FINAL GP Book(exoskeleton)

Book · August 2020

5 authors, including:

Yusef Okasha
Modern Academy

Ahmed Aly Eldin
Modern Academy

Mostafa Gomaa
Modern Academy

Exoskeleton Hand

Bachelor's Degree Project in Electronics and Communication Engineering

2019-2020

Under supervision of:

Dr. Abdelmonem Mahdy

Prepared by:

Ahmed Ali El-din

Ahmed Mohamed El-Shazly

Mohamed Ashraf Abd El-Sayed

Mostafa Ahmed Gomaa

Yousef Okasha Abdallah

Acknowledgement

In the accomplishment of this project successfully, many people have best owned upon us their blessings and the heart pledged support, this time we are utilizing to thank all the people who have been concerned with this project. Primarily we would thank god for being able to complete this project with success. Then we would like to thank our supervisor Dr. Abdelmonem Mahdy, whose valuable guidance has been the ones that helped us patch this project and make it full proof success. His suggestions and his instructions have served as the major contributor towards the completion of the project. Then we would like to thank our parents and friends who have helped us with their valuable suggestions and guidance has been very helpful in various phases of the completion of the project. Last but not the least we would like to thank our classmates who have helped us a lot and doctors who gave us the information about this project.

Content

Appendices

Abstract

Human hands combine unique dexterity (البراعة) and somatosensory abilities (القدرات الحسية الجسدية) which are exceptional in living organisms and not reached in technology. How important hands are for our daily living becomes clear when we temporarily or permanently lose the ability to use them and simple tasks like eating or grooming suddenly require assistance from a person. Worldwide, for millions of individuals suffer from upper limb impairment following neurological diseases or trauma such as stroke or spinal cord injury. Pain, spasticity(التشنج), contractures (التقلصات), and weakness lead to disabilities ranging from poor finger individuation to complete paralysis (شلل تام) of the hand and, consequently, to a major loss of independence and quality of life.

In case of neuromotor hand impairment, it is well acknowledged that through robotics high dose and intensity in the training and a positive effect on the outcome of the rehabilitation therapy can be reached. However, many individuals remain with persistent hand impairment when they are discharged from the hospital after the typical treatments to restore the hand function have been applied. Whether or not there is potential for further recovery of the nervous system, the use of the hand and the therapy must be continued to at least save the residual hand function.

The potential of wearable hand exoskeletons for assistance in daily living and continued home rehabilitation therapy has been recognized in research and industry, and much devices have been developed. However, the human hand is highly skillful, powerful, and of smallest design. The emulation of its functions requires complex mechanical systems, and the trade-off between functionality and usability in daily life is challenging.

The aim of this work is to make and evaluate a wearable robotic system fusing the concepts of assistance and therapy in the design of EXO, a wearable actuated hand exoskeleton for assistance in daily life and with the potential for wearable all-day rehabilitation in the clinic or at home for subjects with neuromotor hand impairments.

We identify the key design factors and requirements for such a device in continuous discussion with expert clinicians, interviews with users, and various usability studies.

We realize that, in a first step, a fully wearable hand exoskeleton for assistance in daily living has to be designed, and only in a second step, through control based on physiological signals, this device can turn into a wearable system for assistance and rehabilitation.

Based on the defined requirements and presented hand exoskeleton, we suggest a detailed design concept for EXO hand. We reduce the dexterity of the hand to few basic functions needed to perform the most used grasp types.

List of Figures

Chapter one
Introduction

Stroke, caused by death of brain cells as a result of blockage of a blood vessel supplying the brain (ischemic stroke) or bleeding into or around the brain (hemorrhagic stroke), is a serious medical emergency. Stroke can result in death or substantial neural damage and is a principal contributor to long-term disabilities. According to the World Health Organization estimates, 15 million people suffer stroke worldwide each year. More than half of stroke survivors experience some level of lasting hemiparesis or hemiplegia resulting from the damage to neural tissues.

1.1. Brain Attack

Stroke or Brain Attack is a disease that involves the blood vessels that supply blood to the brain, stroke occurs when a blood vessel that brings oxygen and nutrients to the brain ruptures or is clogged by a blood clot or some other mass, without adequate supply of oxygen, nerve cells of the brain can't work and die within minutes, when nerve cells can't work, the area of the body they control can't work either. TIA are transient ischemic attacks or mini-strokes that occur when the inadequate blood supply to the brain is recovered after few minutes of an occlusion of a vessel, they represent warning signs of more serious or permanent strokes, causes of stroke and TIA include wandering clots from the heart, fatty buildups on the aortic arch or the vessels of the neck or brain, stenosis or narrowing of the arteries of the neck and the brain or primary diseases affecting the arteries of the central nervous system known as vasculitis(التهاب الأوعية الدموية). (1)

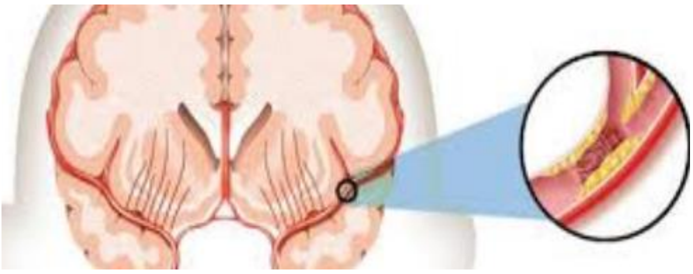


Figure 1.1: Brain attack(stroke).

Diseases associated with strokes include diabetes, high blood pressure, high cholesterol, carotid disease (مرض السباتي) , atrial fibrillation (الرجفان الأذيني)or other heart disease, history or TIAs (mini-strokes), sickle cell anemia, obesity, fibromuscular dysplasia, connective tissue disorders, other family members with history of strokes.

1.1.1. Symptoms and Diagnosis

Symptoms Sudden numbness or weakness of the face, arm or leg, especially on one side of the body, Sudden confusion, difficulties speaking or understanding, Sudden loss of vision in one or both eyes, Sudden gait troubles, dizziness or loss of balance and coordination.

Diagnosis MRI and CT scans can recognize the presence of strokes MR-angiograms, CT-angiograms and Cerebral angiography provide complementary, important information regarding the location of the vessel occlusion, the degree of brain tissue affected, and the collateral circulation to the affected part of the brain. muscle atrophy is diagnosed if muscle atrophy is caused by another condition need to under go testing of diagnose the condition doctor will request complete medical history of patient, doctor should know about old or recent injuries and previously diagnosed medical conditions list prescriptions, over-the counter medications, and supplements and taking detailed description for symptoms of patient, doctor may also order tests to help with the diagnosis and to rule out certain diseases.

Tests may include blood tests, X-rays, magnetic resonance imaging (MRI), computed tomography (CT) scan, nerve conduction studies, muscle or nerve biopsy, electromyography (EMG)

1.1.2. Muscle Atrophy

Is when muscles waste away, usually is caused by a lack of physical activity, when a disease or injury makes it difficult or impossible for you to move an arm or leg, the lack of mobility can result in muscle wasting over time, without regular movement, your arm or leg can start to appear smaller but not shorter than the one you’re able to move. In some cases, muscle wasting can be reversed with a proper diet, exercise, or physical therapy.

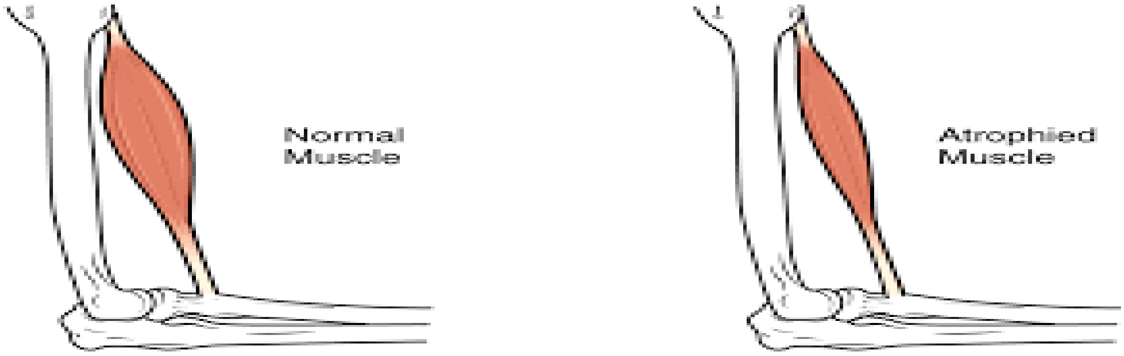


Figure 1.2: Muscle Atrophy.

1.1.3. Causes of neuromuscular atrophy

Nervous atrophy is the most severe type of muscle atrophy and can be an injury or disease of the nerve that connects the muscles, and this type of muscle atrophy tends to occur quickly and suddenly. There are a number of diseases that affect the nerves that control the muscles, namely:

- Amyotrophic lateral sclerosis.
- Damage to the nerve such as carpal tunnel syndrome.
- Guillain-Barré syndrome, also called acute myeloid neuritis, an acquired multiple neuropathy that affects the nerve roots and then moves to the peripheral nerves.
- Nerve damage caused by diabetes, toxins, or alcohol.
- poliomyelitis.
- Spinal cord injury.

1.2. Myasthenia Gravis

Myasthenia gravis is a chronic autoimmune neuromuscular disease that causes weakness in the skeletal muscles, which are responsible for breathing and moving parts of the body, including the arms and legs. The name myasthenia gravis, which is Latin and Greek in origin, means "grave, or serious, muscle weakness."

The hallmark of myasthenia gravis is muscle weakness that worsens after periods of activity and improves after periods of rest. Certain muscles such as those that control eye and eyelid movement, facial expression, chewing, talking, and swallowing are often (but not always) involved in the disorder. The muscles that control breathing and neck and limb movements may also be affected.

There is no known cure but with current therapies most cases of myasthenia gravis are not as "grave" as the name implies. Available treatments can control symptoms and often allow people to have a relatively high quality of life. Most individuals with the condition have a normal life expectancy.

1.2.1. Causes of Myasthenia Gravis

Myasthenia gravis is caused by an error in the transmission of nerve impulses to muscles. It occurs when normal communication between the nerve and muscle is interrupted at the neuromuscular junction—the place where nerve cells connect with the muscles they control.

Neurotransmitters are chemicals that neurons, or brain cells, use to communicate information. Normally when electrical signals or impulses travel down a motor nerve, the nerve endings release a neurotransmitter called acetylcholine. Acetylcholine travels from the nerve ending and binds to acetylcholine receptors on the muscle. The binding of acetylcholine to its receptor activates the muscle and causes a muscle contraction.

In myasthenia gravis, antibodies (immune proteins البروتينات المناعية) block, alter, or destroy the receptors for acetylcholine at the neuromuscular junction, which prevents the muscle from contracting. In most individuals with myasthenia gravis, this is caused by antibodies to the acetylcholine receptor itself. However, antibodies to other proteins, such as Musk (Muscle-Specific Kinase) protein, can also lead to impaired transmission at the neuromuscular junction.

These antibodies are produced by the body's own immune system. Myasthenia gravis is an autoimmune disease because the immune system—which normally protects the body from foreign organisms—mistakenly attacks itself.

The thymus (الغدة الصعترية) is a gland that controls immune function and maybe associated with myasthenia gravis. Located in the chest behind the breast bone, the gland is largest in children. It grows gradually until puberty, and then gets smaller and is replaced by fat. Throughout childhood, the thymus plays an important role in the development of the immune system because it is responsible for producing T-lymphocytes or T cells, a specific type of white blood cell that protects the body from viruses and infections. In many adults with myasthenia gravis, the thymus gland remains large. People with the disease typically have clusters of immune cells in their thymus gland similar to lymphoid hyperplasia—a condition that usually only happens in the spleen and lymph nodes during an active immune response. Some individuals with myasthenia gravis develop thymomas (tumors of the thymus gland أورام الغدة الصعترية). Thymomas are most often harmless, but they can become cancerous.

The thymus gland plays a role in myasthenia gravis, but its function is not fully understood. Scientists believe that the thymus gland may give incorrect instructions to developing immune cells, ultimately causing the immune system to attack its own cells and tissues and produce acetylcholine receptor antibodies—setting the stage for the attack on neuromuscular transmission. (2)

1.3. Hand Tendon Injuries

when tendons in your hand are damaged, surgery must be needed to repair them and help restore movement in the affected fingers or thumb.

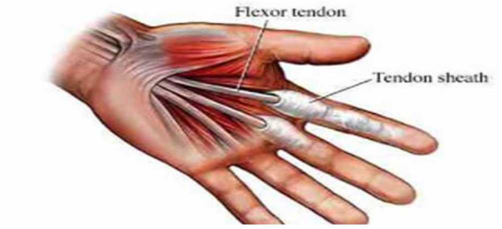


Figure 1.3: Hand Tendons

Tendons are tough cords of tissue that connect muscles to bones. When a group of muscles contract (tighten), the attached tendons will pull on certain bones, allowing you to make a wide range of movements. (3)

There are 2 groups of tendons in the hand:

- **extensor tendons** – which run from the forearm across the back of your hand to your fingers and thumb, allowing you to straighten your fingers and thumb.
- **flexor tendons** – which run from your forearm through your wrist and across the palm of your hand, allowing you to bend your fingers.

Surgery can often be carried out to repair damage to both these groups of tendons.

Hand tendon repair is carried out when one or more tendons in your hand rupture or are cut, leading to the loss of normal hand movements.

If your extensor tendons are damaged, you'll be unable to straighten one or more fingers. If your flexor tendons are damaged, you'll be unable to bend one or more fingers.

Tendon damage can also cause pain and inflammation (swelling تورم) in your hand.
In some cases, damage to the extensor tendons can be treated without the need for surgery, using a rigid support called a splint that's worn around the hand.
Common causes of tendon injuries include:

- cuts – cuts across the back or palm of your hand can result in injury to your tendons.

- sports injuries – extensor tendons can rupture when stubbing finger, such as trying to catch a ball; flexor tendons can occasionally be pulled off the bone when grabbing an opponent's jersey, such as in rugby; and the pulleys holding flexor tendons can rupture during activities that involve lots of strenuous gripping, such as rock climbing.
- bites – animal and human bites can cause tendon damage, and a person may damage their hand tendon after punching another person in the teeth.
- crushing injuries – jamming a finger in a door or crushing a hand in a car accident can divide or rupture a tendon.
- rheumatoid arthritis – rheumatoid arthritis can cause tendons to become inflamed, which can lead to tendons rupturing in severe cases.

1.3.1. Tendon Repair

Tendon repair may involve the surgeon making a cut (incision) in your wrist, hand or finger so they can locate the ends of the divided tendon and stitch them together.
Extensor tendons are easier to reach, so repairing them is relatively straightforward, depending on the type of injury, it may be possible to repair extensor tendons the affected area.
Repairing flexor tendons is more challenging because the flexor tendon system is more complex, flexor tendon repair usually needs to be carried out under either general an aesthetic or regional an aesthetic (where the whole arm is numbed) in an operating theatre by an experienced plastic or orthopedic surgeon who specializes in hand surgery.

1.3.2. Recovering from surgery

Both types of tendon surgery require a lengthy period of recovery (rehabilitation) because the repaired tendons will be weak until the ends heal together.

Depending on the location of the injury, it can take up to 3 months for the repaired tendon to regain its previous strength.

Rehabilitation involves protecting your tendons from overuse using a hand splint. You'll usually need to wear a hand splint for several weeks after surgery.

You'll also need to perform hand exercises regularly during your recovery to stop the repaired tendons sticking to nearby tissue, which can prevent you being able to fully move your hand.

1.3.3. Results

After an extensor tendon repair you should have a working finger or thumb, but you may not regain full movement. The outcome is often better when the injury is a clean cut to the tendon, rather than one that involves crushing or damage to the bones and joints.

A flexor tendon injury is generally more serious because they're often put under more strain than extensor tendons.

After a flexor tendon repair, it's quite common for some fingers to not regain full movement. But the tendon repair will still give a better result than not having surgery, in some cases, complications develop after surgery, such as infection or the repaired tendon snapping or sticking to nearby tissue.

1.4. Hand Rehabilitation Robotics

Hand Rehabilitation Robotics used in Poststroke Motor Recovery and the most diseases and surgeries affecting on hand

The recovery of hand function is one of the most challenging topics in stroke, surgeries and in many diseases rehabilitation. Although the robot-assisted therapy has got some good results in the latest decades, the development of hand rehabilitation robotics is left behind. Existing researches of hand rehabilitation robotics focus either on the mechanical design on designers 'view or on the training paradigms on the clinicians' view, while these two parts are interconnected and both important for designers and clinicians.

we should explore the current hand rehabilitation robots, to know better choices among varied components and thus promoting the application of hand rehabilitation robots. An overview of hand rehabilitation robotics is provided, to give a general view of the relationship between subjects, hand rehabilitation robots, and its evaluation. (4)

1.5. Classification of Hand Rehabilitation Robots

There exist many kinds of ways for the classification of hand rehabilitation robots, some of which follow the convention in mechanical design (focus on the hardware system), while others follow the convention in rehabilitation (which focus on the training paradigms). In fact, each of these ways of classification has its own value, and they are dependent to each other. For example, the hardware system depends on the basic abilities of the rehabilitation robotics (e.g., possible movements and feedback information), while the training paradigms are the main functional components in the recovery (e.g., the application of specific rehabilitation theories). The overall overview in this classification is shown in Figure 1.4. (5)

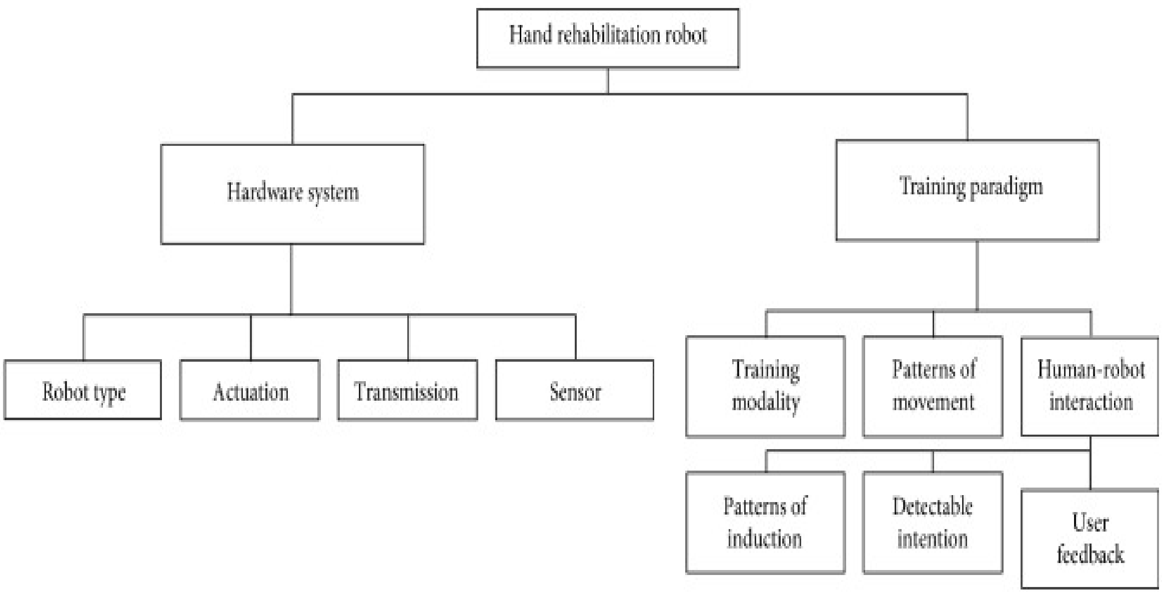


Figure1.4: Classification of Hand Rehabilitation Robots

1.6. Exoskeleton Hand

In recent years, technology has improved healthcare, which means that the life expectancy of patients has increased. Despite this, some patients can live with functional dependencies, which implies a limitation or decrease in their welfare, it is at this point, where the rehabilitation appears. Technology is a branch, which is constantly advancing and developing. It is involved in almost all fields of work, such as military, social, among others. Throughout the last decades, medicine and engineering have worked together in order to develop new systems of rehabilitation, studies of the human body and its limits.

In this way, patients could use the new technologies to recover mobility affected or lost due to illness or accidents. Stroke represents the main cause for the motor disability, millions of persons have a permanent motor deficit due to this disease. With all that, the rehabilitation, known as the therapy with the purpose to regain totally or partially the basic motor abilities or tasks, has become increasingly important.

According to the webpage Exoskeleton Report: "*The field of exoskeleton systems is continuously evolving and re-inventing itself, so it is still difficult to create a singular definition".*

In general:

- Exoskeletons are wearable devices that work in tandem with the user. The opposite of an exoskeleton device would be an autonomous robot that works instead of the operator.
- Exoskeletons are placed on the user's body and act as amplifiers that augment, reinforce or restore human performance. The opposite would be a mechanical prosthetic, such as a robotic arm or leg that replaces the original body part.
- Exoskeletons can be made from rigid materials such as metal or carbon fiber, or they can be made entirely out of soft and elastic parts.
- Exoskeletons can be powered and equipped with sensors and actuators, or they can be entirely passive.
- Exoskeletons can be mobile or fixed/suspended (usually for rehabilitation or teleoperation).
- Exoskeletons can cover the entire body, just the upper or lower extremities, or even a specific body segment such as the ankle or the hip.

In summary, robotics is the application of engineering towards replacing humans from menial tasks, while exoskeletons are the application of robotics and biomechatronic towards the augmentation(تعزيز) of humans in the performance of a variety of tasks.

Therefore, in biomedicine the exoskeletons are one of the tools which have been created to improve both the rehabilitation and to discover the new limits of the human body. To explain this in easier words, an exoskeleton is basically a "wearable mechanic device". The present thesis consists of the design of an exoskeleton especially developed for the fingers rehabilitation with passive motion, in order to help people with limited mobility in their hands. (6)

1.6.1. Overview

Several studies have demonstrated that the use of these rehabilitation exoskeletons, either with active or passive movement, give benefits to the patients, the majority of them have been produced focusing on solving problems regarding the inferior extremities of the human body. Given that, the hand movements are extremely complex to simulate by a device and expensive to be developed and commercialized.

However, other models focus solely on hand injuries. The main difference between the mechanisms are the degrees of freedom (DOF), type of sensors or user control, and used materials. In the following lines, some studies performed in this area will be described.

Concerning the fingers, the article “Evolutionary synthesis of mechanisms applied to the design of an exoskeleton for finger rehabilitation” by A. Bataller, J.A. Cabrera and J.J. Castillo (2006). The device was designed for the index finger with 4 DOF, and by a developed algorithmic system, the authors from the University of Málaga recreated in an accurate way the natural hand-finger movement. Also, J. Wang, J. Li, Y. Zhang, S. Wang (2009) developed another exoskeleton for the index finger.

It has 4 DOF, can generate bidirectional movement and it is adjustable to different measurements of patients. It also uses different sensors to determine the angular position of the finger, and in this way to control, analyze and evaluate the effects of the physical therapy.

Also focused on rehabilitation, A. Wege, K. Kondak, and G. Hommel (2005) worked in a hand exoskeleton. This mechanism has 4 DOF, it is moved by a linear actuator, and it receives information by sensors of Hall Effect in each structure articulation. The model can be observed in Figure 1.5.

Figure.1.5: One finger prototype. 1: Hall Effect sensor. 2: Actuator. (Wege, Kondak & Hommel, 2005)

A variation with medical purposes but with a preventive view can be found in “An anthropomorphic hand exoskeleton to prevent astronaut hand fatigue during extravehicular activities أنشطة خارج المركبة” by B.L. Shields, J.A. Main, S.W. Peterson, and A.M. Strauss (1997). This model tries to reduce the hardness of the space suit by movements monitored with pressure sensors localized between the exoskeleton and the hand.

Current exoskeletons used in rehabilitation processes are made from steel and other heavy and difficult to manufacture plastics. A clear example of current exoskeleton made of steel can be seen in Figure 1.6. The “hand of hope” is a device constructed by Rehab Robotics (Rehab-robotics.com, 2018), which is a company that operates in partnership with the Polytechnic University of Hong Kong.

Figure.1.6: Exoskeleton comparison. (Rehab-Robots 2018)

They work developing new technologies in rehabilitation area in order to help patients with reduced mobility and to improve maximum recovery outcomes.

It uses steel material and linear servo to achieve the movement, This structure is complex and includes several sensors that use the patient's own muscle signals to activate their desired movement.

The “hand of hope” needs a personal computer with specific programs and professionals to manage the program and help the patient if necessary.

Surely, there are exoskeletons printed in 3D. ZMorph (ZMorph Blog, 2018) created the first example, Figure 1.7 shows it. ZMorph is a company developed with the aim of introducing the 3D printing technology to all type of users, even not familiar with electronics, mechanics or engineering.

Figure.1.7: Exoskeleton printed by 3D printer 1.(ZMorph Blog, 2018)

They use a multitool 3D printer with different fabrication methods and materials (ZMorph3d.com, 2018). This device was designed by Eliza Wrobel in the University of Wroclaw; it has a mechanism for each finger including the thumb.
It does not use any actuator or motor to move the exoskeleton, it uses a manual mechanism.

With this exoskeleton, the rehabilitation is performed for all the fingers at the same time, which is a limitation in order to make the therapy for just one finger. This prototype was created to facilitate to a particular patient grab objects and perform physical activities, on balance the rehabilitation of a patient with reduced mobility in their hands. (7)

Another example of 3D printed exoskeleton is shown in Figure 1.8. Lei Cui in the University of Curtin, Australia, (Curtinnovation.com, 2017) created a prototype for one finger rehabilitation; however, it can only perform the downward movement of the finger since the design does not have any type of adjustment to enable an upward movement. As well as the device created by Rehab Robotics named previously, this exoskeleton uses a linear actuator.

Figure.1.8: Exoskeleton printed by 3D printer 2. (Curtinnovation.com, 2017)

Furthermore, most of the hand prototypes were design with other objectives, the main applications of them are to create virtual environments with haptic interaction. In this area, M. Bouzit, G. Burdea and G.P. Rares Boian (2002), describe the construction of a glove-exoskeleton that interacts with a 3D virtual environment in real time by pneumatic sensors, infrared sensors and Hall Effect sensors to measure the angle. Also, B.H. Choi and H.R. Choi in the article “A semi-direct Drive Hand Exoskeleton Using Ultrasonic Motor” describe this procedure. This device consists of a glove to sense objects in virtual environments using ultrasonic motors and sensors. To perform this project many different types of articles have been considered. (8)

On the following table, the features of the previously mentioned exoskeletons are shown:

Table.1.1 Exoskeleton features.

	Rehab Robotics	ZMorph Blog	Curtinnovation
Material	Steel	ABS	ABS
Complexity	High	Low	Medium
Weight	High	Low	Low
Size	Large	Large	Small
Individual finger rehabilitation	YES	NO	YES
Multiple fingers rehabilitation	YES	YES	NO
5 Fingers rehabilitation	YES	YES	NO
Portable	NO	YES	YES
Handling by professionals	YES	YES/NO	YES/NO
Requires a computer	YES	NO	NO
Actuator	Linear	Manual	Linear
Design modifications	NO	YES	YES
Multiples patients	YES	NO	NO
Control by sensor	YES	NO	NO
Nail control (Blood flow cut)	NO	NO	NO
Easy manufacturing	NO	YES	YES
Price	Expensive	Cheap	Cheap

Chapter Two
EMG Signal

2.1. Definition

Electromyography (EMG) signals can be used for clinical and biomedical applications development.

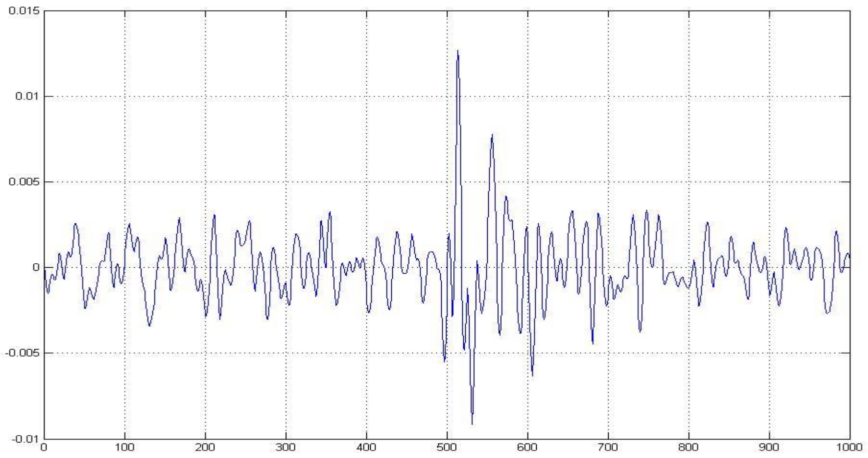


Figure 2.1 : Raw of EMG signal

This signal is normally a function of time and is describable in terms of its amplitude, frequency and phase.

The EMG signal is a biomedical signal that measures electrical currents generated in muscles during its contraction representing neuromuscular activities.The nervous system always controls the muscle activity (contraction/relaxation).Hence, the EMG signal is a complicated signal, which is controlled by the nervous system and is dependent on the anatomical and physiological properties of muscles.

The main reason for the interest in EMG signal analysis is in clinical diagnosis and biomedical applications. The field of management and rehabilitation of motor disability is identified as one of the important application areas. (9)

2.2. Anatomical and Physiological Background

When detecting and recording the EMG signal, there are two main issues of concern that influence the fidelity of the signal. The first is the signal-to-noise ratio. That is, the ratio of the energy in the EMG signals to the energy in the noise signal. In general, noise is defined as electrical signals that are not part of the desired EMG signal.

The other issue is the distortion of the signal, meaning that the relative contribution of any frequency component in the EMG signal should not be altered.

Two types of electrodes have been used to acquire muscle signal: invasive electrode and non-invasive electrode.

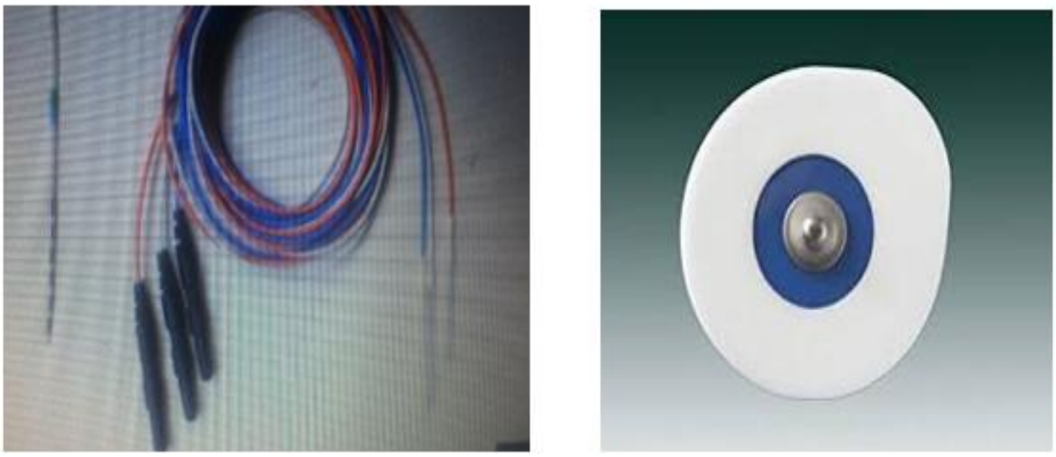


Figure 2.2: invasive electrode and non-invasive electrode.

When EMG is acquired from electrodes mounted directly on the skin, the signal is a composite of all the muscle fiber action potentials occurring in the muscles underlying the skin. These action potentials occur at random intervals. So at any one moment, the EMG signal may be either positive or negative voltage.

Individual muscle fiber action potentials are sometimes acquired using wire or needle electrodes placed directly in the muscle. The signal is picked up at the electrode and amplified. Typically, a differential amplifier is used as a first stage amplifier. Additional amplification stages may follow. Before being displayed or stored, the signal can be processed to eliminate low-frequency or high-frequency noise.

The nervous system is both the controlling and communications system of the body. This system consists of a large number of excitable connected cells called neurons that communicate with different parts of the body by means of electrical signals, which are rapid and specific.
The nervous system consists of three main parts: the brain, the spinal cord and the peripheral nerves.

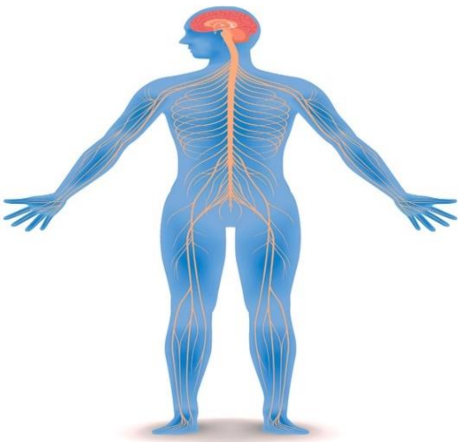


Figure 2.3: Nervous System.

Neurons are highly specialized cells that conduct messages in the form of nerve impulses from one part of the body to another.
The primary function of these specialized cells is to generate forces, movements and the ability to communicate such as speech or writing or other modes of expression.
Muscle tissue has extensibility and elasticity. It has the ability to receive and respond to stimuli and can be shortened or contracted. Muscle tissue has four key functions:
Producing motion, moving substance within the body, providing stabilization, and generating heat.
Three types of muscle tissue can be identified on the basis of structure, and control mechanisms:

(i) Skeletal muscle.
(ii) Smooth muscle.
(iii) Cardiac muscle.

The EMG is applied to the study of skeletal muscle. The skeletal muscle tissue is attached to the bone and its contraction is responsible for supporting and moving the skeleton.

2.3. Electrical noise and factors affecting EMG signal

The amplitude range of EMG signal is 0-10 mV (+5 to -5) prior to amplification.

EMG signals acquire noise while traveling through different tissue. It is important to understand the characteristics of the electrical noise. Electrical noise, which will affect EMG signals, can be categorized into the following types:
1. Inherent noise in electronics equipment: All electronics Equipment generate noise. This noise cannot be eliminated; using high quality electronic components can only reduce it.

2. Ambient noise: Electromagnetic radiation is the source of this kind of noise. The surfaces of our bodies are constantly inundated with electric-magnetic radiation and it is virtually impossible to avoid exposure to it on the surface of earth. The ambient noise may have amplitude that is one to three orders of magnitude greater than the EMG signal.
3. Motion artifact: When motion artifact is introduced to the system, the information is skewed. Motion artifact causes irregularities in the data. There are two main sources for motion artifact: (1) electrode interface and (2) electrode cable. Motion artifact can be reduced by proper design of the electronics circuitry and set-up.

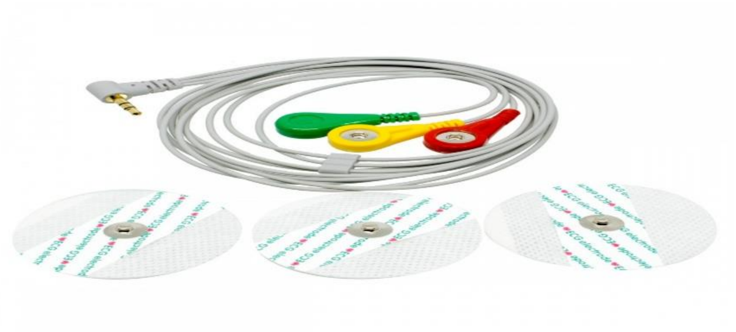


Figure 2.4: Electrodes.

2.4. Applications of EMG

EMG signals can be used for variety of applications like clinical/biomedical applications, human machine interaction, etc. Clinical applications of EMG as a diagnostics tool can include neuromuscular diseases.

EMG can be used to sense isometric muscular activity (type of muscular activity that does not translate into movement).This feature makes it possible to define a class of subtle motionless gestures to control interface without being noticed and without disrupting the surrounding environment .The device for this purpose includes a high input impedance amplifier connected to electrodes, an anti-aliasing filter, a microcontroller to sample and process the EMG signal, and a Bluetooth communication module to transmit the processing results. When activation is detected, the controller sends a signal wirelessly to the main wearable processing unit, such as a mobile phone. Using EMG, the user can react to the cues in a subtle way, without disrupting their environment and without using their hands on the interface. The EMG controller does not occupy the user's hands and does not require them to operate it; hence it is "hands free".

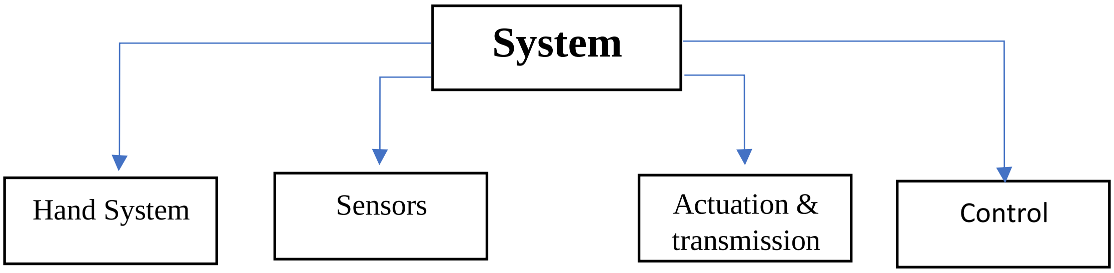
Interactive computer gaming offers another interesting application of bio-signal based interfaces. The game system would have access to heart rate, galvanic skin response, and eye movement signals, so the game could respond to a player's emotional state or guess his or her level of situation awareness by monitoring eye movements.

At the NASA Arms Research Center at Moffett Field, California, the extension of the Human Senses Group uses bio-control systems interfaces. They have used EMG/EEG signal in their research program on human interfaces to flight systems. The group seeks to advance man-machine interfaces by directly connecting a person to a computer via the human electrical nervous system. Based on EMG and EEG signals, this research applies pattern recognition system to interpret these signals as computer control commands.

These NASA researchers have used EMG signal to substitute for mechanical joysticks and keyboards. As an example, they developed a method for flying a high-fidelity flight simulator of a transport aircraft using EMG based joystick. Shows the flight control using EMG technology. The virtual joystick was actuated through an armband implanted with eight electrodes connected to sensors as the pilot gestures to land the aircraft. The pilot could also make emergency landings of a simulated aircraft that had been damaged.

Chapter Three
Project System

3.1. System classification



3.2. Hand System

3.2.1. Actuators on The Hand Of The Patient

we preferred to start with the above pictures as a try to give a self-explanation for this model, and here is another picture to give a clear and comprehensive about this model.

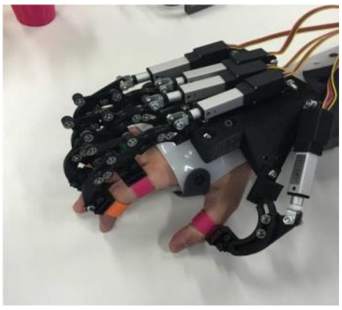


Figure 3.1:Actuators on The Hand of The Patient **1.**

As we can notice this model has some characteristics as follow:

The linear actuators are fixed on the package or we can say on the gloves which covers the hand of the patient according to this the freedom of movement of the hand of the patient is restricted because of this or we can say the patient will face some difficulties in movement because of the weight of these actuators.

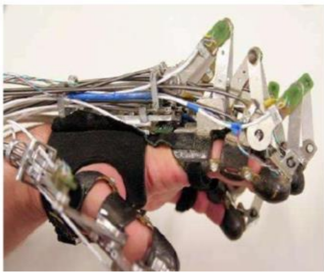


Figure 3.2: Actuators on The Hand of The Patient **2.**

These actuators may be heavy on the hand of the patient which may result in problems or we can say this design lost the advantage of flexibility.

3.2.2. Actuators are gathered in one bag

To be able in this model to present my point of view, I take the help from some pictures with different angles to be able to explain the characteristics of this model and explain the differences between this model and the last one as follow:

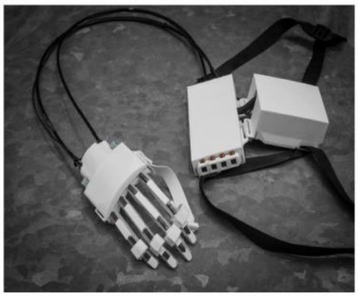


Figure 3.3: Actuators are gathered in one bag.

1-Flexibility of this model is very high compered to the last model because you gather all the linear actuators used to move the motors which are used to control the movement of the fingers and the hand.

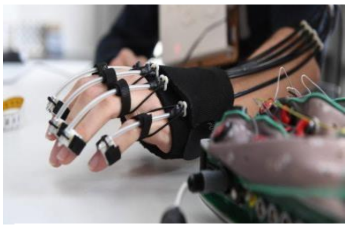


Figure 3.4: Actuators are gathered in one bag.

2-This model reduced the load on the patient hand because we gather all the linear actuators in a bag fixed on the back of the patient which results in more freedom in the movement of the hand of the patient.

3.3. Sensors

There is a lot of EMG sensors available in the market with different sizes, specifications and prices.

I. **Olimex ECG/EKG/EMG Arduino Shield**

Description: This SHIELD allows boards like Arduino to capture Electrocardiography and Electromyography signals.

It converts the analog differential signal (the ECG/EMG bio potentials generated by muscles), attached to its CH1_IN+/CH1_IN- inputs, into a single stream of data as output.

Figure 3.5: Olimex ECG/EKG/EMG Arduino Shield.

The output signal is analog and has to be discretized further with the aim to give the option of digital processing By detecting the electromyogram (EMG), measuring muscle activity has traditionally been used in medical research, however, With shrinking but more powerful microcontrollers and integrated circuits advent EMG power.

The shield opens new possibilities to experiment with biofeedback. You can monitor your heartbeat and log your pulse, recognize gestures by monitoring and analyze the muscle activity. (10)

Features:

1-Supply voltage (3v:12v)

2-Operating temperature (0-85)

3-Can be used in several applications as follow

4-Calibration signal generation by D4/D9 digital output

5-Input connector for normal or Active electrodes.

6-Works with both 3.3V and 5V Arduino boards.

Application:

-Fitness and sports heart rate monitoring

-Portable ECG

-Remote health monitor

-Game peripheral equipment

-Bioelectricity signal collection

II. Gforce Gesture Armband

Figure 3.6: Gforce Gesture Armband.

Description: GForce 100 Armband is a smart wearable Human Interface Device for gesture recognition. It recognizes gestures according to the EMG signals of human forearms, and as well as calculates orientation data in quaternions or Euler Angles from its built-in 6-axis IMU and tri-axis magnetometer.

GForce is used to identify gestures mainly through EMG signals. (11)

Features:

1- Re-charging sensor

GForce 100 Armband is equipped with Li-ion battery (200mAh). The USB port on the main block is used for battery re-charging. During re-charging, the red light LED on the main block is on. Recharging will maximally take 2 hours, and after re-charging completes, the red light LED is turned off

GForce 100 Armband is NOT designed to work during re-charging, as this brings in electrical noise which contaminates the weak EMG biometric signals.

2- This sensor has 8 channels EMG signal.

3- It supports wireless communication (Bluetooth range of up to 50 feet).

4- Its processor is ARM Cortex M4 with FPU.

5- Its battery can afford to ten hours, rechargeable with micro USB cable.

6- It supports only sex movements.

III. MyoWare Muscle Sensor

Description: This sensor will measure the filtered and rectified electrical activity of a muscle; outputting 0-Vs Volts depending the amount of activity in the selected muscle, where Vs signifies the voltage of the power source. (12)

Figure 3.7: MyoWare Muscle Sensor.

Features:

1- Single supply: MyoWare won't need +/- voltage power supplies! Unlike our previous sensor, it can now be plugged directly into 3.3V through 5V development boards.
2- Embedded Electrode Connector: Electrodes now snap directly to MyoWare, getting rid of those pesky cables and making the MyoWare wearable!
3- RAW EMG Output: A popular request from grad students, the MyoWare now has a secondary output of the RAW EMG waveform.
4- Polarity Protected Power Pins: Our #1 customer request was to add some protection so the sensor chips don't burn out when the power is accidentally connected backwards.
5- ON/OFF Switch: Speaking of burning out the board, we've also added an on-board power switch so you can test your power connections more easily.
6- LED Indicators: We've added two on-board LEDs one to let you know when the MyoWare's power is on and the other will brighten when your muscle flexes.

3.4. Actuation & transmission

3.4.1. Actuation

The function of actuation is to transform different kinds of energy to actuate the motion of robots. There are 3 kinds of actuation mentioned here: electrical motor, hydraulic, pneumatic.

I. Electrical Motor

The electrical motor is almost the most widely used actuation in design of hand rehabilitation robots, because they are easily available, reliable, and easy to control and with high precision.

Mostly used electrical motor types:

- **Servo motors:** A servo motor is a rotary actuator or a motor that allows for a precise control in terms of the angular position, acceleration, and velocity. Basically, it has certain capabilities that a regular motor does not have. Consequently, it makes use of a regular motor and pairs it with a sensor for position feedback.

Figure3.8: Servo Motor.

- **Stepper motor:** Stepper motors are DC motors that move in discrete steps. They have multiple coils that are organized in groups called "phases". By energizing each phase in sequence, the motor will rotate, one step at a time.

II. Pneumatic

The pneumatic actuators are used much less than the electrical motor in hand rehabilitation robots, such as the ASSIST designed by Sasaki et al. This actuator has advantages such as less requirement of maintenance and can be stopped under a load without causing damages. Although problems like noise can be overcome by using recompressed air storage, the problem of size cannot be settled because the air storage chamber is necessary. Thus, the pneumatic actuator might better be used for systems with lower mobility.

Figure3.9: pneumatic actuator.

The development of pneumatic artificial muscle makes the actuation another choice. The pneumatic muscle made of rubber inner tube with a shell can inflate or contract. For example, the commercial hand rehabilitation robotic system produced by Kinetic Muscles Inc. (USA) is actuated by air muscle actuator. There is also another kind of pneumatic muscle, namely, the bending type pneumatic muscle. For example, the pneumatic rubber muscle was designed by Noritsugu et al. The disadvantage of pneumatic actuation is that the actuators are difficult to control for its time variability and nonlinear.

III. Hydraulic

The hydraulic actuators are very good in performance such as can generate higher torque compared with the electric or pneumatic systems and can be controlled in high precision and frequency. But, the requirement of a wider space to accommodate the oil transmitting pipes and conduits makes the hydraulic actuators seldomly used in hand rehabilitation robots.

3.4.2. Transmission

The function of transmission is to transform the motion of actuator into a desired direction to complete the execution of a hand's motion. Most of these are a consequence of the choices of actuator or the mechanism.

I. Linkage

Linkages are popular choices in the hand rehabilitation robot system, the same as in the traditional mechanical design. The linkages are light, convenient, and can be easily controlled in a given trajectory. On the one hand, the problem of coincidence of the rotational axis can be settled by using the cross-joint structure. On the other hand, the complexity of device can be reduced according to the concept of fDOF by using the linkage structure.

II. Cable

The cable is also frequently used as the transmission in the hand rehabilitation robots, including the pulley cable and Bowden cable. The pulley requires a continuous tension to maintain traction on the pulleys, which limits the use. On the other hand, the Bowden cable is better for its cable conduit and is flexible. Disadvantages are the variable and high

Figure 3.10: Bowden cable.

friction force caused by the curve. These devices can be easily controlled by force while not so convenient in usage.

3.5. Control

The market offers a wide range of Microcontroller with different specs according to the used application.

I. AVR

AVR is a family of microcontrollers developed since 1996 by **Atmel**, acquired by Microchip Technology in 2016. These are modified Harvard architecture 8-bit RISC single-chip microcontrollers. AVR was one of the first microcontroller families to use on-chip flash memory for program storage, as opposed to one-time programmable ROM, EPROM, or EEPROM used by other microcontrollers at the time. (13)

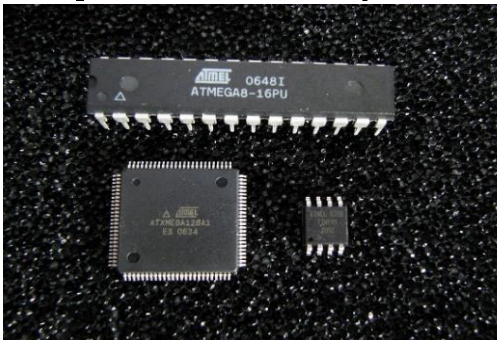


Figure 3.11: AVR microcontroller.

II. ARM

The ARM-Cortex microcontroller is a most popular microcontroller in the digital embedded system world and most of the industries prefer only ARM microcontrollers since it consists of enormous features to implement products with an advanced appearance. The ARM microcontrollers are cost sensitive and high performance devices which are used in a wide range of application such as industrial instrument control systems, wireless networking and sensors and automotive body system etc. (14)

Figure 3.12: ARM-Cortex microcontroller.

III. PIC

PIC (Programmable Interface Controllers) microcontrollers are the world's smallest microcontrollers that can be programmed to carry out a huge range of tasks. These microcontrollers are found in many electronic devices such as phones, computer control systems, alarm systems, embedded systems, etc.

Figure 3.13: PIC (Programmable Interface Controllers) microcontrollers.

Various types of microcontrollers exist, even though the best are found in the GENIE range of programmable microcontrollers.
These microcontrollers are programmed and simulated by a circuit-wizard software. (15)

3.6. Survey

I. Hand system

In this model, a complete glove will surround the hand of the patient with totally way, and we thought to make this glove we need to print it using 3D printer.

We take this way and we made design and printed gloves as follows:

Figure 3.14: a complete glove 3D Printed.

But after all we did, we started to think about new method to do the hand system because our design is failed according to these reasons:

1- This design is very complex and it needed special 3D printer to print it with all its details.
2- It is very expensive
3- The used material to print it is rubber which may be abuse the patient.
So we asked ourselves a question, what is the solution now?
We have switched to another way using finger rings and wrist bracket to be more flexible.

The hand system, mainly consists of two rings fixed on each finger these rings are used to move the finger of the patient backward and forward. The measured distance will be 5cm forward and 5cm backward to get a total distance which is equal to 10cm.

The used rings are made by 3D printer and the used material in it is plastic to reduce load on the fingers of the patient.

Figure 3.15: Hand system consists of two rings.

The rings will used to move the finger by Bowden cables, fixed with these rings and these cables will deliver the motion generated from the motors to control the motion. So, from what we mentioned we used hand system using rings as our hand system in our project.

II. Sensors

After searching we have found that the Olimex Arduino shield is best one for our project as it is more accurate than others, it’s price is low and it availability in EGYPT.

We used ECG/EKG/EMG Shield on one of our team member where two electrodes are fixed on middle of muscle and the third one on the end of the muscle.

And we managed to get some results as follow:

1-our mate begins to move his fingers and his hand and we recorded some results as follow

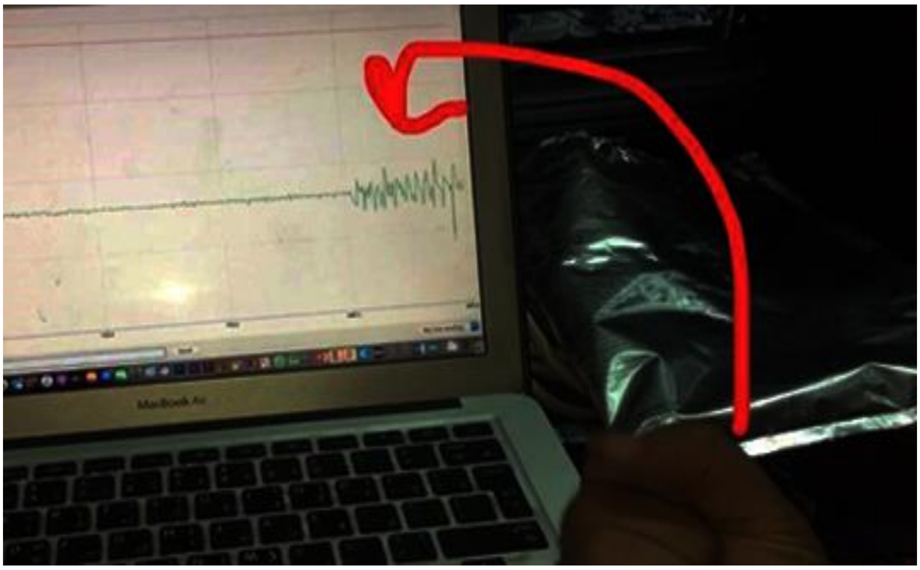


Figure 3.16: Recording results 1.

2-In case of there is no movement in the hand or fingers

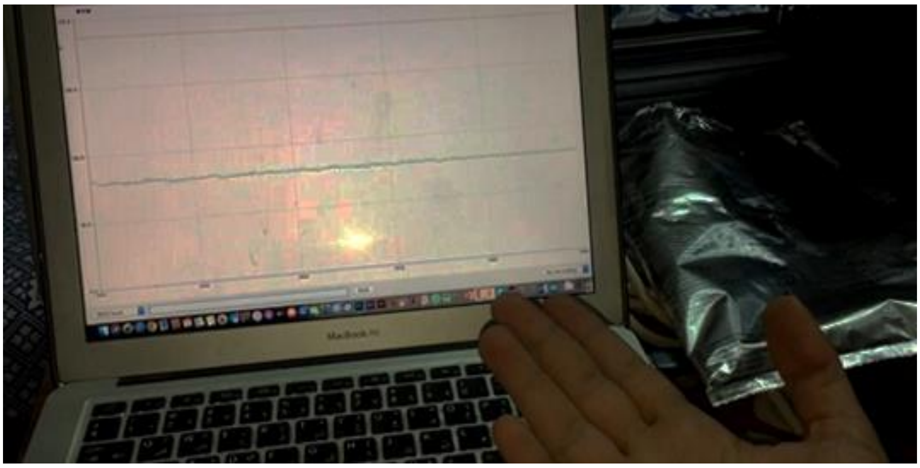


Figure 3.17: Recording results 2.

III. Actuation

We have used Continuous servo motors (Figure 3.5) in our project instead of using pneumatic or hydraulics because they need bump for either oil or gas one and managed to convert its rotational motion to linear motion using gear and linear grid (design in chapter 5).

IV. Transmission

Cables are traditional way to be used in transmission because it is very simple, flexible and suitable under any conditions where there are not any restrictions to use this type of transmission. Bowden cables are the type of cables that we are going to use it(Figure 3.7).

V. Control

We are restricted in control part because the used shield to get raw EMG signal is programmed only with Arduino so to deal with EMG signal, we used Arduino board and we programmed it with Arduino c. (16)

Figure 3.18: Arduino board

We have an ambition which is in next part of our project we are planning to apply several process on our signal like (correlation, mapping and recording) to differentiate between each finger. This can be done by using raspberry pi which will be used to do all of this enhancement. (17)

Figure 3.19: Raspberry pi board.

Chapter Four
System Control

4.1. Introduction

The system control is about taking the EMG signal as input from the Arduino (EMG Shield) and digital process this signal then take a decision and generate the output which is the motor actuation.

4.2. Processing of EMG data

Raw EMG data needs to be processed before it can be used for analysis.

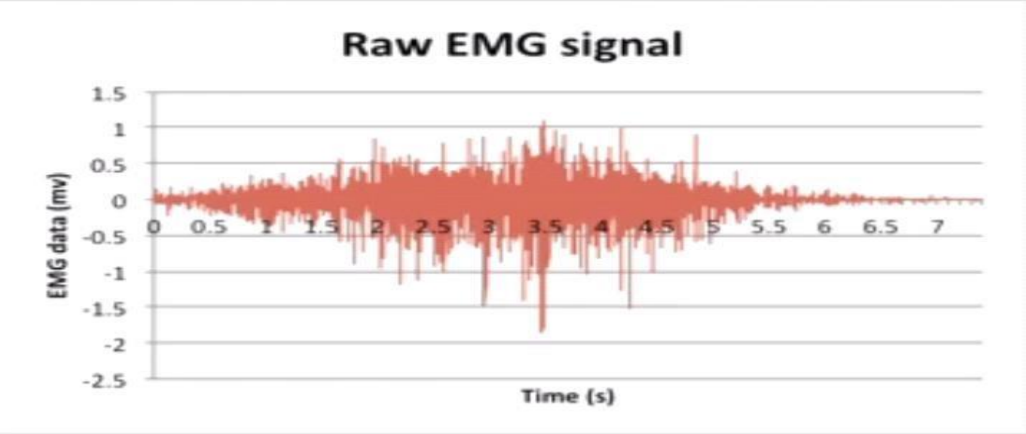


Figure 4.1: Raw EMG signal

Data Processing:

- **Noise filtering**
- **Rectification.**
- **Smoothing**
- **Normalization**

I. Noise filtering

Removing noise that is contaminating EMG signal 60 Hz noise
"From electronic and electric outlets"
-low frequency (<10 HZ) and high frequency (>350 HZ)

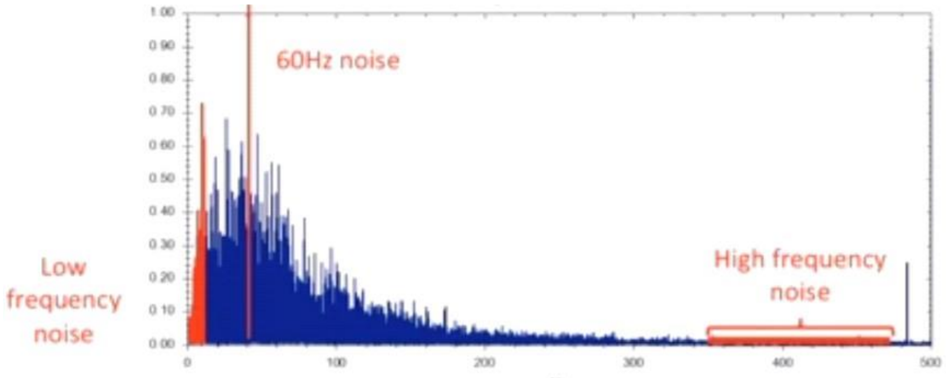


Figure 4.2: The expected affected noise

I. Rectification

Rectification means to take the absolute values

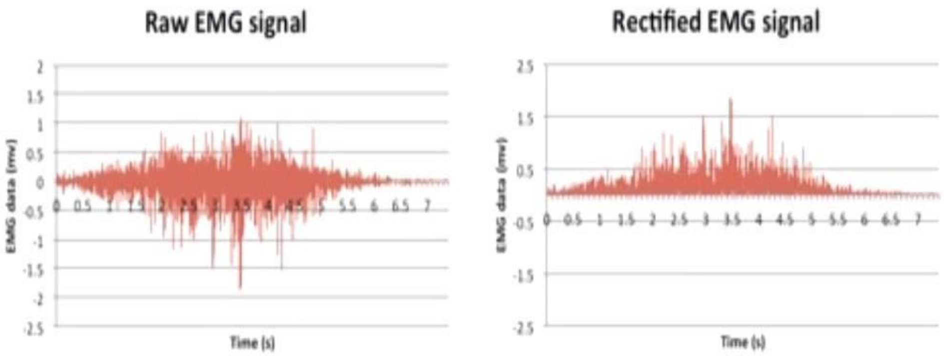


Figure 4.3: EMG signal rectification.

II. Smoothing

Smooth the data using moving average window.

-taking the average of certain duration (window) of data 50-100ms

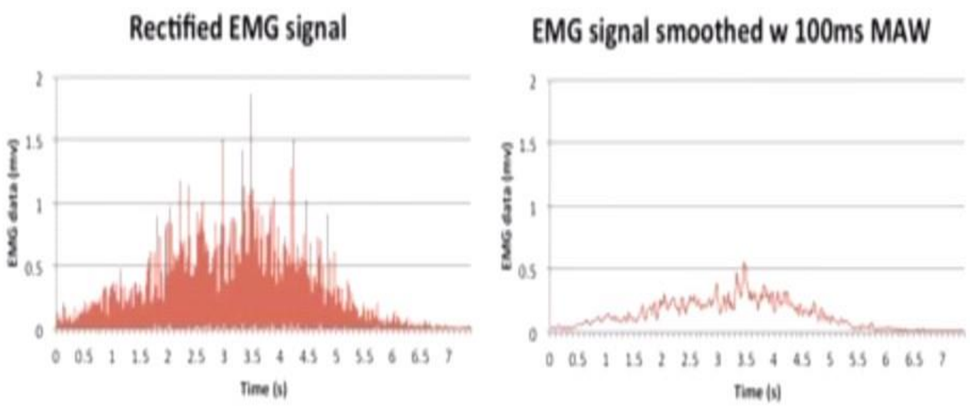


Figure 4.4: EMG signal smoothing.

III. Normalization

Expressing the EMG signal data as a percentage of maximum voluntary muscle contraction

-normalized EMG value=Raw EMG value /MVC *100

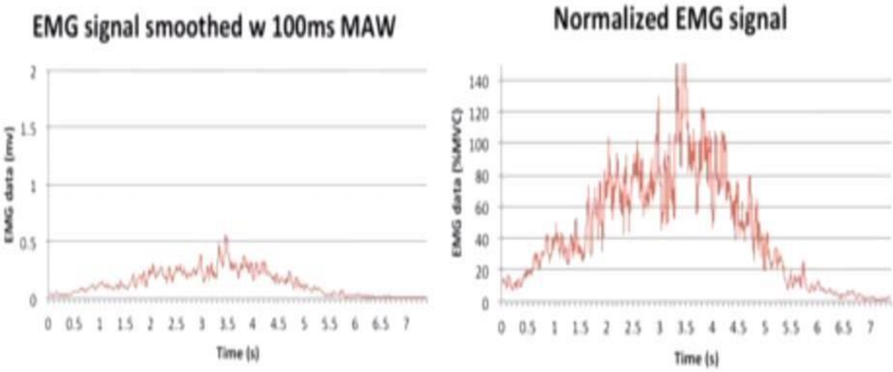


Figure 4.5: EMG signal normalization.

4.3. Correlation

In general, correlation describes the mutual relationship which exists between two or more things. The same definition holds good even in the case of signals. That is, correlation between signals indicates the measure up to which the given signal resembles another signal.

In other words, if we want to know how much similarity exists between the signals 1 and 2, then we need to find out the correlation of Signal 1 with respect to Signal 2 or vice versa**.**

4.3.1. Types of Correlation

Depending on whether the signals considered for correlation are same or different, we have two kinds of correlation: autocorrelation and cross-correlation.

I. Autocorrelation

This is a type of correlation in which the given signal is correlated with itself, usually the time-shifted version of itself.

Similarly, the autocorrelation of the discrete time signal $x[n]$ is expressed as

$$Rxx[m] = \infty \sum n = -\infty x[n] x \star [n-m]$$

II. Cross-Correlation

This is a kind of correlation, in which the signal in-hand is correlated with another signal so as to know how much resemblance exists between them.

The cross-correlation of the discrete time signals $x[n]$ and $y[n]$ is expressed as

$$Rxy[m] = \infty\sum n = -\infty x[n]y \star [n - m]$$

4.4. Signal Mapping

Re-maps a number from one range to another. That is, a value of fromLow would get mapped to toLow, a value of fromHigh to toHigh, values in-between to values in-between, etc.
Does not constrain values to within the range, because out-of-range values are sometimes intended and useful. The constrain () function may be used either before or after this function, if limits to the ranges are desired.
Note that the "lower bounds" of either range may be larger or smaller than the "upper bounds" so the map () function may be used to reverse a range of numbers, for example
y = map(x, 1, 50, 50, 1);
The function also handles negative numbers well, so that this example
y = map(x, 1, 50, 50, -100);
Is also valid and works well.
The map () function uses integer math so will not generate fractions, when the math might indicate that it should do so. Fractional remainders are truncated, and are not rounded or averaged. (18)

TIP: map () can convert from positive ranges to negative ranges.

Syntax
map (value, from Low, from High, to Low, to High)

Parameters
Value: the number to map.
fromLow: the lower bound of the value’s current range.
fromHigh: the upper bound of the value’s current range.
toLow: the lower bound of the value’s target range.
toHigh: the upper bound of the value’s target range.

4.5. Notes and Warnings

As previously mentioned, the map () function uses integer math. So fractions might get suppressed due to this. For example, fractions like 3/2, 4/3, 5/4 will all be returned as 1 from the map () function, despite their different actual values. So if your project requires precise calculations (e.g. voltage accurate to 3 decimal places), please consider avoiding map () and implementing the calculations manually in your code yourself.

4.6. Recording Process

One of the most important and vital process or stages in our project is recording, but firstly we have to ask ourselves a crucial question which is **Why we need the recording stage,** to answer this question we need to know how our project works, we take EMG signal from the muscles of a certain patient and apply certain operations to performs the functions he did not have the ability to do because of his disability, so to do these operations we need a reference for it and we can take this reference by recording process.

As we mentioned the main target from this process is helping the patient doing his normal functions this can be done for example by capturing signal from the patient and record it then comparing it with another new one to know the similarity between them and starting doing certain function or we can use different other ways to enable the patient doing his normal functions. Another reason foe needing recording is that each muscle or organ in our bodies as each organ or muscle has a certain and unique one so to deal with these signals to enable the patient to do a certain function we need to understand which signal related to which muscle to enable me apply the processing that helps the patient at the end to do its original functions.

4.6.1 How we will record EMG signal

As we have searched, we found many ways to do this process, and know we will mention some of these ways and their advantages and disadvantages.

4.6.1.1 Recording using built in functions in MATLAB

One of the easiest ways to record a signal thus MATLAB provides some built in functions like(record blocking, audio recorder,…) to perform this task, but we found that this method is good in case of voice signal "if you want to record a voice" where this method allows you to record several seconds and ends recording after this time and storing the recorded data in numeric array, but as we mentioned this method is good in case of voice signal where it allowed to record using microphone which it is not available in our project where we can record our data using USB cable not a the microphone in laptops so we searched for a new method that can be suitable and compatible with our project. (19)

```
Command Window
>> f=audiorecoder(8000,8,1);
Undefined function or variable 'audiorecoder'.

Did you mean:
>> f=audiorecorder(8000,8,1);
>> record(f,5)
>> b=getaudiodata(f);
>> plot(b)
```

Figure 4.6: Recording Code using MATLAB.

4.6.1.2. Recording EMG Signals on a Computer Sound Card

In this method, we are going to use hardware and software not only software like in the previous method. This can be done by cheap external sound card and a laptop computer to be used as a portable data acquisition system for recording EMG. The circuit uses a common audio amplifier integrated circuit to increase the gain of the EMG signals recorded from EMG surface electrodes and to match the impedance of the electrode-skin interface with the input impedance of the sound card. Data can be recorded using open source sound editing software and analyzed offline using simple Python code.

In this method Arduino-powered MyoWare system is used to capture the EMG signal from human muscles to perform operations on it but as we mentioned in chapter 2 the MyoWare system is not a good choice for capturing raw EMG signal as it cannot be used in human uses because its very low quality and it is launched to be used with animals. But we solved this problem by using the EMG shield “the one we used to generate EMG signal and we spoke about it in chapter 2” so we can replace MyoWare system by EMG shield.

In this method we need to adjust some things which this method will not work without it as follows:

1- surface EMG requires a sampling rate of 1 kHz or higher because Sampling at a lower frequency can lead to signal fidelity problems, especially aliasing this occurs when the rate of change of the waveform is more rapid than the sampling rate.

2-We need to make impedance matching between the electrode-skin interface with the input impedance of the sound card, where the sound card has an input impedance that is typically 10 kΩ and Dry skin can have a resistance as high as 100 kΩ.

When recording EMG, we are dealing with two resistances in series (skin and sound card). Two resistors in series make a voltage divider; when a voltage is applied across these resistors, the voltage drop across each resistor is proportionate to the magnitude of that particular resistor so the need of a buffer amplifier which is a device that matches the impedance of one device to another so An ideal operational amplifier would have an infinite input resistance and an output resistance of zero (although real amplifiers are never perfect). The bigger the input impedance of the amplifier, the more of the EMG signal's voltage is fed to the amplifier. The lower the output impedance of the amplifier, the more of the EMG signal's amplified signal is fed to the sound card. In short, we want the skin resistance to be very small compared to the amplifier's input resistance, and we want the sound card's resistance to be very high compared to the amplifier's output circuit.

So, in this design the output resistance is designed to drive loads with resistances as low as 4Ω, so it is plenty low enough to match our 10kΩ sound card input resistance.

The LM386 integrated circuit is an easy to use audio amplifier that can be used to record EMG on a sound card.

Figure 4.7: The LM386 integrated circuit.

To build this circuit ins one and eight are used to modify the gain (volume) of the chip. The gain can be either 20× (if nothing is attached to these pins) or 200x, if these two pins are connected by a 10 μF capacitor. Pins two and three are the two inputs, which will be configured for use as a differential amplifier. The two skin surface electrodes arranged along the muscle from which the EMG is to be recorded are connected to these two pins by way of ceramic capacitors. It does not really matter which electrode is connected to which pin. Pin four is the ground, and it is connected to four very important points. First, it is connected to the electrode, which might be over the elbow if recording from the biceps brachii muscles. Second, it is connected to the ground plate, a roughly square sheet of aluminum foil, the use of which will be described in greater detail in the “Minimizing Noise” section. Third, it is connected to the negative pole of the 9V battery. Finally, it is connected to the grounded sheath of the connector that plugs into the microphone port on the USB sound card.

There are two additional paths to ground in this circuit. The first is from pin 5, by way of a 10 ohm resistor and a ceramic capacitor in series, such that pins 4 and 5 are essentially connected together via a resistor and a capacitor. The second path to ground is from pin 6 via an electrolytic capacitor. Again, all the ground points indicated in the circuit are connected to each other, to the ground electrode on the skin, to the negative side of the battery, to the ground of the sound card (as described below) and to the ground plate (aluminum foil). Note that the electrolytic capacitors are polarized, and the positive sides should be oriented toward pins 5, 6 or 8, depending on which pin the electrolytic capacitor is connected to. Pin five is the output of the amplifier integrated circuit and is connected to the USB sound card by way of an electrolytic capacitor.

Pin 6 is connected to the positive pole of the 9V battery. The LM386 takes 5V however, so pin 6 will be connected via a 1K resistor. Pin 7 is left open (unconnected).

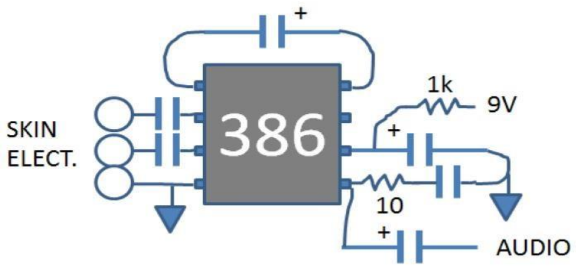


Figure 4.8: Simplified diagram of the LM386 circuit.

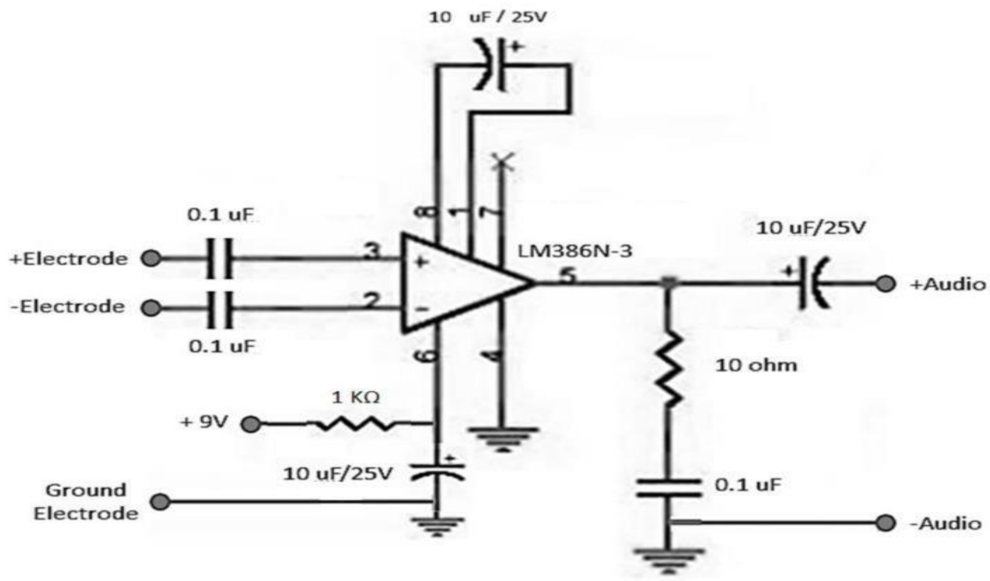


Figure 4.9: Schematic of the LM386 circuit.

One of the benefits of using a computer sound card for data acquisition is that there are numerous software packages for analyzing and filtering signals recorded in this method. We have worked principally with Audacity (www.audacityteam.org) because it is platform-independent, free, and has many useful features, we have also experimented with writing our own custom software for data acquisition and analysis in the Python programming language using the pyaudio and wave libraries. Simple algorithms for removing DC offset, high and low pass filters, notch filters for 60 Hz line noise and plotting power spectrums are not terribly difficult to code, and there are many examples on the web of how to do this in Python.

EMG signals can be adequately sampled at as low as 1000 Hz. However, Audacity defaults to a sample rate of 44.1 kHz (CD quality). We typically turn the sampling rate down to the minimum, 8 kHz which is still well above the necessary rate for EMG.

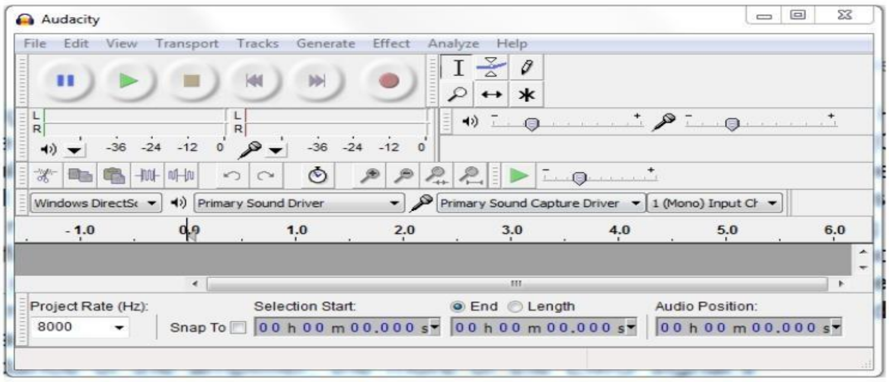


Figure 4.10: Audacity user interface.

When the amplifier circuit has been connected to a USB sound card, this USB sound card is selected as the input. There are actually three microphones on the Audacity user interface. The one to the right with the +/− slider controls the amount of volume amplification coming through the acquisition device. This can be adjusted to optimize signal-to-noise by visual inspection. The red circle button is the recording button; as soon as this button is clicked, so long as the device is connected, powered by a battery, and connected to a human, signal should be visible as it is recorded in the signal window of Audacity.

But there is a problem we face which is noise and Fortunately, there are some simple ways to eliminate noise using this circuit. First, acquiring data on a laptop that is running off of a battery will eliminate the direct connection between the amplifier circuit and the 60 Hz AC power lines that cause so much disturbance. There will still be 60 cycle noise from various proximal radiators, such as fluorescent lights and equipment in the lab space. The grounding plate of aluminum foil reduces this radiated noise considerably. Another noteworthy free software package for processing this type of signal is Backyard Brain's Spike Recorder application (www.backyardbrains.com).

It is fairly easy to construct Python scripts to perform custom analysis of EMG recordings. However, because many python libraries are likely to be involved, it is recommended that users take advantage of the Anaconda (www.anaconda.com) development environment for Python (Spyder) as this package automates the library installation and handling process.

A very easy way to work with audio recordings in Python is by recording the signal in Audacity as a 32-bit mono signal with a sampling frequency of 8000 Hz, selecting the EMG burst you wish to analyze and saving it as a .wav file format. By using free open source software like we mentioned we can apply many process on the required raw EMG signal like filtration, amplification and rectification to enable me to deal with the required signal with smooth way but fortunately we do not need to do this as our EMG shield getting out the raw EMG signal after applying these process on it so it save several lines of codes it would have written in case of we do not have this shield.

After clearly explaining this method, we need now to ask ourselves if this method suitable to our project or not? And the answer is “no” as this method enabled us to require the raw EMG signal, we want but this circuit is very complex.

that we need to make impedance matching between the skin and the used sound card and constructing a circuit with some components which may be has a bad quality which may affect our process then we need to fix this circuit on the hand of the patient which may abuse him, so as a result of all these reasons we found it is important to search for new method and we managed to find one and we will explain it in next lines.

4.6.1.3. Recording using Simulink on MATLAB

This is the method we used to record our EMG signal using it and it can be done by software only using “Arduino support package” on MATLAB and without using any hardware.
This method can be done by two steps as follow:

i) Capturing the wanted raw EMG signal
As we mentioned, this can be done on Simulink of MATLAB by using the two blocks (Analog i/p & to file), these two blocks enable me to record the signal.

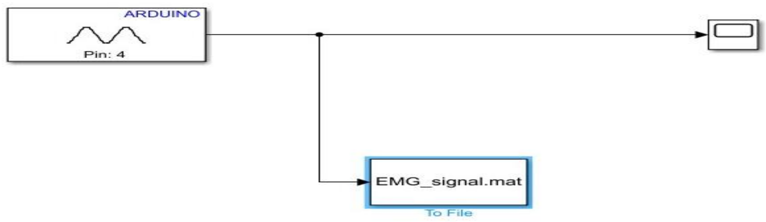


Figure 4.11: Capturing raw EMG signal.

After applying this stage, we managed to capture the wanted signal and here is a photo for what we succeeded to do.

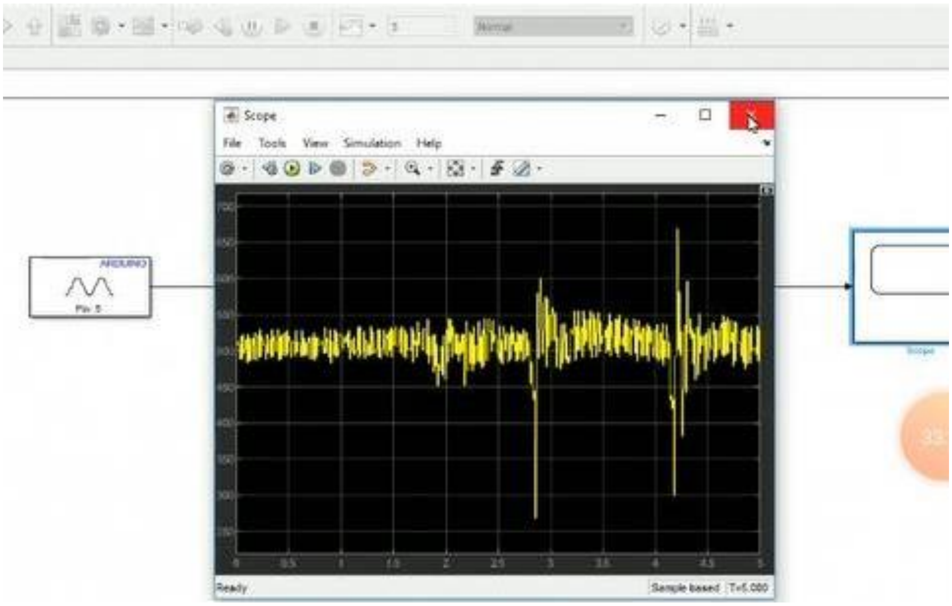


Figure 4.12: Captured raw EMG signal.

ii) Calling the recorded EMG signal

After captured the raw EMG signal and recorded in to “file”, and to apply our process on it we need now to call the captured signal by using “from file” block.

Figure 4.13: Calling the captured EMG signal.

After applying these blocks on Simulink and adjusting the name of the file we recorded the signal on it, we finally managed to call and display our EMG signal that we will use to apply our operations on it.

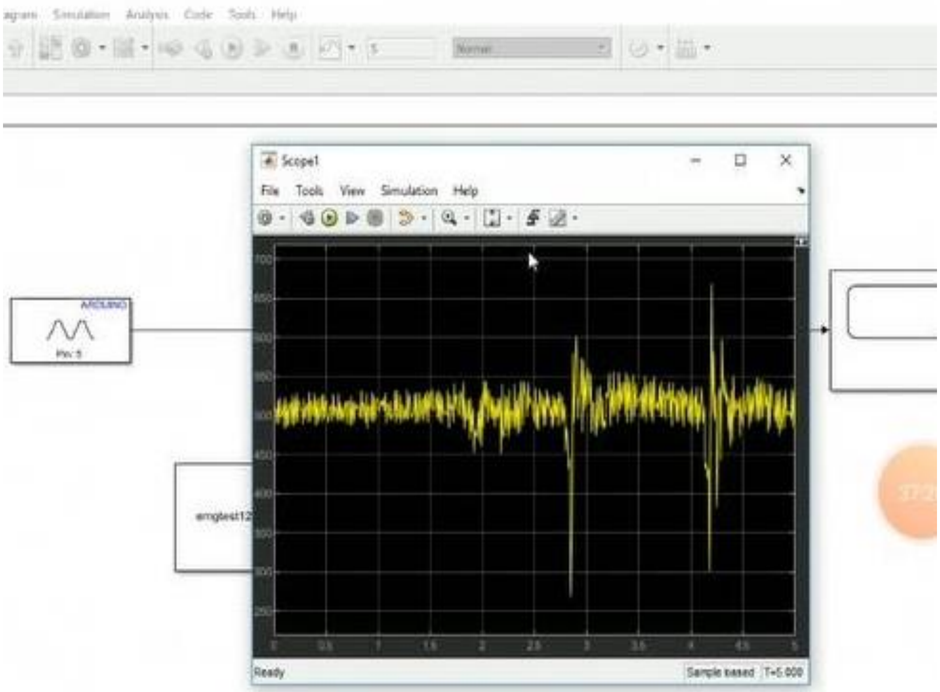
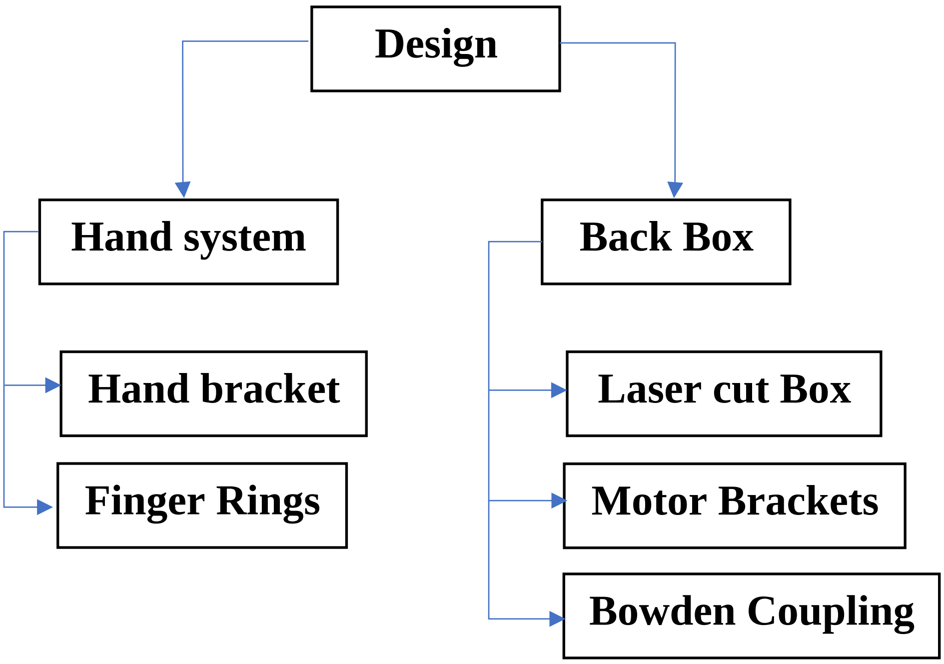


Figure 4.14: Recorded EMG signal.

As a conclusion for what we did in this part, we searched for several methods as a try to evaluate the best scenario and method to record the raw EMG signal, and we finally found what we seek for, we did this process with minimum cost and complexity away from using the second way and we managed to do what we want “To record EMG signal to use it in our process” to enable any patient using his hand with normal way.

Chapter Five
System Design

5.1. Design Classification



Design materials

We are using acrylic and PLA in our design as a material, so our system is half 3d printed and half laser cut acrylic.

The Box itself is made from the laser cut acrylic and the other parts are 3d printed with PLA (Hand bracket, Finger rings, motor brackets, Bowden cable coupling).

In the next pages we will present all the engineering drawings for our parts and its 3d models

5.2. Hand System

I. Hand bracket

Hand bracket is used to fix the Bowden cable cover on the wrist so it can give motion to finger rings.

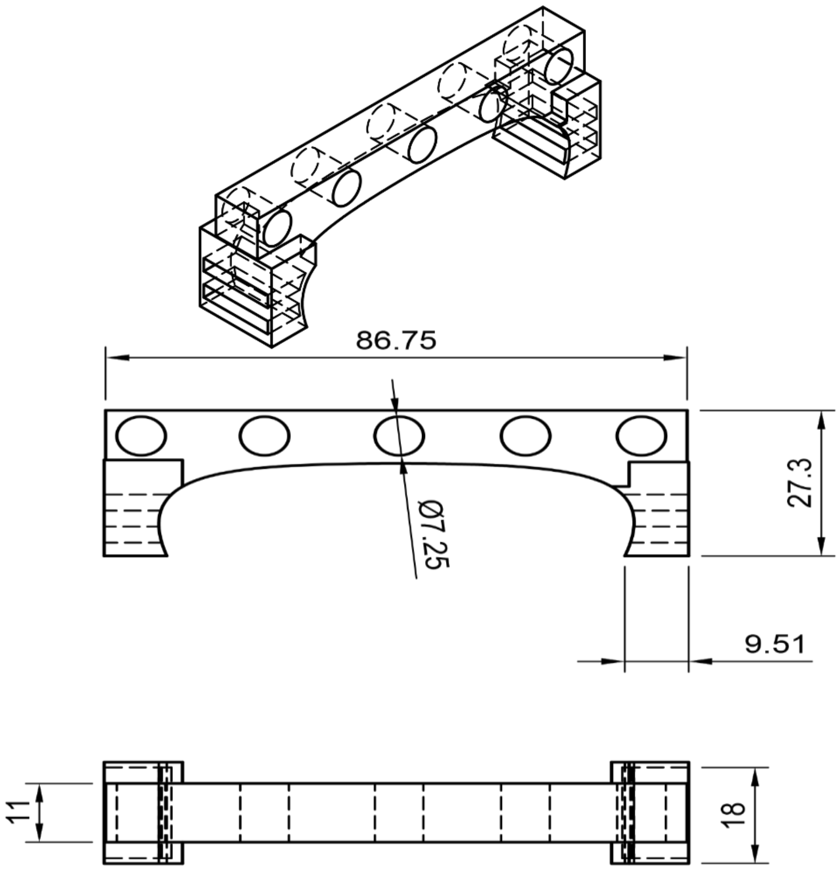


Figure 5.1: Hand bracket.

II. Finger Rings

It was made for the inner wire of the Bowden cable to make the motion on the fingers by bending it.

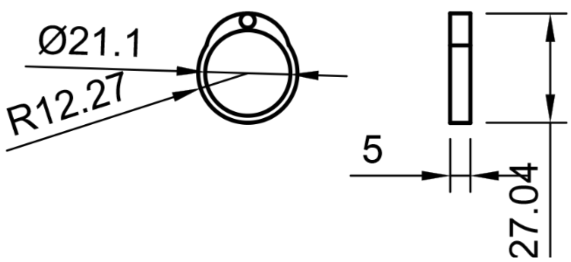


Figure 5.2: Finger Rings.

5.3. Back Box

I. Laser Cut Box

The box itself made of 6mm Acrylic sheet for reducing Cost instead of 3d printing the whole box, it is made with interlocks for assembly after laser cutting.

Box Front

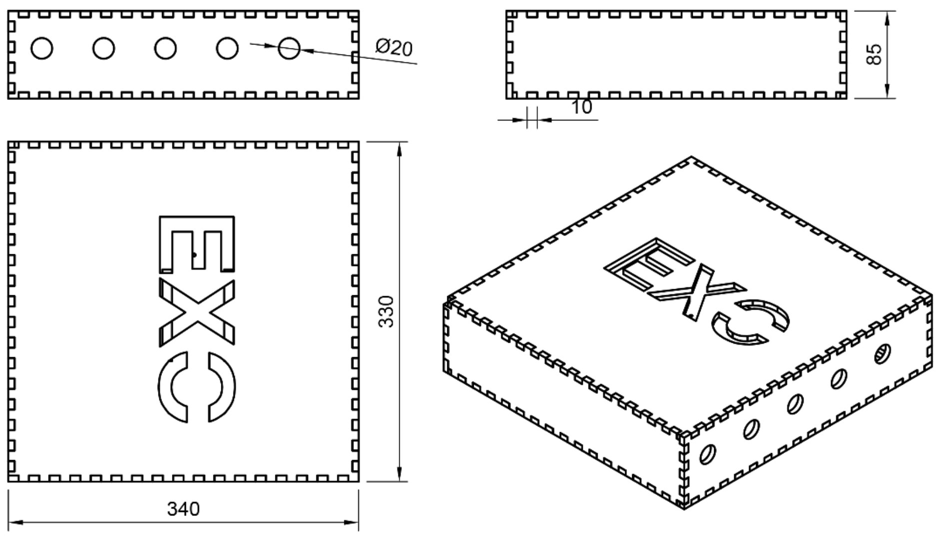


Figure 5.3: Back Box front view.

Box Back

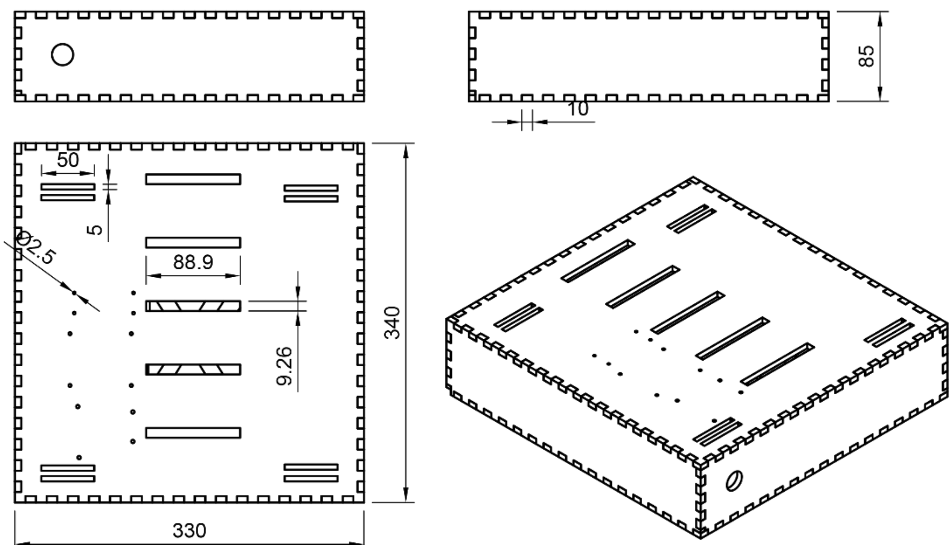


Figure 5.4: Back Box back view.

II. Motor Brackets

It is made to fix the servo motors to box and also it can change the rotational motion of the motor to a linear motion by using gear and a linear grid.

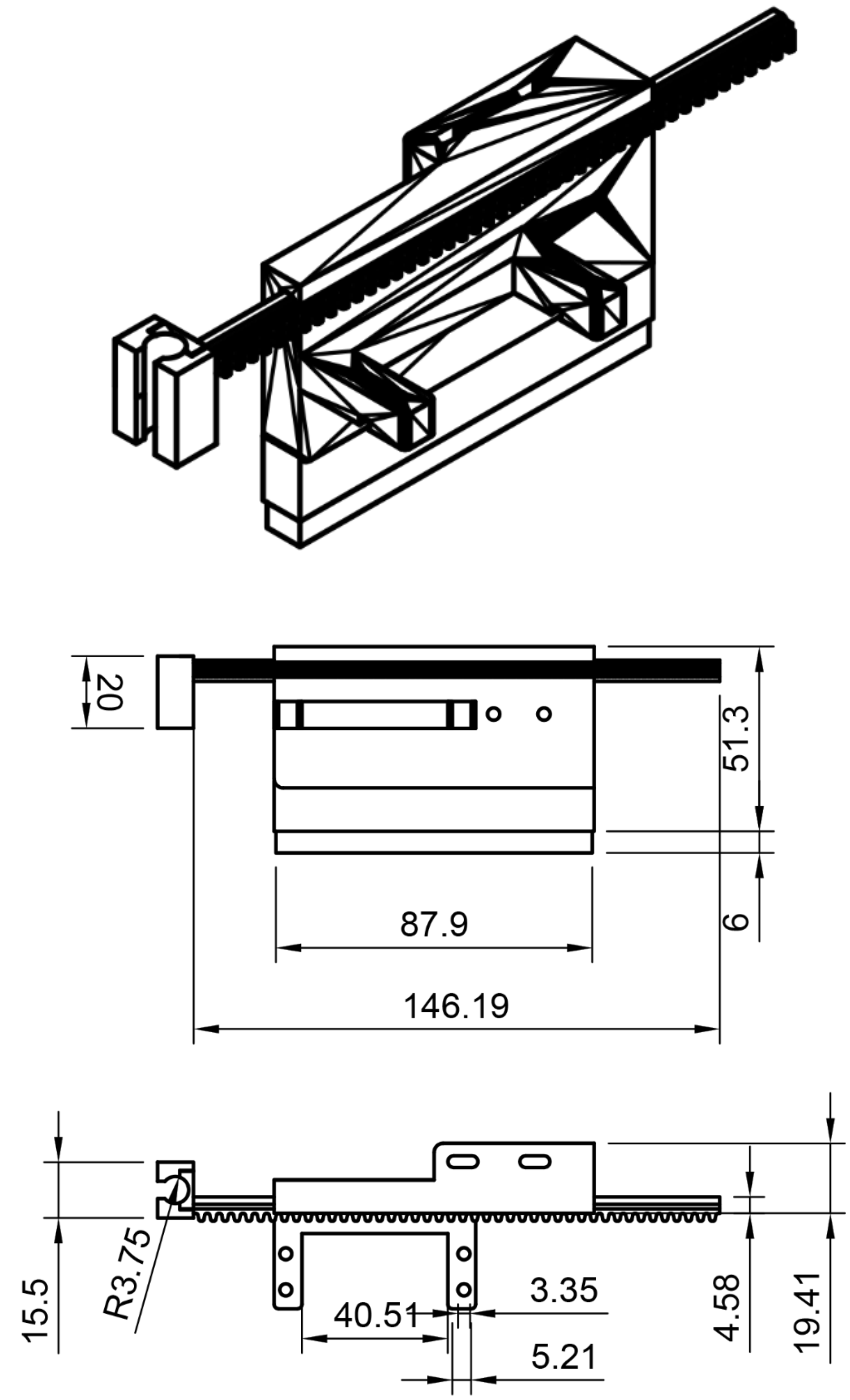


Figure 5.5: Motor Brackets.

III. Bowden Coupling

It is made to fix the other end of the Bowden cable to box it self to receive the motion from the motor and transmit it to the hand system.

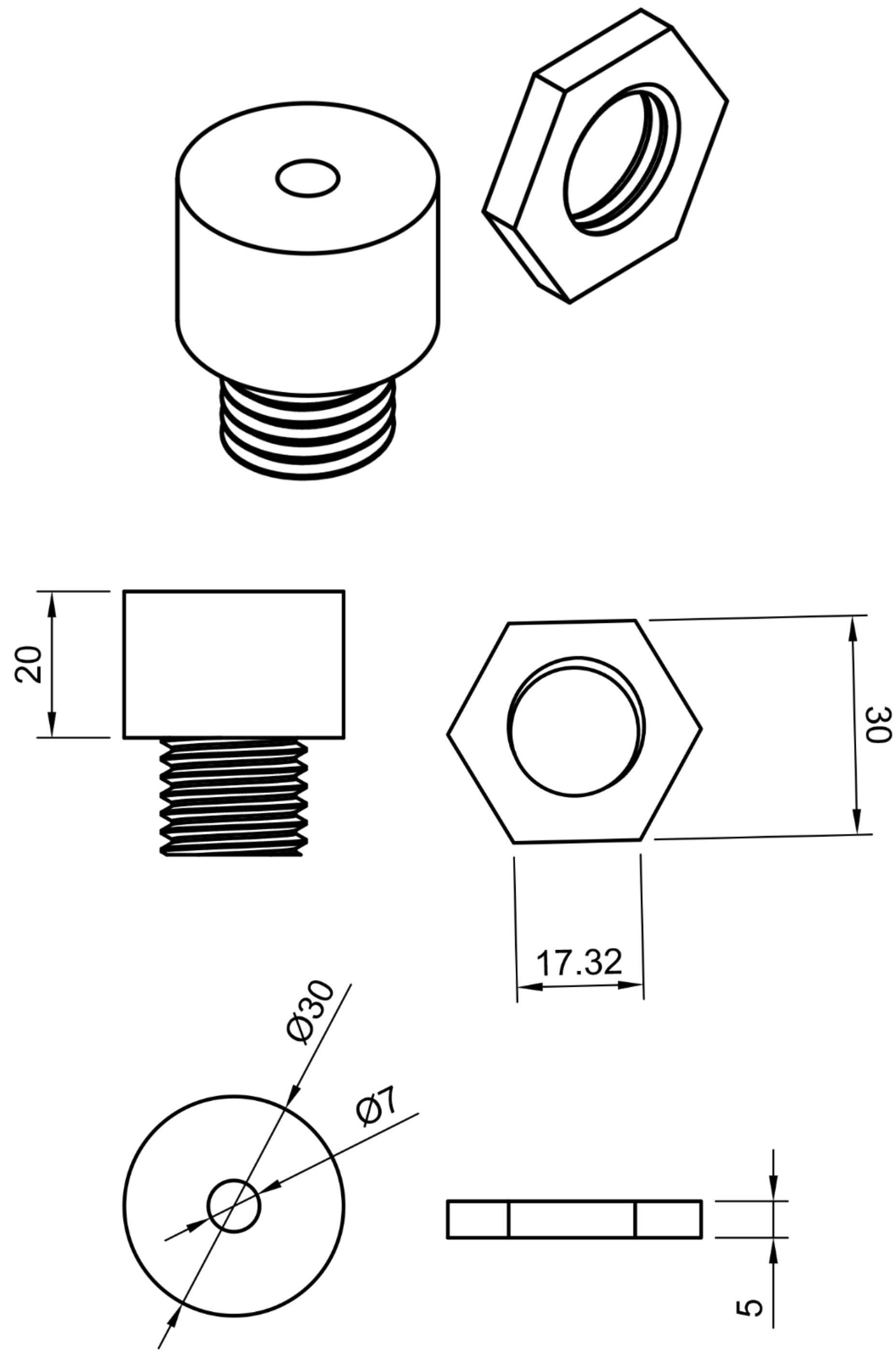
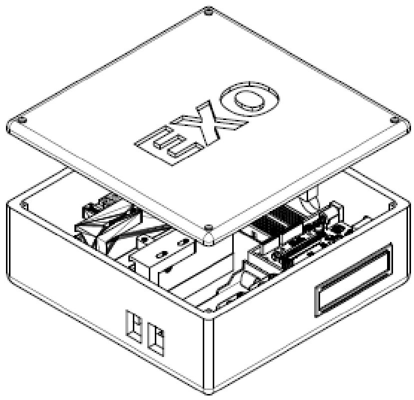


Figure 5.6: Bowden Coupling.

5.4. Enhancement of System Design



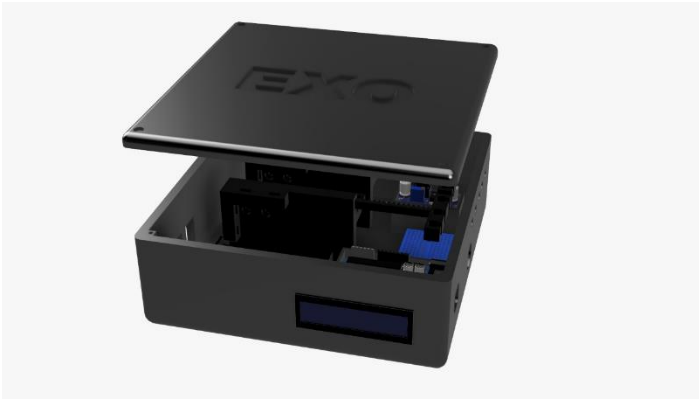


Figure 5.7: New design for box.

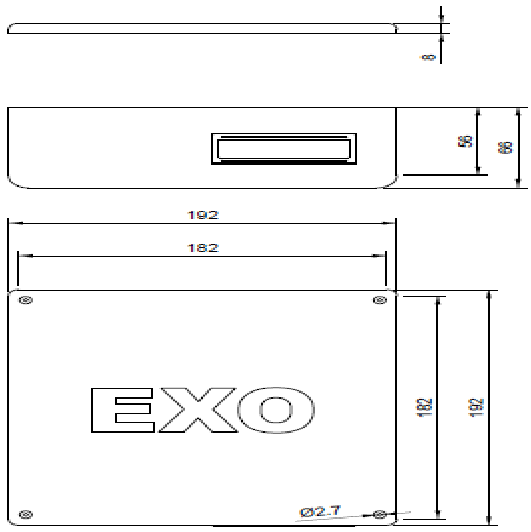


Figure 5.8: Elevation and plan view for new Design.

Chapter Six
System Implementation

From the need to work with electrical biopotentials to control exoskeleton mechanisms, signals are needed without noise, a conditioned and optimal for their use. In the case of electromyographic signals, as in any Bio potential, at the moment of acquiring the analogic signal, noise that interferes with the information of interest can be captured, for this reason, it's necessary to make digitally filters to filter this signal after the acquisition.
There are different forms of interference, the most common is the noise of 50Hz or 60Hz of the electrical network (depending on the geographical location), other interferences are also presented as a result of events that occur during the measurement, such as the movement of the cables, the movement of the person to whom the data is taken or even intrinsic problems of the measuring equipment, other bio signals different to the EMG, own of the person to whom the study is done also produce noise, an example of this is the electrical potential of the heart, known as EKG.

To perform the digital processing of the EMG signal, it must be taken in consideration that the most of its information is in the frequency range from 4Hz to 500 Hz, in addition, because this signal seems to be stochastic, it must be processed with techniques of characterization, which allow to know defined characteristics of the signal, through which to apply different control methodologies on electromechanical systems to allow an adequate human-machine interaction, enabling a person to control a robotic system at will, using only their muscular tissues.

6.1. Interfacing between MATLAB and development boards.

We have found that MATLAB can directly control many development boards such as Arduino and Raspberry pi and also it can convert all the Simulink blocks to compatible programming language for the controllers on the board, So that we see that doing the whole system on the MATLAB Simulink then convert it to a compatible code will be much easier and more professional. We use MATLAB and Simulink to acquire data from the muscle sensor which is connected to the Arduino board through the EMG shield. (20)

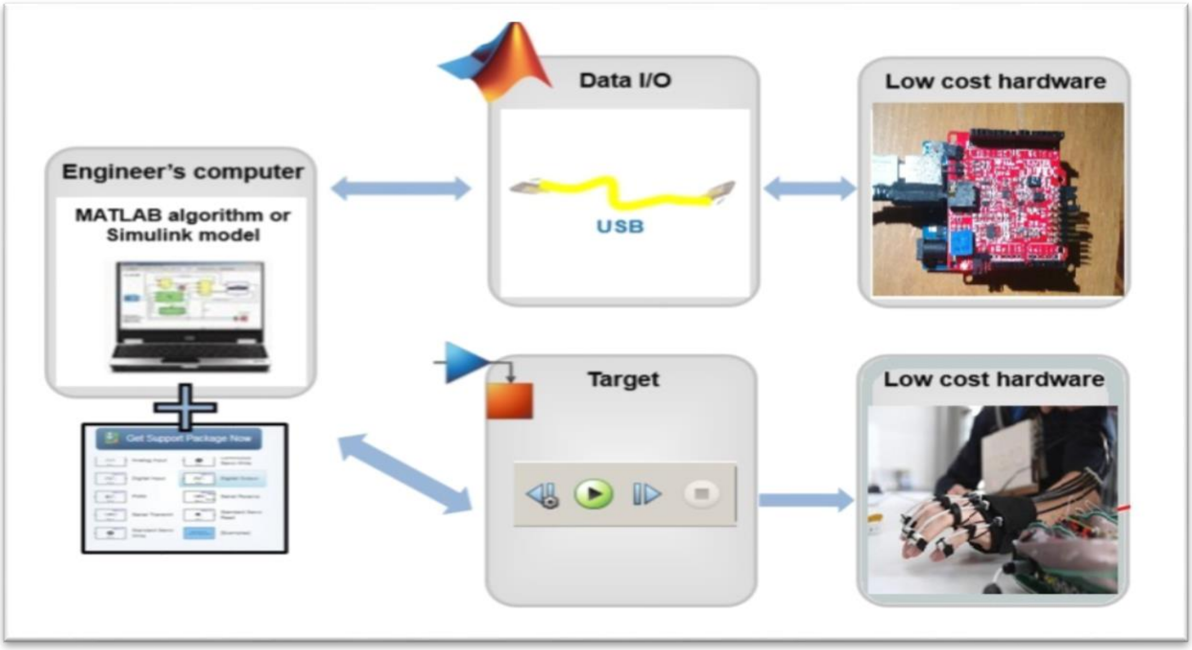


Figure 6.1: Interfacing between MATLAB and System.

6.1.1. Acquisition of the EMG signal

The acquisition of the EMG signal is obtained by means of a bio instrumentation (Olimex ECG/EKG/EMG Arduino Shield) capable of discriminating the wanted signals by means of common mode rejection ratio filters, which isolate the majority of common noises between the bipolar configuration of the acquisition electrodes, and a cascade configuration of filters bandstop and bandpass. However, part of these noises may continue to prevail after acquisition and subsequent digitization, either due to inefficiency of the electronic system or to noise outside the filtering frequencies, acquisition of EMG signal is shown in Figure 3.16 and Figure 3.17. To demonstrate the processing of the EMG signal in MATLAB, EMG signals that were previously acquired and digitized will be used.

Data acquisition can be done in Simulink using the blocks provided in the ‘’Arduino support package for MATLAB’’ like the Arduino analog input block which reads from any analog pin on the Arduino board and we can set this pin from the block properties.

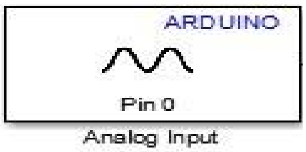


Figure 6.2:Analog input Simulink block.

Data logging through a block called ‘’ To file ‘’ which stores the data in format called time series which can be post processed by MATLAB.

Figure 6.3: TO file Simulink block.

After that we took a mat file for processing on recorded signal, we loaded mat file in script file to start processing.

6.1.2. Digital Processing of EMG Signals.

After loading the signal into MATLAB, the discrimination of its intrinsic characteristics is performed, such as time, magnitude and sampling frequency, which is a fundamental value for subsequent processing. The sampling period ***ts*** in seconds (s) is calculated from the sampling frequency ***fs*** in Hertz (sample/sec), following the equation of the period $ts = 1/fs$. (21)

Fs = 1000 sample/sec (according to Nyquist rate 2*FM)

ts=1/fs

ts=0.001

The number of samples or data of the EMG signal is obtained with the function ***length***, which calculates the length of a vector, and is stored in the vector ***N***. N=length (EMGR)

EMGR: Data read from mat file.

```
figure(1)
plot(t,EMGR,'k')
title('Raw EMG Signal')
xlabel('Time(s)'),ylabel('Magnitude(mV)'),grid on
ylim([-100 100])
xlim([0 time])
```

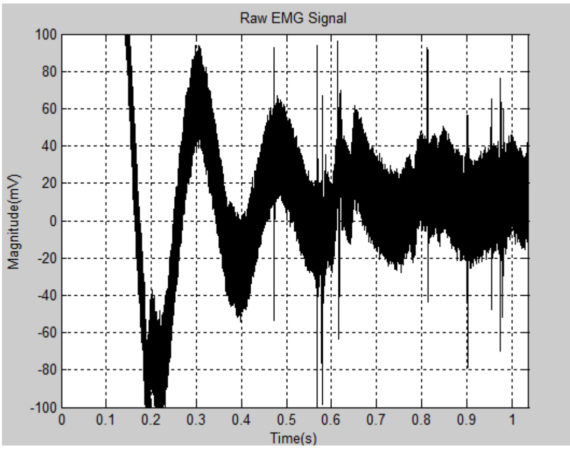


Figure 6.4: Raw EMG Signal.

On the other hand, the calculation of the frequency spectrum of the ***EMGR*** signal is performed by the function ***fft*** that develops the fast Fourier transform

$$\mathrm{EMG}F_k = \sum_{n=0}^{N-1} \mathrm{EMGR}_n e^{-\frac{i2\pi kn}{N}}, \quad k = 0, \ldots, N-1.$$

MATLAB provide fast Fourier transform function (EMGF=fft(EMGR)).

6.1.2.1. Filtration of EMG Signals.

As was specified at the beginning, EMG signals have their most relevant information in the frequency band between 4Hz and 500Hz, for that reason, this is taken as a criterion for the design of a band pass filter, which can to be understood as a filter passes high in series with a low pass filter, with cutoff frequencies of 4Hz to pass high frequencies ***cb*** and 500Hz for low frequencies ***ca***.

For the design of this filter the Butterworth configuration is chosen, since it allows the most flat response possible up to the specified cutoff frequency. For mathematical design, being *H* the frequency response, it must be fulfilled that the $2N-1$ first derivatives of $|H(W)|^2$ must be zero for w=0 & w =∞.

Its transfer function is:

$$|H(\omega)|^2 = \frac{1}{1+(\omega/\omega_c)^{2N}}$$

where *N* is the filter order, is the cutting frequency (in which the response falls 3 dB below the pass band) and w is the complex analogic frequency w=jw.

Although the mathematical design of a Butterworth filter can become complex to calculate, especially by increasing its order, MATLAB provides a very powerful tool that does it in seconds, it is the function ***butter*** given the cutoff frequencies and a type of filter configuration specified, such as band-stop filter that will be used later, provides the filter coefficients ***a*** and ***b*** that are used by the function filter, responsible for implementing the filter designed on the study signal.

```
%B.P.F
cb=4;
ca=500;
[b,a]=butter(4,[cb*2/fs ca*2/fs]);
EMGf1=filter(b,a,EMGR);
figure(3)
plot(t,EMGf1,'k')
title('Filtered EMG signal (4Hz a 500Hz)')
xlabel('Time(s)'),ylabel('Magnitude(mV)'),grid on
xlim([0 time])
```

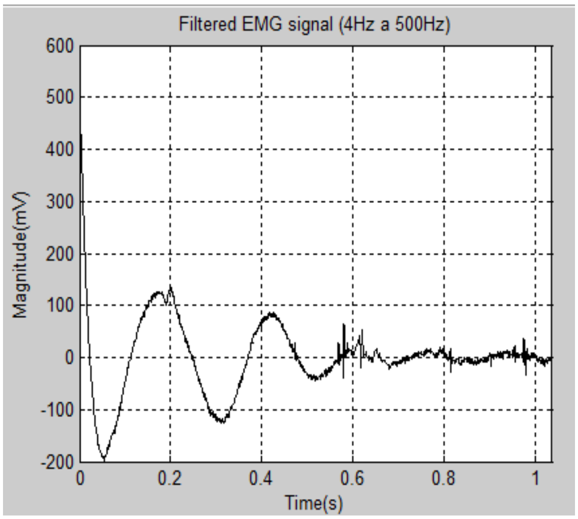


Figure 6.5: Filtered EMG signal (4Hz a 500Hz).

Then design Band-stop filter50Hz, the same procedure is carried out for the design of a 50Hz band-stop filter, since this EMG signal has been acquired in Egypt and in this country the frequency of the electric line is 50Hz. It should be noted that now a specification is added to the function *butter*, called *'stop'*, and that the cutting frequencies are *cb*=47 and *ca*=53 with order 3, to ensure the stability of the filter. This filter is applied to the *EMGf1* signal, this is called cascade filtering, one after another, and stored in *EMGf2*.

```
%STOP
cb=47;
ca=53;
[b,a]=butter(3,[cb*2/fm ca*2/fm],'stop');
EMGf2=filter(b,a,EMGf1);
EMGF2=fft(EMGf2);



%Filtered Signal in time
figure(4)
plot(t,EMGR,'r')
hold on
plot(t,EMGf2,'k')
title('Comparison between Raw EMG and EMG Filtered Signal in time')
xlabel('Time(s)'),ylabel('Magnitude(mV)'),grid on
xlim([0 time])
legend('Raw EMG','Processed EMG')
```

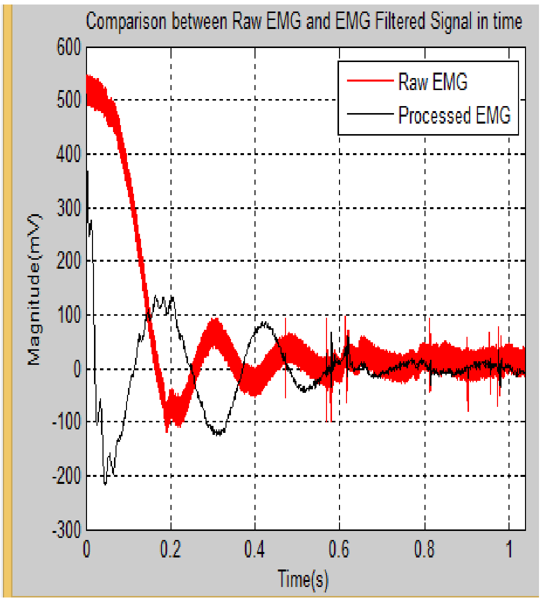


Figure 6.6: Comparison between Raw EMG and EMG Filtered Signal in time.

6.1.3. Characterization of the EMG signal for control of Exoskeleton system.

Once the EMG signal has been a conditioned, **characterization techniques** are implemented to express the information of the signal in a more smoothed way, this is achieved by mathematical tools such as the effective value (RMS), the mean absolute value (MAV). or the sum of the absolute value (IAV), tools that allow calculating an equivalent signal known as "envelope" of the study signal.

This envelope represents the average energy of the signal and is calculated to characterize the EMG signal with defined data and without abrupt changes of high frequency, which can be used as a control signal in bio robotic systems such as Exoskeletons, prostheses, whether robotic arms or other type of instruments.

This is necessary because the EMG signal is highly oscillating and redundant, it can go from a value of 0mv to a value of 10mV in a thousandth of a second, so it is not advisable to directly use the filtered EMG to control a robotic system In addition, obtaining a set of defined characteristics that represent the signal allows the modeling and prediction its behavior.

6.1.3.1 Characterization Techniques.

The effective value or the root mean square (RMS) of an electrical signal such as the EMG, is the quadratic mean of the *N* values of the discrete variable EMG, which is calculated by the function rms and it is represented by the following equation:

$$EMGE_{\mathrm{RMS}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} EMG_i^2} = \sqrt{\frac{EMG_1^2 + EMG_2^2 + \cdots + EMG_N^2}{N}}$$

The mean absolute value (MAV) is an estimate of the absolute average of the EMG signal in a segment *i* with *L* length samples.

$$\overline{\mathrm{EMG}E_{\mathrm{MAV}\,i}} = \frac{1}{N}\sum_{k=1}^{N} |\mathrm{EMG}_k|, \quad i = 0, \ldots, L-1.$$

The integral or sum of the absolute value (IAV) takes the sum of the L absolute values taken from each of the data contained in each of the W sections or windows in which the EMG study signal is divided, when the window method.

$$\overline{\mathrm{EMG}E_{\mathrm{IAV}\,i}} = \sum_{k=1}^{N} |\mathrm{EMG}_k|, \quad i = 0, \ldots, L-1.$$

In this way, the **windowing method** is then used to calculate the envelope of the EMG signal, this method consists of processing the signal by segments delimited by a length L, called windows, one segment at a time. To obtain adequate values in the envelope, it's suggested to use a length **L** of samples per window in milliseconds equal to 200ms **≤ L**≤ 300ms and an overlap SV between windows from 25% to 50% of the length **L** of the window. It should be understood that at lower **L** and higher **SV** there is a greater number of windows and that the greater the number of windows **W** is greater the sampling on the signal, so that a better characterization is obtained.

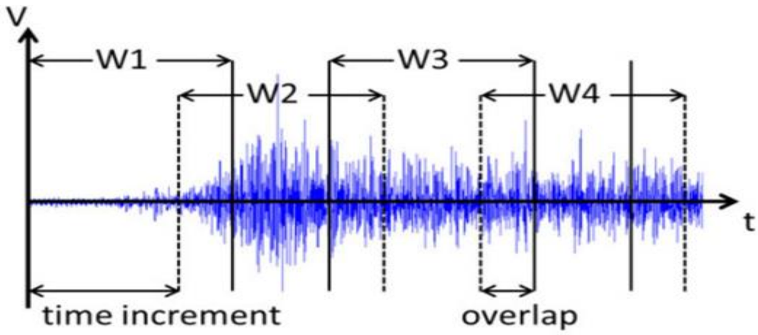


Figure 6.7: windowing method.

Characterization of EMG signals by means of the windowing method, where it is possible to observe four windows **W** that evaluate the signal separated by an increment time, so that they overlap a defined time **SV**.

So, for the design implemented, it's considered a L=40ms and an overlap of 50%, that is SV=20ms, because this configuration provides windows as possible within the accepted criteria.
Next, a standardized envelope (values of magnitude between 0 and 1) will be found, as shown in the following equation, for each one of then parameterization techniques mentioned above, they will be compared to each other to evaluate their differences and the most suitable for working with EMG signals will selected.

$$\text{NormalizedSignal} = \frac{\text{Signal}}{\text{Max}(\text{Signal})}$$

The variables are generated with the previously stipulated values and the number of windows is calculated with $W = \frac{Tms}{L-SV}$ being the value in milliseconds of the time of the signal.

```
L=40;
SV=round(L/2);
EMG=EMGf2';
Tms=time*1000;
W=floor(Tms/(L-SV));
```

The vectors that will content the envelope of the signal according to its characterization technique are created and the respective standardized envelopes are calculated.

```
EMGE_MAV(W) = 0;
EMGE_RMS(W) = 0;
EMGE_IAV(W) = 0;
Start=1;
End=L;
for i = 1:W
    EMGE_MAV(i) = mean(abs(EMG(Start:End)));
    EMGE_RMS(i) = rms(EMG(Start:End));
    EMGE_IAV(i) = sum(abs(EMG(Start:End)));
    Start=Start+SV;
    End=End+SV;
end
EMGE_MAV=EMGE_MAV/max(EMGE_MAV);
EMGE_RMS=EMGE_RMS/max(EMGE_RMS);
EMGE_IAV=EMGE_IAV/max(EMGE_IAV);
```

After obtaining the characterized signals ***EMGE ( )***, they are compared graphically at time ***T***, which is a new time vector, which ranges from T=0 until T=Time and is according to the number of windows **W**

```
T=linspace(0,time,W);
plot(T,EMGE_MAV,'-g*')
hold on
plot(T,EMGE_RMS,'-mo')
plot(T,EMGE_IAV,'-k')
title('MAV vs RMS vs IAV')
xlabel('Time(s)'),ylabel('Normalized Magnitude'),grid on
xlim([0 time])
legend('MAV','RMS','IAV')
```

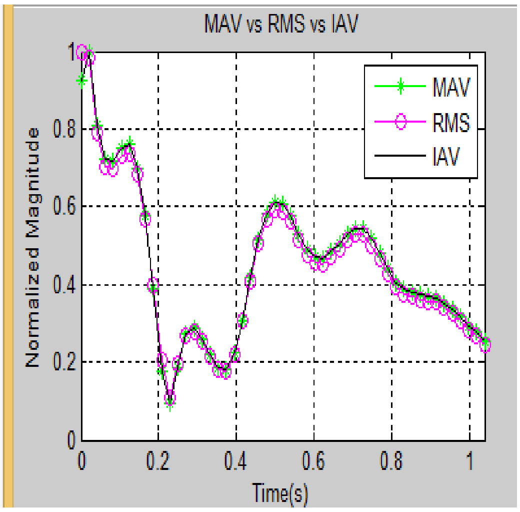


Figure 6.8: MAV vs RMS vs IAV for EMG signal.

When observing the graphical results, it can be understood that the three characterization techniques reach similar results, in fact, IAV and MAV give identical results, so the comparison is made between the IAV and the RMS. When observing that the IAV presents greater differentiation between its magnitudes, it's considered the best mathematical tool for the characterization of the EMG signal

since it allows to better discriminate between the values of the user's contraction, which allows to categorize appropriately according to the magnitude of the EMG, but also RMS and MAV considered best mathematical tools for the characterization of the EMG signal.
The envelope signal EMGE can be categorized according to its magnitude and thus labeled with a certain action, so that thresholds of activation are created. The MATLAB code that executes this logic and processes the signal **EMGE** window by window, is presented below:

```
for i = 1:W
    if EMGE(i)==1
        x(i)=120; % Hand angle
    elseif EMGE(i)>0.75
        x(i)=90;    % Hand angle
    elseif EMGE(i)>0.6
        x(i)=30;     % Hand angle
    else
        x(i)=0;
    end


end
```

This control technique of bio robotic systems can be implemented in systems that require precise and limited movements, so that they are created from two categories to implement ON/OFF controls, which can be implemented in a exoskeleton hand.

```
for i = 1:W
    if EMGE(i)==1
        x(i)=120; % Hand angle
    elseif EMGE(i)>0.75
        x(i)=90;    % Hand angle
    elseif EMGE(i)>0.6
        x(i)=30;     % Hand angle
    else
        x(i)=0;
    end


end

figure(7)
plot(T,x)
title('Angular displacement of an engine based on the EMG categorization')
xlabel('Time(s)'),ylabel('Angular displacement(°)'),grid on
xlim([0 time])
```

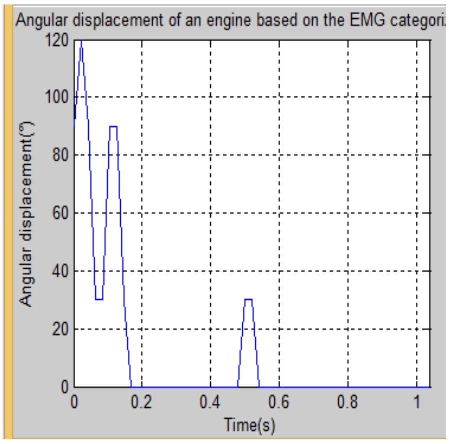


Figure 6.9: Angular displacement of an engine based on the EMG.

On the other hand, is possible design a robotic system control that is proportional to muscle contraction, so the magnitude is not categorized, but a mathematical transformation of the data with the equation

$$G = \frac{\text{EMGE}}{\left(\dfrac{\max(\text{EMGE})}{\max(M)}\right)}$$

where G corresponds to the angular displacement of the motor, $max(M)$ at the maximum angular displacement allowed for the motor rotation M, EMGE corresponds to the envelope of the EMG signal and $max(EMGE)$ to one, which is the maximum value of the EMG envelope signal, $max(M)$ is 120°. As can be seen, the resulting angular displacement is identical to the waveform of the **EMG** signal.

```
G=EMGE/(max(EMGE)/120);
plot(T,G)
title('Angular displacement of a motor proportional to the EMG signal')
xlabel('Time(s)'),ylabel('Angular displacement(°)'),grid on
hold on
xlim([0 time])
```

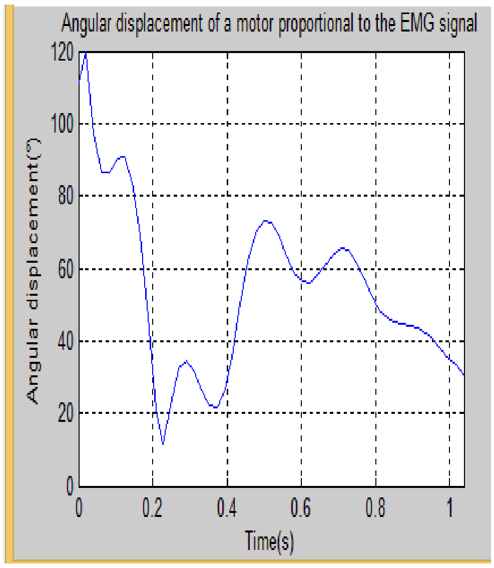


Figure 6.10: Angular displacement of a motor proportional to the EMG signal.

6.2. General Block Diagram of The System

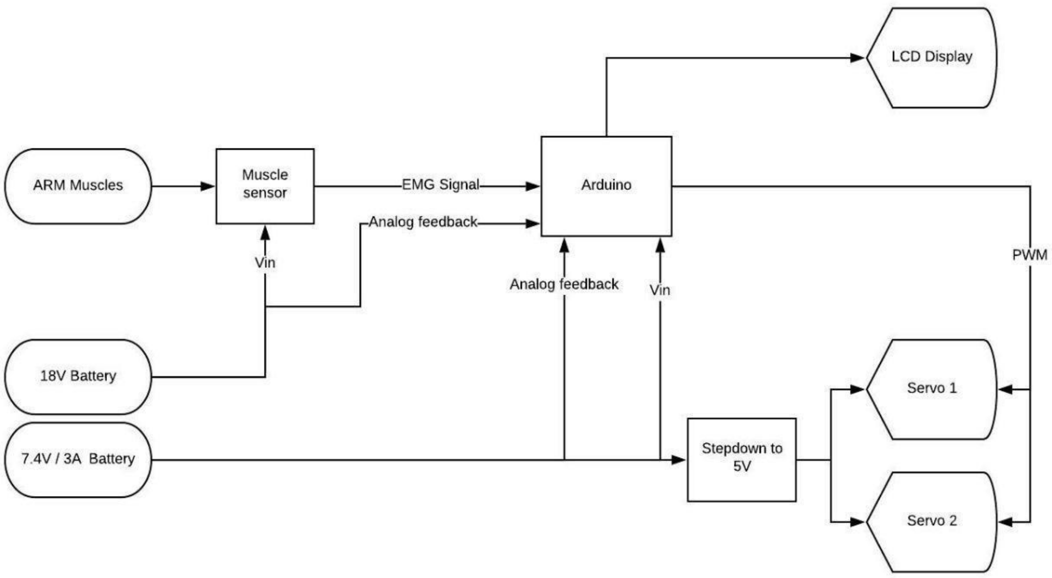


Figure 6.11: General block diagram of the system

Now we are going to explain each block in our system in details

6.2.1. Arm Muscles

As we are dealing with EMG signal so we will get them from any muscle on the arm of the patient where our two electrodes are put on anu two muscles on the arm of the patient and we will use any bone on the arm to be our ground for example the clear bon on the elbow will be regarded as a ground.

6.2.2. Muscle Sensor

We used EMG Arduino Shield but when we suffer from external noise we use MyoWare Muscle Sensor v3 to get our raw EMG signal from the muscles of the patient and we clearly explain this part in chapter two. We can also use EMG shield and MyoWare Muscle Sensor V3 together such as two different channel.

Figure 6.12: EMG Arduino Shield.

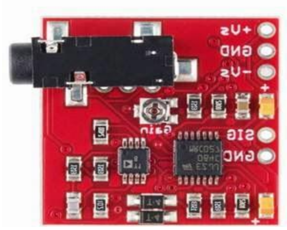


Figure 6.13: MyoWare Muscle Sensor v3.

6.2.3. Arduino Uno

Our Microcontroller we used to perform the required operations on EMG signal is “Arduino Uno” as it is very simple, cheap and can do what we want from it, also we clearly explained it in chapter two.

Figure 6.14: Arduino Uno Board.

6.2.4. Battery(18V)

We get 18V Battery we need from two 9V Battery connected to achieve our needs where these two 9V Battery supplies the EMG shield and the Arduino board with low current they need to power on.

Figure 6.15: Two 9V Battery.

6.2.5. LCD Display

We used LCD (Character LCD 2×16) Blue Display as an indicator for several things for example as an indicator for the state of the system, for example if the user using it or not or as an indicator for the percentage voltage remaining in the batteries or to warn the user if the batteries are run out the power or it can be used in several other cases.

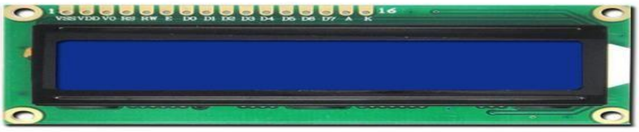


Figure 6.16: LCD (Character LCD 2×16) Blue Display.

6.2.6. Li-ion Battery (3.7/3000mAh)

We used "IMR 18650 AWT Rechargeable Li-ion Battery (3.7V, 3000mAh)" not a normal battery like the one we used with the Arduino board and the shield because motors need high current and these two batteries as we used two 3.7V battery can supply this high current.

Figure 6.17: Two 3.7V Battery.

6.2.7. Voltage Step down

We used “lm2596s-adj dc-dc” as a voltage down converter because max voltage can operate the two servo motors is 6V so we decided to make step down to the voltage with 5V to be in the safe side so we get the voltage and current needed to operate the motors.

Figure 6.18: LM2596s-adj dc-dc module.

6.2.8. Two Servo Motors

We used two servo motors to control the five fingers as we use one of them to control the thumb and index finger and the other three fingers, we will control them using another servo but it has a higher torque to perform this task, and by using them we can control the angles for hand closure by the PWM coming out from the Arduino board.

Figure 6.19: Servo Motor Continuous (360) 6 kg.cm Plastic Gears.

Conclusion

The main purpose of the project consisted of designing and manufacturing an exoskeleton to rehabilitation for patients who suffer any type of limitation movements in their hands, caused by an accident or illness. The model including all the fingers in order to achieve an accurate natural closing movement in the most comfortable way as possible.

we use EMG shield and MyoWare Muscle Sensor V3 to take signal then control motors by it to produce hand movement. Also, the prototype has a compact and strong design, it allows for the possibility of handling it with safety and simplicity. The materials used PLA.

The physiotherapist recommended remove the piece that it is attached to the end of finger, as it is necessary to see the condition of the nail and check that there are no problems related to blood flow.

The use of the rotary motors and the addition of an extra piece make it possible to perform the complete linear movement. Also, the installation of one motor for each finger has been achieved, due to the motor dimensions and modification of the hand support. To secure the device to the user's hand, a strap has been implemented, covering the entire palm and making a complete, adjustable and comfortable support.

Printing the model in one piece saves assembly time and money. The low cost is essential to produce the exoskeleton, by measuring the cost of production; the personnel needed for patient's rehabilitation would be reduced.

Finally, and concerning the environment, the used and wasted materials in 3D printing can be completely recycled and can be used to produce new printing filaments.

References

1. Rosamond W., Flegal K., Furie K., et al. Heart disease and stroke statistics—2008 update: a report from the American heart association statistics committee and stroke statistics subcommittee. *Circulation*. 2007;115(5):e25–e146. DOI: 10.1161/CIRCULATIONAHA.107.187998.

2. National Institute of Health Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6153005/

3. Hand tendon overview available at: https://www.nhs.uk/conditions/hand-tendon-repair/

4. Hand Rehabilitation Robotics on Poststroke Motor Recovery available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5688261/

5. Ma, Z., Ben-Tzvi, P. and Danoff, J. (2015). Hand Rehabilitation Learning System with and Exoskeleton Robotic Glove. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 24(12), pp.1323-1332.

6. Exoskeleton Report (2015) What is an exoskeleton Available at: https://exoskeletonreport.com/what-is-an-exoskeleton/ [Accessed 12 May 2018].

7. ZMorph Blog - articles, case studies and news on 3D printing. (2018). The Making of a 3D Printed Rehabilitation Orthosis. [online] Available at: http://blog.zmorph3d.com/3d-printed-rehabilitation orthosis/ [Accessed 15 Apr. 2018].

8. curtinnovation.com. (2017). Exoskeleton Robotic Finger Orthosis (Dr Lei Cui) | Curt innovation. [online] Available at: http://www.curtinnovation.com/project/exoskeleton-robotic-finger-orthosis/ [Accessed 15 Apr .2018].

9. EMG signal analysis available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1455479/

10. Olimex ECG/EKG/EMG Arduino Shield data sheet Available at: https://www.olimex.com/Products/Duino/Shields/SHIELD-EKG-EMG/resources/SHIELD-EKG-EMG.pdf

11. GForce Arm band data sheet Available at:
https://oymotion.github.io/assets/downloads/gForcePro_spec_v1.0-eng.pdf

12. EMG signal analysis available at:
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1455479/ And available at: Concepts and Trends in Healthcare Information Systems.

13. AVR Data Sheet available at:
https://datasheetspdf.com/pdf/219613/ATMELCorporation/ATMEGA32/

14. ARM Data Sheet available at: https://www.arm.com/products/silicon-ip-cpu.

15. PIC Data Sheet available at: https://www.microchip.com/design-centers/microcontrollers.

16. Arduino Data Sheet available at: https://www.arduino.cc/

17. Raspberrypi Data Sheet available at: https://www.raspberrypi.org/

18. Mapping available at:
https://www.arduino.cc/reference/en/language/functions/math/map/

19. MathWorks Website Available at:
https://www.mathworks.com/products/matlab.html

20. Interfacing between MATLAB and development boards Available at: https://www.mathworks.com/matlabcentral/fileexchange/32374-legacy-matlab-and-simulink-support-for-arduino

21. Digital Processing of Electromyographic Signals for Control Available at:
https://www.mathworks.com/matlabcentral/fileexchange/68245-digital-processing-of-electromyographic-signals-for-control?focused=8007e7ee-29d3-419f-a58a 8dc7b6f9d933&tab=example

Appendices

Appendix 1: Code of the Project

```
#include <TrueRMS.h>
#include <digitalWriteFast.h>
#include <LiquidCrystal.h>
#include <Servo.h>

#define LPERIOD 1000            // loop period time in us. In this
case 1.0ms
#define ADC_INPUT 0             // define the used ADC input
channel
#define battery_input_pin 0   // battery percentage input pin
#define RMS_WINDOW 40           // rms window of 40 samples,
means 2 periods @50Hz
#define PIN_DEBUG 4

unsigned long nextLoop;
int adcVal;
int cnt=0;
int mapped;
float VoltRange = 5.00; // The full scale value is set to 5.00 Volts
but can be changed when using an
                        // input scaling circuit in front of the ADC.
float value;         // value of procesed signal
int flagopen = 0;
int flagclose = 0;
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
Servo myservo1;
Servo myservo2;

float battery_analog_value;
float battery_analog_value2;
float battery_voltage_value;
float battery_voltage_value2;
int   battery_per;
int   battery_per2;
int   lcd_char_location;
```

```
int   lcd_char_location2;

Rms readRms; // create an instance of Rms.

byte battery100[] =
{
  B01110,
  B11111,
  B11111,
  B11111,
  B11111,
  B11111,
  B11111,
  B11111
};
byte battery80[] =
{
  B01110,
  B11011,
  B10001,
  B11111,
  B11111,
  B11111,
  B11111,
  B11111
};
byte battery60[] =
{
  B01110,
  B11011,
  B10001,
  B10001,
  B11111,
  B11111,
  B11111,
  B11111
};
```

```
byte battery40[] =
{
  B01110,
  B11011,
  B10001,
  B10001,
  B10001,
  B11111,
  B11111,
  B11111
};
byte battery20[] =
{
  B01110,
  B11011,
  B10001,
  B10001,
  B10001,
  B10001,
  B11111,
  B11111
};
byte battery0[] =
{
  B01110,
  B11011,
  B10001,
  B10001,
  B10001,
  B10001,
  B10001,
  B11111
};

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

```
void setup() {
  Serial.begin(115200);
  myservo1.attach(9);
  myservo2.attach(10);
////////////////////////////////Battery% & LCD//////////////////////
  lcd.begin(16, 2);      //// set up the LCD's number of columns and
rows
  analogWrite(6,50);
  lcd.createChar(1, battery100);
  lcd.createChar(2, battery80);
  lcd.createChar(3, battery60);
  lcd.createChar(4, battery40);
  lcd.createChar(5, battery20);
  lcd.createChar(6, battery0);

  lcd.setCursor(9, 0);
  lcd.write("B1");
  lcd.setCursor(9, 1);
  lcd.write("B2");
  lcd.setCursor(15, 0);
  lcd.write("%");
  lcd.setCursor(15, 1);
  lcd.write("%");

  pinMode(PIN_DEBUG, OUTPUT);


////////////////////////////////RMS//////////////////////

  // configure for automatic base-line restoration and continuous
scan mode:
 readRms.begin(VoltRange, RMS_WINDOW, ADC_10BIT,
BLR_ON, CNT_SCAN);
  digitalWriteFast(PIN_DEBUG, HIGH);
  readRms.begin(VoltRange, RMS_WINDOW, ADC_10BIT,
BLR_OFF, CNT_SCAN);
  digitalWriteFast(PIN_DEBUG, LOW);
  readRms.start(); //start measuring
  nextLoop = micros() + LPERIOD; // Set the loop timer variable
for the next loop interval.  }
```

```
////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////

void loop() {

///////////////////////////////RMS//////////////////////

  adcVal = analogRead(ADC_INPUT); // read the ADC.
  readRms.update(adcVal);
  cnt++;
  if(cnt >= 50) { // publish every 0.5s
    readRms.publish();
   if(readRms.rmsVal>=0.2 && readRms.rmsVal<=0.35)
   {
    value = 0.25;
    }
    else
    {
      value = readRms.rmsVal*1000;
    }

    Serial.println(value);

    cnt=0;
    if(value >750 && value<=1000 && flagclose == 0)
    {
      myservo1.write(180);  // set servo to mid-point
      myservo2.write(180);  // set servo to mid-point
      delay(2000);
      myservo1.write(90);  // set servo to mid-point
      myservo2.write(90);  // set servo to mid-point
      lcd.setCursor(0 , 0);
      lcd.write("Hand");
      lcd.setCursor(0 , 1);
      lcd.write("Closed");
      flagclose = 1;
      flagopen = 0;

    }
```

```
      else if(value >=500 && value<=750 && flagopen == 0)
     {
      myservo1.write(0);  // set servo to mid-point
      myservo2.write(0);  // set servo to mid-point
      delay(2000);
      myservo1.write(90);  // set servo to mid-point
      myservo2.write(90);  // set servo to mid-point
      lcd.setCursor(0 , 0);
      lcd.write("Hand");
      lcd.setCursor(0 , 1);
      lcd.write("Opened");
      flagopen = 1;
      flagclose = 0;
     }

    }

    while(nextLoop > micros());  // wait until the end of the loop time
interval
    nextLoop += LPERIOD;  // set next loop time to current time +
LOOP_PERIOD

   ////////////////////////////////Battery% & LCD///////////////////////

     //  Conversion formula for voltage for battery 1
     battery_analog_value = analogRead (A5);
     battery_voltage_value = (battery_analog_value * 4.9) / 1024.0;
     if(battery_voltage_value >= 0 && battery_voltage_value < 3.6)
     {
      battery_per=0;
      lcd.setCursor(12, 0);
      lcd.write("0");
       lcd_char_location = 6;
     }
      else if(battery_voltage_value >= 3.6 && battery_voltage_value
< 3.9)
     {
      battery_per=20;
      lcd.setCursor(12, 0);
      lcd.write("20");
```

```
        lcd_char_location = 5;
       }
       else if(battery_voltage_value >= 3.9 && battery_voltage_value
< 4.2)
       {
       battery_per=40;
       lcd.setCursor(12, 0);
       lcd.write("40");
        lcd_char_location = 4;
       }
       else if(battery_voltage_value >= 4.2 && battery_voltage_value
< 4.6)
       {
       battery_per=60;
       lcd.setCursor(12, 0);
       lcd.write("60");
        lcd_char_location = 3;
       }
       else if(battery_voltage_value >= 4.6 && battery_voltage_value
< 4.9)
       {
       battery_per=80;
       lcd.setCursor(12, 0);
       lcd.write("80");
        lcd_char_location = 2;
       }
       else if(battery_voltage_value >= 4.9)
       {
       battery_per=100;
       lcd.setCursor(12, 0);
       lcd.write("100");
       lcd_char_location = 1;
       }

      lcd.setCursor(11, 0);
      lcd.write(lcd_char_location);
```

```
    //  Conversion formula for voltage for battery 2
   battery_analog_value2 = analogRead (A4);
   battery_voltage_value2 = (battery_analog_value2 * 5) / 1024.0;
   if(battery_voltage_value2 >= 0 && battery_voltage_value2 <
1.2)
   {
    battery_per2=0;
    lcd.setCursor(12, 1);
    lcd.write("0");
     lcd_char_location2 = 6;
   }
    else if(battery_voltage_value2 >= 1.2 &&
battery_voltage_value2 < 1.6)
   {
    battery_per2=20;
    lcd.setCursor(12, 1);
    lcd.write("20");
     lcd_char_location2 = 5;
   }
    else if(battery_voltage_value2 >= 1.6 &&
battery_voltage_value2 < 1.9)
   {
    battery_per2=40;
    lcd.setCursor(12, 1);
    lcd.write("40");
     lcd_char_location2 = 4;
   }
    else if(battery_voltage_value2 >= 1.9 &&
battery_voltage_value2 < 2.2)
   {
    battery_per2=60;
    lcd.setCursor(12, 1);
    lcd.write("60");
     lcd_char_location2 = 3;
   }
    else if(battery_voltage_value2 >= 2.2 &&
battery_voltage_value2 < 2.5)
   {
    battery_per2=80;
```

```
    lcd.setCursor(12, 1);
    lcd.write("80");
     lcd_char_location2 = 2;
    }
    else if(battery_voltage_value2 >= 2.5)
    {
    battery_per2=100;
    lcd.setCursor(12, 1);
    lcd.write("100");
    lcd_char_location2 = 1;
    }

  lcd.setCursor(11, 1);
  lcd.write(lcd_char_location2);


//   Serial.print("batt1 = ");
//  Serial.println(battery_per);
    // Serial.print("batt2 = ");
 // Serial.println(battery_per2);
// Serial.println(battery_voltage_value2);
 // delay(1000);



}

/////////////////////////////////////////////////////////////////////////////////////////////////
/////////////
```

Appendix 2: Making Of the 3D Printer

- **Introduction**

The three-dimensional printer is especially made for products designing and prototyping and also for parts which is not mass manufactured. The printer is made in this project to make our control box and parts easier to make.

- **Printer Type**

3D printers have many types but the most used types are:

1. **FDM (Fused Deposition Modeling)**
   1) Cartesian FDM printers.
   2) Delta FDM printers.
   3) Polar FDM printers.
   4) Robotic arms printer.
2. **SLA (Stereolithography).**

Our printer is from the Cartesian **FDM** printers' type.

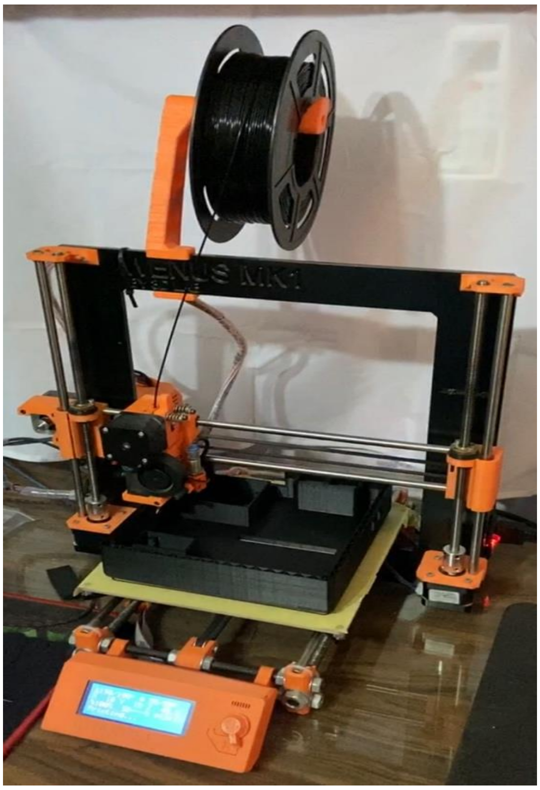


Figure 1:WENUS MK1 Printer.

We will discuss most of the printer parts and how the making process is made and its firmware

1. Printer body and frame
2. Motion
3. Extruder
4. Electronics
5. PSU (Power supply unit)
6. Firmware

1. Printer Body and Frame

The printer frame is based on the Prose i3 style. It is made of CNC machined 6mm aluminum as a main frame parts and then it is painted by electrostatic painting.

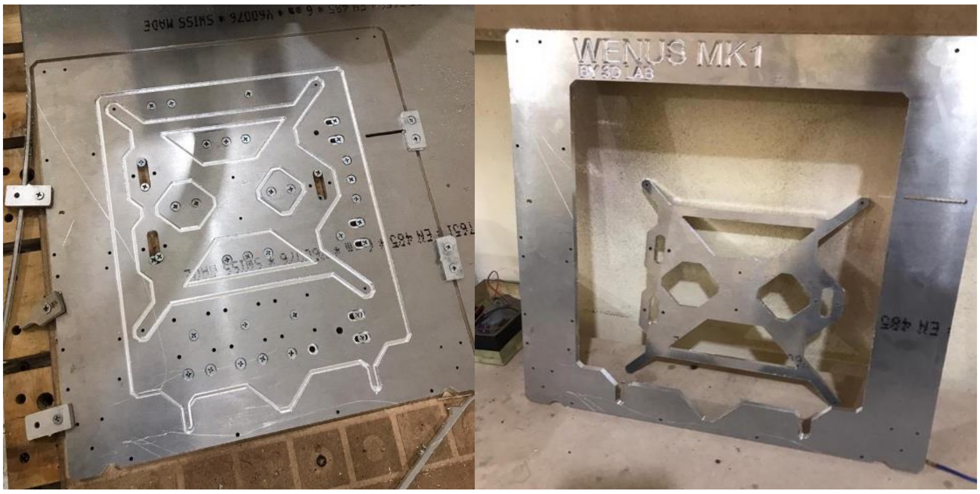


Figure 2: Printer Body and Frame.

The base was made from 10 and 8 mm Thread rods connected together using 3d printed parts as shown in the picture.

Also the X-axis, Y-axis and Z-axis is made of 3d printed parts.

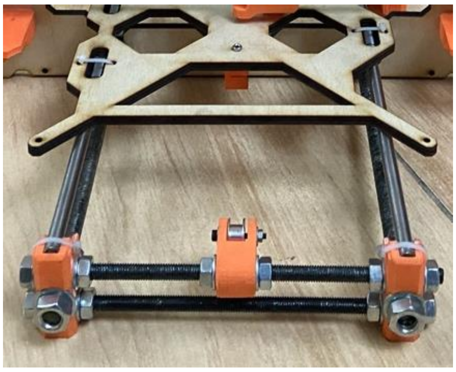


Figure 3: Printer Base.

2. Motion

The printer motion mechanism is made by timing belts (GT2) and pulleys for the X and Y-axis and 8mm Lead screw with flexible coupling (5mm to 8mm) for the Z-axis.

Figure 4: Timing belts (GT2) and Pulleys.

The motors used was 5 NEMA17 motor (2 for z axis, 1 for X-axis, 1 for Y-axis, 1 for Z-axis) with 0.29 N.M torque and rated current 1A.

Figure 5: NEMA17 motor.

3. Extruder

The extruder uses the 1.75mm E3D V6 hot end with the j head heatsink which provides perfect heat dissipation with cooling fan ,also using part cooling fan to cool the extruded filament immediately after the extrusion process, all is assembled with printed parts as a body for all extruder parts.

Figure 6: Extruder used in printer.

4. Electronics

The Arduino mega 2560 is main controller for the whole printer with help of Ramps 1.4 shield which can handle the inputs, output and biasing of the board.

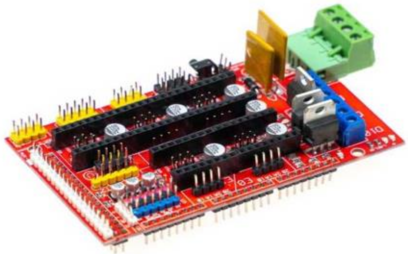


Figure 7:Ramps 1.4 shield.

It works on 12V and two current line (5A and 11A), also it uses A4988 as a Stepper motor drivers which have micro step feature (1/16 of the motor step angle), an extra heatsink was added to the heated bed MOSFET switch due to very high temperature which can lead to MOSFET failure and also cooling fan was added to the whole electronics to provide perfect working temperature.

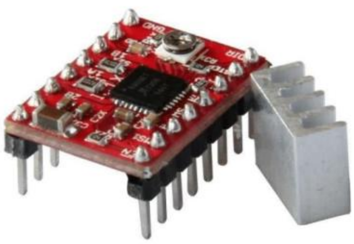


Figure 8:A4988 Stepper motor & heatsink.

The motors limit is defined by limit switches as end stops which tell the motor it has reached its limit

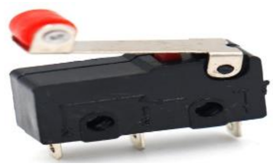


Figure 9: Limit switch.

Also and NTC (temperature sensor) is used for hot end and heated bed temperatures.

5. PSU (Power supply unit)

The printed motors, heated bed, hot end and all the electronics need a PSU than can handle the needed voltage and current for perfect operation. The printer uses a 252W 12V 21A PSU.

Figure 10: 252W 12V 21A PSU (Power supply unit).

6. Firmware

The printer uses the latest MARLIN firmware and modified to reach the printer specifications. Also used some of marlin best features as the Auto bed leveling which uses a proximity sensor to take 9 point and generate a grid on its software to represent all bed bended point.

Also uses the PID control settings to control heating the bed and hot end. it is more efficient than the normal heating technique because it keeps the temperature value stable instead of waiting for it to decrease then heating again in normal technique.

Also enabled the EEPROM setting to allow for saving the PID values and other printer parameters and values.