

Prosthetic Arm Control using Microcontroller

A dissertation submitted in partial fulfilment of the requirements for the
award of degree of

**Master of Engineering
In
Electronic Instrumentation and Control**



Submitted By:

Babli

Roll No. 801051004

Under the Guidance of:

Dr. Ravinder Agarwal

Professor, EIED

Thapar University, Patiala

Department of Electrical and Instrumentation Engineering

Thapar University, Patiala

(Established under the section 3 of UGC act, 1956)

Patiala, 147004, Punjab, India

June 2012

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, "**Prosthetic arm control using microcontroller**", in partial fulfilment of the requirements for the award of degree of Master of Engineering in Electronics Instrumentation and Control Engineering submitted in Electrical and Instrumentation Engineering Department, Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of **Dr. Ravinder Agarwal** (Professor) EIED, Thapar University, Patiala and refers other researcher's works which are duly listed in the reference section.

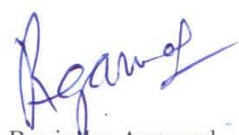
The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

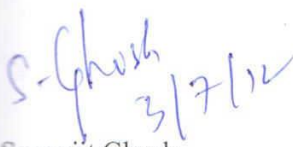
Date:


Babli
Roll No. 801051004


I certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Date:


Dr. Ravinder Agarwal
Professor
Department of Electrical and
Instrumentation Engineering
Thapar University, Patiala
Punjab


Dr. Smarajit Ghosh
Head of Department
Department of Electrical and
Instrumentation Engineering
Thapar University, Patiala
Punjab

Countersigned By


Dr. S K Mishra
Dean of Academic Affairs
Thapar University, Patiala
Punjab

ACKNOWLEDGEMENT

I express my deep sense of gratitude and respect to my guide **Dr. Ravinder Agarwal**, Professor, Electrical and Instrumentation Engineering Department, Thapar University, Patiala for his keen interest and valuable guidance, strong motivation and constant encouragement throughout this thesis work. I thank him for his great patience, constructive criticism and useful suggestions apart from invaluable guidance to me. I am sure that the knowledge gained through my association with my supervisor shall go a long way in helping me to realize my goals in life. I owe my thanks to **Dr. Smarajit Ghosh**, Head, Department of Electrical and Instrumentation Engineering for his kind support.

I also thanks to **Mr Ajay Kumar** for his suggestions, guidance and encouragement.

Date:

Place: Thapar University, Patiala

Babli

ABSTRACT

A prosthetic limb is an artificial device extension that replaces a missing body part. A prosthetic arm is a fake arm for those who amputated their arm. In addition to the standard artificial limb for every-day use, many amputees have special limbs and devices to aid in the participation of sports and recreational activities.

Prosthetic arm is a boon for those persons who have lost their arm due to some accident. The main requirement is that its function should be as near to the natural arm as possible. There are various designs of artificial arm that are available in the market, categorised as electrical, mechanical and Myo-electric arm. Mechanical prostheses use some motion of the body to provide the force necessary to control the prosthetic component. Electrical arms operate the hand by a motor which is driven by micro switches and relays. Myo-Electric arm is stimulated by muscle signal available from the amputee.

EMG signal detected by sensors or electrodes. The amplitude of the surface EMG signal (SEMG) varies from the μV to the few mV range. The rms value for the upper movement of the arm is more than the rms value for the down movement of arm. Similarly, rms value for the clockwise movement of the arm is more than the rms value for the anticlockwise movement of arm. Based on this a microcontroller was programmed to perform up/down and clockwise / anticlockwise movements in steps depending on the dc voltage level. The levels of the dc voltage corresponding to the EMG signal were taken.

Table of Contents

S.No.	Title	Page No.
1.	Declaration	ii
2.	Acknowledgement	iii
3.	Abstract	iv
5.	List of Figures	vii
6.	List of Tables	ix
7.	Abbreviation	x
Chapter 1	Introduction	1-6
	1.1 Prosthetic arm	1
	1.2 Prosthesis in the world	2
	1.2.1 Aesthetics prostheses	2
	1.2.2 Mechanical prostheses	2
	1.2.3 myoelectric prostheses	3
	1.3 Types of prosthetic arm	3
	1.3.1 Mechanical arm	4
	1.3.2 Electrical arm	4
	1.3.3 Hybrid arm	5
	1.3.4 Myo-electric arm	5
Chapter 2	Literature review	7-14
Chapter 3	Material and method	15-38
	3.1 Introduction	15
	3.2 Component description	15
	3.3 System description	29
	3.4 System operation	33
	3.4.1 Selection switch in manual position	34
	3.4.2 Selection switch in EMG position	35
Chapter 4	Result and discussion	39-43
	4.1 Manual control	39
	4.2 EMG control	40
	4.2.1 Arm up/down movement	41

	4.2.2 Arm clockwise/anticlockwise movement	42
Chapter 5	Conclusion and future scope	44
	5.1 Future scope	44
REFERENCES		45-46
APPENDIX A: Code for microcontroller		47-58

1. INTRODUCTION

1.1 Prosthetic Arm

A prosthetic limb is an artificial device or a replacement of missing body part. A prosthetic arm is a fake arm for those who amputated their arm. Earlier Armorers used prostheses mainly in battle to hold sword and shield. The Alt-Ruppin hand was a typical hand that used springs and push button to hold and release fingers. Modern prosthetic principles evolved after II world war. In 1949 first myoelectric switch was developed. Earlier body powered prosthesis components have not much changed because most of the research has focused on externally powered prosthesis and high cost of manufacturing investment [1].

The number of the handicapped has been increasing due to many reasons like traffic accidents, accidents in workshops and diseases. Hand parts are very important parts and have complicated functions. Loss of these parts makes humans anxious and causes many functional troubles. The prostheses with the same performances as natural upper extremity shall make the physical handicapped have the same daily life as they had one before they lost their extremities. Thus it is hoped to make the prostheses serving for their lost one. Recent advancements in technology of mechatronics have helped us develop the mechanism of the externally powered upper extremity prostheses close to that of natural extremity. At present, the prostheses with multi degree-of-freedom and multi-function have been studied. There are many actuators which are electric-motor, gas pressure cylinder, hydraulic cylinder, and so on [2].

In the United States 41,000 persons are registered who had hand amputation or a complete arm. There would be 1,000,000 such persons worldwide with the same frequency of occurrence. Main factors for a loss of an upper extremity are accidents followed by general diseases and injuries from war. The loss of an upper limb results in a drastic restriction of function for an individual. Therefore, in the last 3 decades an increasing number of handicapped persons have been provided with prosthetic hands that have the shape of a human hand.

However, surveys on using such artificial hands revealed that 30 to 50% of the handicapped persons do not use their prosthetic hand regularly. The main factors for the rejection of conventional prosthetic hands were:

- Heavy weight: although commercial prosthetic hands have about the same mass as human hands they appear to be unpleasantly heavy because the mass is transmitted by a lever arm to the short stump of the amputated arm.
- Low functionality: The gripping abilities are restricted with the conventional prosthetic hand as it can only perform a single pincer-like grip movement. For example it is impossible to pick up a pinball with the prosthetic hand. The fingers have only one degree of freedom and can not adapt to the shape of an object. The consequence is an increased force that is necessary to hold an object stable.
- Robot-like movement: Movement appears unnatural because of limited DOF

To overcome these disadvantages a lot of efforts have been done worldwide [3].

1.2 Prosthesis in the World

The different prostheses developed by the main prosthetic societies: UTAH, OTTA BOCK and PROTEOR concentrate 90% of the market. These are classified in three categories namely; aesthetics prostheses, mechanical prostheses and myoelectric prostheses.

1.2.1 Aesthetics Prostheses

This type of prosthesis is generally used by patients and their aim is only aesthetics. The prosthetic part is created from a standard mould and resemblance to the healthy member. This kind of prosthesis does not carry out any movement. It only serves to restore the patient body appearance. This kind of prosthesis is for instance manufactured by the OTTO BOCK society.

1.2.2 Mechanical Prostheses

Currently three kinds of mechanical elbow products are offered to the patients. This kind of prostheses tries to approach the functionality of the lost member. It can be manual (use with the assistance of the healthy member) or with cable. The first kind is the elbow with toothed rack. The pushbuttons are actuated by the valid hand or by a cable. Many drawbacks are attached to

this mechanical elbow such as noise of the toothed rack, the limited number of positions of the front arm and the bad aesthetic of the pushbutton.

The second kind of elbow is the elbow with friction. This moves on the friction of a spiral spring on the axis of the elbow. A cable ordered by the other shoulder actuates blocking: one traction locks it, another unbolts it. It maintains the position less firmly but more functional than the previous. In addition, it needs a double order from the amputee, which is not always easy to carry out.

The third kind is an automatic elbow from OTTO BOCK. The front part is manufactured out of plastic and is not very solid. Its distal part (near to the wrist) is cylindrical and is simply cut to the length of the healthy member.

1.2.3 Myoelectric Prostheses

Myoelectric signals (Electromyogram or EMG) are electrical signals that are registered from the muscles activities. A great number of applications are possible with these signals. The functional motor activities can be measured by placing the surface electrodes directly on the skin. The EMG signals are complex with noise and they are easily influenced by many factors. Then, from the interpretation to the use, the EMG needs several specific treatments. The consequences of the Viet Nam war were at the origin of the development of the UTAH products. This society was the first to propose the EMG technology to control the prosthesis. The OTTO BOCK society also proposes prosthesis of hand coupled with a myoelectric elbow. Unfortunately, the whole system proposed by this society is too expensive for patient. The hand is a tree legs grip with an aesthetic glove [4].

1.3 Types of Prosthetic Arms

Prosthetic arms may be categorized as:

- Mechanical Arm
- Electrical Arm
- Hybrid Arm
- Myo-Electric Arm

1.3.1 Mechanical Arm

Mechanical prostheses are functional prostheses that use some motion of the body to exert the force required to control the prosthetic component. Bowden cable is used in the prosthetics field. It consists of an inner core cable that is free to move within a sleeve cable which is fixed in place at either end. These devices require a harness, to be worn about the shoulders, to which one or more Bowden cables are attached. The conventional below-elbow, body-powered prosthesis has a single control cable that runs from the harness to a terminal device. Terminal-device opening and closing is then controlled by shoulder shrug and/or flexion of the residual upper arm. An above-elbow amputee has additional control cable. That cable is used to switch control of the harness from terminal device opening to elbow flexion by unlocking the elbow. Body-powered prostheses are the most common kind of prosthesis used all over the world, due to the intimate connection of the control cable, which is provided between input and output. It helps the user of a body-powered limb to feel closely connected to the operation of the prosthesis. These prostheses are also lightweight, durable and of relatively low cost. However, body-powered prostheses have a number of shortcomings. The major issues are the uncomfortable harness mechanism, the somewhat ungainly control motions, particularly in the case of above-elbow prostheses, restricted range of motion and limited load-lifting capacity.

1.3.2 Electrical Arm

These are externally powered devices. They receive their power from an external electric source to the body. These are relatively new (last 15 to 20 years) addition to the armamentarium of prosthetic devices.

- a) **Touch Switches:** A pair of touch switches remains in contact with antagonistic wrist muscles flexors and extensors. The wrist flexors activate the 'CLOSE' switch while extensors operate the 'OPEN' touch switch.

- b) **Control Circuit:** Each micro switch is connected to a Flip-flop which is configured to operate in set-reset mode. It is 'SET' by the micro switch and 'RESET' by the limit switches provided at the extremities of hand positions.

c) **Control Relay:** A relay working at 6V or 9V operates the motor. When the flip-flop has one polarity, DC voltage is applied by the relay to the motor and the motor rotates in one direction. When flip-flop output changes its polarity due to operation of second touch switch, the relay also changes the polarity of DC voltage being applied to the motor. Consequently, motor rotates in opposite direction. In this way, the hand closes and opens.

1.3.3 Hybrid Arm

When body-powered and externally powered systems are linked together they are called hybrid systems. Hybrid systems are used most frequently with persons who have amputations above the elbow or who have bilateral arm amputations. Such systems can provide the user with high gripping and/or high lifting capacities of powered systems and fine control of body power. Providing the amputee with a body-powered limb on one side and an electric-powered limb on the other side, they enable the wearer to use the limb that is most appropriate for a specific task. This method also enables the limbs to be operated independently of each other i.e. the body motions required to operate the body-powered side do not influence the state of the powered side of limb and vice versa, as they are decoupled.

1.3.4 Myo-electric Arm

The electrically powered prosthesis under the control of myoelectric signals from residual muscles did not become commercially available until late 1960s and did not gain widespread clinical acceptance until the early 1980s. Myoelectrically controlled upper limb prosthesis offers the highest level of rehabilitation available today. Myoelectric signals are produced by a muscle when it contracts.

- Raw EMG signal acquired by the needle electrodes is of the order of 0.5 mV.
- The surface electrodes signals lies in the range of tens of microvolt. Three surface electrodes are generally used in this type of arm, one acting as reference electrode, another as active electrode and the third as ground electrode. The difference signal between reference and active electrode is processed to reduce noise in the system.
- A normally innervated muscle shows no electrical activity at rest. These signals provide important information on the physiological status of the skeletal muscle and its nerve supply. The intensity of EMG signal increases as the muscle tension increases.

- The frequency range of EMG signal which shows change with opening and closing of hand is 15 to 500 Hz.
- There are two sets of muscles in forearm which get activated whenever an object is grasped or left by our fingers. These muscle groups are called flexors and extensors. In myoelectric prosthesis, it is achieved by making the use of these two muscles to open or close the terminal device. The electrodes accommodated in the prosthetic socket pick up the signals from these muscles which after conditioning are used to control the prosthesis. This type of system results in high grip force and grip speed.

Prosthetic arm is one of the main requirements for those persons who have lost their arm due to some mishap. The main requirement of prosthetic arm is to provide the functionality of the natural hand.

2. LITERATURE REVIEW

In the domain of handicapped people, the amputees are one of the most important groups in the world. The aim of developing prostheses is mainly to improve their conditions of life and to help them recover independence and dignity. Replacing an amputated limb is a complex problem, as it is necessary to replace the osseous structure, to collocate a locomotive system and to give to the prosthesis a command system. The real challenge for the engineering is to make the system occupy the same space that the arm occupied originally, to make it weight less than it weighted before, to make it produce a similar force to that produced by a sound am and in addition to control the whole system by a simplified control scheme[5].

Daniel Graupe *et. al.*, [6] employs parallel filtering to discriminate between the various limb functions of interest to achieve fast discrimination and control as required for practical applications, since this allows the identification itself to be performed off line. Intel 8080 microprocessors at double precision (incorporating hardware multipliers), have yielded an 85 percent success rate in discrimination between four to five limb functions using a single electrode.

George N. Saridis *et. al.*, [7] deals with the statistical analysis and pattern classification of electromyographic signals from the biceps and triceps of a below-the-humerus amputated or paralyzed person. Such signals collected from a simulated amputee are synergistically generated to produce discrete lower arm movements. Results show very good separability of classes of movements when a learning pattern classification scheme is used, and a superposition principle seems to hold which may provide a means of decomposition of any composite motion to the six basic primitive motions, e.g., humeral rotation in and out, elbow flexion and extension, and wrist pronation and supination. Since no synergy was detected for the hand movements, different inputs have to be provided for a grip. For amputees with shorter stumps, synergistic signals could be obtained from the shoulder muscles.

H.T. Law *et. al.*, [8] An electrically driven locking mechanism has been built, which is controlled by the electromyogram (EMG) of the surviving muscles in the upper arm. Hybrid technology is suited for the construction of the associated electronic circuitry. Many similar applications are now being considered in attempts to improve the performance of upper-limb prostheses.

David T. Gibbons *et. al.*, [9] , used the EMG signals from residual muscles to control above-elbow prosthesis has been tried, but presents many problems, not the least being that the prosthesis is under open-loop control. They have developed a prototype of a self-contained above-elbow prosthesis which has electric-powered wrist and elbow joints controlled by an Intel 8751 microcomputer. A more satisfactory control technique is extended physiological proprioception where the inherent proprioceptive feedback present within an intact joint is used to provide closed-loop control. Their technique is to control the positioning of this above elbow prosthesis using the motion of the intact shoulder. Grasp, which does not involve positioning in space is separately controlled using EMG signals from biceps and triceps muscles. A choice from a range of linkages can enable the user to perform different tasks in different situations.

Yiorgos A. Bertost *et. al.*, [10] presented anew extended physiological proprioception (e.p.p.) position controller based on a microprocessor. In an e.p.p. controller the user is continuously coupled to the action of the prosthesis by means of a mechanical linkage. In this application, the most distal intact physiological joint (or muscle/tendon cineplasty) is mechanically linked to the prosthetic joint being controlled. The user applies a force through a cable to move the e.p.p. configured prosthesis. A sensor along with the interface-signal conditioning electronics converts the applied force to a dc voltage that is applied to the microcontroller. A linearization routine, based on a lookup table of force and voltage data of the particular FSR previously collected and stored in the microcontroller, converts the nonlinear response of the FSR to a linear integer for the input voltage. A PWM signal is generated based on the voltage and output by the microcontroller to drive the H-bridge and dc motor.

Marcello Mulas *et. al.*, [11], presents the development, testing and experimentation of a device for the hand rehabilitation. The system we designed is intended for people who have partially lost the ability to control correctly the hand musculature, for example after a stroke or a spinal cord injure. Based on EMG signals the system can "understand" the subject volition to move the hand and thanks to its actuators can help the fingers movement in order to perform the task. The exoskeleton is designed to be adaptable and it is actuated by two servomotors. Two potentiometers are used as position sensors, in order to control the real status of the hand patient. It is necessary to program a test series with several patients in order to prove the real effectiveness of the hand rehabilitation system we have developed.

Kentaro Nagata *et. al.*, [12] describe the classification method of hand movements using 96 channels matrix-type (16x6) of multi channel surface electrode. System that has 96 channels electrode is not need to select a particular electrode position. Only attaching this electrode, one can obtain correct EMG and this way means providing with a simple and easy way. The purpose of the study is the development of the EMG pattern recognition method using multi channel electrode. From measured 96 channels EMG data, we choose one line (16channels) of this electrode with the smallest noise. The EMG signal is recognized by canonical discriminant analysis. In order to recognize the EMG signal, the first three eigenvectors are chosen to form a discriminant space. And Euclidean distance is applied to classify the EMG. From the experiment in this method, we can discriminate 12 movements of the hand including four finger movements.

Shahjahan Shahid *et. al.*, [13] applied the higher order statistics based system reconstruction algorithm can be applied to the EMG signal to characterize the motor unit action potential. The electromyographic (EMG) signal provides information about the performance of muscles and nerves. At any instant, the shape of the muscle signal, motor unit action potential (MUAP), is constant unless there is movement of the position of the electrode or biochemical changes in the muscle due to changes in contraction level. The rate of neuron pulses, whose exact times of occurrence are random in nature, is related to the time duration and force of a muscle contraction. It is observed that the appearance of MUAPs estimated from any EMG signal clearly shows evidence of motor unit recruitment and crosstalk, if any, due to activity in neighbouring muscles. It is also found that the shape of MUAPs remains the same on loading.

HaeOck Lee *et. al.*, [14] discussed an externally powered upper extremity prosthesis as a system. The necessary components to design a better prosthetic arm are divided into four subsystems: input, effector, feedback. Current research is reviewed in terms of these subsystems. Each subsystem performs its own task, but they are related to each other and together they function to make up a prosthetic upper extremity, which provides the movement to the amputee.

Guanglin Li *et. al.*, [15] proposed a new approach to improve the control of prosthetic arm rotation in amputees. Arm rotation is sensed by implanting a small permanent magnet into the distal end of the residual bone, which produces a magnetic field. The position of the bone rotation can be derived from magnetic field distribution detected with magnetic sensors on the

arm surface, and then conveyed to the prosthesis controller to manipulate the rotation of the prosthesis. A sensitivity analysis of the magnet size and arm size was presented. The influence of relative position of the magnet to the magnetic sensors, orientation of the magnet relative to the limb axis, and displacement of the magnetic sensors on the magnetic field was evaluated. The simulation results suggest that the direction and angle of rotation of residual humerus could be obtained by decoding the magnetic field signals with magnetic sensors built into a prosthetic socket. This pilot study provides important guidelines for developing a practical interface between the residual bone rotation and the prosthesis for control of prosthetic rotation.

Subrata Kumar Kundu *et. al.*, [16] proposed a 5 DOF externally powered prosthetic arm for AE amputees to increase amputee's mobility in daily life activities. The proposed prosthesis is designed to generate natural human like arm motion while performing a daily life task. This paper summarizes the design and controllability of the proposed prosthesis. Currently available commercial prosthetic arm could not gain wide acceptance in the amputee society due to their low functionality with limited DOF motions. In order to increase the mobility of the AE amputee in their daily life activities, a five DOF transhumeral prosthesis using conventional DC motor is presented in this paper. In order to make the appearance closer to human arm, the proposed prosthesis is supposed to be covered with an artificial skin and the rectangular socket is replaced with a half spherical structure like commercial prosthesis. Moreover, light weight polymer material is expected to be used in future to reduce the weight of the prosthesis.

Shuxiao Wang *et. al.*, [17] presents a robotic arm for stroke patients. Two SEMG (Surface Electromyography) signals collected from bicipital muscle and triceps muscle of arm, which are used to control robotic arm. When patients want to flex arm, the SEMG signal of bicipital muscle is larger than that of triceps muscle. On the other hand, when patient want to extend arm, the SEMG signal of triceps muscle is larger than that of bicipital muscle. The robotic arm's rotation direction is decided by the difference of two SEMG signals. The torque and speed of the robotic arm are controlled by the amplitudes of SEMG signals. The system is based DSP (Digital Signal Processor). H-bridge is used to drive DC motor.

Hardeep S. Ryait *et. al.*, [18] presents review paper that gives the historical developments in three main sections. First part describes the EMG signal properties. Second part deals with the mathematical models developed till now for EMG signal analysis. In the third part different design approaches have been reviewed for artificial hand. First approach discussed here is on

the body-powered terminal devices which are controlled by the user's pull on the control cable to open the hand or hook and for the grip strength. Other being myoelectric controls type, an externally-powered system which uses electrical impulses, generated by contraction of the amputees own remaining muscles to operate a motor in a mechanical hand, hook or elbow. This paper presents a brief overview of above mentioned issues with regard to artificial hands.

Hiroyuki Takeda *et. al.*, [19] developed a novel prosthetic arm with a five-fingered prosthetic hand using our original pneumatic actuators and a slender tendon-driven wrist using a wire drive and two small motors. Because the prosthetic hand's driving source is comprised of small pneumatic actuators, the prosthetic hand is safe when it makes contact with people; it can also operate flexibly. In addition, the arm has a tendon-driven wrist to expand its motion space and to perform many operations.

Deepak Joshi *et. al.*, [20] discussed the trends undergoing in all the various steps involved in EMG (Electromyogram) based prosthetic hand development. In order to overcome some limitations of current prosthetic hands mainly related to the proper functionality and controllability, the prosthetic hand has been designed following a biomechatronic approach based on biologically inspired design solutions. The majority of electrically powered prosthetic hands are based on a simple design that limits motion to one degree of freedom. Designs of multi-articulated prosthetic hands have had limited success due to their complexity and number of mechanical components. Classical EMG (myoelectric) controllers have failed in the past, since they were based on only determining existence or non-existence of an EMG signal. Recent work has approached this multifunctional control problem using a large number of electrodes, though still considering only a limited part of the EMG spectrum.

Tanmay Pal *et. al.*, [21] on-line characterization system has been developed for DC motors. These sensitive applications require high precision and high speed of response. The system has been programmed on ARM microcontroller, it actuates DC motor and automatically collects data while it is being accelerated and attains a steady speed ; the embedded routines process it instantly and returns the current values of inertia, friction coefficient, back-emf constant and torque constant. A prototype system for DC motor control has been developed in laboratory, it has been characterized and control experiments have been performed.

Hardeep S. Ryait *et. al.*, [22] SEMG signals can be detected by sensors placed over the skin surface onto the muscle tissue. Here single electrode is used with SEMG signal as actuator. Muscle activities from a single location (over group of muscles near triceps brachii) on arm during gripping and elbow movement were carried out. The electrode was placed on the upper arm, one hand width under the armpit. Movements were analysed by deriving RMS values from the acquired SEMG. The microprocessor controlled prosthetic arm is designed to perform the operation of gripping.

S. Herle *et. al.*, [23] present an algorithm based on an autoregressive (AR) model representation and a neural network, for EMG signal classification. The results have shown that combining a low-order AR model with a feed forward neural network, a rate of classification of 98% can be achieved, while keeping the computational cost low. The solution proposed is capable of controlling three joints (i.e. six movements) of the upper limb prosthesis. The inputs of the high-level controller are obtained from the classifier, while its outputs are applied as input signals for the low-level controller.

Hideyuki Uehara *et. al.*, [24] implemented a mobile robotic arm for people with severe disabilities. The system is composed of a robotic arm, microcontroller, and its controller. The main body of the robotic arm can be contained in a briefcase to carry a laptop computer. The robotic arm has seven servomotors that are controlled by the pulse width modulation. Controllers for the robotic arm system, have been developed e.g., head-controlled interface, eye-controlled interface, vision-based interfaces, and electromyogram (EMG)-based controller. In this study a notebook PC, which was connected to the microcontroller through an RS-232C cable, was used as the controller. Furthermore, a control program written in the form of C language was developed with gcc compiler. The program was then written into a flash memory of the microprocessor.

Hardeep S. Ryait *et. al.*, [25] carried out the study of SEMG signals at different below elbow muscles for four operations of the hand wrist/grip-like opening (op)/closing (cl)/down (d)/up (u). Myoelectric signals were extracted by using a single-channel SEMG amplifier consisting of a differential amplifier, non inverting amplifier, and interface module. Matlab softscope was used to acquire the SEMG signal from the hardware. After acquiring the data from six selected locations, interpretations were made for the estimation of parameters of the SEMG using the Matlab- filter algorithm and the fast Fourier transform technique. An interpretation of

wrist/grip operations using principal component analysis (PCA) was carried out. PCA was used to identify the best SEMG signal capturing system out of two-channel, three-channel, and four-channel systems. Two acupuncture points (on wrist) were also selected for the analysis with other points on the arm.

Vedran Vajnberger *et. al.*, [26] explains the whole process of making a system for remote control of a robot arm with five degrees of freedom (DOF). For this purpose a hardware structure was fully designed and implemented. The hardware structure is based on microcontroller PIC16F877A and surrounding architecture that controls movement of different axis of the arm. The arm has no sensors, so the visual information from the camera was used as feedback. Two communications were used to operate robot arm. The first one is realized by serial RS-232 protocol between PC and Microcontroller, and this communication is used to operate the arm. The second communication uses TCP/IP protocol for remote control. The TCP/IP protocol provides communication between server and client computers and sends information of position of robot arm.

Hardeep S. Ryait *et. al.*, [27] Measurement of SEMG depends on a number of factors/parameters like amplitude, time and frequency domain properties. In the present investigation, analysis was carried firstly; to study the grip force vs. SEMG parameters at acupuncture points on arm, using single channel approach. At all the selected acupuncture points a linear increment of SEMG was observed. Secondly; to discriminate four elbow movements from different locations on arm using two channel approach with single parameter. The parameter for the analysis chosen was the root of mean of square (RMS) value of SEMG. Further; principal component analysis was used to verify the elbow movement discrimination. Extension and supination were the two operations which were observed to be easy to realize by prosthetic devices. The selection of these locations was done on the basis of acupuncture points and anatomy of elbow.

Luay Fraiwan *et. al.*, [28] presents a start point for training patient on using prosthetic devices using virtual reality prosthesis. The proposed system consists mainly of an electromyography (EMG) system connected to the patient arm (biceps and triceps muscles) and interfaced with a PC using a data acquisition system. The PC uses Matlab to enhance the EMG signals and detect the presence of events in them. These events are used to control a virtual hand with two

movements; grasping and wrist rotation. The system was tested on a subject who performed the grasping and wrist rotation for 90 trials. The overall success rate was found to be 84%.

3. MATERIAL AND METHOD

3.1 Introduction

SEMG is a non invasive method for the measurement of muscle activity. The amplitude of the surface EMG signal (SEMG) varies from the μV to the few mV depending on the muscle under observation. In prosthetic hand small motor provide functions of hand and control is being done through proper sensing of movements of muscles. This is called myo-electric controlled or electrically powered Prosthesis. The signal can be acquired by electrodes and passed through amplifier section. The signal can be passed through filter section to remove noise. The rms to dc converter is used to output the dc signal. The level of that dc signal represents the EMG signal. It has been observed that the rms value for the upper movement of the arm is more than the rms value for the down movement of arm. Similarly rms value for the clockwise movement of the arm is more than the rms value for the anticlockwise movement of arm. Based on this a microcontroller is programmed to perform up/down and clockwise/anticlockwise movements in steps depending on the level of dc voltage.

3.2 Component Description

The main components used in this dissertation are microcontroller, motors, motor driver, encoders and switches.

Microcontroller

Atmel ATmega16 is a low-power CMOS microcontroller. It is based on the AVR enhanced RISC architecture. Atmega16 is an 8-bit microcontroller. The pin diagram of Atmega16 is shown in Figure 3.1. By executing powerful instructions in a single clock cycle; the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the designer to optimize power consumption versus processing speed. The Atmel ATmega16 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

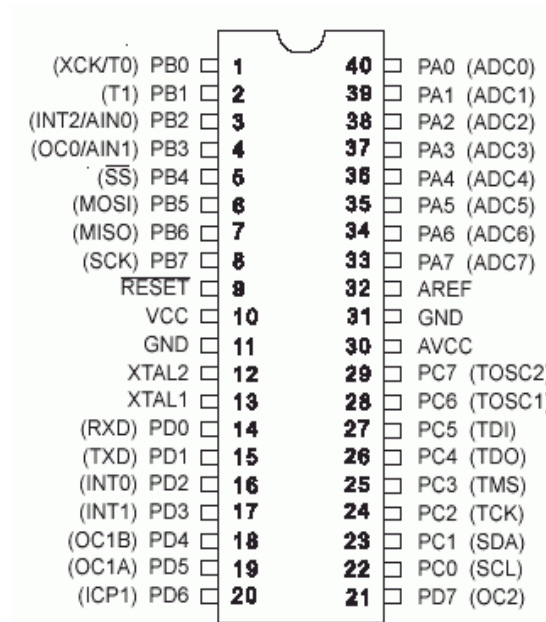


Figure 3.1: Pin diagram of Atmega16

The main features of ATmega16 are as:

- 40 pins,
- 16K bytes of In-System Programmable Flash Program memory,
- 512 bytes EEPROM,
- 1K byte SRAM,
- 32 general purpose I/O lines,
- Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes,
- One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode, Internal and External Interrupts,
- 8-channel 10-bit ADC,
- Four PWM Channels,
- Speed grades 0-16 MHz
- Operates between 4.5 - 5.5 volts.

Port Operation Registers

There are three registers that are related to the various port operations that we can perform.

- DDRx – Data Direction Register

- PORTx – Pin Output Register
- PINx – Pin Input Register

DDRx Register

The DDRx Register: Data Direction Register. The DDxn bit in the DDRx Register selects the direction of this pin. If DDxn is written logic one, pin is configured as an output pin. If DDxn is written logic zero, pin is configured as an input pin.

PORTx Pins

If PORTxn is written logic one when the pin is configured as an input pin, the pull-up resistor is activated. To switch the pull-up resistor off, PORTxn has to be written logic zero or the pin has to be configured as an output pin. If PORTxn is written logic one when the pin is configured as an output pin, the port pin is driven high (one). If PORTxn is written logic zero when the pin is configured as an output pin, the port pin is driven low (zero).

PINx Register

The PINx register gets the reading from the input pins of the microcontroller.

ADC

ATmega16 has a 10-bit successive approximation ADC. The ADC is connected to an 8-channel Analog Multiplexer that allows 8 single-ended voltage inputs from the pins of Port A. The ADC converts an analog input voltage to a 10-bit digital value through successive approximation. The minimum value represents GND and the maximum value represents the voltage on the AREF pin minus 1 LSB. Optionally, AVCC or an internal 2.56V reference voltage may be connected to the AREF pin by writing to the REFSn bits in the ADMUX Register. The internal voltage reference may thus be decoupled by an external capacitor at the AREF pin to improve noise immunity.

ADMUX: ADC Multiplexer Selection Register

ADMUX is 8 - bit register. The bitwise description of ADMUX register is shown in Figure 3.2

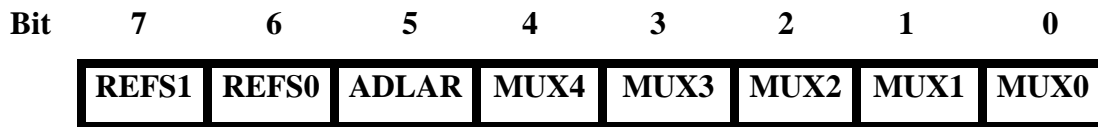


Figure 3.2: Bitwise description of ADMUX

Bit 7:6 – REFS1:0: Reference Selection Bits

These two bits select the voltage reference for the ADC. Voltage reference selection based on the status of these two pins is shown in Table 3.1. If these bits are changed during a conversion, the change will not go in effect until this conversion is complete.

Table 3.1: Voltage reference selection for ADC

REFS1	REFS0	Voltage reference selection
0	0	AREF, internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V with external capacitor at AREF pin

Bit 5 – ADLAR: ADC Left Adjust Result

These two bits are used to adjust the result. To left adjust the result write one to ADLAR. Otherwise, the result is right adjusted.

Bits 4:0 – MUX4:0: Analog Channel Selection

The value of these bits selects which combination of analog inputs are connected to the ADC. The table 3.2 shows the selection of analog channel based on the combination of these 4 bits. These bits also select the gain for the differential channels. If these bits are changed during a conversion, the change will not go in effect until this conversion is complete

Table 3.2: Analog channel selection table

MUX 4..0	Single ended input
00000	ADC0
00001	ADC1
00010	ADC2
00011	ADC3
00100	ADC4
00101	ADC5
00110	ADC6
00111	ADC7

ADCSRA: ADC Control and Status Register A

ADCSRA is 8 - bit register. The bitwise description if ADCSRA register is shown in Figure 3.3



Figure 3.3: Bitwise description of ADCSRA

Bit 7 – ADEN: ADC Enable

ADC is enabled by putting one to ADEN bit

Bit 6 – ADSC: ADC Start Conversion

In Single Conversion mode, this bit is written with one to start each conversion. In Free Running Mode, this bit is written with one to start the first conversion.

Bit 5 – ADATE: ADC Auto Trigger Enable

Auto Triggering of the ADC can be enabled by writing this bit to one. The ADC will start a conversion on a positive edge of the selected trigger signal.

Bit 4 – ADIF: ADC Interrupt Flag

This bit is set to one when an ADC conversion completes and the Data Registers are updated.

Bit 3 – ADIE: ADC Interrupt Enable

The ADC Conversion Complete Interrupt is activated when this bit is written to one and the I-bit in SREG is set.

Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits

These bits determine the division factor between the XTAL frequency and the input clock to the ADC as in Table 3.3

Table 3.3: ADC prescaler select bits

ADPS2	ADPS1	ADPS0	Division factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Timer

Timer/Counter0 is a general purpose, single channel, 8-bit Timer/Counter module.

TCCR0: Timer/Counter Control Register

TCCR0 is 8 – bit register. Figure 3.4 shows a bitwise description of TCCR0 register.

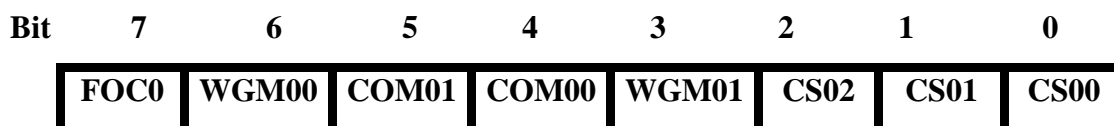


Figure 3.4: Bitwise description of TCCR0

FOC0: Force Output Compare

The FOC0 bit is only active when the WGM00 bit specifies a non-PWM mode

Bit 6, 3 – WGM01:0: Waveform Generation Mode

These bits control the counting sequence of the counter, the source for the maximum (TOP) counter value, and what type of Waveform Generation to be used. The operation mode selection is according the status of these two bits as in Table 3.4

Table 3.4: Operation mode selection

Mode	WGM01	WGM00	Timer/Counter Mode of operation
0	0	0	Normal
1	0	1	PWM, Phase correct
2	1	0	CTC
3	1	1	Fast PWM

Bit 5:4 – COM1:0: Compare Match Output Mode

These bits control the Output Compare pin (OC0) behaviour as in Table 3.5

Table 3.5: Output compare (OC) behaviour

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected
0	1	Toggle OC0 on compare match
1	0	Clear OC0 on compare match
1	1	Set OC0 on compare match

Bit 2:0 – CS02:0: Clock Select

These three bit select the clock as per Table 3.6

Table 3.6: Clock selection

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clkI/O/(No prescaler)
0	1	0	clkI/O/8 (From prescaler)
0	1	1	clkI/O/64 (From prescaler)
1	0	0	clkI/O/256 (From prescaler)
1	0	1	clkI/O/1024 (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

Timer/Counter Register – TCNT0

The Timer/Counter Register gives direct access, both for read and writes operations, to the Timer/Counter unit 8-bit counter.

Output Compare Register – OCR0

The Output Compare Register contains an 8-bit value that is continuously compared with the counter value (TCNT0). A match can be used to generate an output compare interrupt, or to generate a waveform output on the OC0 pin [29].

DC Geared Motor

DC motors are inexpensive, small, and powerful motors that are widely used. DC geared motor is shown in Figure 3.5. Gear-train reductions are typically needed to reduce the speed and increase the torque output of the motor. A DC motor is an electric motor that runs on direct current (DC) electricity. DC motors are normally very easy to reverse simply by changing the polarity of the DC input. This changeover process can be achieved via a simple changeover switch or for remote or electronic control, via a suitable relay. A big advantage of DC motors is

that variable speed control is easy and can achieve with just a suitable variable resistor / rheostat or variable DC power supply.



Figure 3.5: DC geared motor

DC Motor Speed Control

The speed of a DC motor is directly proportional to the supply voltage, so if the supply voltage reduces from 12 Volts to 6 Volts, the motor will run at half the speed. Pulse-width modulation (PWM) or duty-cycle variation methods are commonly used in speed control of DC motors. The duty cycle is defined as the percentage of digital 'high' to digital 'low' plus digital 'high' pulse-width during a PWM period. The average DC voltage value for 0% duty cycle is zero; with 25% duty cycle the average value is 1.25V (25% of 5V). With 50% duty cycle the average value is 2.5V, and if the duty cycle is 75%, the average voltage is 3.75V and so on. The maximum duty cycle can be 100%, which is equivalent to a DC waveform. From this method one can obtain a smooth speed variation without reducing the starting torque of the motor. PWM technique also eliminates harmonics. Thus by varying the pulse-width, we can vary the average voltage across a DC motor and hence its speed.

Motor Driver

The L293 and L293D are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. It is a 16 pin IC. The pin diagram of this IC is shown in Figure 3.6 L293D is a dual H-bridge motor driver integrated circuit (IC). Motor drivers act as current amplifiers since they take a low-current

control signal and provide a higher-current signal. This higher current signal is used to drive the motors.

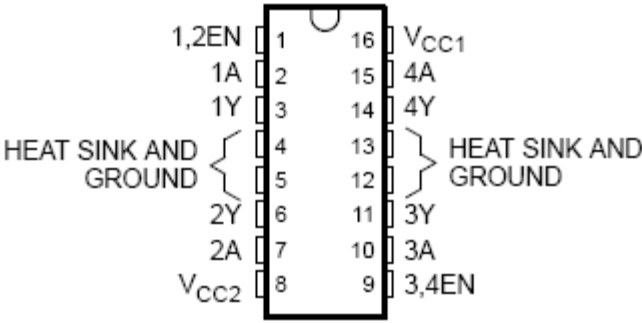


Figure 3.6: Pin diagram of L293D

L293D contains two inbuilt H-bridge driver circuits. In its common mode of operation, two DC motors can be driven simultaneously, both in forward and reverse direction. The motor operations of two motors can be controlled by input logic at pins 2 and 7 and 10 and 15. Input logic 00 or 11 will stop the corresponding motor. Logic 01 and 10 will rotate it in clockwise and anticlockwise directions, respectively. Enable pins 1 and 9 (corresponding to the two motors) must be high for motors to start operating. When an enable input is high, the associated driver gets enabled. As a result, the outputs become active and work in phase with their inputs. Similarly, when the enable input is low, that driver is disabled, and their outputs are off and in the high-impedance state. The motor action according to the input pins is shown in Table 3.7.

Table 3.7: Motor action based on status of input pins

Input(1A,4A)	Input(2A,3A)	Motor action
Logic 0	Logic 1	Moves clockwise
Logic 1	Logic 0	Moves anticlockwise
Logic 1	Logic 1	Stop
Logic 0	Logic 0	Stop

Thus, the motors behave as per the control signals generated using the microcontroller with the excitation from the external battery voltage [31].

Shaft Encoder

Shaft encoder is used to measure the angular position. It is also called as rotary encoder. This device measures the angular position and outputs the analog or digital code. The shaft encoder is shown in Figure 3.7



Figure 3.7: Shaft encoder

This is the absolute encoder. The shaft of such encoders can be attached to the shaft of any rotating device. In this way the output of such encoders gives the current position of shaft. They produce a unique binary code for each different position of shaft. There is a metal disc that contains the concentric rings openings as in Figure 3.8

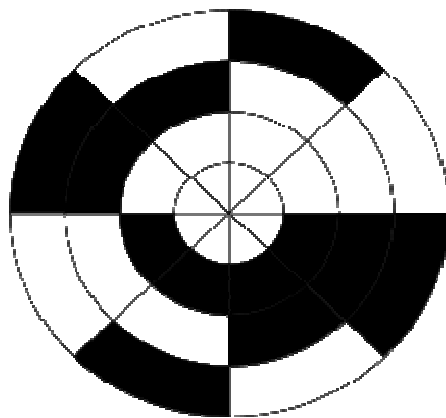


Figure 3.8: 3-Bit binary coding disc

Those rings are fixed to an insulating disc, which is fixed to the shaft. Each concentric ring is divided into segments. The number of segments depends upon the code being used. There are some sliding contacts that are fixed to a stationary object and wiper against the metal disc. There are some areas where the metal has been taken out. The disc will rotate with the rotation of shaft. Due to this some of the contacts will touch metal and some may touch the area where the metal has been taken out. As the disc rotates with the shaft, some of the contacts touch metal and some fall in the gaps where the metal has been cut out. The metal sheet is connected

to a source of electric current. Each contact is connected to a separate electrical sensor. The metal pattern is designed so that each possible position of the axle creates a unique binary code in which some of the contacts are connected to the current source and others are not. This code can be read by a controlling device, such as a microprocessor or microcontroller to determine the angle of the shaft. If there are n contacts, the number of distinct positions of the shaft would be 2^n e.g. for n=3 there would be 8 positions. The standard binary coding for 3 contacts is shown in Table 3.8

Table 3.8: Standard binary coding

Sector	Contact1	Contact 2	Contact 3	Angle
0	Off	Off	Off	0° to 45°
1	Off	Off	On	45° to 90°
2	Off	On	Off	90° to 135°
3	Off	On	On	135° to 180°
4	On	Off	Off	180° to 225°
5	On	Off	On	225° to 270°
6	On	On	Off	270° to 315°
7	On	On	On	315° to 360°

As the disc rotates the contacts produce a standard binary count. This has the drawback that if the disc stops between two adjacent sectors, or the contacts are not perfectly aligned, it can be impossible to determine the angle of the shaft. To avoid this problem, Gray encoding is used.

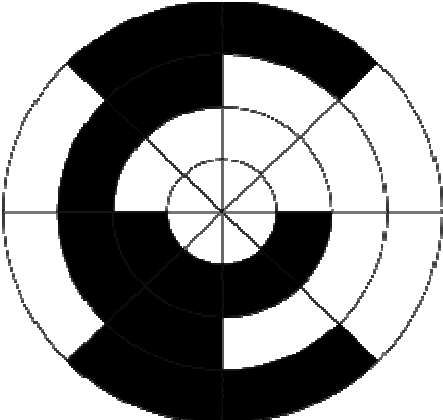


Figure 3.9: Gray coding disc

This is a system of binary counting in which adjacent codes differ in only one position. The metal disc for the gray coding is shown in Figure 3.9. Table 3.9 shows the angle according to the gray coding [30].

Table 3.9: Gray coding

Sector	Contact1	Contact 2	Contact 3	Angle
0	Off	Off	Off	0° to 45°
1	Off	Off	On	45° to 90°
2	Off	On	On	90° to 135°
3	Off	On	Off	135° to 180°
4	On	On	Off	180° to 225°
5	On	On	On	225° to 270°
6	On	Off	On	270° to 315°
7	On	Off	Off	315° to 360°

Switches

In electronics, a switch is an electrical component that can make or break an electrical circuit, interrupting the current or diverting it from one conductor to another. The most familiar form of switch is a manually operated electromechanical device with one or more sets of electrical contacts. Each set of contacts can be in one of two states: either 'closed' meaning the contacts are touching and electricity can flow between them, or 'open', meaning the contacts are separated and not conducting. Figure 3.10 shows the push button type switch.



Figure 3.10: Push button switch

A switch may be directly manipulated by a human as a control signal to a system, such as a computer keyboard button, or to control power flow in a circuit, such as a light switch. Automatically-operated switches can be used to control the motions of machines. All switches are connected to appropriate ports for a desired function

Software Used

CodeVisionAVR:

CodeVisionAVR Evaluation V2.05.4 compiler used for programming of microcontroller. It is a C cross-compiler, Integrated Development Environment and Automatic program Generator designed for the Atmel AVR family of microcontrollers. The program is designed to run under the Windows 98, NT 4, 2000, XP and Vista 32 bit operating systems. The C cross-compiler implements all the elements of the ANSI C language, as allowed by the AVR architecture, with some features added to take advantage of speciality of the AVR architecture and the embedded system needs. The compiled COFF object files can be C source level debugged, with variable watching, using the Atmel AVR studio debugger.

Besides the standard C libraries, the CodeVision AVR C compiler has dedicated libraries for:

- Alphanumeric LCD modules
- Philips I2C bus
- SPI
- Delays
- ADCs etc

CodeVision AVR also contains Automatic Program Generator that allows us to write, in a matter of minutes, all the code needed for implementing the following functions:

- Timers/counters initialization
- ADC initialization
- Input/output port initialization
- LCD module initialization
- External interrupt initialization etc
- SPI initialization

3.3 System Description

The basic blocks of the whole system is shown in Figure 3.11. The main blocks are microcontroller, motors, motor driver and position sensing elements.

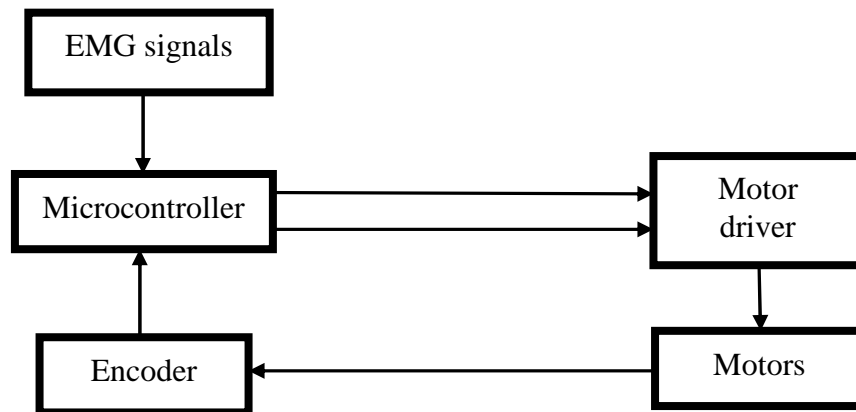


Figure 3.11: Basic block diagram of the system

DC signals from the potentiometer were fed to the microcontroller. The whole operation of the system was controlled by microcontroller. The whole code written for the specified operation was burnt into the microcontroller chip. The control signals for motor were provided by microcontroller. Motor driver chip was in between the microcontroller and motors. Encoder here was working as a positions sensing element. Encoder was used to sense the motor position and to provide those signals microcontroller. So the microcontroller made action according to the current position of motor.

The connection diagram for the whole system is in Figure 3.12. The microcontroller in this dissertation was Atmel ATmega16. It has 8 channels 10 bit successive approximation ADC. Port A pins are used as ADC pins. Two ADC channels were used. Port D pins were used to drive the motor. Some pins of port D were connected to input pins of motor controller to control the motor. Port B pin (PB3) was connected to enable pins of motor driver for both the motors. PB3 pin is the default pin to output the PWM signal in ATmega16. The width of the signal was controlled by the timer registers. So the speed of both motors was controlled by this

pin. The two motors for arm up/down and arm clockwise/anticlockwise were connected to the output pins of motor driver. There were two shaft encoders. They were used as a position sensing elements. They were used to sense the position of both the motors and send the signals to microcontroller. So the microcontroller makes action according to the signal fed by these shaft encoders.

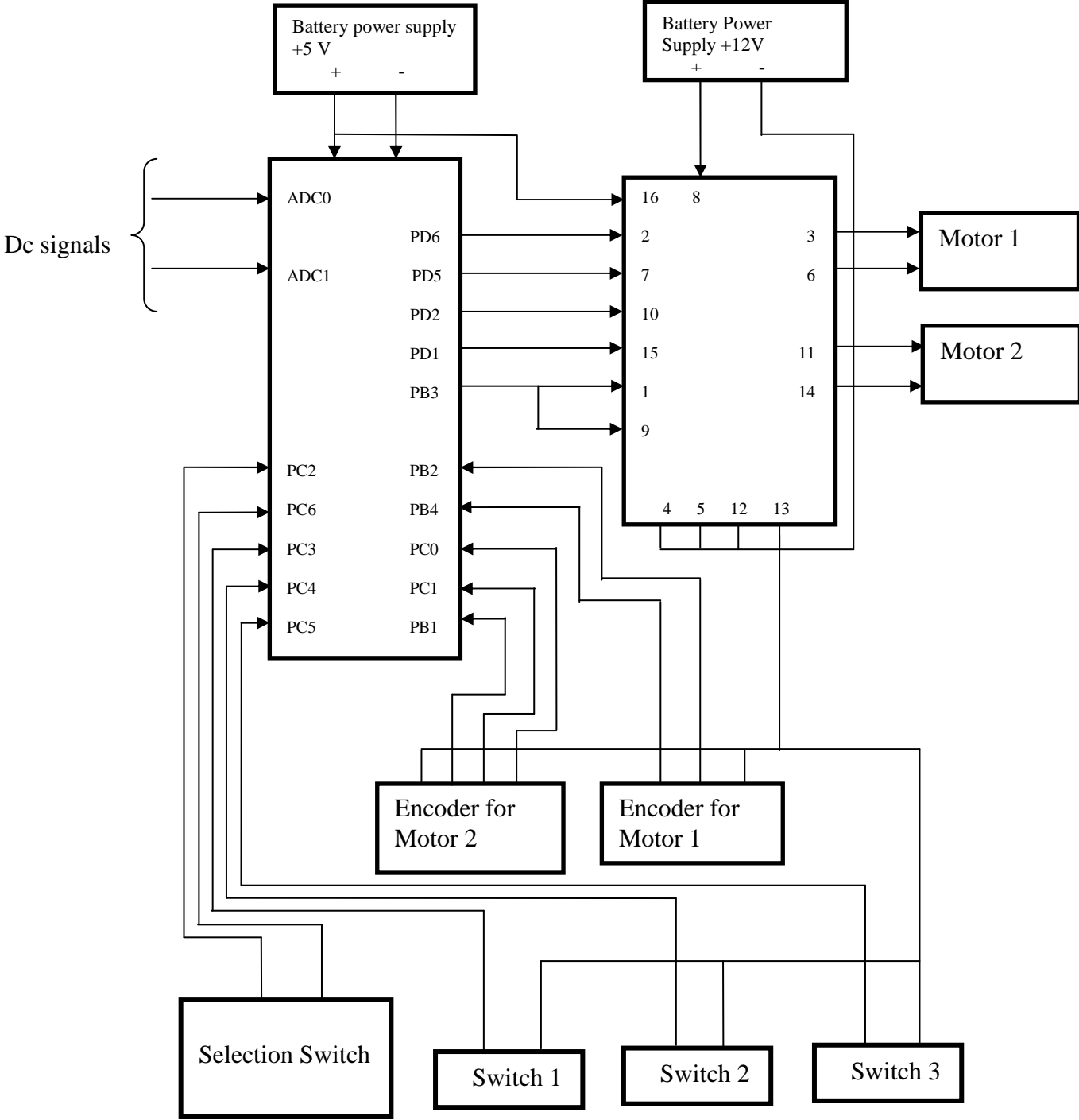


Figure 3.12: Connection diagram

Table 3.10: Port pin logic

Port pins	Status				Function
PD6	Logic 1	Arm up	Logic 0	Arm down	Microcontroller instruct the driver to control Elbow motor
PD5	Logic 0		Logic 1		
PD2	Logic 1	wrist clockwise	Logic 1	Wrist anticlockwise	Microcontroller instruct the driver to control wrist motor
PD1	Logic 0		Logic 0		
PB3					PWM signal for motor speed control
PB2					Encoder output for motor 1
PB4					
PC0					Encoder output for motor 2
PC1					
PB1					
PA0					Analog input from potentiometer representing the EMG signal
PA1					Analog input from potentiometer representing the EMG signal
PC2					Selection switch
PC6					Selection switch
PC3					Switch S1
PC4					Switch S2
PC5					Switch S3

In the present study two potentiometers were used to represent dc level of EMG signals. The program to control the the whole system burnt into the microcontroller. The microcontroller kit is shown in Figure 3.13

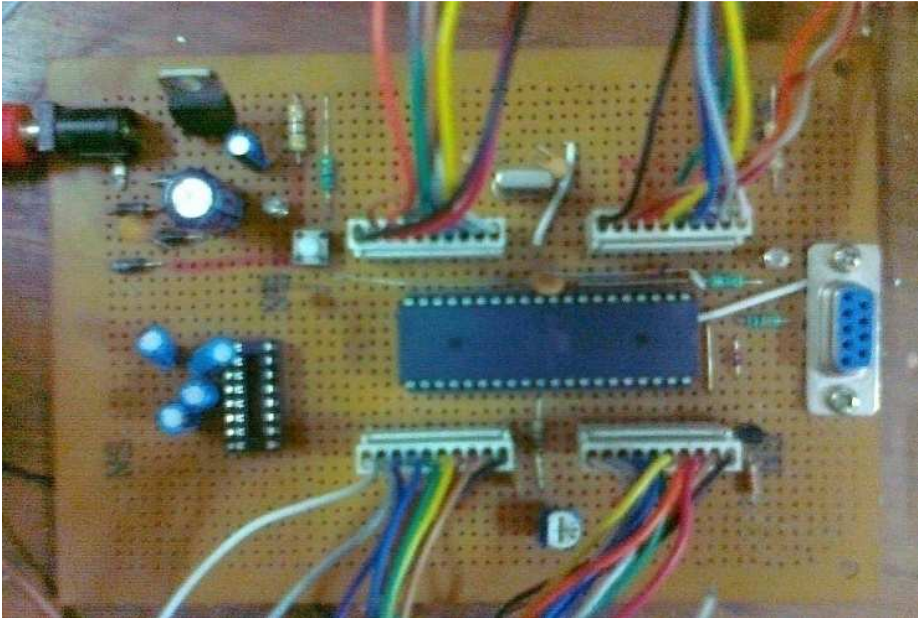


Figure 3.13: Microcontroller kit

The command signals were generated by microcontroller according to the code burnt into it. These command signals fed to the H bridge chip. The H bridge chip is shown in Figure 3.14.

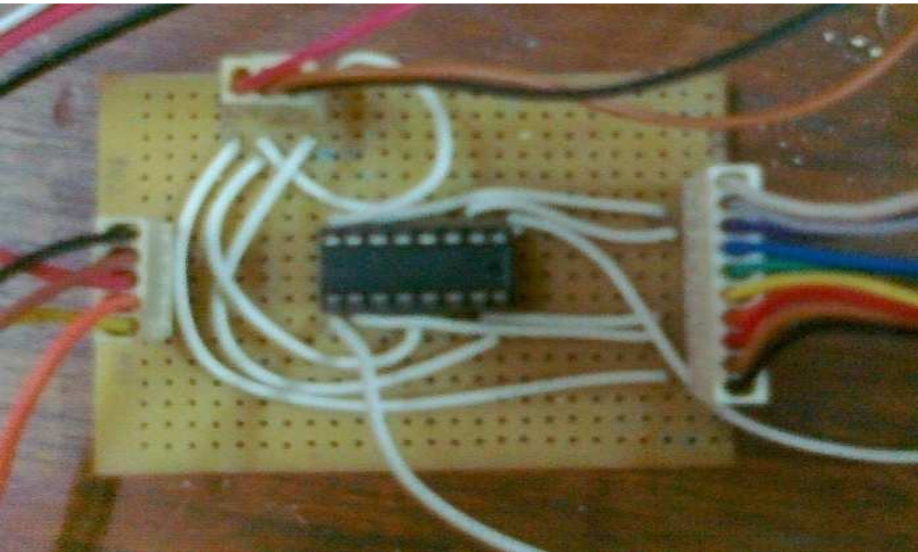


Figure 3.14: H bridge circuit

H bridge chip generate the output signals to drive the specified motor in a specific direction for a specific time. There were two motors for arm up/down movement and clockwise/anticlockwise operation of the arm as in Figure 3.15.

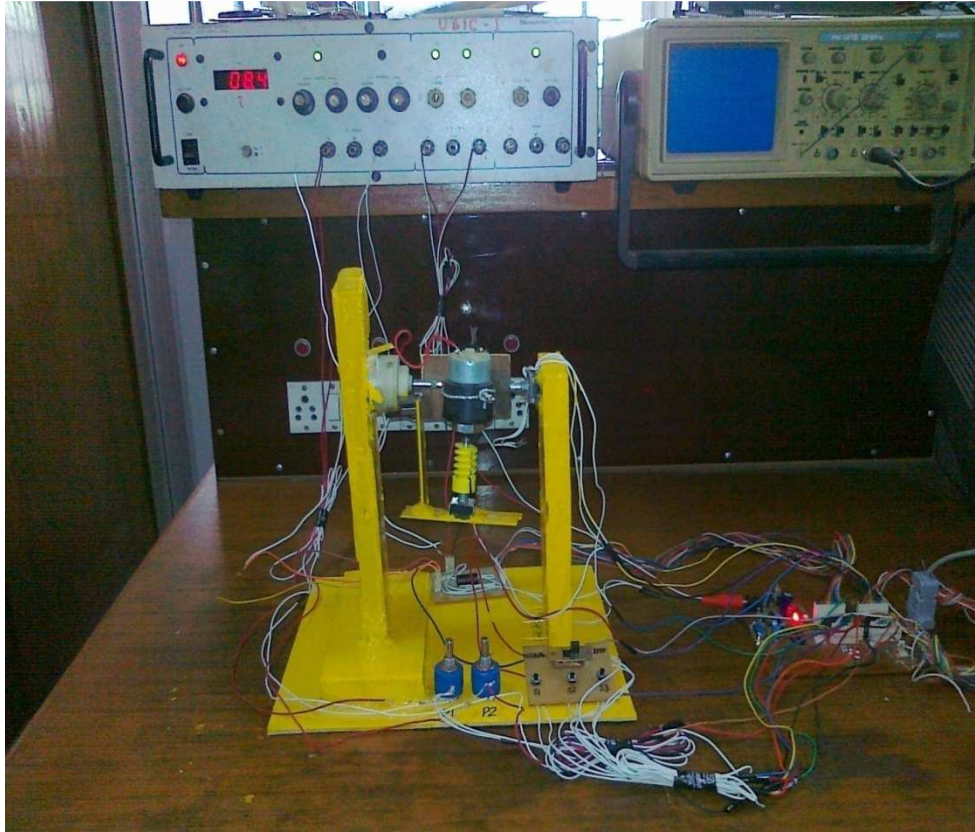


Figure 3.15: Arm prototype having motors and encoders

The motors were moved for a specific period of time and in specific direction depending on the voltage level of EMG signal. The shaft encoders were attached to the motors to read out the current motor position.

3.4 System Operation

The system can operate in two ways. This is based on the position of selection switch. Initially the selection switch is in manual position. Both the motors are at rest position. Both the potentiometers are at initial position. The system can be operated in two modes manual and EMG. This is explained in Figure 3.16. The selection of the mode is done through selection switch.

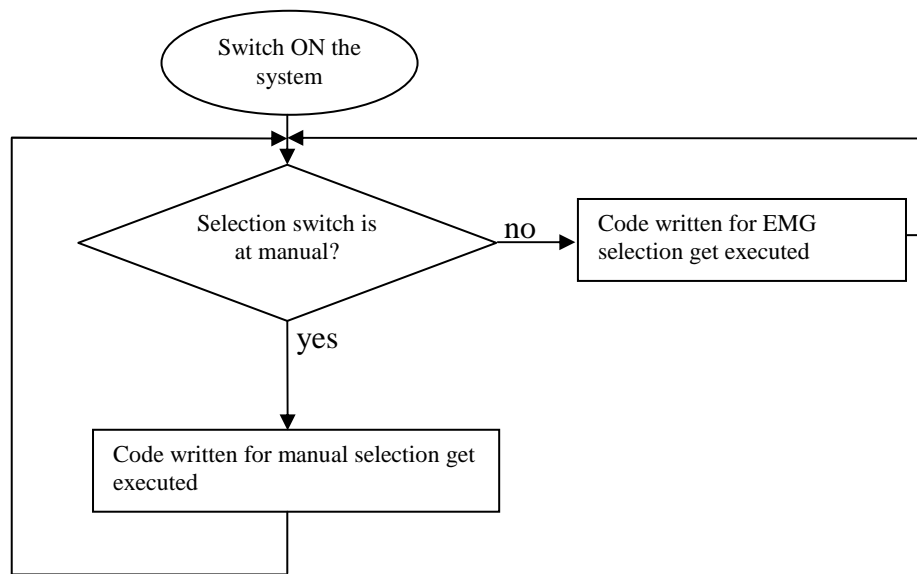


Figure 3.16: Selection between manual and EMG

It was done through the position of manual switch. When the switch was on the manual position, the system is operated through manual switches S1 S2 and S3. When the switch is on the EMG position, the system is operated by two potentiometers that indicate the dc level of EMG signals.

3.4.1 Selection Switch in Manual Position

When the selection switch was on manual position the system is operated through manual switches. There were 3 manual switches that were used to perform the desired function. These switches were marked as S1, S2 and S3. The whole operation is depicted in Figure 3.17. Two movements i.e. arm up/down and arm clockwise/anticlockwise movement were performed by whole system. When switch S2 was pressed alone the system perform arm up movement. The switch was push button type. So the amputee presses the switch until get the movement up to the desired extent. Similarly if amputee presses switch S3 alone the system perform arm down movement. To perform the arm clockwise/anticlockwise movement all three switches were used. The combination of two switches was used to perform the desired function. If the amputee presses switch S1 and S2 together the system perform the arm clockwise movement.

Similarly if the amputee presses switch S1 and S3 together the system perform the arm anticlockwise movement.

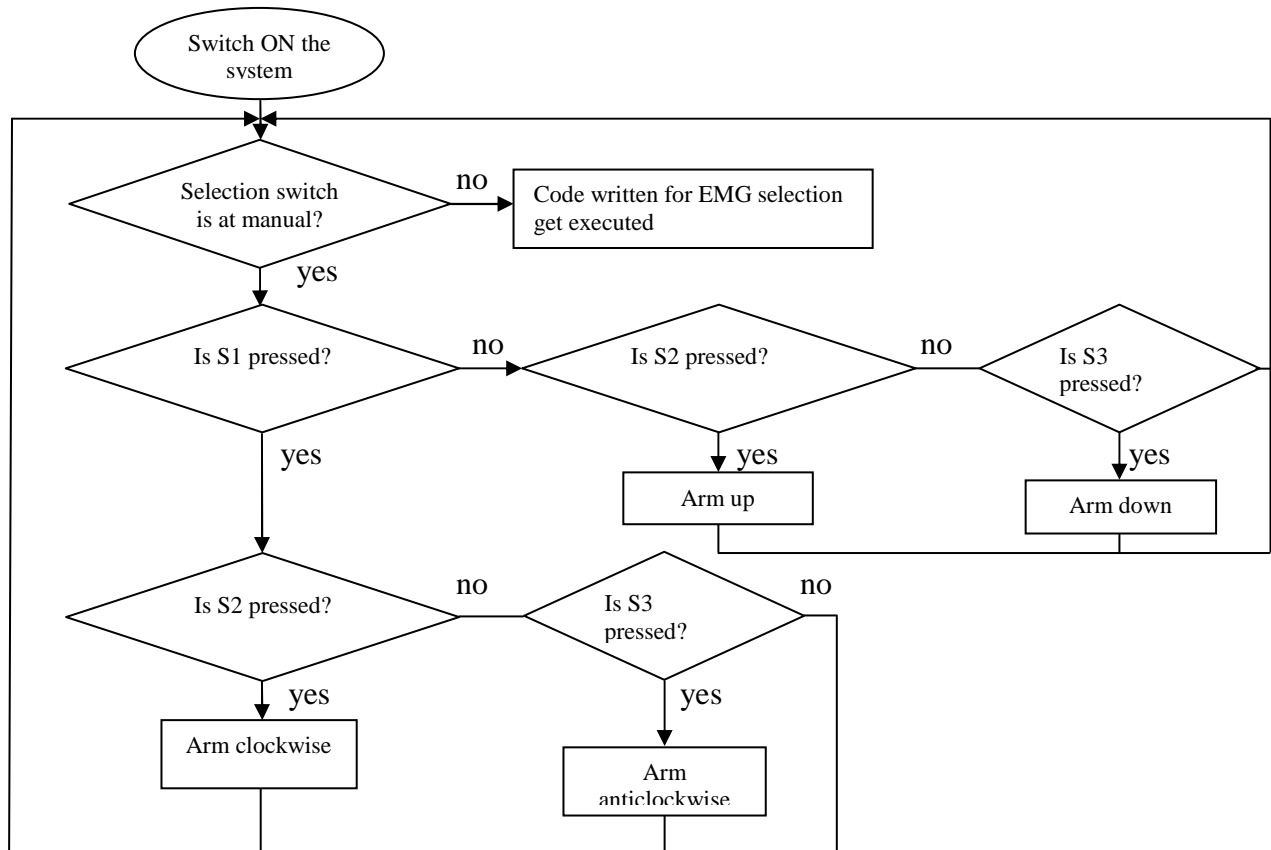


Figure 3.17: Manual operation

3.4.2 Selection Switch in EMG Position

When the selection switch was not at manual position the code written for EMG selection get executed. The two potentiometers were used to indicate the amplified EMG signal. Initially the two potentiometers and both motors were at initial position. The two motors were for up/down movement and clockwise/anticlockwise movement. Here, 3 positions were taken for both motors, *i.e.*, initial position, 1st position and 2nd position, as shown in Figure 3.18.

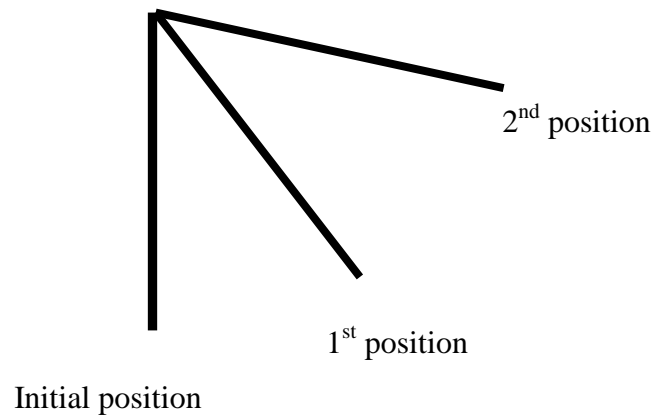
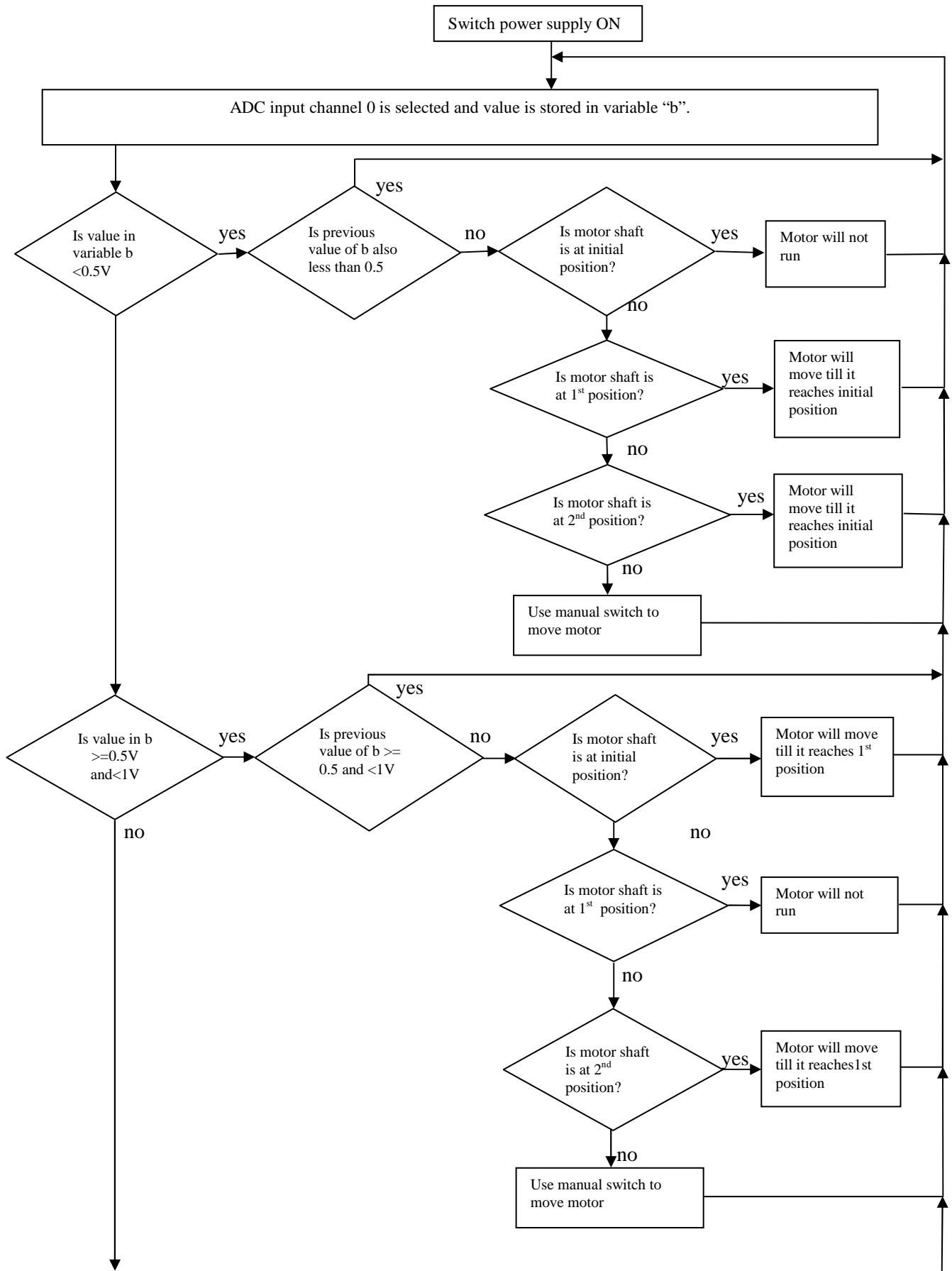


Figure 3.18: Different position for motor

Initially when both potentiometers were at initial position, the analog voltage to both the ADC channels was 0 volts. The dc level varied with the potentiometer connected to the ADC channel. The microcontroller read the dc level at the ADC channel. The command signal was generated by microcontroller according to the motor shaft position. The motor shaft position was detected by the shaft encoder connected to the shaft of the motor. The whole operation is explained through flow chart as in Figure 3.19. The command signal and the duration of the delay were according to the current position of motor. In this way arm up/down movement performed according to the level of the dc signal. Similarly arm clockwise/anticlockwise movement was performed by variation in the dc level of the signal provided by second potentiometer. For arm clockwise/anticlockwise movement three positions were fixed. The movement of the motor was according to the input dc signal and shaft encoder. Separate encoder was attached to the arm clockwise/anticlockwise motor shaft. Two manual switches were used to correct the position of arm in case of any failure. Flowchart shown in Figure 3.19 is for arm up/down movement. For arm clockwise/anticlockwise movement the flow chart would be similar.



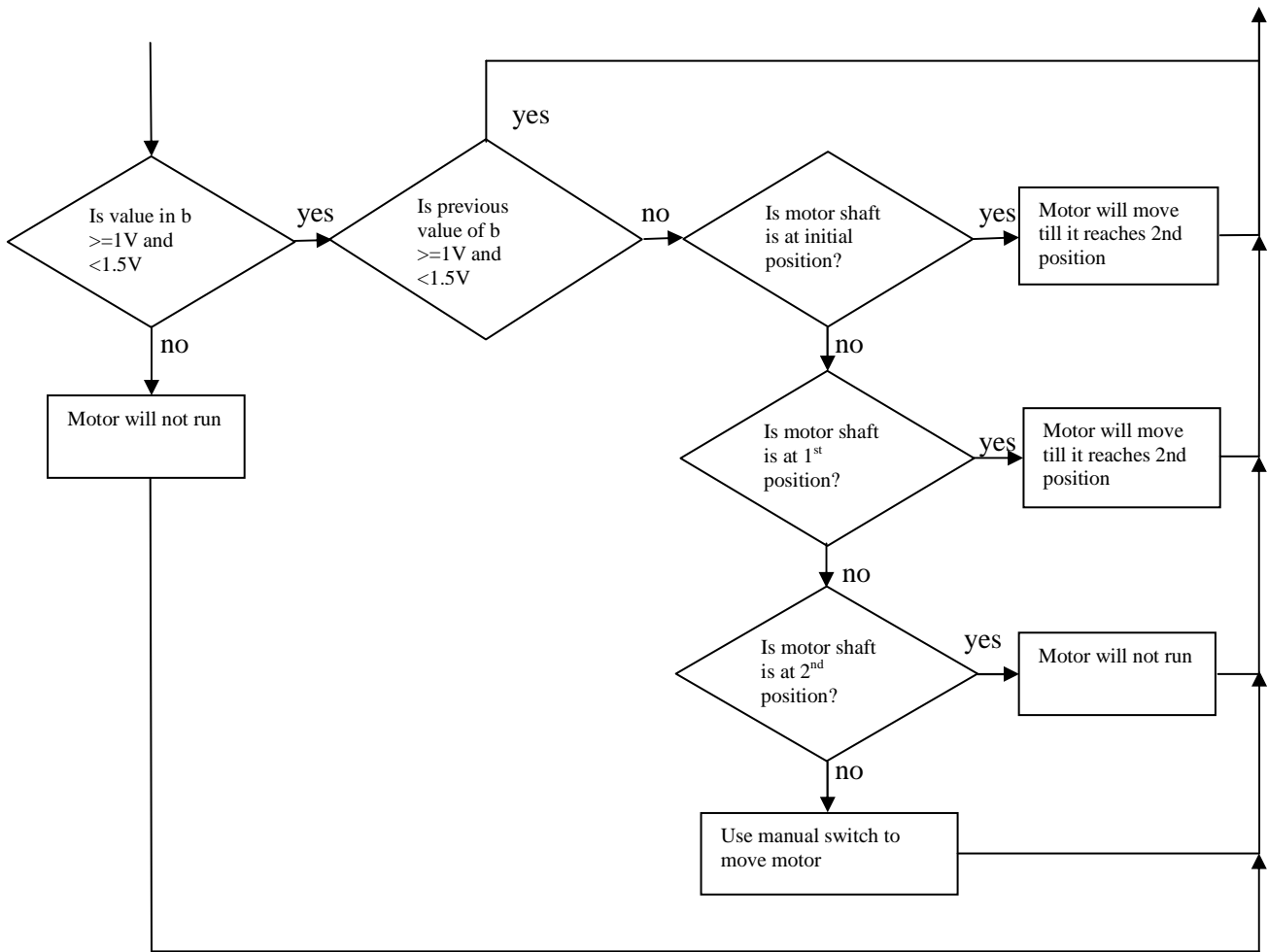


Figure 3.19: System operation for EMG selection

4. RESULTS AND DISCUSSIONS

The amputee can choose between manual control by switches or by EMG signal by the selection switch.

4.1 Manual Control

If the amputee put the switch in “manual” the arm is controlled by switches S1, S2 and S3



Figure 4.1: Selection switch in manual position

When switch S2 is pressed alone the system will perform arm up movement. The switch is push button type. So the amputee will press the switch until get the movement up to the desired extent. Similarly if amputee presses switch S3 alone the system will perform arm down movement. To perform the arm clockwise/anticlockwise movement all three switches are used. The combination of two switches will perform the desired function. If the amputee presses switch S1 and S2 together the system will perform the arm clockwise movement. Similarly if the amputee presses switch S1 and S3 together the system will perform the arm anticlockwise movement.

4.2 EMG Control

If the amputee put the switch in “EMG” the arm is controlled by EMG signals.



Figure 4.2: Selection switch in EMG position

Here potentiometers are used to represent amplified dc signal corresponding to EMG signal. The two potentiometers are marked as P1 and P2



Figure 4.3: Potentiometers to provide input signal

4.2.1 Arm Up/Down Movement

Switch P1 control the arm up/down movement. When level of the dc signal is below 0.5V the arm up/down motor should be at rest. The movement of motor for the arm up/down movement depend on the motor position. The dedicated position is rest position. If at that time arm is at 1st or 2nd position then the arm up/down motor will move and arm will come to the rest position.

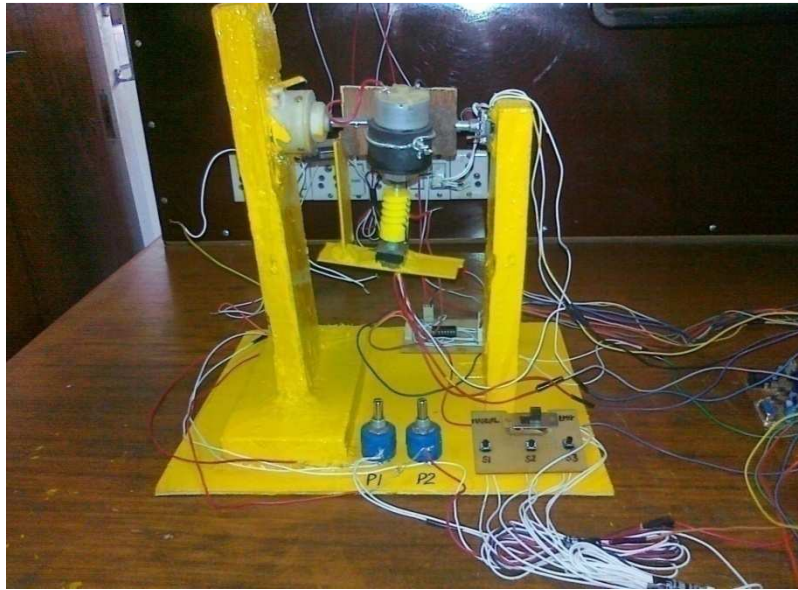


Figure 4.4: Arm up/down movement (at rest)

When level of the dc signal is between 0.5V and below 1V then movement of motor for the arm up/down movement depend on the motor position. The dedicated position is 1st position. If at that time arm is at rest or 2nd position then arm up/down motor will move and arm will come to the 1st position.

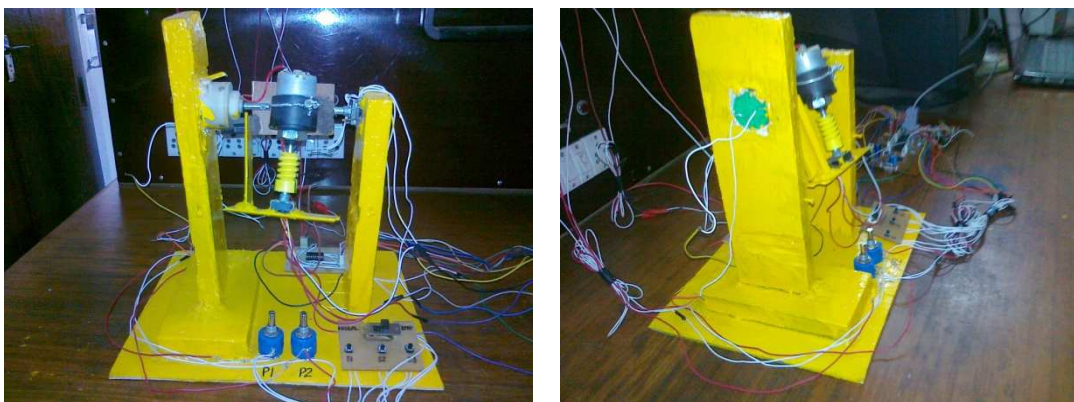


Figure 4.5: Arm up/down movement (at 1st position)

When level of the dc signal is between 1V and below 1.5V, then movement of motor for the arm up/down movement depend on the motor position. The dedicated position is 2nd position. If at that time arm is at rest or 1st position then arm up/down motor will move and arm will come to the 2nd position.

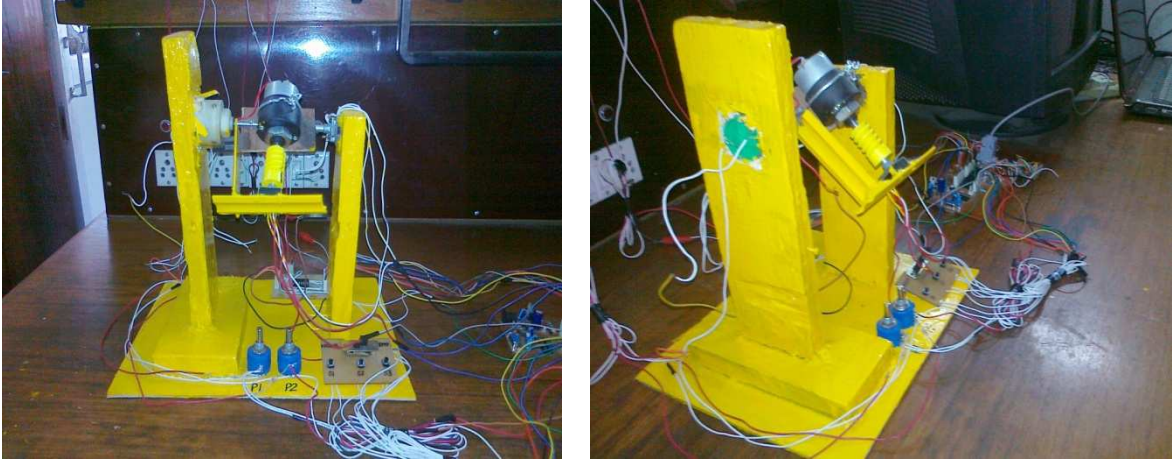


Figure 4.6: Arm up/down movement (at 2nd position)

4.2.2 Arm Clockwise/Anticlockwise Movement

Switch P2 control the arm up/down movement. When level of the dc signal is below 0.5V, then movement of motor for the arm clockwise/anticlockwise movement depend on the motor position. The dedicated position is rest position. If at that time arm is at 1st or 2nd position then the arm clockwise/anticlockwise motor will move and arm will come to the rest position.

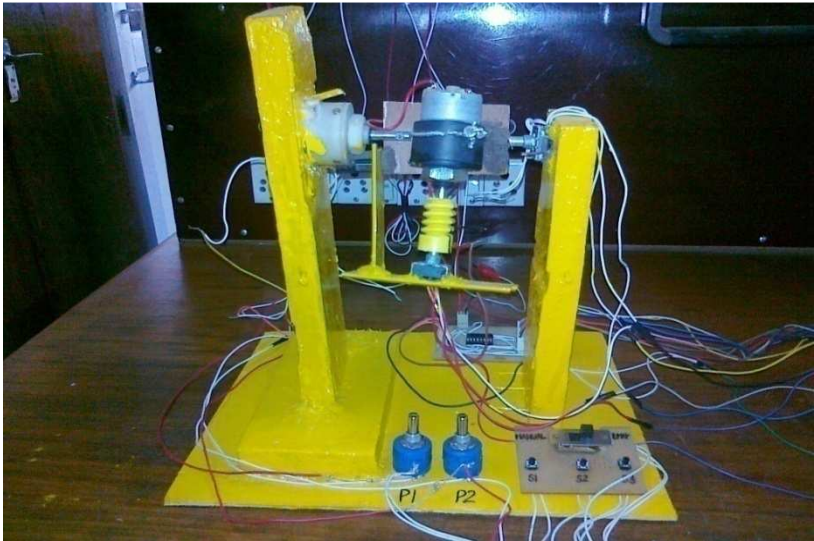


Figure 4.7: Arm clockwise anticlockwise movement (at rest)

When level of the dc signal is between 0.5V and below 1V then movement of motor for the arm up/down movement depend on the motor position. The dedicated position is 1st position. If at that time arm is at rest or 2nd position then arm up/down motor will move and arm will come to the 1st position.

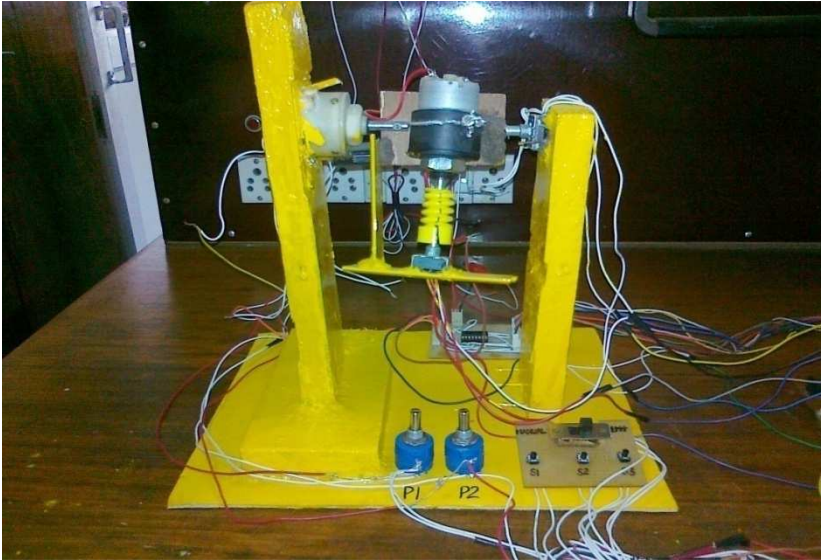


Figure 4.8: Arm clockwise anticlockwise movement (at 1st position)

When level of the dc signal is between 1V and below 1.5V then movement of motor for the arm up/down movement depend on the motor position. The dedicated position is 2nd position. If at that time arm is at rest or 1st position then arm up/down motor will move and arm will come to the 2nd position.



Figure 4.9: Arm clockwise anticlockwise movement (at 2nd position)

5. CONCLUSION AND FUTURE SCOPE

Prosthetic arm is a boon for those persons who have lost their arm due to some mishap. One of the main requirements of artificial arm is it's functionality. It should be near to natural hand as possible. In the present study a prototype of prosthetic arm has been developed for two movements – arm up/down movement and arm clockwise/anticlockwise movement. This system was designed, fabricated and has been tested successfully.

5.1 Future Scope

- More precision work can be done. The prototype can be tested using actual EMG signals instead of inputs from potentiometer.
- More arm functions can be added to improve DOF.

REFERENCES

- [1] Ashok Muzumdar, Powered Upper Limb Prostheses, Chapter 1, 1st Ed, Ottawa, Canada, pp 1-10, 1996
- [2] Katsutoshi Kuribayashi, Makoto Takahashi, Takao Taniguchi, An Upper Extremity Prosthesis Using SMA Actuator, *IEEE International Workshop on Robot and Human Communication*, pp 52-57, Japan, September 1992
- [3] S. Schulz, C. Pylatiuk and G. Bretthauer, A New Ultra Light Anthropomorphic Hand, *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 3, pp 2437-2441, Korea, May 2001
- [4] V. Artigue, G. Thomann, Development of a Prosthetic Arm: Experimental Validation with the User and Adapted Software, *IEEE International Conference on Robotics and Automation*, pp 3079- 3084, Japan, May 2009
- [5] Z. Escudero, L. Leija, Ja. Alvarez, R. Munoz, Upper Limb Prosthesis Controlled by Myoelectric Signal, *Proceedings of The First Joint BMES/EMBS Conference Serving Humanity, Advancing Technology*, Vol. 1, pp 636, Atlanta, October 1999
- [6] Daniel Graupe, Jan Magnussen, Aloysius A. Beex, A Microprocessor System for Multifunctional Control of Upper-Limb Prostheses via Myoelectric Signal Identification, *IEEE Transactions on Automatic Control*, Vol. 23, pp 538-544, USA, August 1978
- [7] George N.Saridis, Thomas P. Gootee, EMG Pattern Analysis and Classification for a Prosthetic Arm, *IEEE Transactions on Biomedical Engineering*, Vol. BME-29, pp 403-412, June 1982
- [8] H.T. Law and J. J. Hewson, An Electromyographically Controlled Elbow Locking Mechanism for an Upper Limb Prosthesis, *Electrocomponent Science and Technology*, Vol. 10, pp. 87-93, UK, Oct 1983
- [9] David T. Gibbons, Micheal D. O'riain, Sebastien Philippe Auguste, An Above-Elbow Prosthesis Employing Programmed Linkages, *IEEE Transactions on Biomedical Engineering*, Vol. 34, pp 493-498, July 1987
- [10] Yiorgos A. Bertost, Craig W. Heckathorne, Richard F.ff. Weir, and Dudley S. Childress, Microprocessor based E.P.P. Position Controller for Electric-Powered Upper Limb Prostheses, *Proceedings of 19th Annual International Conference of IEEE/EMBS*, Vol. 5, pp 2311-2314, USA, October 1997
- [11] Marcello Mulas, Michele Folgheraiter and Giuseppina Gini, An EMG-Controlled Exoskeleton for Hand Rehabilitation, *Proceedings of IEEE 9th International IEEE Conference on Rehabilitation Robotics*, pp 371-374, USA, July 2005
- [12] Kentaro Nagata, Keiichi Adno, Masafumi Yamada, Kazushige Magatani, A Classification Method of Hand Movements Using Multi Channel Electrode, *Proceedings of 27th Annual IEEE Conference Engineering in Medicine and Biology Society* , pp 2375-2378, China, September 2005
- [13] Shahjahan Shahid, Jacqueline Walker, Gerard M. Lyons, Ciaran A. Byrne, and Anand Vishwanath Nene, Application of Higher Order Statistics Techniques to EMG Signals to Characterize the Motor Unit Action Potential, *IEEE Transactions on Biomedical Engineering*, Vol. 52, pp 1195-1209, July 2005
- [14] HaeOck Lee, Dawnlee J. Roberson, A Systemic View of an Upper Extremity Prosthesis, *IEEE International Conference on System of Systems Engineering*, pp 1-6, April 2007
- [15] Guanglin Li and Todd A. Kuiken, Modeling of Prosthetic Limb Rotation Control by Sensing Rotation of Residual Arm Bone, *IEEE Transactions on Biomedical Engineering*, Vol. 55, pp 2134 – 2142, September 2008

- [16] Subrata Kumar Kundu, and Kazuo Kiguchi, Development of a 5 DOF Prosthetic Arm for Above Elbow Amputees, *Proceedings of 2008 IEEE International Conference on Mechatronics and Automation*, pp 207- 212, August 2008
- [17] Shuxiao Wang, K.W.E Cheng, K. Ding, Jiongang Lin, C.D. XU, A Robotic Arm Design for Stroke Patients, *3rd International Conference on Power Electronics Systems and Applications*, pp 1-4, May 2009
- [18] Hardeep S. Ryait, A. S. Arora, and Ravinder Agarwal, Study of Issues in the Development of Surface EMG Controlled Human Hand, *J. Materials Science: Materials in Medicine*, Springer, Vol. 20, pp S107- S114, 2009
- [19] Hiroyuki Takeda, Nobutaka Tsujiuchi, Takayuki Koizumi, Hiroto Kan, Development of Prosthetic Arm with Pneumatic Prosthetic Hand and Tendon-Driven Wrist, *31st Annual International Conference of the IEEE EMBS*, pp 5048 – 5051, September 2009
- [20] Deepak Joshi, Sonal Atreya, A. S. Arora, Sneha Anand, Trends in EMG Based Prosthetic Hand Development: A Review, *Indian J. Biomechanics: Special Issue (NCBM)*, pp 228-232, March 2009
- [21] Tanmay Pal, Chandra Shekhar and Himanshu Dutt Sharma, Design and Development of Embedded System on ARM for On-Line Characterization of DC Motor, *International Conference on Advances in Computing, Control, and Telecommunication Technologies*, pp 502-506, December 2009
- [22] Hardeep S.Ryait, A.S. Arora, Ravinder aggarwal, SEMG-Prosthetic Elbow Controlled with RMS Identification, *A Journal of Bioinformatic and its Applications*, Vol 5, pp 72- 79, October 2010
- [23] S. Herle, S. Man, Gh. Lazea, C. Marcu, P. Raica and R. Robotin, Hierarchical Myoelectric Control of a Human Upper Limb Prosthesis, *19th International IEEE Workshop on Robotics in Alpe-Adria-Danube Region - RAAD*, pp 55-60, June 2010
- [24] Hideyuki Uehara, Hiroki Higa, Takashi Soken, A Mobile Robotic Arm for People with Severe Disabilities, *3rd IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, pp 126-129, September 2010
- [25] Hardeep S. Ryait, A. S. Arora, and Ravinder Agarwal, Interpretations of Wrist/Grip Operations From SEMG Signals at Different Locations on Arm, *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 4, pp 101-111, April 2010
- [26] Vedran Vajnberger, Tarik Terzimehic, Semir Silajdzic and Nedim Osmic, Remote Control of Robot Arm with five DOF, *Proceedings of 34th International Convention MIPRO*, , pp. 1707- 1711, Opatija, Croatia, May 2011
- [27] Hardeep S. Ryait, A. S. Arora, and Ravinder Agarwal, SEMG Signal Analysis at Acupressure Points for Elbow Movement, *Journal of Electromyography and Kinesiology*, Vol 21, pp 868-876, October 2011
- [28] Luay Fraiwan, Mohammed Awwad, Ma'en Mahdawi, Shaher Jamous, Real Time Virtual Prosthetic Hand Controlled Using EMG Signals, *1st Middle East Conference on Biomedical Engineering – MECBME*, pp 225 – 227, February 2011
- [29] Atmega16 microcontroller datasheet at www.atmel.com/Images/doc2466.pdf
- [30] Shaft encoder www.en.wikipedia.org/wiki/Rotary_encoder
- [31] L293 datasheet at <http://www.ti.com/lit/ds/symlink/l293d.pdf>

Microcontroller Code

```
#include <mega16.h>
#include <stdlib.h>
#include <delay.h>

#define ADC_VREF_TYPE 0xC0

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    delay_us(10);
    ADCSRA|=0x40;
    while ((ADCSRA & 0x10)==0);
    ADCSRA|=0x10;
    return ADCW;
}

void main(void)
{
    unsigned int b,c,m=0,n=0,t=0,v=0,r=0,s=0,u=0,w=0,y=0;
    unsigned int val[7]={0xC8,0x118,0x190,0x1E0,0x258,0x2A8,0x320};
    PORTA=0xFF;
    DDRA=0x00;
    PORTB=0xF7;
    DDRB=0x08;
    PORTC=0xFF;
    DDRC=0x00;
    PORTD=0x00;
    DDRD=0xFF;

    TCCR0=0x13;
    TCNT0=0x00;
    OCR0=0x1E;

    ACSR=0x80;
    SFIOR=0x00;

    ADMUX=ADC_VREF_TYPE & 0xff;
    ADCSRA=0xA6;
    SFIOR&=0x1F;
```

```

while (1)
{
    if(PINB.0==0)
    {
        PORTD=0x21;
        delay_ms(10);
        PORTD=0x00;
    }

    if(y==0)
    {
        PORTD=0x00;
        delay_ms(5000);

        if(PINB.2==0 && PINB.4==1)
        {
            while(PINB.2!=1 || PINB.4!=1)
            {
                PORTD=0x40;
                delay_ms(100);
                PORTD=0x00;
                delay_ms(100);
            }
        }
        delay_ms(5000);

        if(PINB.2==1 && PINB.4==1)
        {
            while(PINB.2!=0 || PINB.4!=0)
            {
                PORTD=0x40;
                delay_ms(100);
                PORTD=0x00;
                delay_ms(100);
            }
        }
        delay_ms(5000);

        if(PINB.2==0 && PINB.4==0)
        {
            while(PINB.2!=1 || PINB.4!=1)
            {
                PORTD=0x20;
                delay_ms(20);
                PORTD=0x00;
                delay_ms(100);
            }
        }
        delay_ms(5000);
    }
}

```

```

if(PINB.2==1 && PINB.4==1)
{
    while(PINB.2!=0 || PINB.4!=0)
    {
        PORTD=0x40;
        delay_ms(100);
        PORTD=0x00;
        delay_ms(100);
    }
}
delay_ms(5000);

PORTD=0x00;
delay_ms(5000);

if(PINC.0==0 && PINC.1==1)
{
    while(PINC.0!=1 || PINC.1!=1)
    {
        PORTD=0x01;
        delay_ms(150);
        PORTD=0x00;
        delay_ms(100);
    }
}
delay_ms(5000);

if(PINC.0==1 && PINC.1==1)
{
    while(PINC.0!=0 || PINC.1!=0)
    {
        PORTD=0x01;
        delay_ms(100);
        PORTD=0x00;
        delay_ms(100);
    }
}
delay_ms(5000);

if(PINC.0==0 && PINC.1==0)
{
    while(PINC.0!=1 || PINC.1!=1 || PINB.1!=0)
    {
        PORTD=0x02;
        delay_ms(100);
        PORTD=0x00;
        delay_ms(100);
    }
}

```

```

        delay_ms(5000);

        if(PINC.0==1 && PINC.1==1)
        {
            while(PINC.0!=0 || PINC.1!=1 || PINB.1!=1)
            {
                PORTD=0x02;
                delay_ms(100);
                PORTD=0x00;
                delay_ms(100);
            }
        }
        delay_ms(5000);
        if(PINB.2==0 && PINB.4==0)
        {
            while(PINB.2!=1 || PINB.4!=1)
            {
                PORTD=0x20;
                delay_ms(20);
                PORTD=0x00;
                delay_ms(100);
            }
        }
        delay_ms(5000);
        if(PINB.2==1 && PINB.4==1)
        {
            while(PINB.2!=0 || PINB.4!=1)
            {
                PORTD=0x20;
                delay_ms(20);
                PORTD=0x00;
                delay_ms(100);
            }
        }
        delay_ms(5000);
        y=1;
    }

while(PINC.2==0)
{
    If(PINC.4==0)
    {
        PORTD=0x40;
        Delay_ms(30);
    }
    If(PINC.5==0)
    {
        PORTD=0x20;
        Delay_ms(30);
    }
}

```

```

If(PINC.3==0 && PINC.4==0)
{
    PORTD=0x01;
    Delay_ms(30);
}
If(PINC.3==0 && PINC.5==0)
{
    PORTD=0x02;
    Delay_ms(30);
}
}
while(PINC.6==0)
{
    If(PINC.5==0)
    {
        PORTD=0x20;
        Delay_ms(30);
    }
    If(PINC.4==0)
    {
        PORTD=0x02;
        Delay_ms(30);
    }
}

b=read_adc(0x00);
if(b<val[0]) //rest
{
    if(v==0)
    {
        if(PINB.2==0 && PINB.4==1)
        {
            PORTD=0x00;
            m=0;
            n=0;
            t=0;
            v=1;
        }
        if(PINB.2==1 && PINB.4==1)
        {
            while(PINB.2!=0 || PINB.4!=1)
            {
                PORTD=0x20;
                delay_ms(20);
                PORTD=0x00;
                delay_ms(50);
            }
            PORTD=0x00;
            m=0;
            n=0;
        }
    }
}

```

```

        t=0;
        v=1;
    }
    if(PINB.2==0 && PINB.4==0)
    {
        while(PINB.2!=0 || PINB.4!=1)
        {
            PORTD=0x20;
            delay_ms(20);
            PORTD=0x00;
            delay_ms(100);
        }
        PORTD=0x00;
        //delay_ms(100);
        m=0;
        n=0;
        t=0;
        v=1;
    }
}
else if(b>=val[0] && b<val[1])
{
    PORTD=0x00;
    delay_ms(1000);
}

else if(b>=val[1] && b<val[2]) //1st
{
    if(m==0)
    {
        if(PINB.2==0 && PINB.4==1)
        {
            while(PINB.2!=1 || PINB.4!=1)
            {
                PORTD=0x20;
                delay_ms(50);
                PORTD=0x00;
                delay_ms(100);
            }
            //PORTD=0x42;
            //delay_ms(600);
            PORTD=0x00;
            m=1;
            n=0;
            t=0;
            v=0;
        }
        if(PINB.2==1 && PINB.4==1)
        {

```

```

        PORTD=0x00;
        m=1;
        n=0;
        t=0;
        v=0;
    }
    if(PINB.2==0 && PINB.4==0)
    {
        while(PINB.2!=1 || PINB.4!=1)
        {
            PORTD=0x20;
            delay_ms(20);
            PORTD=0x00;
            delay_ms(100);
        }
        //PORTD=0x21;
        //delay_ms(200);
        PORTD=0x00;
        m=1;
        n=0;
        t=0;
        v=0;
    }
}
else if(b>=val[2] && b<val[3])
{
    PORTD=0x00;
    delay_ms(1000);
}
else if(b>=val[3] && b<val[4]) //2nd
{
    if(n==0)
    {
        if(PINB.2==0 && PINB.4==1)
        {
            while(PINB.2!=0 || PINB.4!=0)
            {
                PORTD=0x40;
                delay_ms(100);
                PORTD=0x00;
                delay_ms(100);
            }
            PORTD=0x00;
            m=0;
            n=1;
            t=0;
            v=0;
        }
        if(PINB.2==1 && PINB.4==1)

```

```

        {
            while(PINB.2!=0 || PINB.4!=0)
            {
                PORTD=0x40;
                delay_ms(100);
                PORTD=0x00;
                delay_ms(100);
            }
            //PORTD=0x42;
            //delay_ms(1200);
            PORTD=0x00;
            m=0;
            n=1;
            t=0;
            v=0;
        }
        if(PINB.2==0 && PINB.4==0)
        {
            PORTD=0x00;
            //delay_ms(500);
            m=0;
            n=1;
            t=0;
            v=0;
        }
    }
}
else if(b>=val[4] && b<val[5])
{
    PORTD=0x00;
    delay_ms(1000);
}
else
{
    PORTD=0x00;
    delay_ms(500);
}

c=read_adc(0x01);
if(c<val[0]) //rest
{
    if(w==0)
    {
        if(PINC.0==0 && PINC.1==1)
        {
            PORTD=0x00;
            r=0;
            s=0;
            u=0;
        }
    }
}

```

```

        w=1;
    }
    if(PINC.0==1 && PINC.1==1)
    {
        while(PINC.0!=0 || PINC.1!=1 || PINB.1!=1)
        {
            PORTD=0x02;
            delay_ms(100);
            PORTD=0x00;
            delay_ms(100);
        }
        PORTD=0x00;
        r=0;
        s=0;
        u=0;
        w=1;
    }
    if(PINC.0==0 && PINC.1==0)
    {
        while(PINC.0!=0 || PINC.1!=1)
        {
            PORTD=0x02;
            delay_ms(100);
            PORTD=0x00;
            delay_ms(100);
        }
        PORTD=0x00;
        //delay_ms(100);
        r=0;
        s=0;
        u=0;
        w=1;
    }
}

else if(c>=val[0] && c<val[1])
{
    PORTD=0x00;
    delay_ms(1000);
}
else if(c>=val[1] && c<val[2]) //1st
{
    if(r==0)
    {
        if(PINC.0==0 && PINC.1==1)
        {
            while(PINC.0!=1 || PINC.1!=1)
            {
                PORTD=0x01;
            }
        }
    }
}

```

```

        delay_ms(150);
        PORTD=0x00;
        delay_ms(100);
    }
    //PORTD=0x42;
    //delay_ms(600);
    PORTD=0x00;
    r=1;
    s=0;
    u=0;
    w=0;
}
if(PINC.0==1 && PINC.1==1)
{
    PORTD=0x00;
    r=1;
    s=0;
    u=0;
    w=0;
}
if(PINC.0==0 && PINC.1==0)
{
    while(PINC.0!=1 || PINC.1!=1 || PINB.1!=0)
    {
        PORTD=0x02;
        delay_ms(100);
        PORTD=0x00;
        delay_ms(100);
    }
    //PORTD=0x21;
    //delay_ms(200);
    PORTD=0x00;
    r=1;
    s=0;
    u=0;
    w=0;
}
}
}
else if(c>=val[2] && c<val[3])
{
    PORTD=0x00;
    delay_ms(1000);
}
else if(c>=val[3] && c<val[4]) //2nd
{
    if(s==0)
    {
        if(PINC.0==0 && PINC.1==1)
        {

```

```

        while(PINC.0!=0 || PINC.1!=0)
        {
            PORTD=0x01;
            delay_ms(100);
            PORTD=0x00;
            delay_ms(100);
        }
        PORTD=0x00;
        r=0;
        s=1;
        u=0;
        w=0;
    }
    if(PINC.0==1 && PINC.1==1)
    {
        while(PINC.0!=0 || PINC.1!=0)
        {
            PORTD=0x01;
            delay_ms(100);
            PORTD=0x00;
            delay_ms(100);
        }
        //PORTD=0x42;
        //delay_ms(1200);
        PORTD=0x00;
        r=0;
        s=1;
        u=0;
        w=0;
    }
    if(PINC.0==0 && PINC.1==0)
    {
        PORTD=0x00;
        //delay_ms(500);
        r=0;
        s=1;
        u=0;
        w=0;
    }
}
}
else if(c>=val[4] && c<val[5])
{
    PORTD=0x00;
    delay_ms(1000);
}
}

```

```
else
{
    PORTD=0x00;
    delay_ms(500);
}
}
```