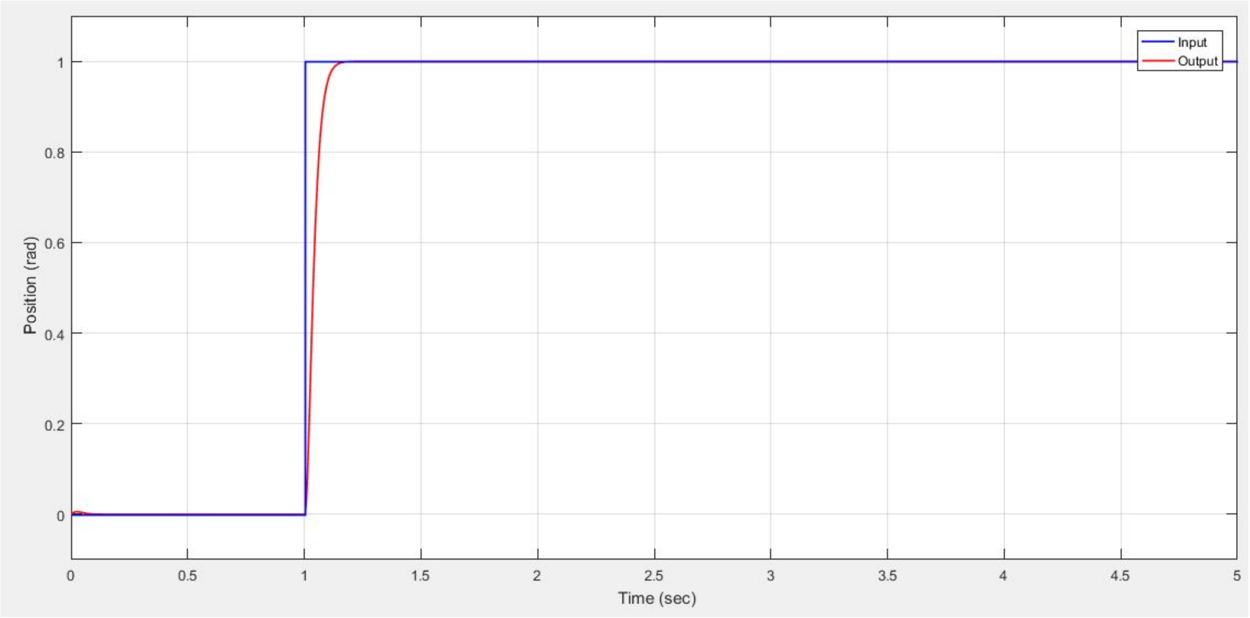


Figure 7.25: Simulation Response with disturbance

- **Response with disturbance**

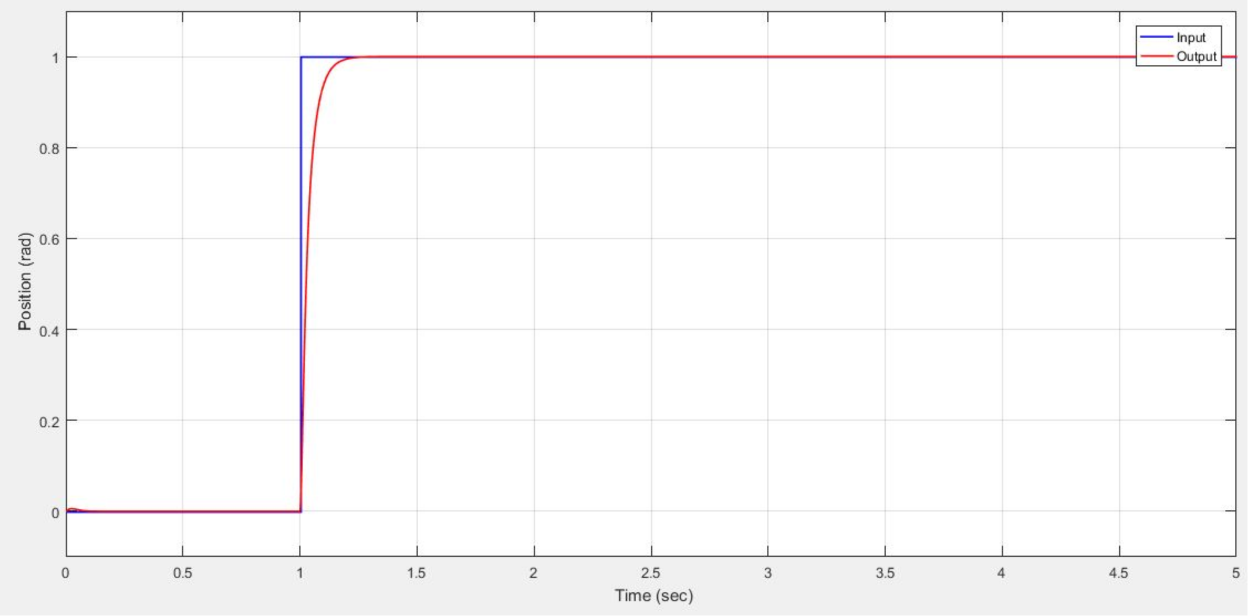


Figure 7.26: Simulation Response without disturbance

As we see with and without the disturbance the arm goes wit step input,but with small delay and 15% overshoot without disturbance, 4% overshot with disturbance, and the simulation simulik is work properly.

2. **step input**

   All part of this test is actual, the output, the feedback, by encoder, the controller by Arduino also the disturbance with 20 *N* are actual, just the input was simulated.

- **Simulink**

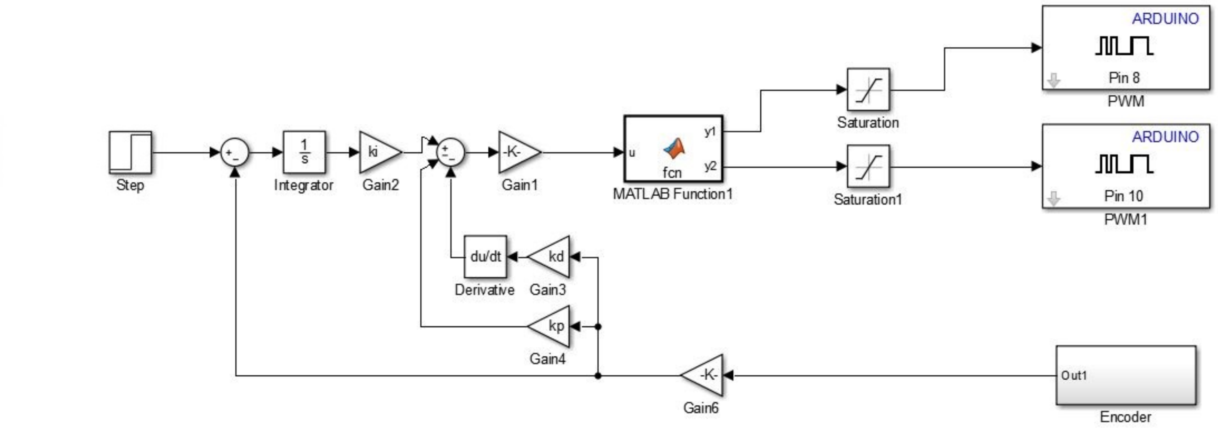


Figure 7.27: *MatLab* simulink with step input

- **Response without disturbance**

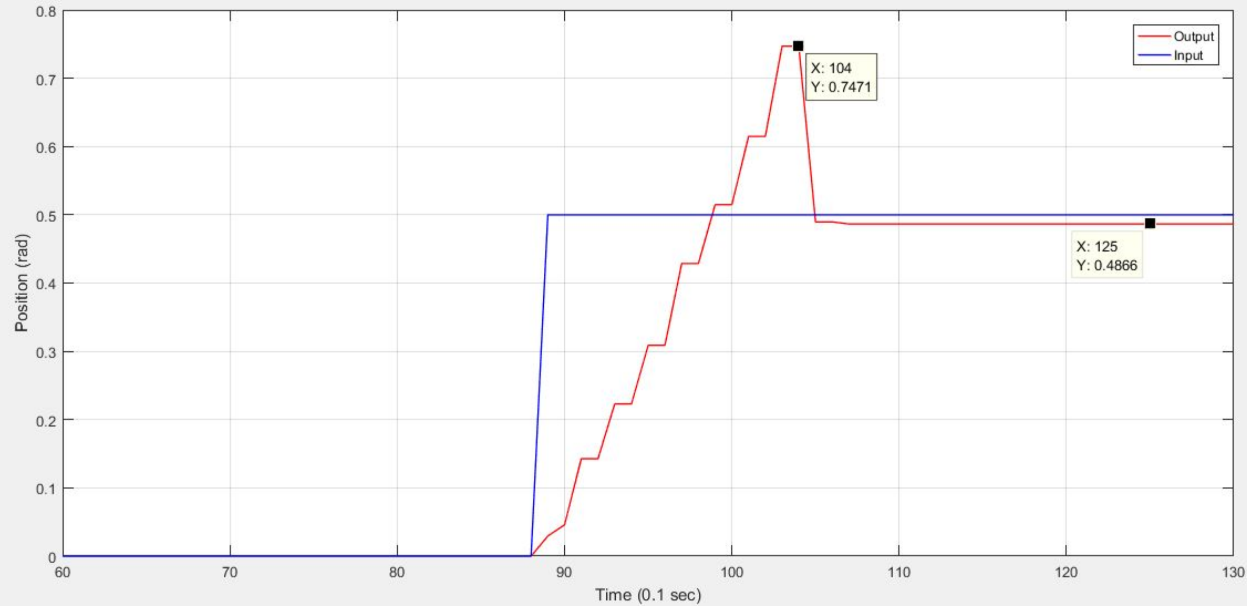


Figure 7.28: Response with simulated step input without disturbance

- **Response with disturbance**

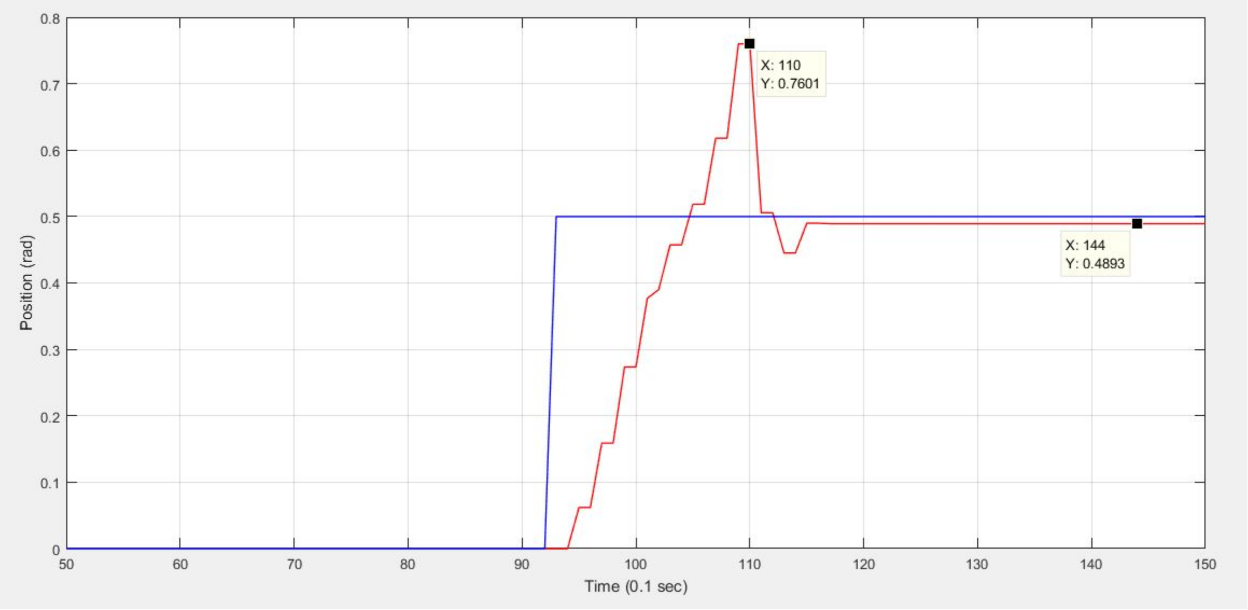


Figure 7.29: Response with simulated step input with disturbance

As we see with and without the disturbance the reposes are the same, and because of the large desired angle which the motor supposed to reach it, the motor goes with overshoot equal 52 %, and the went bake with very small error equal 0.01 *rad*.

3. **Ramp input**

   All part of this test is actual, the output, the feedback, by encoder, the controller by Arduino also the disturbance with 20 N are actual, just the input was simulated.

- **Simulink**

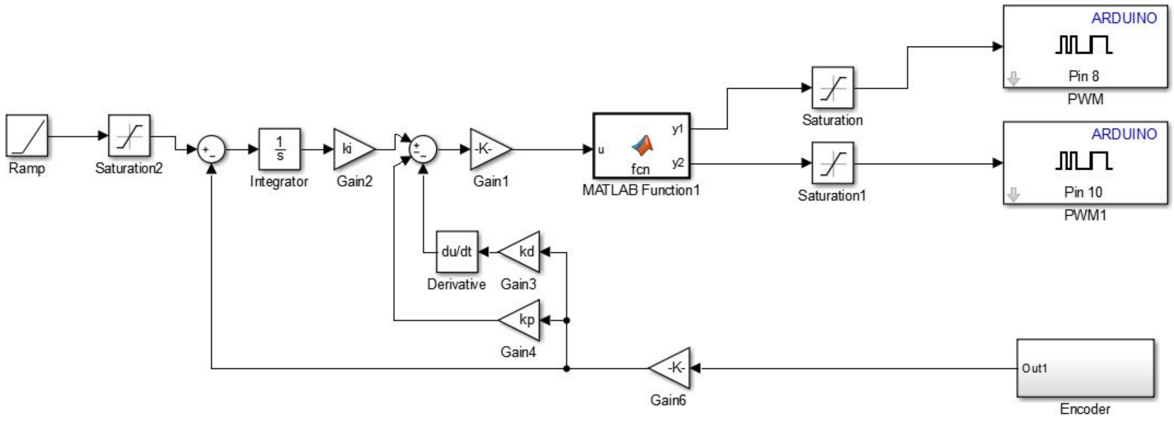


Figure 7.30: *MatLab* simulink with ramp input

- **Response without disturbance**

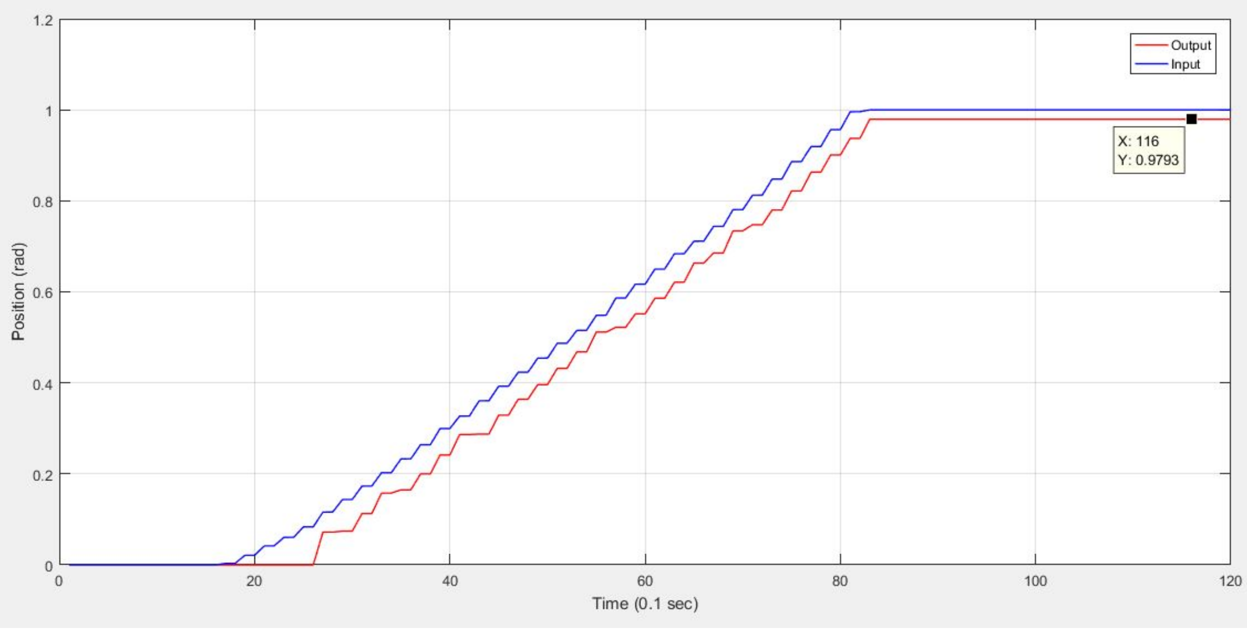


Figure 7.31: Response with simulated ramp input without disturbance

- **Response with disturbance**

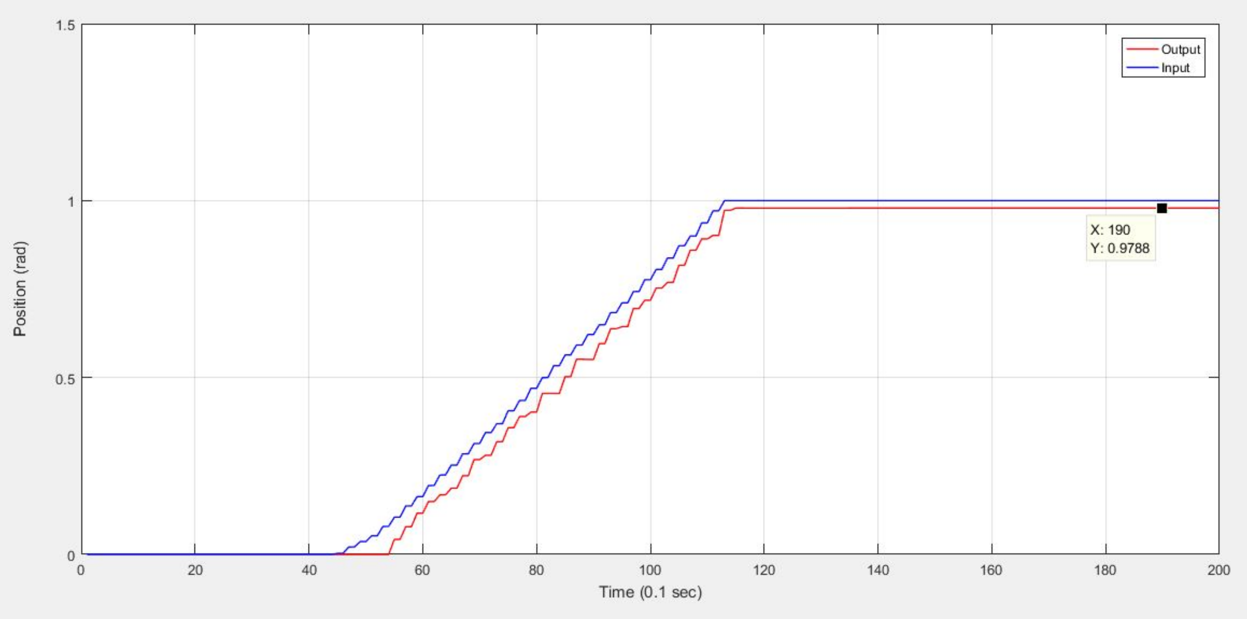


Figure 7.32: Response with simulated ramp input with disturbance

As we see with and without the disturbance the reposes are the same, the motor went with ramp input as desired, but with very small error equal 0.02 *rad*.

4. **Force input**

   All part of this test is actual, the input, output, the feedback, by encoder, the controller by Arduino also the disturbance with 20 *N* are actual.

- **Simulink**

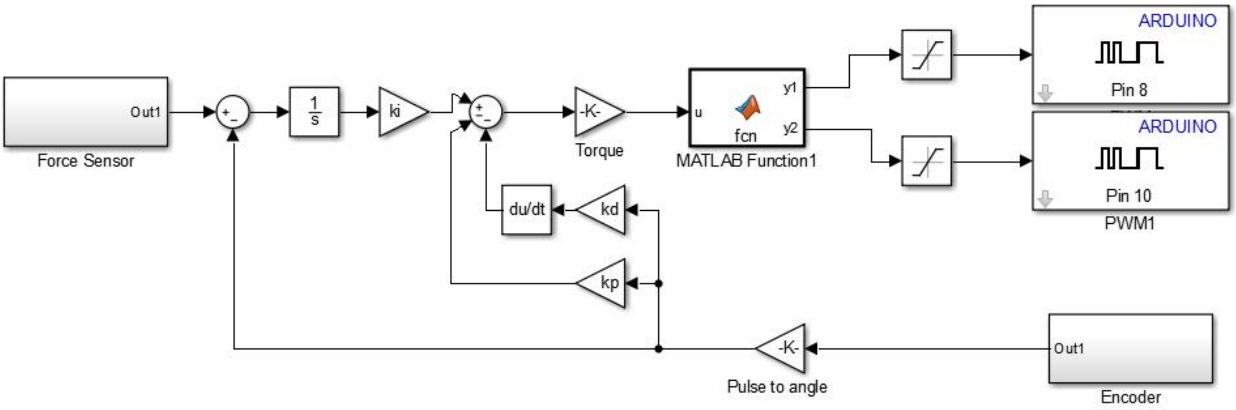


Figure 7.33: *MatLab* simulink with force sensors input

- **Response without disturbance**

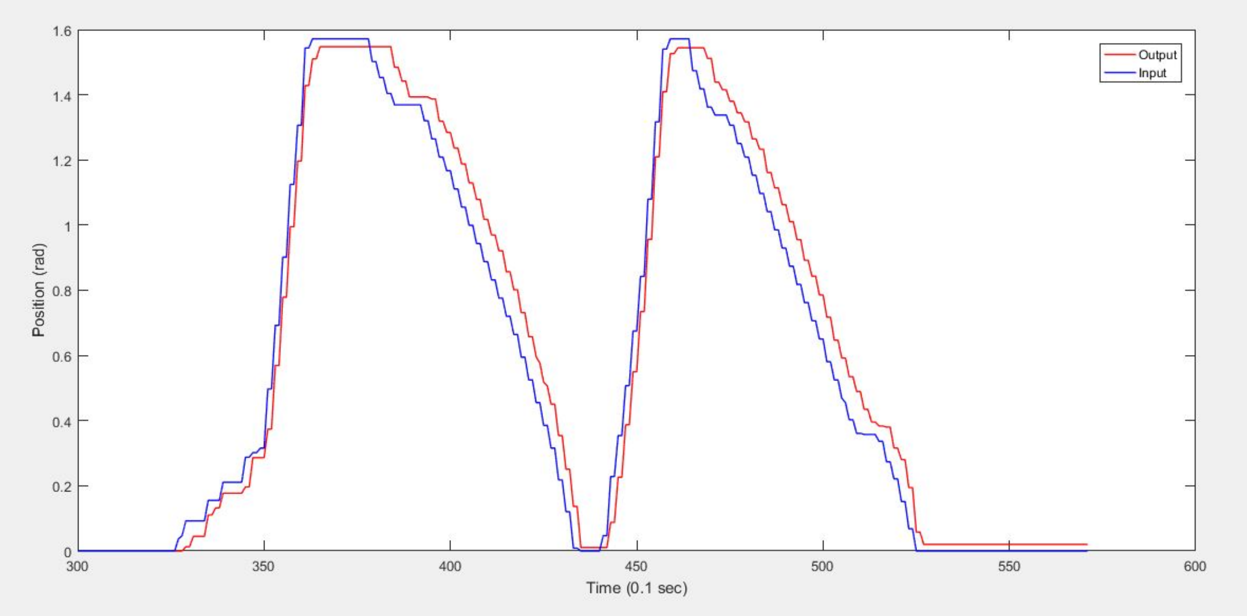


Figure 7.34: Response with force sensors input without disturbance

- **Response with disturbance**

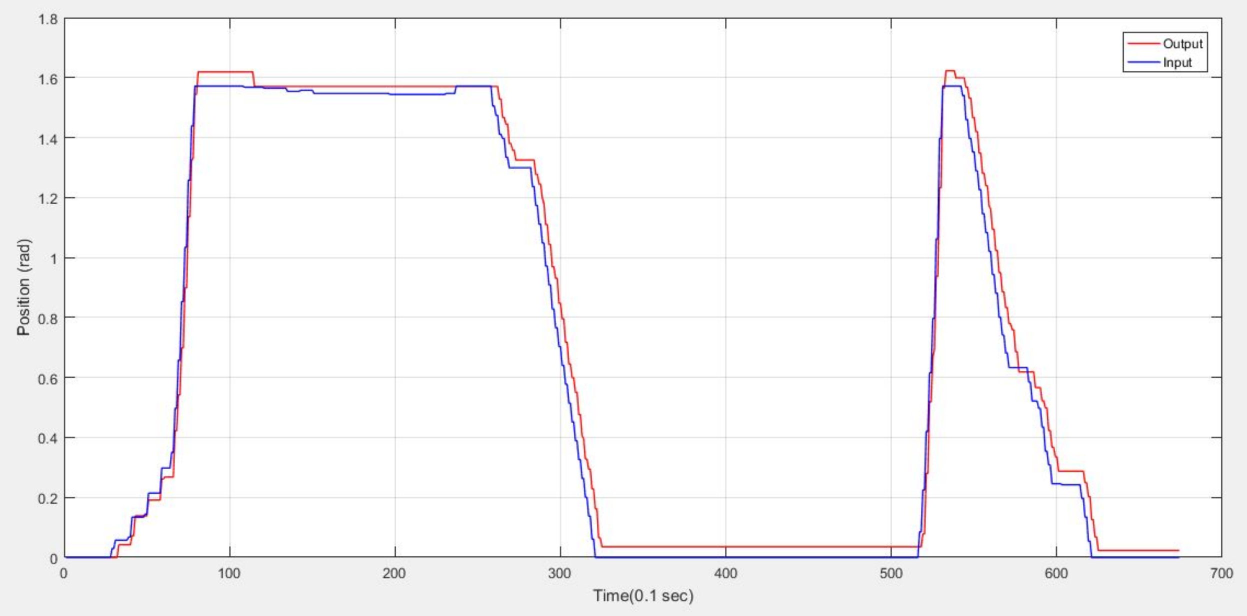


Figure 7.35: Response with force sensors input with disturbance

As we see with and without the disturbance the reposes are the same, the motor went with force sensors inputs wherever they went, but with very small error equal.

Chapter 8

Evaluation and Conclusion

8.1 Introduction

This Chapter provide the project evaluation, conclusion of the Exoskeleton prototype and also future works are found here.

8.2 Evaluation

According to the previous result, and whole work on Exoskeleton prototype, we can evaluate the project. First of all we build the *MatLab* similink, on simulation mode we take the result and analyzed them, and the result have been as proper and satisfy the prototype requirements.

Then we went to experiment work we build the prototype as its deign, we also build *MatLab* similink for experiment mode, and then took the result of its in deferent inputs step,ramp and by force sensors, the result also have been excellent result, and satisfy the prototype requirement.

8.3 Conclusion

After we finished designing, building and analyzing the exoskeleton prototype we conclude these point.

- Before build any prototype, you should design it and check the validity of it.
- We can design and makes stress and deflection analysis for the prototype, using *CATIA* software.

- We should bring a suitable component (motor, mechanism, sensor, controller ...) for the prototype, to make your work easier.
- We can use torque and position torque together, by chose a motor and motor drive has a torque control mode.
- We can build *MatLab* similink and connect it with Arduino microcontroller, by download and install the Arduino *MatLab* package, and the same thing for *LabVIEW* by install the LabVIEW interface with Arduino.

8.4 Future work

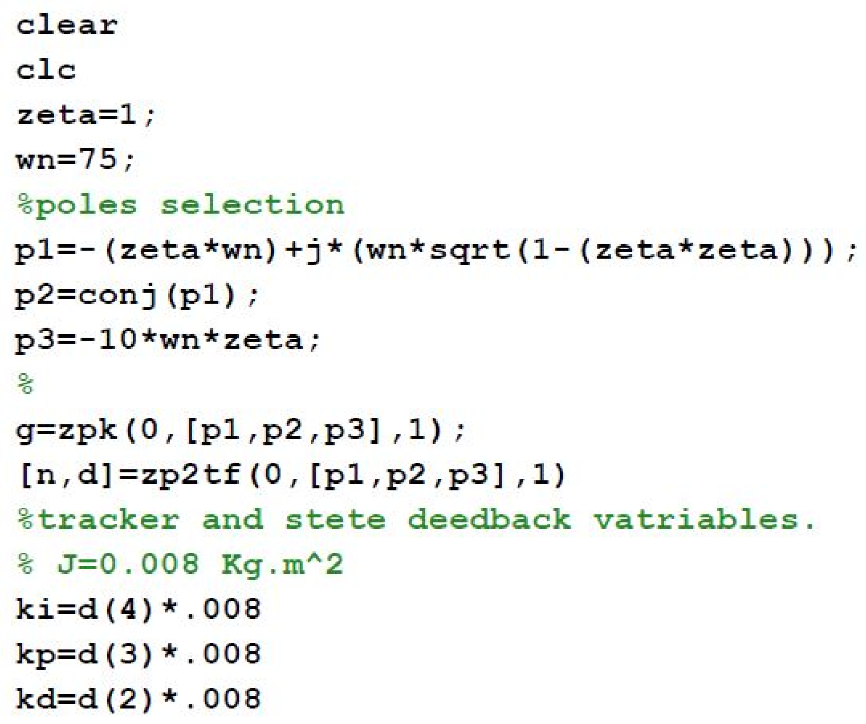
This project can be finished by chain work. At first as we see by design the Upper-Limn Exoskeleton with force dissipation to earth, then build a prototype to check the control validity, and we can use the prototype for educational processes. The next step is to build a two-degree of freedom Upper-Limb Exoskeleton. Then Design and Build Lower-Limb Exoskeleton. Then makes a compensation of them to build Full Exoskeleton. And that work is possible for Palestine Polytechnic University student, to compete With large production companies.

Appendices

Appendix A

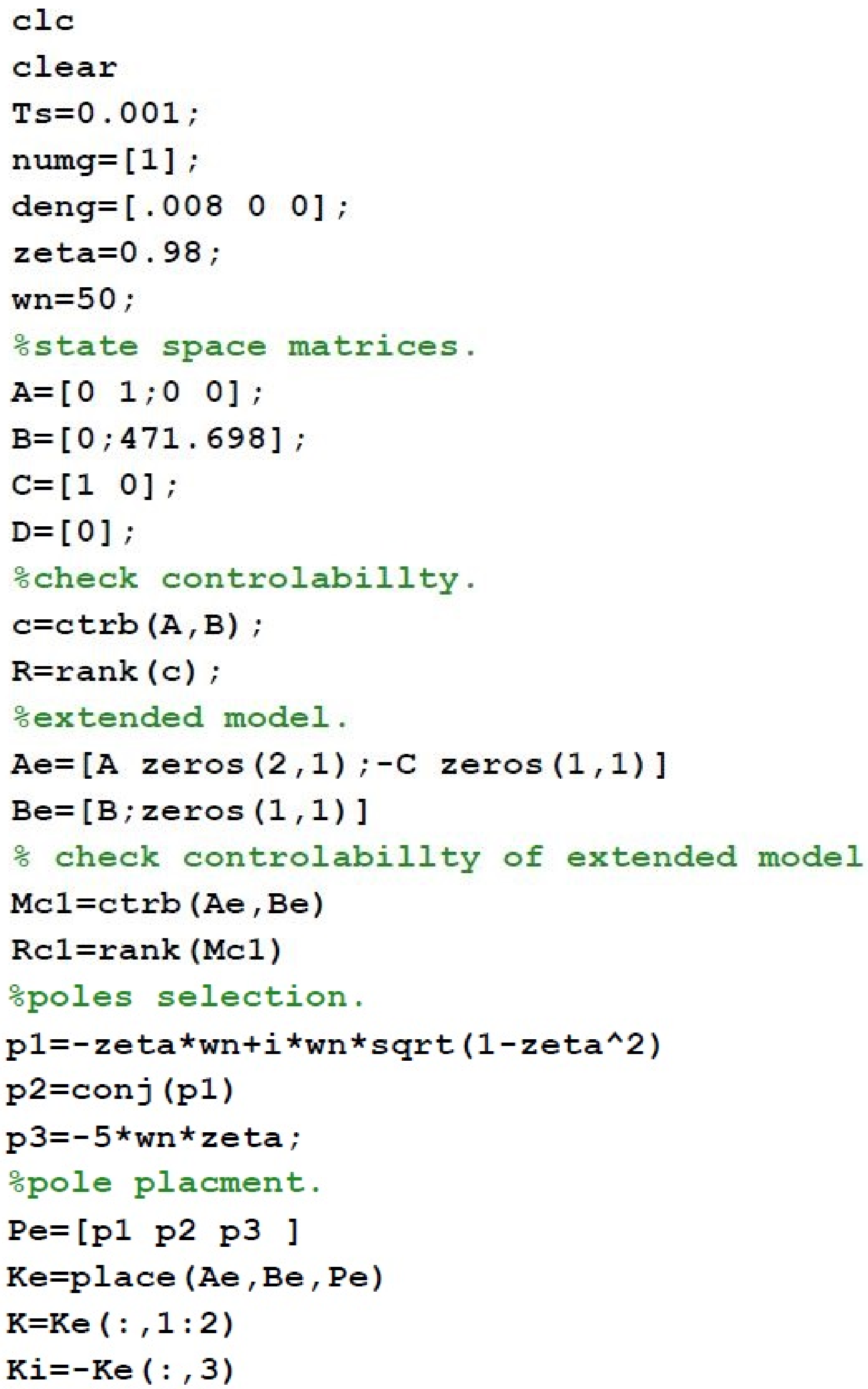
MatLab m-files

- ***MatLab* m-files for experiment 1DoF prototype**



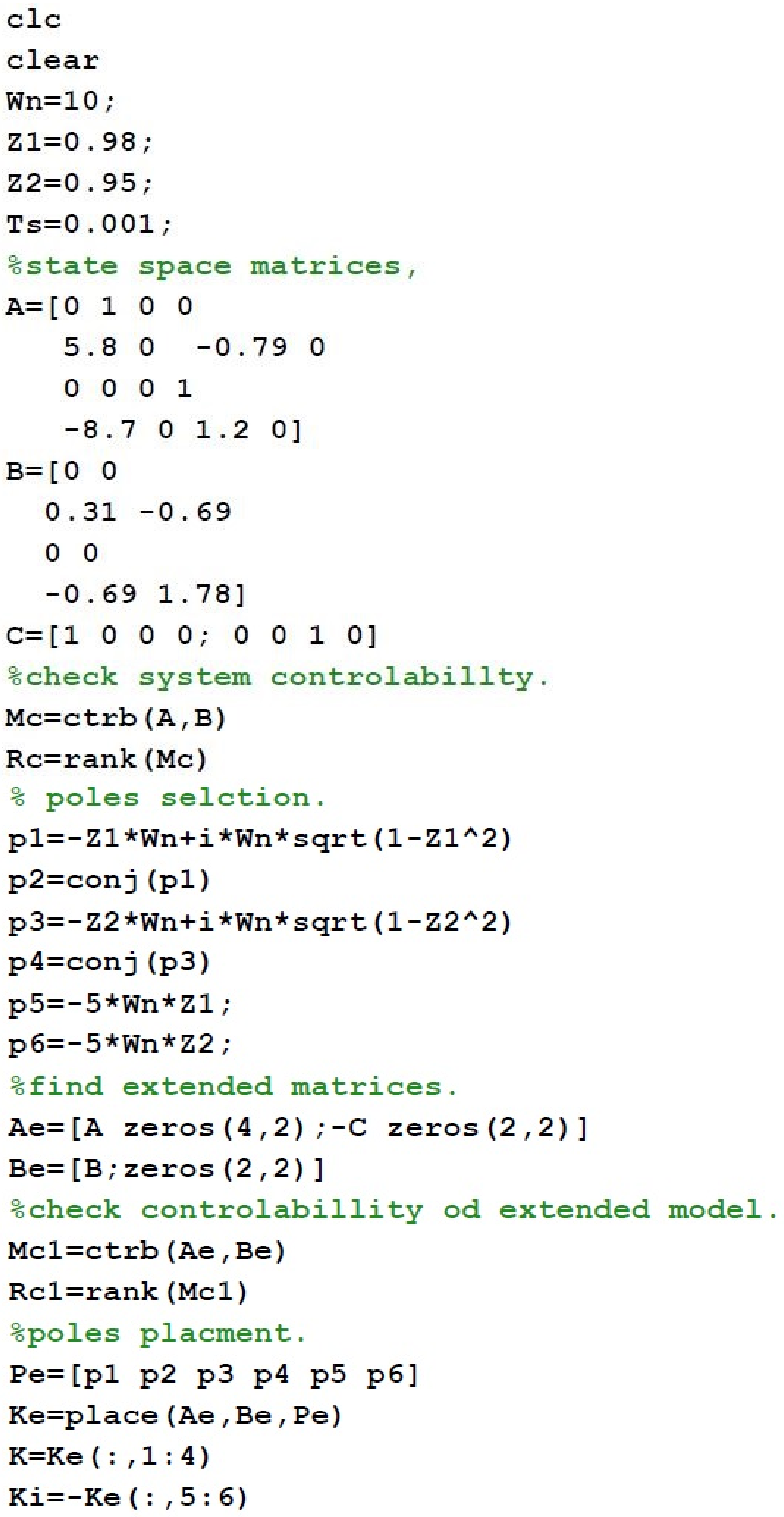
```
clear
clc
zeta=1;
wn=75;
%poles selection
p1=-(zeta*wn)+j*(wn*sqrt(1-(zeta*zeta)));
p2=conj(p1);
p3=-10*wn*zeta;
%
g=zpk(0,[p1,p2,p3],1);
[n,d]=zp2tf(0,[p1,p2,p3],1)
%tracker and stete deedback vatriables.
% J=0.008 Kg.m^2
ki=d(4)*.008
kp=d(3)*.008
kd=d(2)*.008
```

- ***MatLab* m-files for simulation 1DoF prototype**



```
clc
clear
Ts=0.001;
numg=[1];
deng=[.008 0 0];
zeta=0.98;
wn=50;
%state space matrices.
A=[0 1;0 0];
B=[0;471.698];
C=[1 0];
D=[0];
%check controlabillty.
c=ctrb(A,B);
R=rank(c);
%extended model.
Ae=[A zeros(2,1);-C zeros(1,1)]
Be=[B;zeros(1,1)]
% check controlabillty of extended model
Mc1=ctrb(Ae,Be)
Rc1=rank(Mc1)
%poles selection.
p1=-zeta*wn+i*wn*sqrt(1-zeta^2)
p2=conj(p1)
p3=-5*wn*zeta;
%pole placment.
Pe=[p1 p2 p3 ]
Ke=place(Ae,Be,Pe)
K=Ke(:,1:2)
Ki=-Ke(:,3)
```

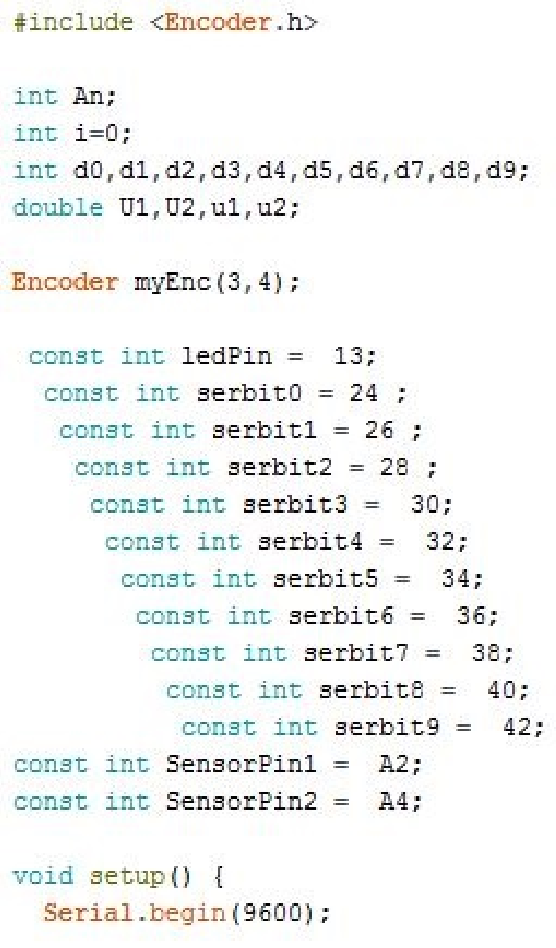
- ***MatLab* m-files for full exoskeleton**



```
clc
clear
Wn=10;
Z1=0.98;
Z2=0.95;
Ts=0.001;
%state space matrices,
A=[0 1 0 0
   5.8 0  -0.79 0
   0 0 0 1
   -8.7 0 1.2 0]
B=[0 0
  0.31 -0.69
  0 0
  -0.69 1.78]
C=[1 0 0 0; 0 0 1 0]
%check system controlabillty.
Mc=ctrb(A,B)
Rc=rank(Mc)
% poles selction.
p1=-Z1*Wn+i*Wn*sqrt(1-Z1^2)
p2=conj(p1)
p3=-Z2*Wn+i*Wn*sqrt(1-Z2^2)
p4=conj(p3)
p5=-5*Wn*Z1;
p6=-5*Wn*Z2;
%find extended matrices.
Ae=[A zeros(4,2);-C zeros(2,2)]
Be=[B;zeros(2,2)]
%check controlabillity od extended model.
Mc1=ctrb(Ae,Be)
Rc1=rank(Mc1)
%poles placment.
Pe=[p1 p2 p3 p4 p5 p6]
Ke=place(Ae,Be,Pe)
K=Ke(:,1:4)
Ki=-Ke(:,5:6)
```

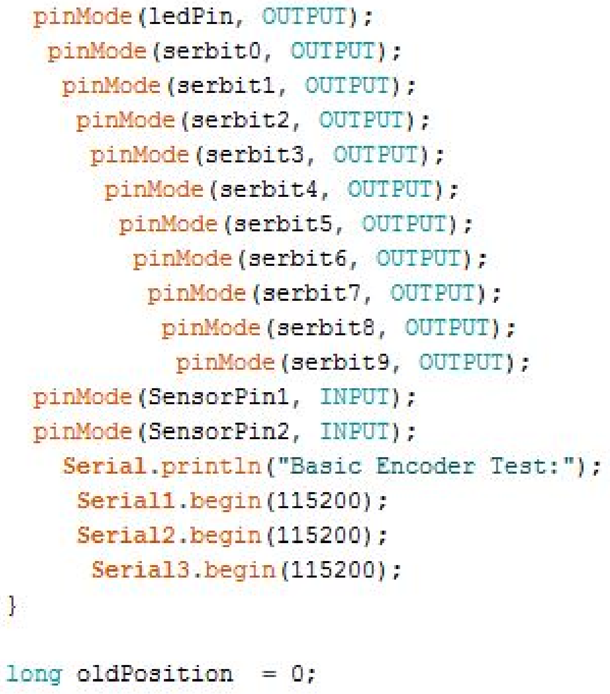
Appendix B

Arduino Code



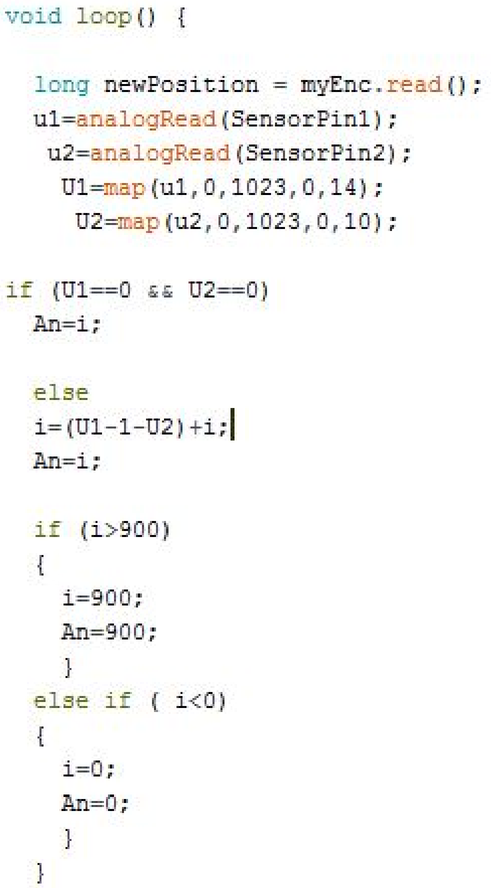
```
#include <Encoder.h>

int An;
int i=0;
int d0,d1,d2,d3,d4,d5,d6,d7,d8,d9;
double U1,U2,u1,u2;

Encoder myEnc(3,4);

 const int ledPin =  13;
  const int serbit0 = 24 ;
   const int serbit1 = 26 ;
    const int serbit2 = 28 ;
     const int serbit3 =  30;
      const int serbit4 =  32;
       const int serbit5 =  34;
        const int serbit6 =  36;
         const int serbit7 =  38;
          const int serbit8 =  40;
           const int serbit9 =  42;
const int SensorPin1 =  A2;
const int SensorPin2 =  A4;

void setup() {
  Serial.begin(9600);
```

Figure B.1: initialization



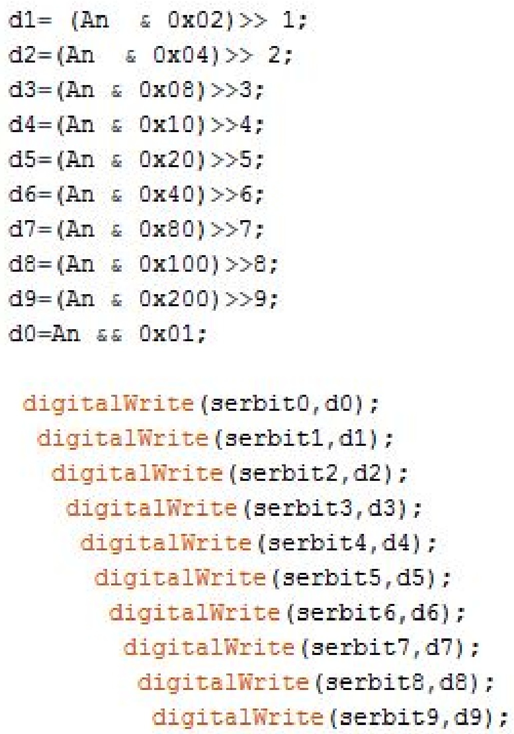
```
  pinMode(ledPin, OUTPUT);
   pinMode(serbit0, OUTPUT);
    pinMode(serbit1, OUTPUT);
     pinMode(serbit2, OUTPUT);
      pinMode(serbit3, OUTPUT);
       pinMode(serbit4, OUTPUT);
        pinMode(serbit5, OUTPUT);
         pinMode(serbit6, OUTPUT);
          pinMode(serbit7, OUTPUT);
           pinMode(serbit8, OUTPUT);
            pinMode(serbit9, OUTPUT);
  pinMode(SensorPin1, INPUT);
  pinMode(SensorPin2, INPUT);
    Serial.println("Basic Encoder Test:");
     Serial1.begin(115200);
     Serial2.begin(115200);
      Serial3.begin(115200);
}

long oldPosition  = 0;
```

Figure B.2: Define Pin mode



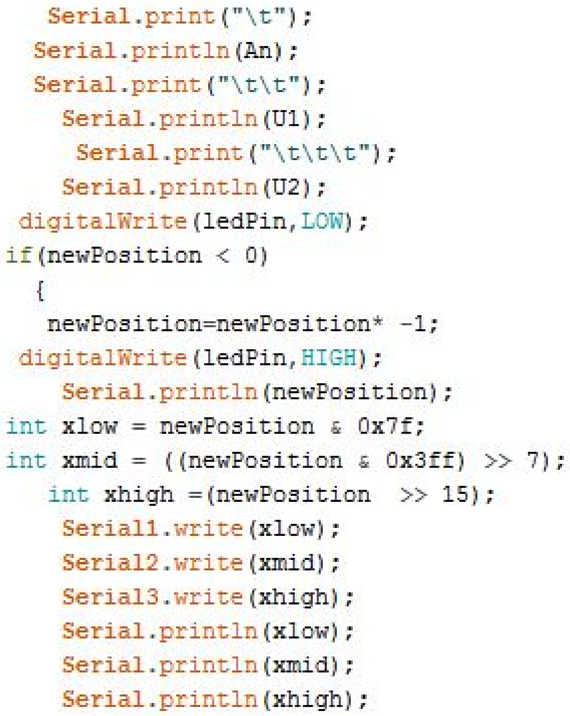
```
void loop() {

  long newPosition = myEnc.read();
  u1=analogRead(SensorPin1);
   u2=analogRead(SensorPin2);
    U1=map(u1,0,1023,0,14);
     U2=map(u2,0,1023,0,10);

if (U1==0 && U2==0)
  An=i;

  else
  i=(U1-1-U2)+i;
  An=i;

  if (i>900)
  {
    i=900;
    An=900;
    }
  else if ( i<0)
  {
    i=0;
    An=0;
    }
  }
```

Figure B.3: Force sensors code



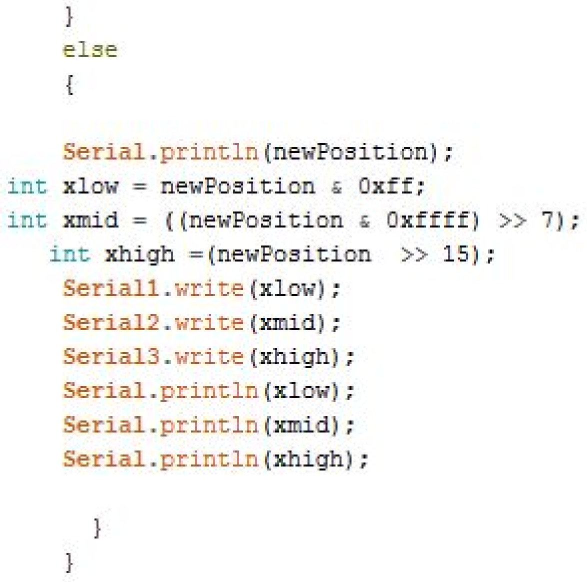
```
d1= (An  & 0x02)>> 1;
d2=(An  & 0x04)>> 2;
d3=(An & 0x08)>>3;
d4=(An & 0x10)>>4;
d5=(An & 0x20)>>5;
d6=(An & 0x40)>>6;
d7=(An & 0x80)>>7;
d8=(An & 0x100)>>8;
d9=(An & 0x200)>>9;
d0=An && 0x01;

 digitalWrite(serbit0,d0);
  digitalWrite(serbit1,d1);
   digitalWrite(serbit2,d2);
    digitalWrite(serbit3,d3);
     digitalWrite(serbit4,d4);
      digitalWrite(serbit5,d5);
       digitalWrite(serbit6,d6);
        digitalWrite(serbit7,d7);
         digitalWrite(serbit8,d8);
          digitalWrite(serbit9,d9);
```

Figure B.4: Force sensors serial send with 10-bit



```
   Serial.print("\t");
  Serial.println(An);
  Serial.print("\t\t");
    Serial.println(U1);
     Serial.print("\t\t\t");
    Serial.println(U2);
 digitalWrite(ledPin,LOW);
if(newPosition < 0)
  {
   newPosition=newPosition* -1;
 digitalWrite(ledPin,HIGH);
    Serial.println(newPosition);
int xlow = newPosition & 0x7f;
int xmid = ((newPosition & 0x3ff) >> 7);
   int xhigh =(newPosition  >> 15);
    Serial1.write(xlow);
    Serial2.write(xmid);
    Serial3.write(xhigh);
    Serial.println(xlow);
    Serial.println(xmid);
    Serial.println(xhigh);
```

Figure B.5: Force sensors and encoder serial monitor



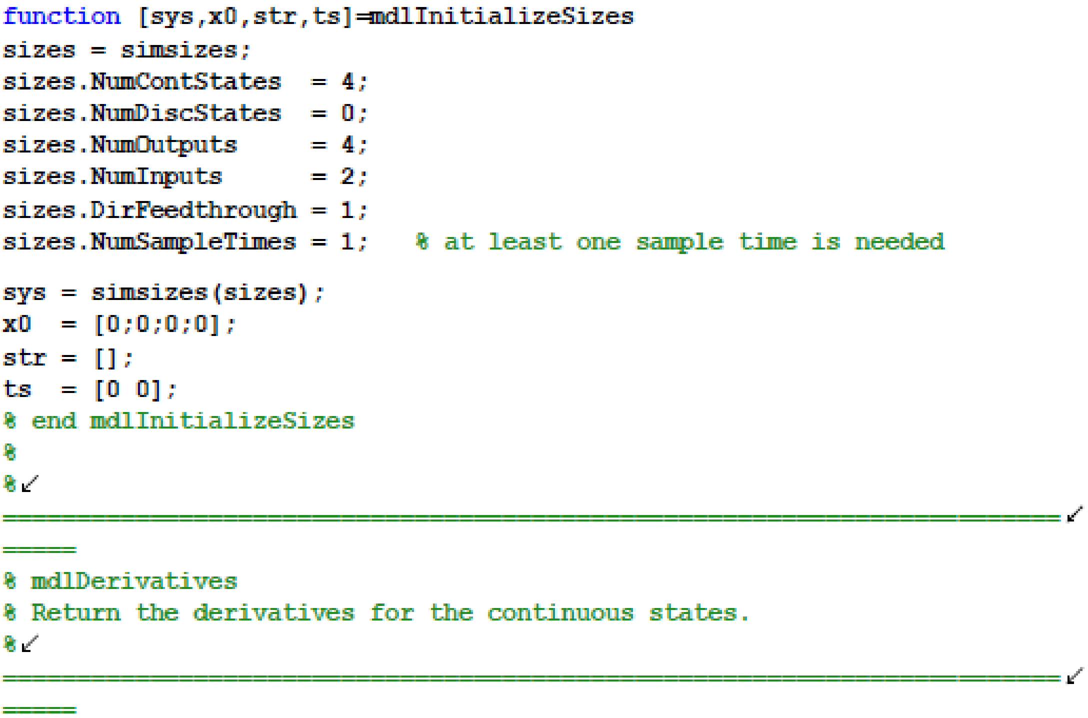
```
    }
    else
    {

    Serial.println(newPosition);
int xlow = newPosition & 0xff;
int xmid = ((newPosition & 0xffff) >> 7);
   int xhigh =(newPosition  >> 15);
    Serial1.write(xlow);
    Serial2.write(xmid);
    Serial3.write(xhigh);
    Serial.println(xlow);
    Serial.println(xmid);
    Serial.println(xhigh);

      }
    }
```

Figure B.6: Encoder send by 24-bit serial

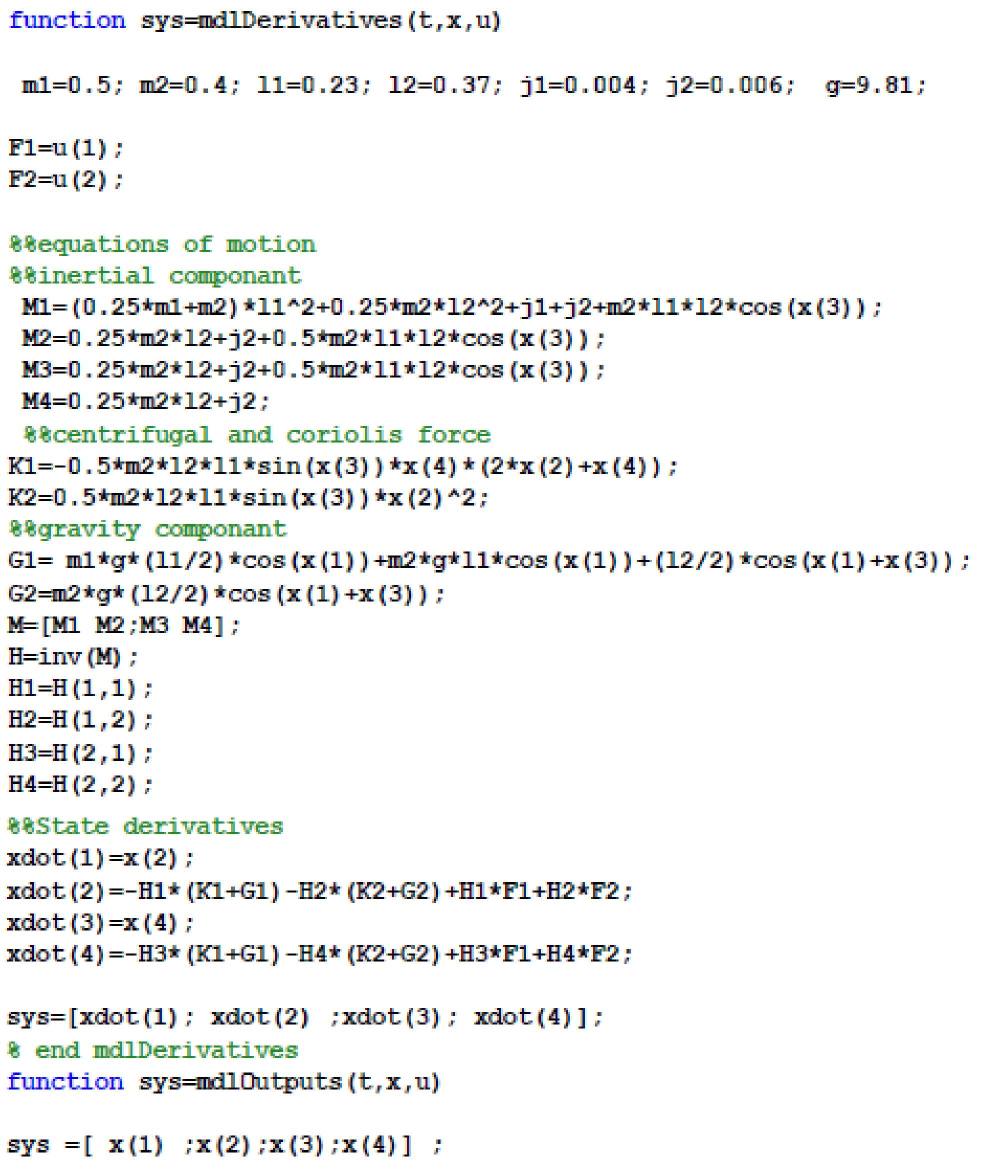
Appendix C

S-Function (Exodynamics file)

```matlab
function [sys,x0,str,ts] = Exodynamics(t,x,u,flag)%%

%%2DOF Exoskeleton nonlinear dynamics
switch flag,
  %%%%%%%%%%%%%%%%%%
  % Initialization %
  %%%%%%%%%%%%%%%%%%
  case 0,
    [sys,x0,str,ts]=mdlInitializeSizes;
  %%%%%%%%%%%%%%%
  % Derivatives %
  %%%%%%%%%%%%%%%
  case 1,
    sys=mdlDerivatives(t,x,u);

  %%%%%%%%%%%
  % Outputs %
  %%%%%%%%%%%
  case 3,
    sys=mdlOutputs(t,x,u);
  %%%%%%%%%%%%%%%%%%%
  % Unhandled flags %
  %%%%%%%%%%%%%%%%%%%
  case { 2, 4, 9 },
    sys = [];
  %%%%%%%%%%%%%%%%%%%%
  % Unexpected flags %
  %%%%%%%%%%%%%%%%%%%%
  otherwise
    DAStudio.error('Simulink:blocks:unhandledFlag', num2str(flag));
end
```



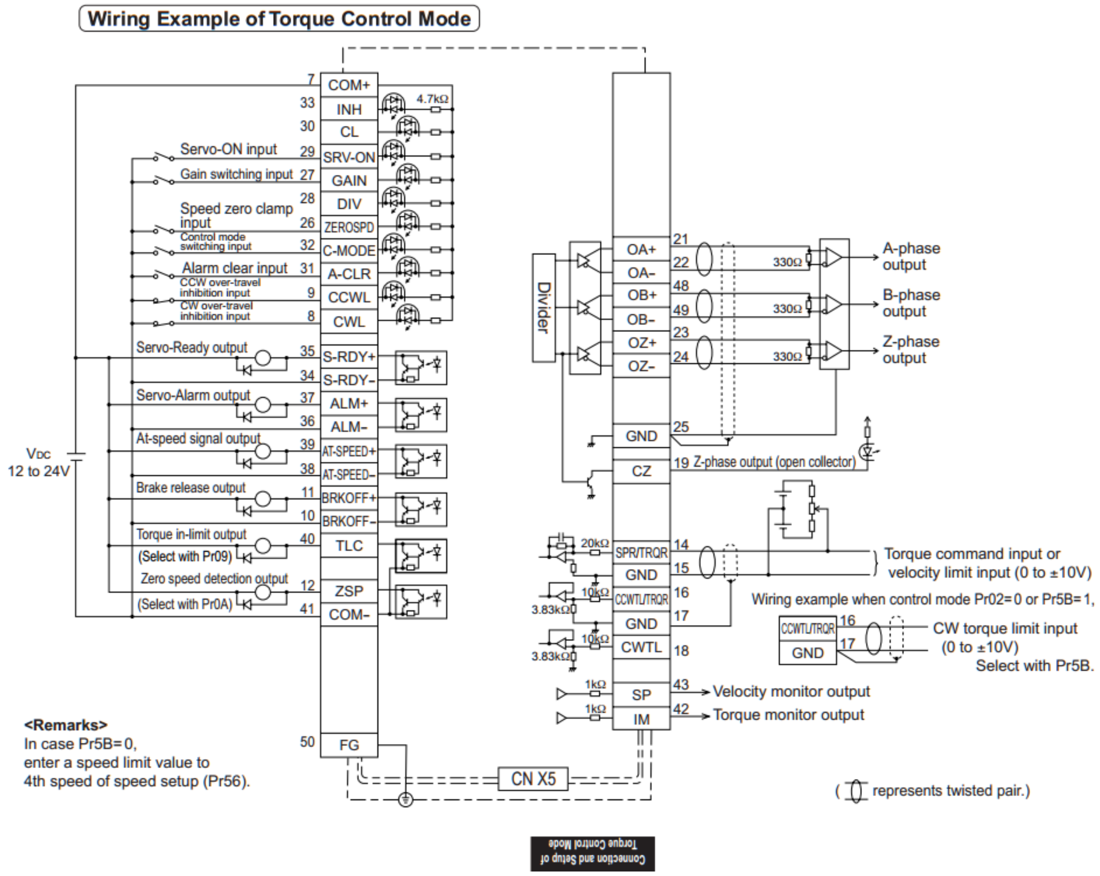
```
function [sys,x0,str,ts]=mdlInitializeSizes
sizes = simsizes;
sizes.NumContStates  = 4;
sizes.NumDiscStates  = 0;
sizes.NumOutputs     = 4;
sizes.NumInputs      = 2;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;   % at least one sample time is needed
sys = simsizes(sizes);
x0  = [0;0;0;0];
str = [];
ts  = [0 0];
% end mdlInitializeSizes
%
%↙
========================================================================↙
=====
% mdlDerivatives
% Return the derivatives for the continuous states.
%↙
========================================================================↙
=====
```

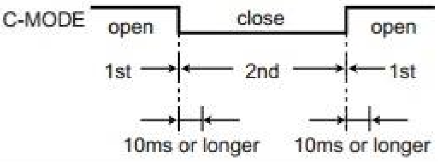


```
function sys=mdlDerivatives(t,x,u)

 m1=0.5; m2=0.4; l1=0.23; l2=0.37; j1=0.004; j2=0.006;  g=9.81;

F1=u(1);
F2=u(2);

%%equations of motion
%%inertial componant
 M1=(0.25*m1+m2)*l1^2+0.25*m2*l2^2+j1+j2+m2*l1*l2*cos(x(3));
 M2=0.25*m2*l2+j2+0.5*m2*l1*l2*cos(x(3));
 M3=0.25*m2*l2+j2+0.5*m2*l1*l2*cos(x(3));
 M4=0.25*m2*l2+j2;
 %%centrifugal and coriolis force
K1=-0.5*m2*l2*l1*sin(x(3))*x(4)*(2*x(2)+x(4));
K2=0.5*m2*l2*l1*sin(x(3))*x(2)^2;
%%gravity componant
G1= m1*g*(l1/2)*cos(x(1))+m2*g*l1*cos(x(1))+(l2/2)*cos(x(1)+x(3));
G2=m2*g*(l2/2)*cos(x(1)+x(3));
M=[M1 M2;M3 M4];
H=inv(M);
H1=H(1,1);
H2=H(1,2);
H3=H(2,1);
H4=H(2,2);
%%State derivatives
xdot(1)=x(2);
xdot(2)=-H1*(K1+G1)-H2*(K2+G2)+H1*F1+H2*F2;
xdot(3)=x(4);
xdot(4)=-H3*(K1+G1)-H4*(K2+G2)+H3*F1+H4*F2;

sys=[xdot(1); xdot(2) ;xdot(3); xdot(4)];
% end mdlDerivatives
function sys=mdlOutputs(t,x,u)

sys =[ x(1) ;x(2);x(3);x(4)] ;
```

Appendix D

Torque Control Parameters



<table>
<tr><td>02
*</td><td>Setup of
control mode</td><td>0 to 6
<1></td><td>You can set up the control mode to be used.
<table>
<tr><th>Setup value</th><th colspan="2">Control mode</th></tr>
<tr><th></th><th>1st mode</th><th>2nd mode</th></tr>
<tr><td>0</td><td>Position</td><td>–</td></tr>
<tr><td><1></td><td>Velocity</td><td>–</td></tr>
<tr><td>2</td><td>Torque</td><td>–</td></tr>
<tr><td>3**1</td><td>Position</td><td>Velocity</td></tr>
<tr><td>4**1</td><td>Position</td><td>Torque</td></tr>
<tr><td>5**1</td><td>Velocity</td><td>Torque</td></tr>
<tr><td>6</td><td>Full-closed</td><td>–</td></tr>
</table>
**1) When you set up the combination mode of 3, 4 or 5, you can select either the 1st or the 2nd with control mode switching input (C-MODE).
When C-MODE is open, the 1st mode will be selected.
When C-MODE is shorted, the 2nd mode will be selected.
Don't enter commands 10ms before/after switching.


</td></tr>
</table>

Standard default : < >

PrNo.	Title	Setup range	Unit	Function/Content
1E	1st notch width selection	0 to 4 <2>	–	You can set up the notch filter width of the 1st resonance suppressing filter in 5 steps. Higher the setup, larger the notch width you can obtain. Use with default setup in normal operation.

Parameters for Auto-Gain Tuning

Standard default : < >

<table>
<tr><th>PrNo.</th><th>Title</th><th>Setup range</th><th>Unit</th><th>Function/Content</th></tr>
<tr><td>20</td><td>Inertia ratio</td><td>0 to 10000
<250>*</td><td>%</td><td>You can set up the ratio of the load inertia against the rotor (of the motor) inertia.
<table><tr><td>Pr20=(load inertia/rotor inertia) X 100 [%]</td></tr></table>
When you execute the normal auto-gain tuning, the load inertial will be automatically estimated after the preset action, and this result will be reflected in this parameter.
The inertia ratio will be estimated at all time while the real-time auto-gain tuning is valid, and its result will be saved to EEPROM every 30 min.
<Caution>
If the inertia ratio is correctly set, the setup unit of Pr11 and Pr19 becomes (Hz). When the inertia ratio of Pr20 is larger than the actual, the setup unit of the velocity loop gain becomes larger, and when the inertia ratio of Pr20 is smaller than the actual, the setup unit of the velocity loop gain becomes smaller.</td></tr>
<tr><td>21</td><td>Setup of real-time auto-gain tuning</td><td>0 to 7
<1></td><td>–</td><td>You can set up the action mode of the real-time auto-gain tuning.
With higher setup such as 3, the driver respond quickly to the change of the inertia during operation, however it might cause an unstable operation.
Use 1for normal operation.
<table>
<tr><th>Setup value</th><th>Real-time auto-gain tuning</th><th>Varying degree of load inertia in motion</th></tr>
<tr><td>0</td><td>Invalid</td><td>–</td></tr>
<tr><td><1>, 4, 7</td><td rowspan="3">Normal mode</td><td>Little change</td></tr>
<tr><td>2, 5</td><td>Gradual change</td></tr>
<tr><td>3, 6</td><td>Rapid change</td></tr>
</table></td></tr>
<tr><td>22</td><td>Selection of machine stiffness at real-time auto-gain tuning</td><td>0 to 15
A to C-frame:
<4>
D to F-frame:
<1></td><td>–</td><td>You can set up the machine stiffness in 16 steps while the real-time auto-gain tuning is valid.
<table>
<tr><td colspan="2">low←machine stiffness→high
low← servo gain →high</td></tr>
<tr><td>Pr22</td><td>0, 1- - - - - - - - - - - - 14, 15</td></tr>
<tr><td colspan="2">low← response →high</td></tr>
</table>
<Caution>
When you change the setup value rapidly, the gain changes rapidly as well, and this may give impact to the machine. Increase the setup gradually watching the movement of the machine.</td></tr>
</table>

Parameters for Adjustment of Time Constants of Gains and Filters

Standard default : < >

PrNo.	Title	Setup range	Unit	Function/Content
11	1st gain of velocity loop	1 to 3500 A to C-frame:<35>* D to F-frame:<18>*	Hz	You can determine the response of the velocity loop. In order to increase the response of overall servo system by setting high position loop gain, you need higher setup of this velocity loop gain as well. However, too high setup may cause oscillation. **<Caution>** When the inertia ratio of Pr20 is set correctly, the setup unit of Pr11 becomes (Hz).
12	1st time constant of velocity loop integration	1 to 1000 A to C-frame:<16>* D to F-frame:<31>*	ms	You can set up the integration time constant of velocity loop. Smaller the setup, faster you can dog-in deviation at stall to 0. The integration will be maintained by setting to "999". The integration effect will be lost by setting to "1000".
13	1st filter of speed detection	0 to 5 <0>*	–	You can set up the time constant of the low pass filter (LPF) after the speed detection, in 6 steps. Higher the setup, larger the time constant you can obtain so that you can decrease the motor noise, however, response becomes slow. Use with a default value of 0 in normal operation.
14	1st time constant of torque filter	0 to 2500 A to C-frame:<65>* D to F-frame:<126>*	0.01ms	You can set up the time constant of the 1st delay filter inserted in the torque command portion. You might expect suppression of oscillation caused by distortion resonance.
19	2nd gain of velocity loop	1 to 3500 A to C-frame:<35>* D to F-frame:<18>*	Hz	Position loop, velocity loop, speed detection filter and torque command filter have their 2 pairs of gain or time constant (1st and 2nd). For details of switching the 1st and the 2nd gain or the time constant, refer to P.226, "Adjustment". The function and the content of each parameter is as same as that of the 1st gain and time constant.
1A	2nd time constant of velocity loop integration	1 to 1000 <1000>*	ms	
1B	2nd filter of velocity detection	0 to 5 <0>*	–	
1C	2nd time constant of torque filter	0 to 2500 A to C-frame:<65>* D to F-frame:<126>*	0.01ms	
1D	1st notch frequency	100 to 1500 <1500>	Hz	You can set up the frequency of the 1st resonance suppressing notch filter. The notch filter function will be invalidated by setting up this parameter to "1500".

<Notes>

- For parameters which No. have a suffix of "*", changed contents will be validated when you turn on the control power.
- Parameters which default values have a suffix of "*" will be automatically set up during real time auto-gain tuning. When you change manually, invalidate the real-time auto-gain tuning first then set, referring to P.239, "Release of Automatic Gain Adjusting Function" of Adjustment.

56	4th speed of speed setup	–20000 to 20000 <0>	r/min	You can set up the speed limit value in unit of [r/min] . **<Caution>** The absolute value of the parameter setup is limited by Pr73 (Set up of over-speed level).
57	Setup of speed command filter	0 to 6400 <0>	10μs	You can set up the time constant of the primary delay filter to the analog speed command/analog torque command/analog velocity control (SPR : CN X5, Pin-14)
5B	Selection of torque command	0 to 1 <0>	–	You can select the input of the torque command and the speed limit.

Pr5B	Torque command	Velocity limit
<0>	SPR/TRQR/SPL	Pr56
1	CCWTL/TRQR	SPR/TRQR/SPL

Standard default : < >

<table>
<tr><th>PrNo.</th><th>Title</th><th>Setup range</th><th>Unit</th><th>Function/Content</th></tr>
<tr><td>25</td><td>Setup of an action at normal mode auto-gain tuning</td><td>0 to 7
<0></td><td>–</td><td>You can set up the action pattern at the normal mode auto-gain tuning.
<table>
<tr><th>Setup value</th><th>Number of revolution</th><th>Rotational direction</th></tr>
<tr><td><0></td><td rowspan="4">2 [revolution]</td><td>CCW → CW</td></tr>
<tr><td>1</td><td>CW → CCW</td></tr>
<tr><td>2</td><td>CCW → CCW</td></tr>
<tr><td>3</td><td>CW → CW</td></tr>
<tr><td>4</td><td rowspan="4">1 [revolution]</td><td>CCW → CW</td></tr>
<tr><td>5</td><td>CW → CCW</td></tr>
<tr><td>6</td><td>CCW → CCW</td></tr>
<tr><td>7</td><td>CW → CW</td></tr>
</table>
e.g.) When the setup is 0, the motor turns 2 revolutions to CCW and 2 revolutions to CW.</td></tr>
<tr><td>28</td><td>2nd notch frequency</td><td>100 to 1500
<1500></td><td>Hz</td><td>You can set up the 2nd notch width of the resonance suppressing filter in 5 steps. The notch filter function is invalidated by setting up this parameter to "1500".</td></tr>
<tr><td>29</td><td>Selection of 2nd notch width</td><td>0 to 4
<2></td><td>–</td><td>You can set up the notch width of 2nd resonance suppressing filter in 5 steps. Higher the setup, larger the notch width you can obtain.
Use with default setup in normal operation.</td></tr>
<tr><td>2A</td><td>Selection of 2nd notch depth</td><td>0 to 99
<0></td><td>–</td><td>You can set up the 2nd notch depth of the resonance suppressing filter. Higher the setup, shallower the notch depth and smaller the phase delay you can obtain.</td></tr>
</table>

Parameters for Adjustment (2nd Gain Switching Function)

Standard default : < >

<table>
<tr><th>PrNo.</th><th>Title</th><th>Setup range</th><th>Unit</th><th>Function/Content</th></tr>
<tr><td>30</td><td>Setup of 2nd gain</td><td>0 to 1
< 1> *</td><td>–</td><td>You can select the PI/P action switching of the velocity control or 1st/2nd gain switching.
<table>
<tr><th>Setup value</th><th>Gain selection/switching</th></tr>
<tr><td>0</td><td>1st gain (PI/P switching enabled) *1</td></tr>
<tr><td>< 1> *</td><td>1st/2nd gain switching enabled *2</td></tr>
</table>
*1 Switch the PI/P action with the gain switching input (GAIN CN X5, Pin-27). PI is fixed when Pr03 (Torque limit selection) is 3.
<table>
<tr><th>GAIN input</th><th>Action of velocity loop</th></tr>
<tr><td>Open with COM–</td><td>PI action</td></tr>
<tr><td>Connect to COM–</td><td>P action</td></tr>
</table>
*2 For switching condition of the 1st and the 2nd, refer to P.243, "Gain Switching Function" of Adjustment.</td></tr>
<tr><td rowspan="3">31</td><td>1st mode of control switching</td><td>0 to 10
< 0> *</td><td>–</td><td>You can select the switching condition of 1st gain and 2nd gain while Pr30 is set to 1.</td></tr>
<tr><td colspan="4">
<table>
<tr><th>Setup value</th><th>Gain switching condition</th></tr>
<tr><td>< 0>*, 4to 10</td><td>Fixed to the 1st gain.</td></tr>
<tr><td>1</td><td>Fixed to the 2nd gain.</td></tr>
<tr><td>2 *1</td><td>2nd gain selection when the gain switching input is turned on. (Pr30 setup must be 1.)</td></tr>
<tr><td>3 *2</td><td>2nd gain selection when the toque command variation is larger than the setups of Pr33 (1st level of control switching) and Pr34 (1st hysteresis of control switching).</td></tr>
</table></td></tr>
<tr><td></td><td></td><td></td><td>*1 Fixed to the 1st gain regardless of GAIN input, when Pr31 is set to 2 and Pr03 (Torque limit selection) is set to 3.
*2 For the switching level and the timing, refer to P.243, "Gain Switching Function" of Adjustment.</td></tr>
</table>

<Notes>

- Parameters which default values have a suffix of "*" will be automatically set up during real time auto-gain tuning. When you change manually, invalidate the real-time auto-gain tuning first then set, referring to P.239, "Release of Automatic Gain Adjusting Function" of Adjustment.

Parameters Which Are Automatically Set

Table of auto-gain tuning

Pr No.	Title	Stiffness value															
		0	**[1]**	**2**	**3**	**[4]**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
10	1st gain of position loop	12	32	39	48	63	72	90	108	135	162	206	251	305	377	449	557
11	1st gain of velocity loop	9	18	22	27	35	40	50	60	75	90	115	140	170	210	250	310
12	1st time constant of velocity loop integration	62	31	25	21	16	14	12	11	9	8	7	6	5	4	4	3
13	1st filter of velocity detection	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1st time constant of torque filter time *2	253	126	103	84	65	57	45	38	30	25	20	16	13	11	10	10
15	Velocity feed forward	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300
16	Velocity FF filter	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
18	2nd gain of position loop	19	38	46	57	73	84	105	126	157	188	241	293	356	440	524	649
19	2nd gain of velocity loop	9	18	22	27	35	40	50	60	75	90	115	140	170	210	250	310
1A	2nd time constant of velocity loop integration	999	999	999	999	999	999	999	999	999	999	999	999	999	999	999	999
1B	2nd filter of speed detection	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1C	2nd time constant of torque filter *2	253	126	103	84	65	57	45	38	30	25	20	16	13	11	10	10
20	Inertia ratio	Estimated load inertia ratio															
27	Setup of instantaneous velocity observer	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	2nd gain setup	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
31	1st mode of control switching *1	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
32	1st delay time of control switching	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
33	1st level of control switching	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
34	1st Hysteresis of control switching	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33
35	Switching time of position gain	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
36	2nd mode of control switching	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

▨ represents parameters with fixed value. Default for A to C-frame is 4, and 1 for D to F-frame.

*1 Stiffness value is 10 for position control and full-closed control, and 0 for velocity control and torque control.

*2 Lower limit for stiffness value is 10 for 17-bit encoder, and 25 for 2500P/r encoder.

References

[1] Powered exoskeleton. (2016, April 29). In Wikipedia, The Free Encyclopedia. Retrieved 23:23, April 30, 2016, from cen.wikipedia.org/w/index.php?title=Poweredexoskeletonoldid = 717702160.

[2] Ministry of health from 29/9/2000 to 31/12/2014. Palestine health information center. Palestine 2015. July 2014.

[3] Heedon Lee1, Wansoo Kim1, Jungsoo Han2, and Changsoo Han1,The Technical Trend of the Exoskeleton Robot System for Human Power Assistance, INTERNATIONAL JOURNAL OF PRECISION ENGINEERING AND MANUFACTURING Vol. 13, No. 8, pp. 1491-1497, AUGUST 2012 / 1491.

[4] S. A. Murray, K. H. Ha, C. Hartigan and M. Goldfarb, An assistive control approach for a lower-limb exoskeleton to facilitate recovery of walking following stroke, IEEE Trans. Neural Syst. Rehabil. Eng., vol. 23, no. 3, pp. 441-449, 2015.

[5] Lei Li; K. H. Hoon; Adela Tow; P. H. Lim; K. H. Low, Design and Control of Robotic Exoskeleton with Balance Stabilizer Mechanism,Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on 2015,pp.3817 - 3823,2015.

[6] Perry , J. Rosen and S. Burns, Upper-limb powered exoskeleton design, IEEE/ASME Trans. Mechatronics, vol. 12, no. 4, pp. 408-417, 2007.

[7] Lorenzo Grazi; Simona Crea; Andrea Parri; Tingfang Yan; Mario Cortese; Francesco Giovacchini; Marco Cempini; Guido Pasquini; Silvestro Micera; Nicola Vitiello, Gastrocnemius Myoelectric Control of a Robotic Hip Exoskeleton, la Vitiello 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 3881 - 3884, 2015.

[8] A. Scano, Static and Dynamic characterization of the LIGHTarm exoskeleton for rehabil- 43 itation, Rehabilitation Robotics (ICORR), 2015 IEEE International Conference on 11 14 August 2015.

[9] U. Onen, F. Botsali, M. Kalyoncu, M. Tinkir, N. Yilmaz and Y. Sahin, Design and actuator selection of a lower extremity exoskeleton, IEEE Trans. Mechatron., vol. 19, no. 2, pp. 623-632, 2014.

[10] A. B. Zoss , H. Kazerooni and A. Chu, Biomechanical design of the Berkeley lower extremity exoskeleton (BLEEX), IEEE/ASME Trans. Mechatronics, vol. 11, no. 2, pp. 128-138, 2006.

[11] Elizabeth Beattie, Nick McGill, Nick Parrotta, Niko Vladimirov Titan arm , http://www . titanarm . com/about.

[12] S. Wang, Design and Control of the MINDWALKER exoskeleton, IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 23, no. 2, pp. 277-286, 2015.

[13] A. Otten, H. C. Voort, A. H. Stienen, E. H. F. van, R. G. K. M and H. van der Kooij, Limpact: A hydraulically powered self-aligning upper limb exoskeleton, IEEE Trans. Mechatronics.

[14] Shetty, D. and Kolk, bookMechatronics System Design, Shetty, D. and Kolk, R., v. 1, isbn.9780534952853, lccn.97013889, series.Mechatronics System Design, url.https://books. google.ps /books?id=XUlSAAAAMAAJ, publisherPWS Pub,1996.

[15] Wansoo Kim; Heedon Lee; Donghwan Kim; Jungsoo Han; Changsoo Han,Mechanical Design of the Hanyang Exoskeleton Assistive Robot(HEXAR)Control, Automation and Systems (ICCAS), 2014 14th International Conference on 2014, pp. 479 - 484, 2014.

[16] Farris R J, Quintero H A, Goldfarb M. Performance evaluation of a lower limb exoskeleton for stair ascent and descent with Paraplegia[C]//Engineering in Medicine and Biology Society (EMBC), 2012: 1908-1911.

[17] K. Kong and J. Doyoung, Design and control of an exoskeleton for the elderly and patients, IEEE/ASME Trans. Mechatronics, vol. 11, no. 4, pp. 428-432, 2006.

[18] Chris Bank, Jerry Cao, The Guide to UX Design Process and Documentation https:// studio.uxpin.com /ebooks/ guide-to-ux-design-process-and-documentation.