

MOVABLE ROBOTIC ARM CONTROL USING ANDROID APPLICATION

Submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Mechanical Engineering

By

P. Subash (Register no: 37150185)

S. Aravind (Register no: 37150015)



**DEPARTMENT OF MECHANICAL ENGINEERING SCHOOL OF
MECHANICAL ENGINEERING SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with grade “A” by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119**

MARCH 2021



SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119

www.sathyabama.ac.in



DEPARTMENT OF MECHANICAL ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of P.SUBASH (Reg. No. 37150185) and S.ARAVIND (Reg. No. 37150015) who carried out the project entitled **"MOVABLE ROBOTIC ARM CONTROL USING ANDROID APPLICATION"** under my supervision from August 2020 to March 2021.

Mr.R.RAJASEKAR, M.E.,

Internal Guide

Dr. G. ARUNKUMAR, M.E., Ph.D.,

Head of the Department

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We P.SUBASH (37150185) and S.ARAVIND (37150015) hereby declare that the Project Report entitled "MOVABLE ROBOTIC ARM CONTROL USING ANDROID APPLICATION" done by us under the guidance of Mr.RAJA SEKAR., M.E., is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Mechanical Engineering.

1.

2.

DATE: 30.03.2021

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my sincere thanks to **Dr. S. Prakash, M.E., Ph.D.**, Dean, School of Mechanical Engineering and **Dr. G. Arun Kumar M.E., Ph.D.**, Head of the Department, Dept. of Mechanical Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mr.RAJA SEKAR., M.E.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I also express my thanks to all Teaching and Non-teaching staff members of the Department of mechanical engineering who were helpful in many ways
The completion of the project .

DESIGN AND FABRICATION OF MOVING ROBOTIC ARM CONTROL USING ANDROID APPLICATION

ABSTRACT:

The main focus of this project was to design and develop the Mechanism for robotic arm for lifting. The robotic arm was designed with four degrees of freedom and programmed to accomplish accurately simple light material lifting task to assist in the production line in any industry. 3D printing method is used in this project to fabricate the components of the robotic arm. Therefore, it provided more precise dimensions and huge time and cost-saving in fabrication. The robotic arm is equipped with 4 servo motors to link the parts and bring arm movement. Arduino, an open-source computer hardware and software is applied to control the robotic arm by driving servo motors to be capable to modify the position. Wireless control was done by using a smart phone with android operating system through a Bluetooth module. The robotic arm was under testing and validating its performance and the results indicates that it can perform the lifting task properly.

Chapter No.	TITLE	Page No.
	ABSTRACT	V
	LIST OF FIGURES	IX
	LIST OF TABLES	XII
1	INTRODUCTION	1
	1.0 Project prototype	4
	1.1 Block diagram	5
	1.2 Hardware and software requirements	6
	1.3 Hardware required	6
	1.4 Software required	6
2	DESING WORK	6
	2.0 Robotic Arm	6
	2.1 Types of joint in Robotic arm	6
	2.1.1 Free body diagram of the robotic arm	6
	2.1.2 Number of degree for specific DOF	6
	2.1.3 Work region of the Robotic arm	8
3	AUTOCAD ASSEMBLY OF PARTS	21
	3.0 Aim and scop	21
	3.1 Power supply	21
	3.2 3D Printing	22
4	EXPERIMENTAL OR MATERIAL AND METHODS	24
	ALGORITHMS	

Chapter No.	TITLE	Page No.
	4.0 Experimental of prototype model	24
	4.1 Working methodology	25
	4.2 Adriano controlling system	26
	4.3 Bluetooth module	27
	4.4 Software Implementation	27
	4.5 Motor Driver L298	28
	4.6 Ball Barings	28
	4.7 Deep-Groove Ball Bearings	29
	4.8 Benefits	29
	4.9 Design specifications	30
	4.10 D.C Motors	32
	4.11 Battery	33
	4.12 Fabrication of ladder	35
	4.13 Fabrication of main frame	38
	4.14 Assembly of M.S angle guides	39
	4.15 Assembly of Rack & Pinions	39
	4.16 Motor support plates	41
	4.17 Threaded rods & roller assembly	44
	4.18 Upper arm	48
	4.19 Lower arm	49
	4.20 Motor shaft coupling	51
	4.21 Motor power calculations for pinion	55
	4.22 Top holding (upper) arm calculation	55
	4.23 Side holding (Lower) arm calculation	56
5	RESULT AND DISCUSSION PERFORMANCE ANALYSIS	57

LIST OF FIGURES

FIGURE No	INDEX	PAGE No.
1 & 2	Risk of falling & injuries	3
3	Ladder climber	12
4	Ladder climbing carrying device	13
4.1	View of ladder climbing carrying robot	25
4.2	Ladder climbing carrying robot on ladder	25
4.3	Arduino uno	27
4.4	Bluetooth connection	27
4.5	L293d	28
4.6	Box type high torque geared dc motors	33
4.7	Battery	35
4.8	Preparing the prop for joining the rungs	36
4.9	Marking the rung material for cutting rungs	36
4.10	Cutting of the rungs	36
4.11	Grinding the faces of the rungs for smoothness	37
4.12	Joining the rungs & prop	37
4.13	Finished ladder	37
4.15	Cutting the main frame from raw material	38
4.16	Marking the dimensions for setting guide ways	38
4.17	View of M.S. angle guides on the main frame	39
4.18	Rack & pinion assembly	40

4.19	View of rack & pinion with in the guideways on main frame	41
4.20	Marking of dimensions on motor support plate	41
4.21	View of motor support plate after marking	42
4.22	Drilling Centre hole using drill bit @12.5 mm	42
4.33	Drilling 4 holes using drill bit @ 5 mm	42

LIST OF FIGURES

FIGURE No	INDEX	PAGE No
4.24	Robotics drilling 4 holes using drill bit @ 6 mm if oversize is required	43
4.25	View of motor support plate after drilling required holes	43
4.26	Reducing screw length to fit exactly into motor threading	43
4.27	View of motor support plate after fitting to a d c motor	44
4.28	View of welding of motor support plate to the main frame	44
4.29	Threaded rods	45
4.31	View of threaded rods with nylon rollers	45
4.32	Nylon rollers	47
4.34	View of nylon rollers on robot	47
4.35	Cutting of arm @ 125 mm	48
4.36	Bending of M.S. flat	48
4.37	View of upper arm on robot	49
4.38	Cutting of arm @ 115 mm	50
4.39	Bending of m.s. flat	50
4.40	View of lower arms on the ladder	50
4.41		

4.42, 4.43	Pinion gear & cylindrical shaft arrangement	51
4.44	Cylindrical shaft surface grinding	51
4.45	Drilling @ 10 mm to provide clearance b/w shaft and pinion	52
4.46	View of cylindrical shaft and pinion gear	52
4.47	Joining of cylindrical shaft & pinion gear using arc welding	52
4.48	Surface grinding after joining process	53
4.49	Fig 4.49 Cylindrical shaft with threading welded to arm	53

LIST OF TABLES

TABLE No.	INDEX	PAGE No.
4.9.0	Dimensions	30
4.9.1	Abutment dimensions	31
4.9.2	Calculation data	31

CHAPTER 1

INTRODUCTION:

Automated is characterized as the investigation, plan and utilization of mechanical frameworks for assembling. With the ascent in assembling modern exercises, a mechanical arm is developed to assist different businesses with playing out an errand or work as opposed to utilizing labor. Robots are for the most part used to perform risky, dangerous, profoundly redundant, and disagreeable assignments. Robot can perform material taking care of, gathering, curve welding, obstruction welding, machine apparatus stack and dump capacity, painting and splashing, etc [3, 4]. It is extremely valuable since it has high exactness, insight and unending energy levels in managing job contrasted with individual. For a model, a mechanical arm is generally utilized in the amassing or loading line by lifting the little items with redundant movement that human couldn't tolerate doing in a significant stretch of time. The light material lifting errand should be possible by the mechanical arm effectively and efficient on the grounds that it isn't confined by weariness or wellbeing chances what man may insight. There are fundamentally two distinct kinds of robots which are administration robot and a modern automated. Administration robot is worked semi or completely self-sufficiently to perform administration helpful to the prosperity of people and hardware with the exception of assembling operation [6-8]. Then again, modern robot is formally characterized by ISO as a naturally controlled and multipurpose controller programmable in at least three hub.

A mechanical robot is a re-programmable multifunctional controller configuration to move material, parts, instruments, or concentrated gadgets through factor modified movement for execution of an assortment of undertakings. This is the

Definition from the Robot Institute of America to reflect primary highlights of current robot frameworks. A modern robot framework can incorporate any gadgets or sensors along with the mechanical robots to play out its undertakings just as sequencing or checking correspondence interfaces [3, 9].

In 1970, Stanford University built up a PC controlled robot arm with electric drive engines, known as Stanford Arm. In 1973, the main modern robot outfitted with a minicomputer-based control framework was created in Cincinnati Milacron Corporation. In 1977, an European organization, ASEA, additionally created electrical fueled modern robots furnished with the microcomputer-based control frameworks. Around the same time, a robot vision framework was created by Stanford Research Institute (SRI) at Stanford University. In 1978, the Puma (programmable widespread machine for gathering) robot was created dependent on the Stanford arm in America. This robot utilizes servomotors furnished with a high level control framework utilizing a couple of microchips and advance programming. In 1979, Sankyo and IBM built up the popular SCARA (particular agreeable explained robot arm) at Yamanashi University in Japan.

In 2007 the world market developed by 3% with around 114,000 new introduced mechanical robots. Toward the finish of 2007, there were around 1,000,000 mechanical robots being used, compared with an expected 50,000 help robots for modern use.

Aside from mechanical utilization, the automated arms likewise intended to mimic human development with high exactness. A plan of mechanical arm which can move, weight lifting, Chinese calligraphy composing and shading order is finished by a gathering of Korea understudies. There is additionally an eight level of opportunity automated arm planned by a gathering of USA engineers. It can get a handle on numerous articles with a great deal of shapes structure a pen to a ball and invigorating additionally the hand of person. Besides, mechanical arm with various level of opportunity have been utilized to perform assortment of assignments in space application. The framework is known as Space Shuttle Remote Manipulator System (SSRMS) or Canadarm. Automated arm's undertakings in the space included investigations of room transport utilizing an extraordinarily conveyed blast with cameras and sensors joined toward the end effector and satellite sending and recovery moves from the payload narrows of the space transport.

This current work is completed at University Malaysia Pahang, Malaysia, the principle center is to configuration, create of a system of mechanical arm for lifting reason with short expense, precise and prevalent control. Automated arm with four levels of opportunity had the option to perform basic errands particularly light material taking care of for modern application

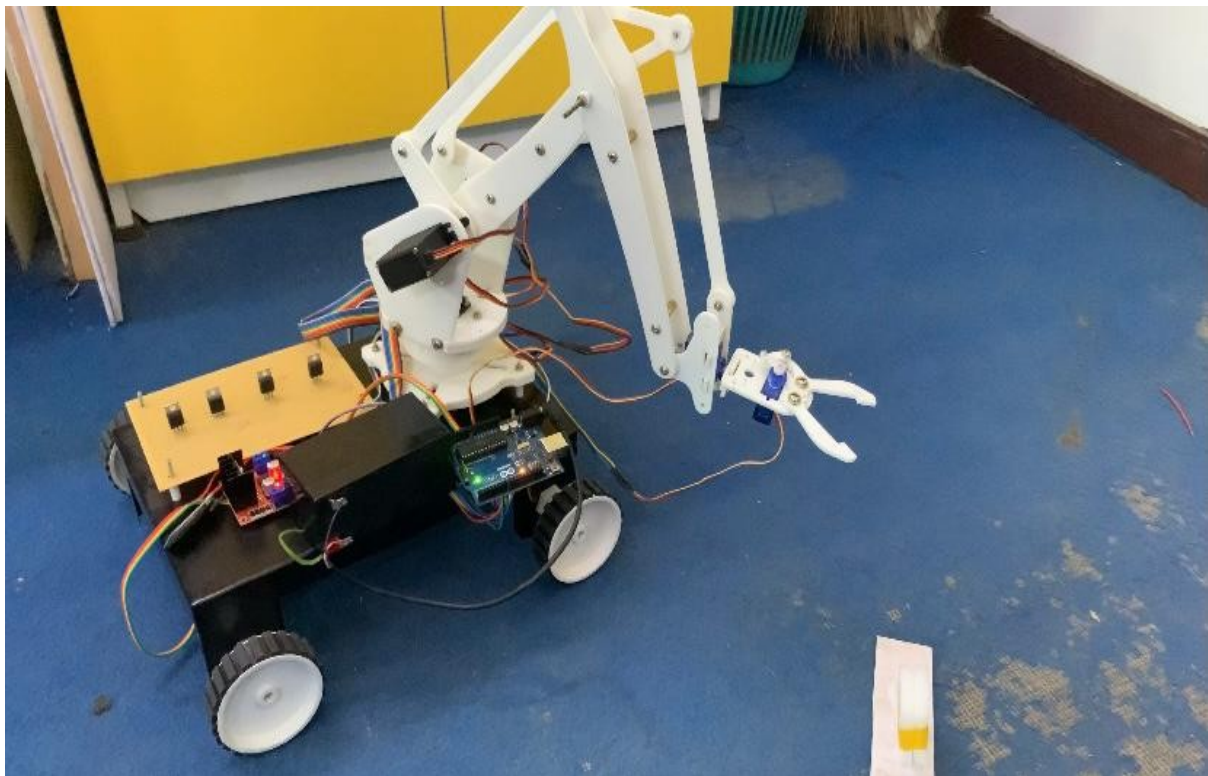


Fig.1 PROJECT PROTOTYPE

BLOCK DIAGRAM:

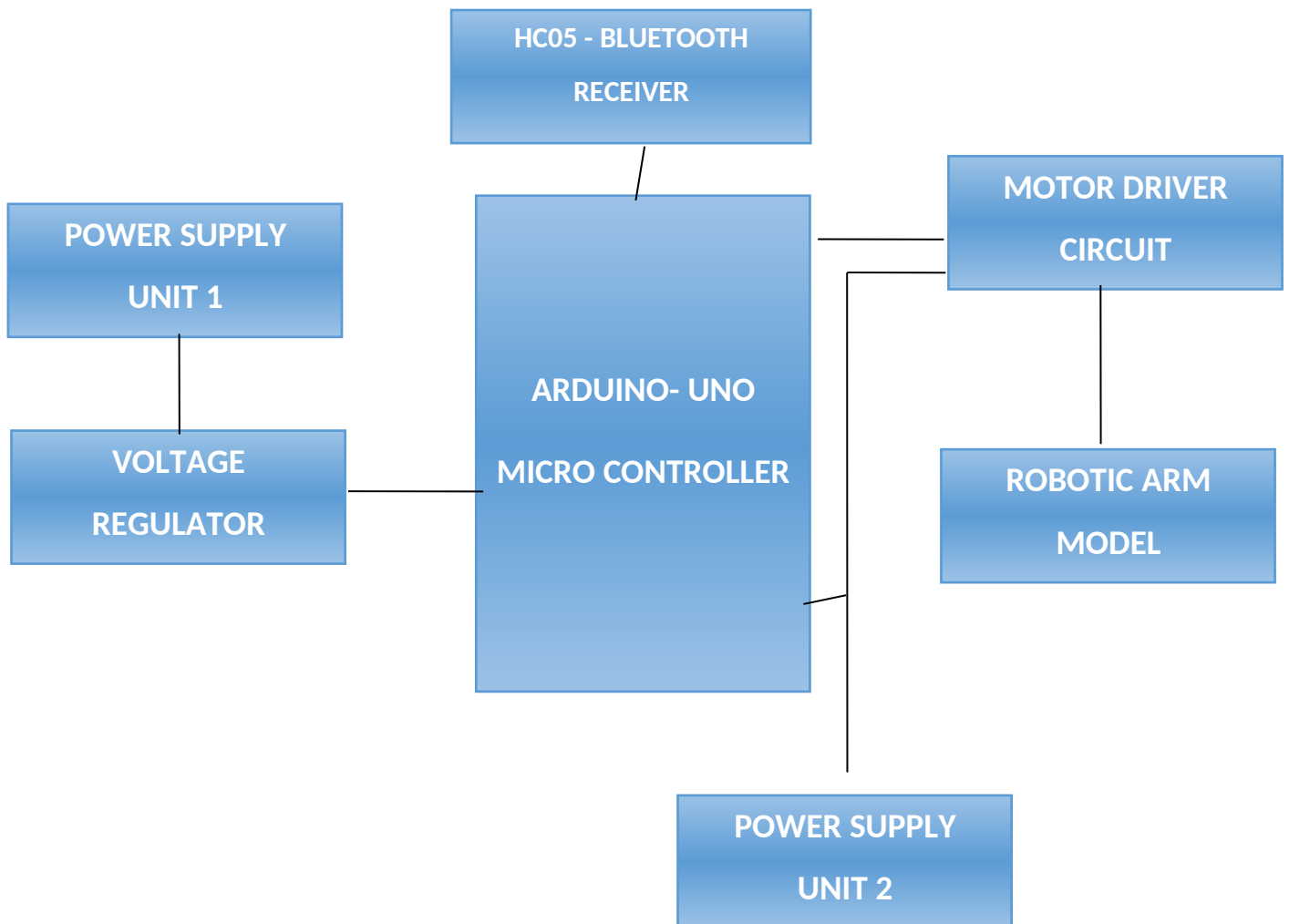


Fig.1.1 BLOCK DIAGRAM

1.1 HARDWARE AND SOFTWARE REQUIREMENTS

1.2 HARDWARE REQUIRED

- ARDUINO UNO MICROCONTROLLER
- HC05 BLUETOOTH RECEIVER
- POWER SUPPLY UNIT
- ROBOTIC ARM MODEL
- CONNECTING WIRES
- SOLDERING KIT

1.2.1 SOFTWARE REQUIRED

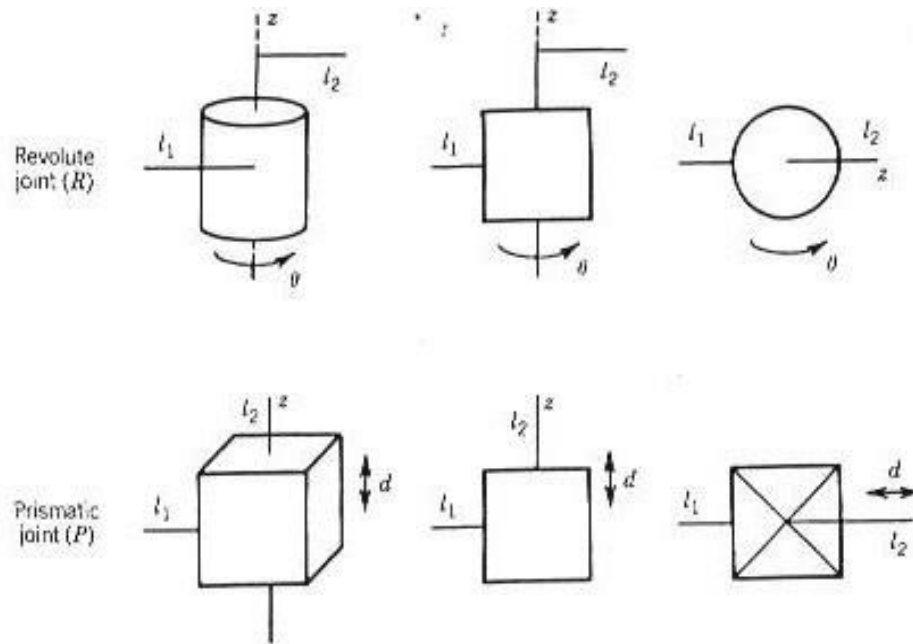
- EMBEDDED C
- ARDUINO – IDE

1.3 DESIGN WORK

1.3.1 ROBOTIC ARM

THE MECHANICAL PLAN OF THE ROBOT ARM DEPENDS ON A ROBOT CONTROLLER WITH COMPARATIVE CAPACITIES TO A HUMAN ARM. AUTOMATED ARM FRAMEWORK REGULARLY COMPRISES OF CONNECTIONS, JOINTS, ACTUATORS, SENSORS AND REGULATORS. THE CONNECTIONS ARE ASSOCIATED BY JOINTS TO FRAME AN OPEN KINEMATIC CHAIN. ONE END OF THE CHAIN IS CONNECTED TO THE ROBOT BASE, AND ANOTHER END IS OUTFITTED WITH A DEVICE (HAND, GRIPPER, OR END-EFFECTORS) WHICH IS COMPARABLE TO HUMAN HAND TO PERFORM GATHERING AND DIFFERENT ERRANDS AND TO INTERFACE WITH THE CLIMATE. THERE ARE TWO KINDS OF JOINT WHICH ARE KALEIDOSCOPIC AND ROTATING JOINTS AND IT ASSOCIATE ADJOINING JOINT.

1.4 TWO TYPE OF JOINT IN ROBBOT



Symbolic representation of robot joints

The connections of the controller are associated by joints permitting rotational movement and the connections of the controller is considered to shape a kinematic chain the Free Body Diagram for mechanical plan of the automated arm. A mechanical arm with just four levels of opportunity is planned on the grounds that it is satisfactory for the vast majority of the important development. Simultaneously, it is serious by its intricacy and cost-saving as number of actuators in the automated arm increments with levels of opportunity. In a mechanical framework, the quantity of levels of opportunity is controlled by the quantity of autonomous joint variable

FREE BODY DIAGRAM OF THE ROBOTIC ARM.

1.5 NUMBER OF DEGREES FOR SPECIFIC DOF

THE AUTOMATED ARM HAS FOUR SERVO ENGINES, EACH ENGINE FOR ONE DOF AND THE ENGINES WORK WITH 6 VOLTS. THERE ARE FOUR PIVOTING DOF IN THE ARM AND THEIR SCOPES OF REVOLUTION ARE DEMONSTRATED IN THE TABLE 1.

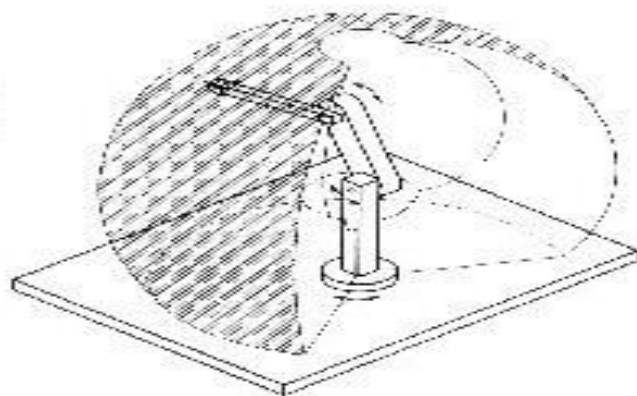
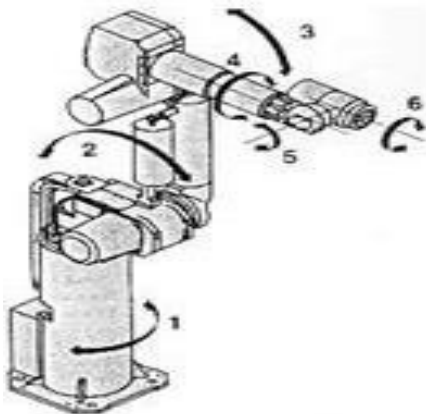
Table:1.5.1

Degrees of Freedom(DOF)	Degrees
1	180
2	90
3	90
4	90

The zone that the gripper alleged as end-effector can reach is known as robot workspace. It relies upon the DOF and interpretation restriction, the arm interface lengths, the point at which something should be gotten up robot arrangement. shows the ordinary work district of the mechanical arm with four level of opportunity (4 DOF)[3].

1.6 WORK REGION OF THE ROBOTIC ARM

By using Autocad software, the finalized design is as follow.

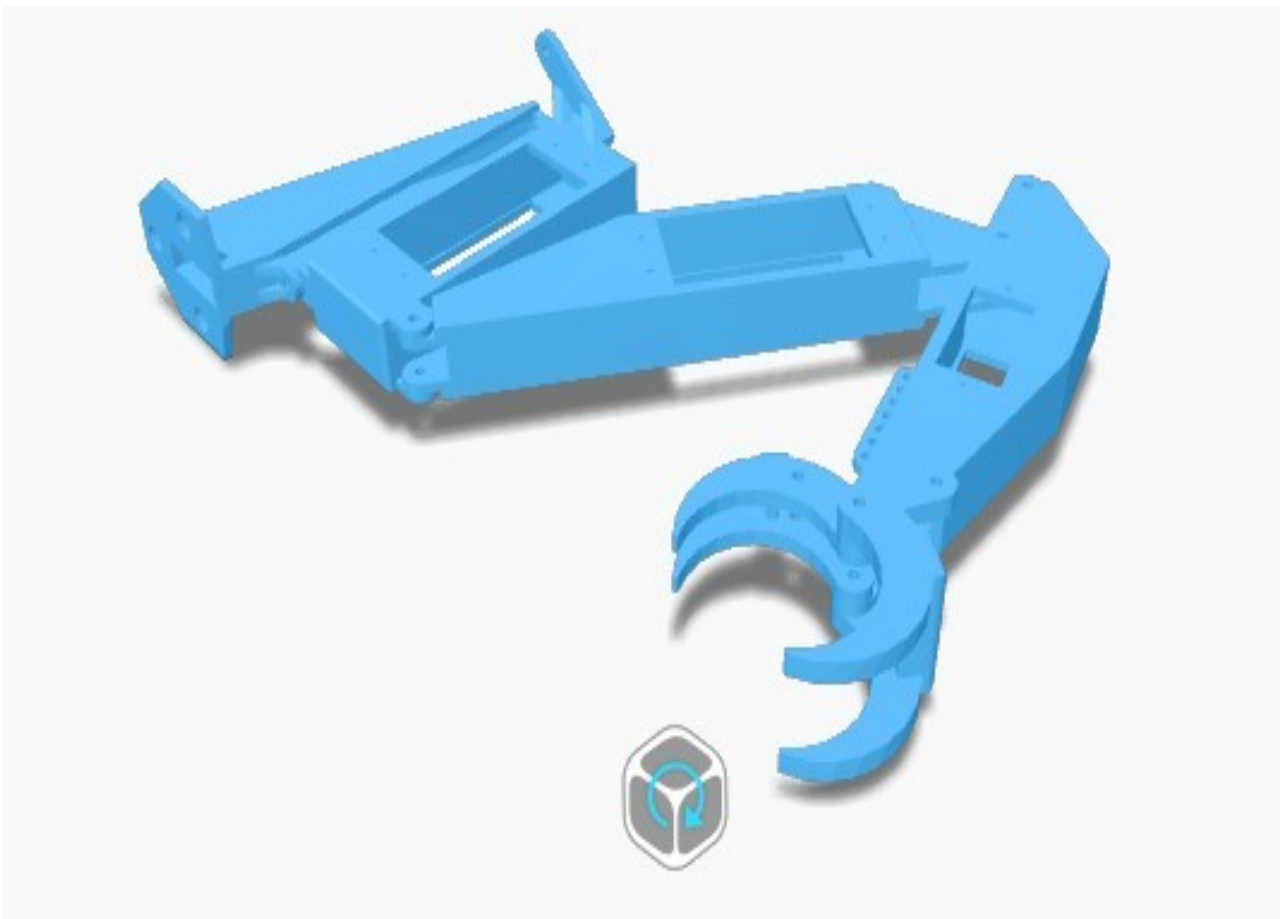


Articulated Robot Arm

FIG 1.6.1

1.7 AUTOCAD ASSEMBLY OF PARTS.

This is a help for the automated arm and will be fixed at base. The plan



is somewhat more extensive at the base and skewed at 20 o degree to the top. The rib configuration is added to the part to build firmness because of burden applied to the part plan. It changed the math of the part and it is down to earth and practical to build the primary strength of a section. In addition, this part has a tetragonal shape to disperse the heap equally as it is needed to help huge burden applied by the item to be lifted,

servo engine and connection. This appendage is intended for rotational reason and furthermore the engine for up-down movement of the mechanical arm. The overhanging part is the place where the wires will be associated and afterward incited by the servo engine at the base. Engine lodging is pleasantly found a way into the empty piece of the appendage. This is moreover

the linkage to the following part for up-down movement. The accompanying appendage appeared underneath is planned with most extreme broadening shape just as the zone for engine to be fit in. The lower part is associated with the past appendage utilizing an interfacing wire. This part comprises of the fundamental piece of the automated arm as it houses the engine for gripper. The gripper will be associated with this appendage and constrained by the engine through the interfacing wire. The gripper comprises of lower and upper paw. It is planned that the lower paw is fixed at a specific point on the past appendage and the moving upper hook to the engine. The state of the hook is croissant and it can expand the grasping presentation.

1.8 ACTUATORS

Actuators are gadgets that cause revolving joints to turn about their movement tomahawks, or drive kaleidoscopic joints to slide along their movement tomahawks. By and large, there are three sorts of central activating frameworks utilized in robot frameworks: water driven impelling frameworks, pneumatic inciting frameworks, and electrical impelling frameworks. In our plan, we are utilizing electrical impelling frameworks. DC engines and stepper engines are utilized to impel the development of the mechanical arm. This is on the grounds that this framework can be controlled effectively and the servo engine can give a quick reaction, high precision, included encoders which consequently give input to the engines and flexibility of position accordingly[3, 7].]. This sort of actuator is reasonable for little robots. Nonetheless, the hindrances of the servo engine is that turn range is under 180o range, in this way the district came to by the arm and potential positions are significantly decrease[7, 24]. As we have 3 joints in the mechanical arm, we need to set up 3 regulators to control every single joint actuator. Choosing appropriate engine for the application is significant in the plan

stage which choice is made dependent on force and the speed of the engine. Power figuring of joint guarantees that the engine picked can uphold the heaviness of the automated arm and furthermore the heap which should be conveyed. Servo engines is suggested when high force and exact speed wanted by the client. For just situating without requiring a high force, stepper engines are utilized in the application.

Force is the propensity of power to turn an item about a hub. Numerically, force is characterized as the cross result of the switch arm distance and power, which will in general create turn for example

$$T = F \cdot L \text{ Nm [25]}$$

Where, F=force acting on the motor

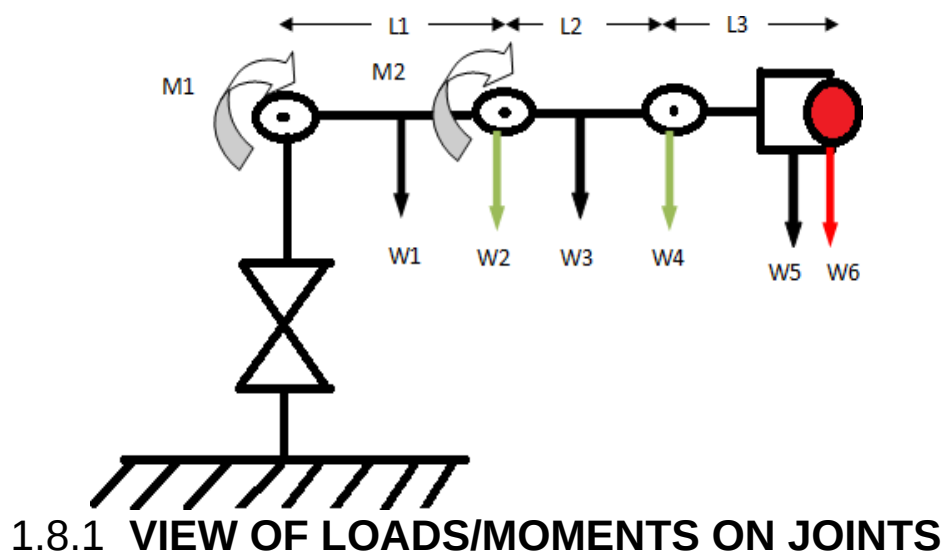
L=length of the shaft

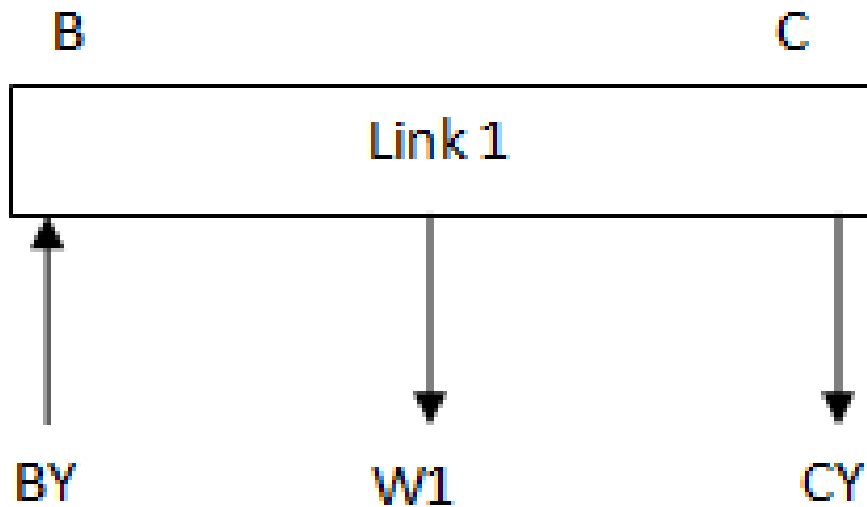
Force, F is given by,

$$F = m \cdot g \text{ N [5]}$$

Where, m=mass to be lifted by the motor

g =gravitational constant=9.8 m/s





1.8.2 FORCE DIAGRAM OF LINK 1.

In light existing apart from everything else determined, an appropriate servo engine was chosen, is, which has a force of 152.8 oz-in to agree with the force necessity of the automated arm. They can turn 60 degrees in 15millionsecond and weight of 55.0 grams each. For the gripper, a miniature servo with 9 grams and 0.12sec/60 degrees is utilized on the grounds that very little force expected to work the grasping interaction because of the light weight of gripper. In the mechanical arm, 4 servo engines are utilized.

1.9 Engines filled various needs at each piece of the mechanical arm:

- Motor 1 for rotational movement.
- Motor 2 for elbow development.
- Motor 3 for wrist development.
- Motor 4 for grasping movement.

CHAPTER -2

2.Writing SURVEY:

2.1 Point: Gesture Controlled Mobile Robotic Arm Using Accelerometer

Creator: Girish Solanki

DATE: 2017

Depiction:

In this paper we have introduced a model to control automated arm through human signals utilizing accelerometer. A three hub accelerometer is mounted on human hand to play out the activity of mechanical arm as per the activity of human hand. Accelerometer is associated with the Atmega 16 Microcontroller which is customized to take simple readings from accelerometer and communicate them utilizing RF transmitter to the accepting unit at mechanical arm. Developments of the automated arm are accomplished through Servo-Motor are a kind of electromechanical actuators that don't pivot ceaselessly like DC/AC or stepper engines; rather, they are utilized to position and hold some article. They are utilized where persistent turn isn't needed so they are not used to drive wheels (except if a servo is altered). The arm is likewise outfitted with a gripper to encourage the pick and drop facility. The entire game plan is put on a portable stage with wheels to encourage development starting with one spot then onto the next which can be controlled utilizing a remote controller. The principle point is to control the automated arm utilizing human signals remotely with smooth development over a reach.

Theme: ROBOTIC VEHICLE MOVEMENT AND ARM CONTROL THROUGH HAND GESTURES USING ARDUINO

2.2 Creator: N Meenatchi Sundaram

DATE: 2018

Portrayal:

Mechanical vehicle development and Arm control through hand signals utilizing arduino is an assistive robot framework for easy to use support, emergency clinic tasks, and dangerous climate and to build the self-rule of actually impeded individuals. It is a sort of practical supplanting hand coordinated with the moving vehicle that guides the amputees frequently experiences actual challenges. The excellent point of this framework is to make it attainable to communicate with the robot through the acknowledgment of hand motions for the genuinely tested individuals. This is an electromechanical framework wherein the mechanical arm can be constrained by a basic inertial route sensor called accelerometer. Also the delicate getting gripper associated with the automated arm can be constrained by flex sensor. The signal orders estimated by the accelerometer and flex sensor are handled by the arduino regulator. The arduino regulator is utilized in both the transmitter and collector circuits, it reacts dependent on the information given through the hand motions. The signs get sent and gotten by cutting edge remote nRF correspondence framework. The mechanical vehicle can move on the whole conceivable four bearings (forward, in reverse, both ways). The various activities performed by the automated arm are picking and setting the articles starting with one area then onto the next area. For the controlling of the engine, engine driver IC and arduino regulator is utilized. This framework gives more regular and advantageous approaches to communicate their aims and associations with the climate.

2.3 Point: Real time hand development controlled automated arm for hazard counteraction

Creator: Shamman Noor

DATE: 2017

Depiction:

In this paper, we propose an "Counterfeit Human Hand Model" which can accurately follow the genuine developments of human hand through picture preparing accordingly keeping away from sensor clamor and harm and giving more customizable reaction. Webcams are utilized to catch the hand pictures followed by Morphological picture preparing to separate data from human hand development. The five levels of opportunity carried out in the proposed technique are turn of base, arm like movements, elbow like movements, grasp movements, and pivot of hold. The robot hand is controlled effectively utilizing human hand development, where the situation can get, control and move objects from one spot to other with accuracy. The client can handle the hand totally dependent on his/her aims to achieve the necessary undertaking. Our proposed constant technique for carrying out human hand development likewise guarantees immediate control and execution. Inaccessible control of the mechanical hand guarantees total security to human, yet additionally permits adaptability in controlling removed items. By and large, our proposed framework can ensure wellbeing, adaptability and precision in performing touchy undertakings in everyday life subsequently offering appropriateness for hazard anticipation.

2.4 Subject: A Gestures Recognition Based Approach for Human-Robot-Interaction

Creator: A. Freddi

DATE: 2018

Depiction:

This work proposes a mechanical controller aide for incapacitated clients as well as for older individuals with restricted engine abilities. In detail, the association among the robot and the client depends on the client motions acknowledgment. The client picks an article among those accessible by moving his/her arm in a particular posture, which is perceived by utilizing an outside camera. At that point, pictures of the articles open to the robot are obtained by means of the robot camera, situated toward the finish of the robot arm, and are dissected by a Support Vector Machine classifier to perceive the client chose object. At last, the controller picks the article and places it on the client's hand, whose area in the Cartesian space is resolved through the outside camera and refreshed on the web.

CHAPTER -3

Aim and Scope:

Moving robotic arm is use to move the one object to a other place and using the bluetooth device , A mechanical robot is a re-programmable multifunctional controller configuration to move material, parts, instruments, or concentrated gadgets through factor modified movement for execution of an assortment of undertakings. This is the definition from the Robot Institute of America to reflect primary highlights of current robot frameworks. A modern robot framework can incorporates any gadgets or sensors along with the mechanical robots to play out its undertakings just as sequencing or checking correspondence interfaces [3, 9].

In 1970, Stanford University built up a PC controlled robot arm with electric drive engines, known as Stanford Arm. In 1973, the main modern robot outfitted with a minicomputer-based control framework was created in Cincinnati Milacron Corporation. In 1977, an European organization, ASEA, additionally created electrical fueled modern robots furnished with the microcomputer-based control frameworks. Around the same time, a robot vision framework was created by Stanford Research Institute (SRI) at Stanford University. In 1978, the Puma (programmable widespread machine for gathering) robot was created dependent on the Stanford arm in America. This robot utilizes servomotors furnished with a high level control framework utilizing a couple of microchips and advance programming. In 1979, Sankyo and IBM built up the popular SCARA (particular agreeable explained robot arm) at Yamanashi University in Japan.

3. POWER SUPPLY

For huge robots, an inside burning motors is utilized to produce the power or to control water driven or pneumatic actuator. In any case, DC power source usually used to supply a steady voltage to the little robots and circuit that should be worked inside certain force limit[5]. Batteries are generally protected, modest, little and simple to use[2]. Nonetheless, there is intricacy in wiring in this manner a force for the mechanical arm is provided by the PC through USB port. This is on the grounds that mechanical arm performs consistently under AC power source.

4. 3D-PRINTING

For this undertaking, the automated arm is created by utilizing 3D printing with the material of PLA. The upsides of 3D printing are time and cost saving and high accuracy on measurement contrasted with metal creation. Makerbot programming was utilized to import the printing information to the machine, Creator. After set up the printer, each part requires around 2 hours to finish the printing interaction. Setup of the setting are set to standard and default, the extruder and stage temperatures are 230 o C and 110o C individually.

5. MICROCONTROLLER

Microcontroller produces beats which are needed by the servo engines to create a movement to make the mechanical arm working. The control for the automated arm comprises essentially of three levels: a microcontroller, a driver, and a PC based UI Thus, the mechanical arm is incorporated by subsystem, microcontroller to empower it to play out a lifting task.

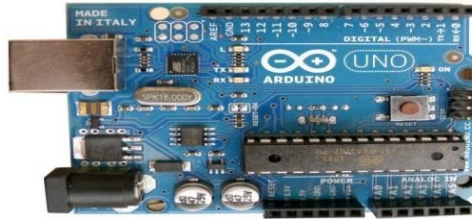
So, the microcontroller interfaces every one of these parts underneath. A short rundown of electrical segments included

5.1 Micro Servo Motor

- 1) Servo Motor
- 2) Voltage Regulator 5V
- 3) Capacitor
- 4) Connecting Wire

- 5) Breadboard
- 6) Arduino UNO

Fig;5.1.1



6.ARDUINO MICROCONTROLLER BOARD USED TO CONTROL THE ARM MOTION.

Arduino board consists of the following parts:

- USB connector
- Power connector
- Automatic power switch
- Digital pins
- Analog pins
- Power pins
- Reset switch

CHAPTER - 4

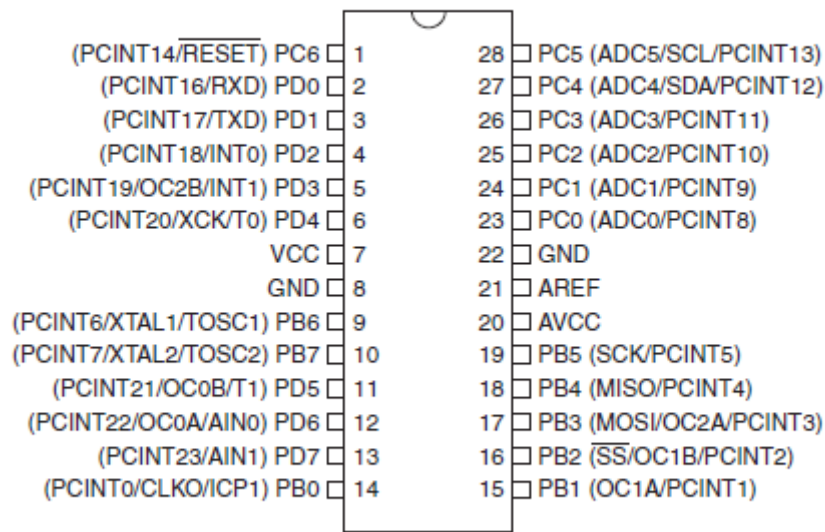
4.1 ATMEGA328 IC

DESCRIPTION

The ATmega328 is a single-chip microcontroller created by Atmel in the mega AVR family. The Atmel 8-bit AVR RISC-based microcontroller combines 32 kB ISP flash memory with read-while-write capabilities, 1 kB EEPROM, 2 kB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. The device achieves throughput approaching 1 MIPS per MHz.



4.1.1 ATmega328P IC



4.1.2 PIN DIAGRAM

High Performance, Low Power Atmel®AVR® 8-Bit Microcontroller Family

- Advanced RISC Architecture
- 131 Powerful Instructions
- Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers
- Fully Static Operation
- Up to 20 MIPS Throughput at 20MHz
- On-chip 2-cycle Multiplier

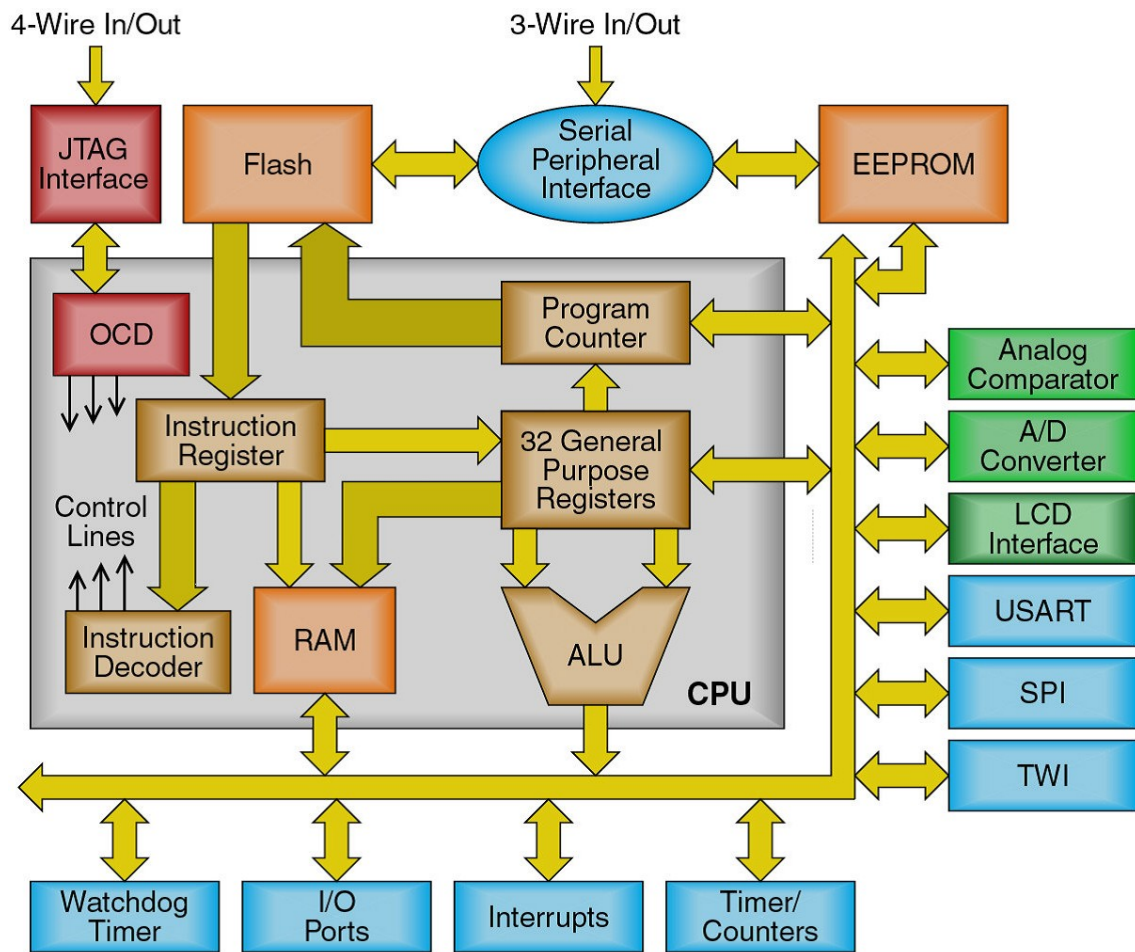
- High Endurance Non-unpredictable Memory Segments

- ☐ 4/8/16/32KBytes of In-System Self-Programmable Flash program memory
- ☐ 256/512/512/1KBytes EEPROM
- ☐ 512/1K/1K/2KBytes Internal SRAM
- ☐ Write/Erase Cycles: 10,000 Flash/100,000 EEPROM C(1)C/100 years at 25□– Data maintenance: 20 years at 85
- ☐ Optional Boot Code Section with Independent Lock Bits In-System Programming by On-chip Boot Program
- ☐ True Read-While-Write Operation
- ☐ Programming Lock for Software Security Atmel® QTouch® library support
- ☐ Capacitive touch catches, sliders and wheels – QTouch and QMatrix® securing
- ☐ Up to 64 sense channels

- Peripheral Features

- ☐ Two 8-cycle Timer/Counters with Separate Prescaler and Compare Mode
- ☐ One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
- ☐ Real Time Counter with Separate Oscillator

- ☐ Six PWM Channels
- ☐ 8-direct 10-bit ADC in TQFP and QFN/MLF bundle
- ☐ Temperature Measurement
- ☐ 6-direct 10-bit ADC in PDIP Package
- ☐ Temperature Measurement
- ☐ Programmable Serial USART—Master/Slave SPI Serial Interface
- ☐ Byte-arranged 2-wire Serial Interface (Philips I2C viable)
- ☐ Programmable Watchdog Timer with Separate On-chip Oscillator
- ☐ On-chip Analog Comparator
- ☐ Interrupt and Wake-up on Pin Change



4.1.3 ARCHITECTURE DIAGRAM

CHAPTER - 5

5.Special Microcontroller Features

- Power-on Reset and Programmable Brown-out Detection
- Internal Calibrated Oscillator
- External and Internal Interrupt Sources
- Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-pin PDIP, 32-lead TQFP, 28-cushion QFN/MLF and 32-cushion QFN/MLF
- Operating Voltage: 1.8 - 5.5V
- Temperature Range: - 40 C□C to 85□
- Speed Grade: 0 - 4MHz@1.8 - 5.5V, 0 - 10MHz@2.7 - 5.5.V, 0 - 20MHz @ 4.5 - 5.5V C□

- Power Consumption at 1MHz, 1.8V, 25
- ☐ Active Mode: 0.2Ma
- ☐ Power-down Mode: 0.1μA
- ☐ Power-save Mode: 0.75μA (Including 32kHz RTC)

5.1 PIN DISCRIPTION

VCC Digital stock voltage

GND Ground.

5.2Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is a 8-bit bi-directional I/O port with inward draw up resistors (chose for each piece). The Port B yield cushions have balanced drive qualities with both high sink and source capacity. As information sources, Port B sticks that are remotely pulled low will source current if the draw up resistors are initiated. The Port B pins are tri-expressed when a reset condition becomes dynamic, regardless of whether the clock isn't running. Contingent upon the clock determination combine settings, PB6 can be utilized as contribution to the transforming Oscillator speaker and contribution to the inside clock working circuit. Contingent upon the clock determination combine settings, PB7 can be utilized as yield from the reversing Oscillator intensifier. On the off chance that the Internal Calibrated RC Oscillator is utilized as chip clock source, PB7..6 is utilized as TOSC2..1 contribution for the Asynchronous Timer/Counter2 if the AS2 chomped in ASSR is set.

5.3 Port C (PC5:0)

Port C is a 7-piece bi-directional I/O port with inward draw up resistors (chosed for each piece). The PC5 0 yield cradles have balanced drive qualities with both high sink and source ability. As sources of info, Port C pins that are remotely pulled low will source current if the draw up resistors are initiated. The Port C pins are tri-expressed when a reset condition becomes dynamic, regardless of whether the clock isn't running.

5.4 PC6/RESET

On the off chance that the RSTDISBL Fuse is modified, PC6 is utilized as an I/O pin. Note that the electrical attributes of PC6 contrast from those of different pins of Port C. On the off chance that the RSTDISBL Fuse is unprogrammed, PC6 is utilized as a Reset input. A low level on this pin for more than the base heartbeat length will produce a Reset, regardless of whether the clock isn't running. The base heartbeat length is given in Table 28-3 on page 308. More limited heartbeats are not ensured to create a Reset.

5.5 Port D (PD7:0)

Port D is a 8-bit bi-directional I/O port with inside pull-up resistors (chosed for each piece). The Port D yield cradles have even drive qualities with both high sink and source capacity. As data sources, Port D pins that are remotely pulled low will source current if the draw up resistors are initiated. The Port D pins are tri-expressed when a reset condition becomes dynamic, regardless of whether the clock isn't running.

5.6 AVCC

AVCC is the inventory voltage pin for the A/D Converter, PC3:0, and ADC7:6. It ought to be remotely associated with VCC, regardless of whether the ADC isn't utilized. On the off chance that the ADC is utilized, it ought to be associated with VCC through a low-pass channel. Note that PC6..4 utilize computerized supply voltage, VCC

5.7 AREF

AREF is the simple reference pin for the A/D Converter

5.8 ADC7:6 (TQFP and QFN/MLF Package Only)

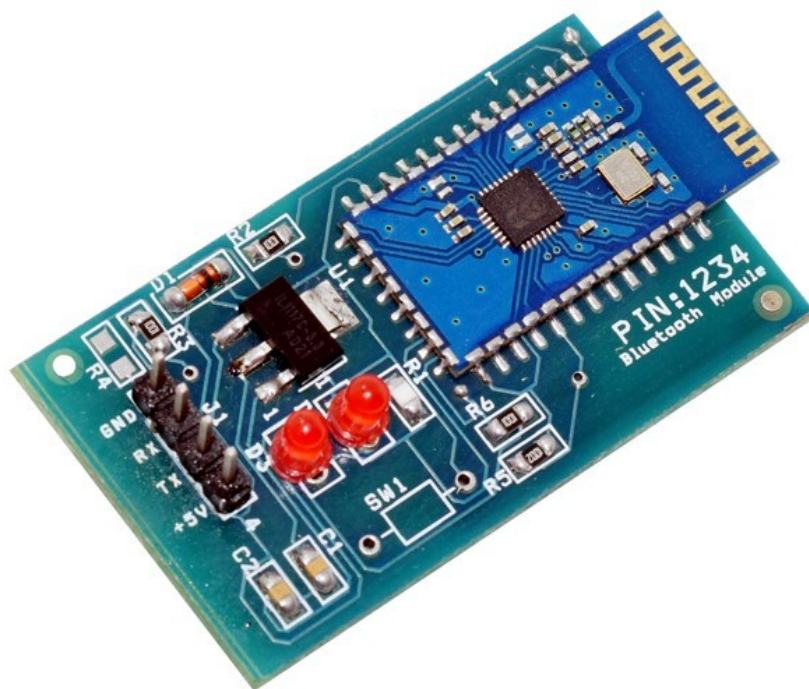
In the TQFP and QFN/MLF bundle, ADC7:6 fill in as simple contributions to the A/D converter. These pins are controlled from the simple stockpile and fill in as 10-digit ADC channels.

CHAPTER – 6

6.BLUETOOTH

6.1 Depiction

Bluetooth is a remote innovation standard for trading information over brief distances (utilizing short-frequency UHF radio waves in the ISM band from 2.4 to 2.485 GHz) from fixed and cell phones, and building individual zone organizations (PANs).

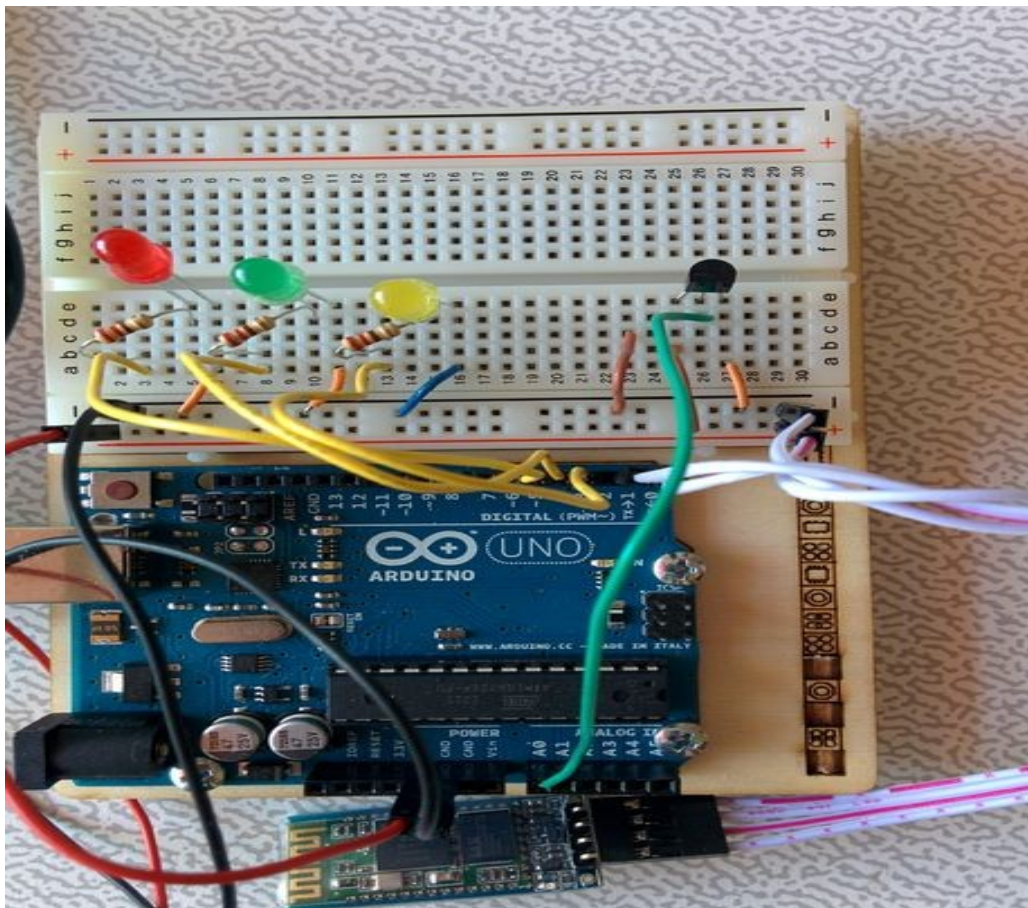


6.1.1BLUETOOTH MODULE

Designed by telecom merchant Ericsson in 1994, it was initially imagined as a remote option in contrast to RS-232 information

links. It can interface a few gadgets, conquering issues of synchronization.

Bluetooth UART empowers you to remote send and get sequential information. Gadgets furnished with Bluetooth innovation support remote highlight point associations, just as remote admittance to cell phones. You can just utilize it for sequential port substitution to set up association among MCU and PC for information move



6.1.2 BLUETOOTH INTERFACE WITH ARDUINO

It delivers the received data and receives the data to be transmitted to and from a host system through a host controller interface.

6.2 FEATURES

- Supply voltage: 5VDC
- Distance range: 20m
- Paired with mobile phone
- UART interface

6.3 APPLICATIONS

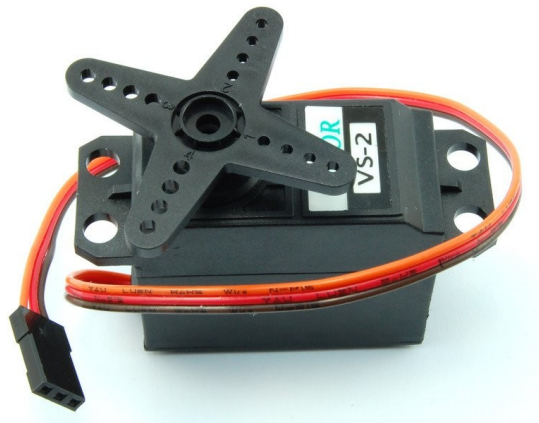
- Wireless Telemetry
- Remote Data Logging
- Robotics

6.4 SERVO MOTOR

6.4.1 DESCRIPTION

Designed by telecom merchant Ericsson in 1994, it was initially imagined as a remote option in contrast to RS-232 information links. It can interface a few gadgets, conquering issues of synchronization.

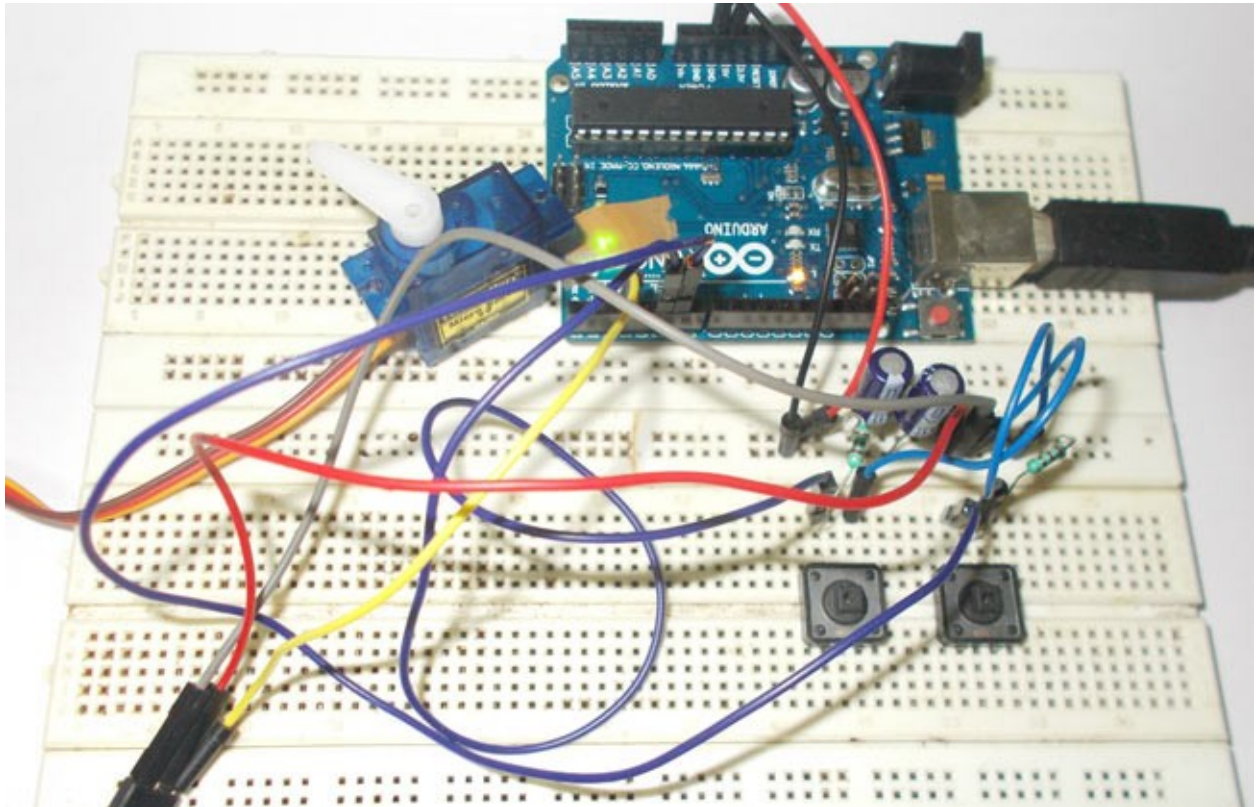
Bluetooth UART empowers you to remote send and get sequential information. Gadgets furnished with Bluetooth innovation support remote highlight point associations, just as remote admittance to cell phones. You can just utilize it for sequential port substitution to set up association among MCU and PC for information move.



6.4.2 SERVO MOTOR

Servomotors are not a particular class of engine albeit the term servomotor is regularly used to allude to an engine appropriate for use in a shut circle control framework. The engine is combined with some sort of encoder to give position and speed input. In the least difficult case, just the position is estimated. The deliberate situation of the yield is contrasted with the order position, the outer contribution to the regulator.

In the event that the yield position varies from that required, a blunder signal is created which at that point makes the engine turn one or the other way, depending on the situation to carry the yield shaft to the fitting position. As the positions approach, the mistake signal decreases to nothing and the engine stops.



6.4.3 SERVO MOTOR INTERFACE WITH ARDUINO

A servomotor is a rotating actuator or straight actuator that considers exact control of precise or direct position, speed and speed increase. It comprises of an appropriate engine coupled to a sensor for position criticism. It likewise requires a generally refined regulator, regularly a devoted module planned explicitly for use with servomotors.

A servomotor is shut circle servomechanism that utilizes position input to control its movement and last position. The contribution to its control is some sign, either simple or advanced, addressing the position directed for the yield shaft.

6.5 FEATURES

- Supply voltage: 5VDC
- Control signal: Analog or Digital
- High-precision positioning.

6.6 APPLICATIONS

- Conveyors
- Solar Tracking System
- Antenna Positioning
- Camera Auto Focus
- It is used to measure the speed of the output shaft

6.7 MOTOR DRIVER

L293D is a regular Motor driver or Motor Driver IC which permits DC engine to drive on either direction. L293D is a 16-pin IC which can handle a bunch of two DC engines all the while toward any path. It implies that you can handle two DC engine with a single L293D IC. Double H-connect Motor Driver incorporated circuit (IC)

L293D is a double H-connect engine driver coordinated circuit (IC). Engine drivers go about as momentum intensifiers since they take a low-ebb and flow control flag and give a higher-ebb and flow signal. This higher current sign is utilized to drive the engines.



L293D contains two inbuilt H-connect driver circuits. In its

regular method of activity, two DC engines can be driven at the same time, both in forward and invert heading. The engine activities of two engines can be constrained by input rationale at pins 2 and 7 and 10 and 15. Information rationale 00 or 11 will stop the comparing engine. Rationale 01 and 10 will turn it in clockwise and anticlockwise ways, separately.

Empower pins 1 and 9 (relating to the two engines) should be high for engines to turn over working. At the point when an empower input is high, the related driver gets empowered. Thus, the yields become dynamic and work in stage with their data sources. Additionally, when the empower input is low, that driver is crippled, and their yields are off and in the high-impedance state.

6.7.1 Features

- Easily compatible with any of the system
- Easy interfacing through FRC (Flat Ribbon Cable)
- External Power supply pin for Motors supported
- Onboard PWM (Pulse Width Modulation) selection switch
- 2pin Terminal Block (Phoenix Connectors) for easy Motors Connection
- Onboard H-Bridge base Motor Driver IC (L293D)

6.7.2 Technical Specification:

- Power Supply : Over FRC connector 5V DC
- External Power 9V to 24V DC
- Dimensional Size : 44mm x 37mm x 14mm (l x b x h)
- Temperature Range : 0°C to +70 °C

CHAPTER – 7

7.POWER SUPPLY UNIT

7.1 DESCRIPTION

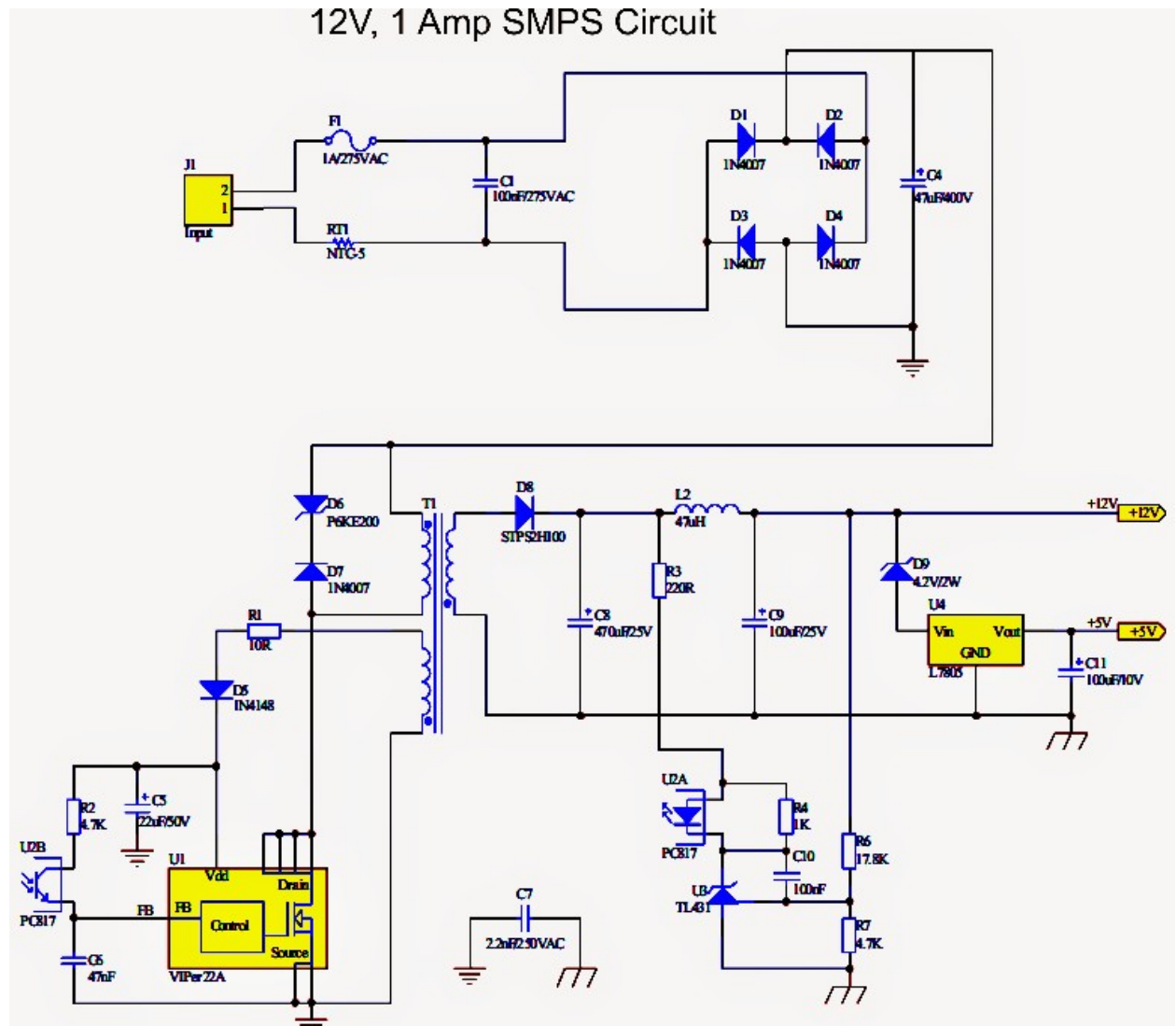
A force supply unit (or PSU) changes mains AC over to low-voltage managed DC power for the inward parts of a PC. Present day PCs generally utilize exchanged mode power supplies. Some force supplies have a manual switch for choosing input voltage, while others naturally adjust to the mains voltage.

A force supply is utilized to decrease the mains power at 240 volts AC down to something more useable, say 12 volts DC. There are two kinds of force supply, straight and switch mode. A direct force supply utilizes a transformer to lessen the voltage. The AC signal is redressed and controlled to deliver a high DC voltage.

An AC connector, AC/DC connector, or AC/DC converter is a kind of outside power supply, frequently encased for a situation like an AC plug. Connectors for battery-controlled hardware might be depicted as chargers or rechargers (see likewise battery charger). AC connectors are utilized with electrical gadgets that require control however don't contain inside segments to infer the necessary voltage and force from fundamental force. The inward hardware of an outside power supply is fundamentally the same as the plan that would be utilized for an implicit or inner stockpile.

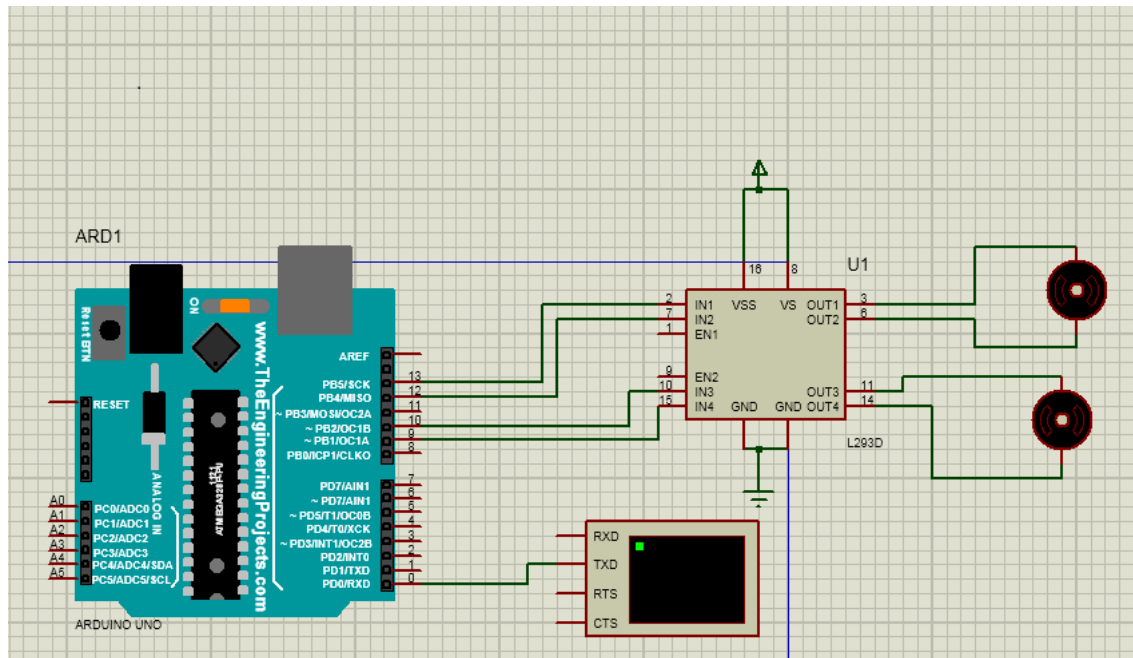
A connector is a gadget that converts ascribes of one electrical gadget or framework to those of a generally incongruent gadget or framework. Some alter force or sign ascribes, while others only adjust the actual type of one electrical connector to another. In a PC, a connector is regularly incorporated into a card that can be embedded into an opening on the PC's

motherboard. The card adjusts data that is traded between the PC's microchip and the gadgets that the card upholds.(11.1.1)



7.1.2 CIRCUIT DIAGRAM:

An electric force connector may empower association of a force plug, some of the time called, utilized in one district to an AC power attachment utilized in another, by offering associations for the dissimilar contact plans, while not changing the voltage. An AC connector, additionally called a "recharger", is a little force supply that changes family electric flow from circulation voltage) to low voltage DC appropriate for purchaser hardware.



Some modify power or signal attributes, while others merely adapt the physical form of one electrical connector to another. For computers and related items, one kind of serial port adapter enables connections between 25-contact and nine-contact connectors, but does not affect electrical power- and signalling-related attributes

7.3 FEATURES

- Output current:1A
- Supply voltage: 220-230VAC
- Output voltage: 12VDC
- Reduced costs
- Increased value across front-office and back-office functions
- Access to current, accurate, and consistent data
- It generates adapter metadata as WSDL files with J2CA extension.

7.4 APPLICATIONS

1)

2) Back-end frameworks which need to send buy request information to prophet applications send it to the mix administration by means of an incorporation worker customer.

3)

4) • SMPS applications.

5)

6) I. SOFTWARE

7)

8) ARDUINO IDE:

9)

10) The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a word processor for composing code, a message region, a book support, a toolbar with catches for basic capacities and a progression of menus. It associates with the Arduino and Genuino equipment to transfer programs and speak with them.

11)

12) Projects composed utilizing Arduino Software (IDE) are called outlines. These representations are written in the word processor and are saved with the document expansion .ino. The supervisor has highlights for cutting/gluing and for looking/supplanting text. The message region gives criticism while saving and sending out and furthermore shows blunders. The support shows text yield by the Arduino Software (IDE), including total mistake messages and other data. The base righthand corner of the window shows the designed board and sequential port. The toolbar catches permit you to check and transfer programs, make, open, and save outlines, and open the chronic

screen.

13)

- 14) Prior to transferring your sketch, you need to choose the right things from the Tools > Board and Tools > Port menus. The sheets are portrayed underneath. On the Mac, the sequential port is presumably something like `/dev/tty.usbmodem241` (for an Uno or Mega2560 or Leonardo) or `/dev/tty.usbserial-1B1` (for a Duemilanove or prior USB board), or `/dev/tty.USA19QW1b1P1.1` (for a sequential board associated with a Keyspan USB-to-Serial connector). On Windows, it's likely COM1 or COM2 (for a sequential board) or COM4, COM5, COM7, or higher (for a USB board) - to discover, you search for USB sequential gadget in the ports part of the Windows Device Manager. On Linux, it ought to be `/dev/ttyACMx`, `/dev/ttyUSBx` or comparable. Whenever you've chosen the right sequential port and board, press the transfer button in the toolbar or select the Upload thing from the Sketch menu. Current Arduino sheets will reset consequently and start the transfer. With more established sheets (pre-Diecimila) that need auto-reset, you'll need to press the reset button on the board not long prior to beginning the transfer. On most sheets, you'll see the RX and TX LEDs flicker as the sketch is transferred. The Arduino Software (IDE) will show a message when the transfer is finished, or show a mistake.

15)

- 16) At the point when you transfer a sketch, you're utilizing the Arduino bootloader, a little program that has been stacked on to the microcontroller on your board. It permits you to transfer code without utilizing any extra equipment. The bootloader is dynamic for a couple of moments when the board resets; at that point it begins whichever sketch was most as of late transferred to the

microcontroller. The bootloader will squint the ready (pin 13) LED when it begins (for example at the point when the board resets).

17)

18) PROGRAMMING LANGUAGE:

19)

20) Inserted C is a bunch of language expansions for the C programming language by the C Standards Committee to address shared trait gives that exist between Cextensions for various implanted frameworks.

21)

22) Inserted C is a bunch of language augmentations for the C programming language by the C Standards Committee to address shared characteristic issues that exist between C expansions for various implanted frameworks. Truly, installed C programming requires nonstandard expansions to the C language to help outlandish highlights, for example, fixed-point math, various particular memory banks, and essential I/O activities

23)

24) An inserted framework is a PC framework with a devoted capacity inside a bigger mechanical or electrical framework, frequently with ongoing figuring requirements. It is implanted as a feature of a total gadget regularly including equipment and mechanical parts. Implanted frameworks control numerous gadgets in like manner use today. 98% of all chip are fabricated as parts of inserted frameworks.

25)

26) Instances of properties of run of the mill implanted PCs when contrasted and universally useful partners are

low force utilization, little size, rough working reaches, and low per-unit cost. This comes at the cost of restricted handling assets, which make them essentially more hard to program and to associate with. Notwithstanding, by building knowledge instruments on top of the equipment, exploiting conceivable existing sensors and the presence of an organization of implanted units, one can both ideally oversee accessible assets at the unit and organization levels just as give enlarged capacities, well past those accessible. For instance, clever strategies can be intended to oversee power utilization of implanted frameworks.

27)

28) Current inserted frameworks are frequently founded on microcontrollers (for example Computer processor's with coordinated memory or fringe interfaces), yet standard microchips (utilizing outside chips for memory and fringe interface circuits) are additionally normal, particularly in more-complex frameworks. Regardless, the processor(s) utilized might be types going from universally useful to those worked in certain class of calculations, or even specially crafted for the current application. A typical standard class of committed processors is the advanced sign processor (DSP).

29)

30) Since the inserted framework is devoted to explicit assignments, plan specialists can advance it to decrease the size and cost of the item and increment the unwavering quality and execution. Some inserted frameworks are mass-delivered, profiting by economies of scale.

31)

32) Implanted frameworks range from versatile gadgets

like computerized watches and MP3 players, to enormous fixed establishments like traffic signals, manufacturing plant regulators, and to a great extent complex frameworks like cross breed vehicles, MRI, and aeronautics. Intricacy changes from low, with a solitary microcontroller chip, to exceptionally high with numerous units, peripherals and organizations mounted inside a huge suspension or nook.

33)

34) Other than utilizing Labview[3], Matlab arduino interfacing and simulink block[37, 41], Arduino IDE gives an all around organized wide scope of capacities those direct the arduino pins and ports straightforwardly. It can encourages programming Arduino board effectively and transfer the coding or order on arduino board. It contains a SERVO library of capacity that can be utilized to control servo engines and the programming language is like JAVA and C++ language. Arduino IDE assists the software engineers with diminishing the programming overhead and improve running season of hidden program[42].

35)

36) I. Bluetooth Communication

37)

38) Remote control framework can be agreed by Zigbee[2, 45], and Bluetooth technology[31, 35, 36, 38-41], and To control the automated arm by utilizing an Android Application, a Bluetooth module is required. Bluetooth is a remote correspondence convention running at the speed of 2.4 Ghz with the design of customer worker and it can make an individual region network[39]. HC-06 Bluetooth module sends information remotely to Bluetooth empowered Arduino.

39)

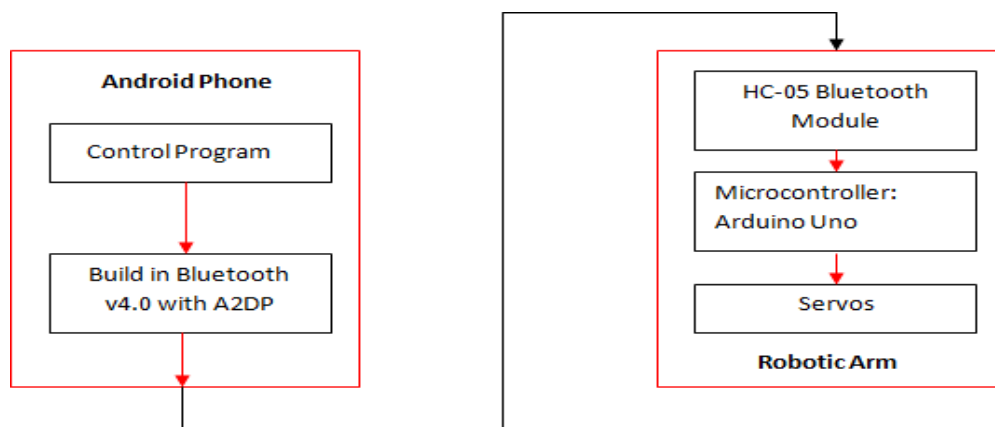
40) II. ANDROID PLATFORM

41)

42) Android gadget offer various correspondence interfaces like USB, Wi-Fi, Bluetooth that can be utilized for association with the mechanical arm. It is most stretched out utilized stage on the planet and reasonable for mechanical framework control. Additionally, it is less expensive whenever contrasted with some other ARM-based handling unit[39].

43)

44) Servoduino application is prepared on the lookout and utilized in this task. With this application, automated framework control should be possible by controlling the servo engines with arduino microcontroller in the wake of sending orders from the cell phone through Bluetooth association with the mechanical arm. Subsequent to forming the programming code in PC programming, it will be imported to servoduino application in cell phone. The benefit of servoduino application is easy to understand as a basic UI is given and significant distance control should be possible utilizing cell phone. Block Diagram



7.4.1 ROBOT CONTROL SYSTEM BLOCK DIAGRAM.

Automated arm spoke with client through Bluetooth to the Bluetooth module introduced on the control arrangement of mechanical arm. At the point when a client slides each catch on the application, input signal or controlling sign is given from the matched Android advanced mobile phone and afterward sent through Bluetooth to the mechanical arm. The information from the advanced mobile phone is sent as ASCII design to be gotten by the Bluetooth module and gave to Arduino for additional activity. The Arduino recognized the contribution of sign and changed over them into order, the robot at that point checked the order with its recently characterized orders and the servo engines acts appropriately relying upon the order got from the microcontroller by pushing ahead, in reverse, left, right or to stop. Accordingly, a PDA Android worked robot is made to perform lifting errands.

7.5 TEST RIG DEVELOPMENT

To approve the robot arm and its segment, barely any tests were done which included testing the two parts and the generally speaking automated arm framework. shown tests led on the automated arm have been done in research facility.

By shifting the situation of the items which should be lifted by the automated arm, the servo engine development range is tried. Distinctive direct motivations to every servo engine are sent by providing an order through PDA. This can assists with checking the reaction of the servo engine whether it can moves to the correct situation as per the order given by the client. This cycle happened when servo engines deciphered the sign from microcontroller through encoder which brought about the turn to the ideal position. The underlying and last position are set apart to rate the

exactness of the actuator.

For generally speaking framework execution, most extreme burden which ready to be lifted by the mechanical arm is resolved utilizing various loads. During the test, the mechanical arm got the weight and moved it to a specific position.

7.6 Equipment TESTING

An installed framework is a PC framework intended to do one or a couple of devoted as well as explicit capacities frequently with ongoing figuring requirements." So inserted framework is only minimal effort, low space and low support framework. It is intended to deal with a particular errand to be constrained by at least one fundamental preparing centers that are normally either microcontrollers or advanced sign processors.

7.7 TESTING

Testing is dubious term and to characterize it we should know and we ought to have a few baselines. Fundamentally to comprehend testing we should know not many things:

1. Testing isn't autonomous module it is interlinked with the framework to be tried.
2. To test the framework you should know the framework, this really implies on the off chance that you need to test anything first you ought to have an intensive information on the framework.

These are extremely fundamental things and can be applied to any framework/scenario. For Embedded Systems testing we ought to know about the testing

climate.

7.7.1 We have basically three categories for testing:

1. White box testing
2. Black Box testing
3. Grey Box testing

7.8WHITE BOX TESTING

It is finished with the genuine code. On the off chance that we are utilizing a few API's some other device to see code inclusion and so forth you are doing white box testing. In the event that we are changing the code to see whether it is functioning true to form isn't, is white box testing.

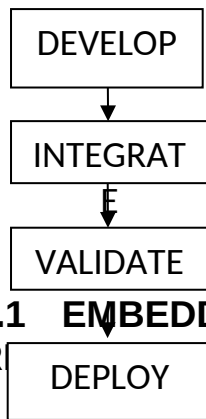
7.9DISCOVERY TESTING

On the off chance that we accept that each framework has a bug and attempts to discover absconds without knowing the inward coding of the framework.

7.10DARK BOX TESTING

On the off chance that we realize how inward coding has been done and composing the experiments dependent on how the framework will be utilized goes under dark box testing.

7.11SOFTWARE TESTING



7.11.1 EMBEDDED SOFTWARE IS WRITTEN FOR SPECIFIC HARDWARE

- Requires specific inputs and outputs
- Target has different architecture
 - Endianness
 - Memory model
 - Hardware accelerators
- Target has limited resources
 - Memory
 - Disk

7.11.2 SIMULATION OVERCOMES LIMITED ACCESS TO HARDWARE

- Different levels of simulation
 - Emulation: qemu (PowerPC, ARM)
 - Virtualization: KVM, Virtual Box
 - Stubbing/hardware abstraction
- PC has much more resources and performance

7.11.3 TESTING AND OPEN SOURCE SOFTWARE

- Open source tools to support testing
 - Unit tests
 - Doctest
 - D-Bus
- Testing of open source tools
 - Gstreamer
 - Linux kernel

7.11.4 LINUX KERNEL HAS STUBS AND TEST FRAMEWORK

- For most device types, stub implementation exists
 - For USB gadget: dummy_hcd, zero
 - For MTD (= flash): block2mtd
- Linux Test Project (LTP) offers a large test suite
 - Mainly for file systems and networking.
 - Mainly regression, load and performance tests.

Developer should write tests from the start, It saves time, Share the tests with other developers, Even more important for open source projects, For embedded systems, hardware abstraction / stubs are essential.

7.11.5 TEST DESIGN TOOLS

Tools that help you decide what tests need to be executed. Test data and test case generators.

7.11.6 GUI TEST DRIVERS

Tools that automate execution of tests for products with graphical user interfaces. Client/server test automation tools, including load testers, also go here.

7.11.7 LOAD AND PERFORMANCE TOOLS

Tools that specialize in putting a heavy load on systems (especially client-server Systems). These tools are often also GUI test drivers.

7.11.8 TEST MANAGEMENT TOOLS

Tools that automate execution of tests for products without graphical user interfaces. Also tools that help you work with large test suites.

CHAPTER 8/ RESULTS & DISCUSSION

I. ROBOTIC ARM MOVEMENT COVERAGE

The greatest reach for the automated arm are recorded during the test and appeared in The further get point of the mechanical arm is 2.10cm, and the most extreme point the mechanical arm can reach is 92o, with range from 36o to 128o.

II. FINAL POSITION

Consequence of the automated arm of lifting with various weight is introduced in this part. The heap to be lifted in this examination is a blockade with various weight. The automated arm is told to lift the block and move it to a particular position. The examination is begun to look at the exactness of situating with a variety in weight of a blockade which is in the scope of 20 grams to 100 grams, the heap with 20 grams go about as a source of perspective. The accuracy of the mechanical arm to lift various loads is recorded in the table beneath. From the information got, the mechanical arm can lift 100 grams true to form bring about this task. In any case, the development of mechanical arm isn't smooth when it lifted 100 grams because of absence of solidarity in the linkage that comprised of aroused wire with 1mm measurement. This issue can be addressed by utilizing a high strength linkage which comprised of steel.

PRECISION IN X AND Y AXIS.

Weight (grams)	Precision (mm)	
	x- axis	y- axis

20	0	0
40	3	7
60	0	10
80	12	17
10	30	32
0		

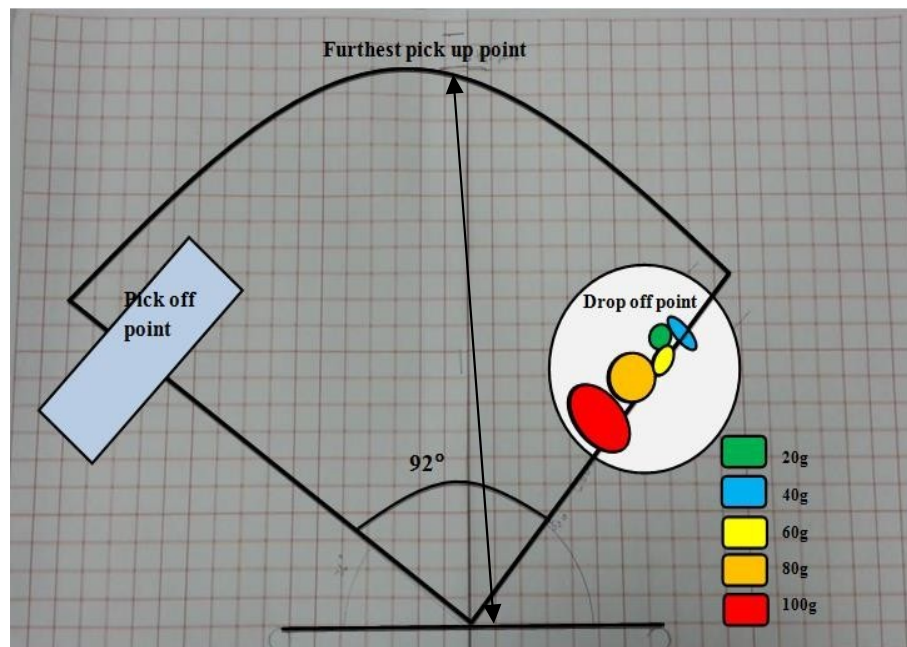
Table;8.1

II. TIME DURATION

A complete lifting cycle can be done at 15.55 seconds and there is an insignificant difference of time taken for various weights.

Table 13.6.2. Duration of lifting process.

Weight (grams)	Duration (second)
20	15.54
40	15.64
60	15.65
80	15.67
100	15.68



8.1.1Maximum range of robotic arm movement.

8.1.2 CONCLUSION

This undertaking presents the plan and improvement of a mechanical arm for light material lifting application. The mechanical arm is drafted utilizing PC helped plan programming, Autocad and creation with advance innovation, 3D-printing to save cost and time. The mechanical arm is comprised of ABS (Acrylonitrile Butadiene Styrene) and one of the linkages between the body and engine is comprised of PLA (Polylactic Acid). During the 3D printing measure, we dealt with an issue in printing an overhanging part which emphatically influenced the joining part by causing an extreme deformity. Meshmixer programming which can create support consequently at the overhanging part in this manner a fulfilled item is gotten. The actuator used to perform arm development is 4 servo engines. The servo engine is controlled by means of an Arduino Uno through coding. Because of the constraint of the servo engine, the development scope of the automated arm is confined inside 180 degrees. The automated arm is planned with four levels of opportunity since it is reasonable for the vast majority of the application included lifting. Therefore, it tends to be savvy and basic development by utilizing just 4 actuators.

The automated arm parts are consolidated by clasp: screws and nuts. For the mechanical arm control, the framework basically comprised of three segments which are microcontroller, android application and servo engine. Arduino microcontroller is less expensive and simpler to be customized with C language contrasted with the other microcontrollers. A UI in Servo arduino, an android application permits the client to control the development with no preparation required so the automated arm is easy to understand. Analysis is set up to test the presentation of the mechanical arm by changing the heap to be lifted; the lifting system of an automated arm is approved upheld by the outcomes acquired in the test.

FUTURE SCOPE

In the further turn of events, the automated arm can be arranged on a versatile stage with 4 wheels to permit convenience and route. Plan of a widespread gripper is intriguing on the grounds that it can lift various states of articles. Mechanical arm has sensors to distinguish the situation of the articles and the entire cycle is robotized and it can likewise speak with client through systems administration.

REFERENCES

- [1] I. Standard, 8373: 1994.
- [2] E. Punna, R. K. Nenavath, and B. Maloth, "Labview Controlled Robotic ARM."
- [3] A. Elfasakhany, E. Yanez, K. Baylon, and R. Salgado, "Design and development of a competitive low-cost robot arm with four degrees of freedom," *Modern Mechanical Engineering*, vol. 1, p. 47, 2011.
- [4] C. F. Olson, "Probabilistic self-localization for mobile robots," *Robotics and Automation, IEEE Transactions on*, vol. 16, pp. 55-66, 2000.
- [5] I. P. MODEL, "Industry Based Automatic Robotic Arm."
- [2] C. Bartneck and J. Forlizzi, "A design-centred framework for social human-robot interaction," in *Proceedings of the 13th IEEE International Workshop on Robot and Human Interactive Communication*, 2004, pp. 31-33.
- [3] G. Katal, S. Gupta, and S. Kakkar, "DESIGN AND OPERATION OF SYNCHRONIZED ROBOTIC ARM."
- [4] J. L. Jones, "Robots at the tipping point: the road to iRobot Roomba," *Robotics & Automation Magazine, IEEE*, vol. 13, pp. 76-78, 2006.
- [5] A. C. McDonald, "Robot technology: theory, design and applications," *Sir Sanford Fleming Coll., Ontario*1986.
- [6] S. Yu, X. Yu, B. Shirinzadeh, and Z. Man, "Continuous finite-time control for robotic manipulators with terminal sliding mode," *Automatica*, vol. 41, pp. 1957-1964, 2005.
- [7] M. Hägele, K. Nilsson, and J. N. Pires, "Industrial robotics," in *Springer handbook of robotics*, ed: Springer, 2008, pp. 963-986.
- [8] R.-J. Wang, J.-W. Zhang, J.-M. Xu, and H.-Y. Liu, "The multiple-function intelligent robotic arms," in *Fuzzy Systems, 2009. FUZZ-IEEE 2009. IEEE International Conference on*, 2009, pp. 1995-2000.
- [9] L. B. Duc, M. Syaifuddin, T. T. Toai, N. H. Tan, M. N. Saad, and L. C. Wai, "Designing 8 degrees of freedom humanoid robotic arm," in *Intelligent and Advanced Systems, 2007. ICIAS 2007. International Conference on*, 2007, pp. 1069-1074.
- [10] C. Carignan, G. Gefke, and B. Roberts, "Intro to Space Mission Design: Space Robotics," in *Seminar of Space Robotics*, University of Maryland, Baltimore, 2002.
- [11] R. Bonitz, J. Slostad, B. Bon, D. Braun, R. Brill, C. Buck, et al.,

- "Mars volatiles and climate surveyor robotic arm," *Journal of Geophysical Research: Planets* (1991– 2012), vol. 106, pp. 17623-17634, 2001.
- [12] R. Bonitz, L. Shiraishi, M. Robinson, J. Carsten, R. Volpe, A. Trebi-Ollennu, et al., "The Phoenix Mars lander robotic arm," in *Aerospace conference*, 2009 IEEE, 2009, pp. 1-12.
 - [13] R. Boyce and J. Mull, "Complying with the Occupational Safety and Health Administration: Guidelines for the dental office," *Dental Clinics of North America*, vol. 52, pp. 653-668, 2008.
 - [14] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*: Springer Science & Business Media, 2009.
 - [15] M. A. Rahman, A. H. Khan, T. Ahmed, and M. M. Sajjad, "Design, Analysis and Implementation of a Robotic Arm-The Animator."
 - [16] P. Singh, A. Kumar, and M. Vashisth, "Design of a Robotic Arm with Gripper & End Effector for Spot Welding," *Universal Journal of Mechanical Engineering*, vol. 1, pp. 92-97, 2013.
 - [17] M. W. Spong, Vidyasagar, *Robot Dynamics and Control*, 1989.
 - [18] A. J. Critchlow, *Introduction to robotics*: Macmillan New York, 1985.
 - [19] M. A. K. Yusoff, R. E. Samin, and B. S. K. Ibrahim, "Wireless mobile robotic arm," *Procedia Engineering*, vol. 41, pp. 1072-1078, 2012.
 - [20] G. Mikell and N. Roger, "Robótica Industrial. Tecnología, Programación y Aplicaciones. Ed," ed: Mc Graw Hill.
 - [21] C. A. Schuler and W. L. McNamee, *Industrial electronics and robotics*: McGraw- Hill, Inc., 1986.
 - [22] S. R. Nair, "Design of a Robotic Arm for picking and placing an object controlled using LabVIEW," *International Journal of Scientific and Research Publications*, vol. 2, pp. 1-4, 2012.
 - [23] F. J. Diaz-Colorado and R. F. Martinez-Gonzalez, "Design and Implementation of a Low-Cost Embedded System to Turn Android Device into a Robotic Arm Controller," *Technology*, vol. 2, pp. 7-12, 2015.
 - [24] J. Olawale, A. Oludele, A. Ayodele, and N. M. Alejandro, "Development of a Microcontroller Based Robotic Arm," in *Proceedings of the 2007 Computer Science and IT Education Conference*, 2007.
 - [25] G. A, N. R, K. S, and G. R, "Design and Control of 3DOF Articulated Robotic Arm using LabVIEW and NI-myRIO," *International Journal of Innovative Research in Electrical*,

- Electronics, Instrumentation and Control Engineering, vol. 3, 2015.
- [26] M. C. C. Mouli, M. P. Jyothi, K. N. Raju, and C. Nagaraja, "Design and Implementation of Robot Arm Control Using LabVIEW and ARM Controller," IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) e-ISSN, pp. 2278- 1676, 2013.
 - [27] R. Pahuja and N. Kumar, "Android Mobile Phone Controlled Bluetooth Robot Using 8051 Microcontroller," Electronics & Communication Engineering, Department, BRCM College of Engineering & Technology, Bahal, India, International Journal of Scientific Engineering and Research (IJSER) www. ijsr. in ISSN (Online), pp. 2347- 3878, 2014.
 - [6] M. Jabir, N. K, N. John, M. Fayas, M. Mohan, M. Sajeev, et al., "Wireless Control of Pick and Place Robotic Arm Using an Andriod Application," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (IJAREEIE), vol. 4, 2015.
 - [7] A. Verma and A. Sharma, "Wireless Robotic ARM," IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), vol. 9, 2014.
 - [8] Z. A. Ali, M. Tanveer, H. Shaukat, and S. Anwar, "Android Operated Robotic Arm," Universal Journal of Control and Automation, vol. 2, pp. 1-3, 2014.
 - [9] N. Rai, P. G. Bhutia, and U. Pradhan, "Remote Learning: Android Operated Educational Robot Arm with 6 DOF," International Journal of Electrical, Electronics and DataCommunication, vol. 2, 2014.
 - [10] D. S. Praneet Singh, PRachi, Jitendra Kumar, "Android Based, Arduino Powered Automated Wall Painting," International Journal of Computer Science and Information Technologies (IJCSIT), vol. 5, pp. 4490-4491, 2014.
 - [11] A. Soriano, L. Marin, M. Valles, A. Valera, and P. Albertos, "Low Cost Platform for Automatic Control Education Based on Open Hardware.*," in World Congress Cape Town, South Africa, 2014, pp. 9044-9050.
 - [12] M. M. Rahman, Ashik-E-Rasul, N. M. A. Haq, and M. M. Hassan, "An Android Controlled Mobile Robot for Stereo Vision and Live Streaming with Robotic Arm," presented at the International Conference on Mechanical, Industrial and Energy Engineering 2014, Khulna, BANGLADESH. , 2014.
 - [13] A. Ansari, "Android App Based Robot," International Journal of

Computer Science and Information Technologies (IJCSIT), vol. 6, pp. 1598-1600, 2015.

- [14] S. Narayanan and C. R. Reddy, "Bomb Defusing Robotic Arm using Gesture Control," in International Journal of Engineering Research and Technology, 2015.
- [15] G. Gautum, A. Ashish, A. Kumar, and Avdesh, "Wirelessly Hand Glove Operated Robot," International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), vol. 3, 2014.
- [16] A. Chavan, A. B. A. Jain, N. Shinde, and M. Salunke, "Implementation of Generic Multi- Purpose Robotic Arm," International Journal of Computer Applications, vol. 116, 2015
- [17] R. Krishna, G. S. Bala, S. S. ASC, B. B. P. Sarma, and G. S. Alla, "Design and implementation of a robotic arm based on haptic technology," Int. J. of Eng. Research and Applications, vol. 2, 2012.
- [18] P. Herman, A. R. Mosteo, and A. C. Murillo, "A low-cost custom-built robotic arm on the TurtleBot platform."
- [19] B. Ganesh Choudhary and B. Chethan Ram, "Real time robotic arm control using hand gestures," in High Performance Computing and Applications (ICHPCA), 2014 International Conference on, 2014, pp. 1-3.

SOURCECODE:

```
int motor1Pin1 = 8; // pin 2 on L293D IC
int motor1Pin2 = 7; // pin 7 on L293D IC
int enable1Pin = 9; // pin 1 on L293D IC
int motor2Pin1 = 5; // pin 10 on L293D IC
int motor2Pin2 = 4; // pin 15 on L293D IC
int enable2Pin = 6; // pin 9 on L293D IC
int motor3Pin1 = 14; // pin 2 on L293D IC
int motor3Pin2 = 13; // pin 7 on L293D IC
int enable3Pin = 15; // pin 1 on L293D IC
int motor4Pin1 = 11; // pin 10 on L293D IC
int motor4Pin2 = 10; // pin 15 on L293D IC
int enable4Pin = 12; // pin 9 on L293D IC
int state;
int flag=0; //makes sure that the serial only prints once the state
void setup() {
  pinMode(motor1Pin1, OUTPUT),pinMode(motor1Pin2, OUTPUT),pinMode(enable1Pin, OUTPUT);
  pinMode(motor2Pin1, OUTPUT),pinMode(motor2Pin2, OUTPUT),pinMode(enable2Pin, OUTPUT);
  pinMode(motor3Pin1, OUTPUT),pinMode(motor3Pin2, OUTPUT),pinMode(enable3Pin, OUTPUT);
  pinMode(motor4Pin1, OUTPUT),pinMode(motor4Pin2, OUTPUT),pinMode(enable4Pin, OUTPUT);
  Stop();
  Serial.begin(9600);
}
```

```

void loop() {
  if(Serial.available() > 0){
    state = Serial.read();
    flag=0;
  }

  if (state == 'F'){
    Forward();
    if(flag == 0)Serial.println("Go Forward"),flag=1;
  }

  else if (state == 'B'){
    Reverse();
    if(flag == 0)Serial.println("Go Reverse"),flag=1;
  }

  else if (state == 'L'){
    Left();
    if(flag == 0)Serial.println("Go Left"),flag=1;
  }

  else if (state == 'R'){
    Right();
    if(flag == 0)Serial.println("Go Right"),flag=1;
  }

  else if (state == 'S'){
    Stop();
    if(flag == 0)Serial.println("STOP!"),flag=1;
  }

  else if (state == 'P'){
    Pick();
    if(flag == 0)Serial.println("Pick"),flag=1;
  }

  else if (state == 'Q'){
    Place();
    if(flag == 0)Serial.println("Place"),flag=1;
  }

  else if (state == 'H'){
    Hold();
    if(flag == 0)Serial.println("Hold"),flag=1;
  }

  else if (state == 'I'){
    Free();
    if(flag == 0)Serial.println("Unhold"),flag=1;
  }
}

void Forward(){
  digitalWrite(motor1Pin1, LOW);
  digitalWrite(motor1Pin2, HIGH);
  digitalWrite(motor2Pin1, HIGH);
  digitalWrite(motor2Pin2, LOW);
}

```

```

    digitalWrite(enable1Pin, HIGH);
    digitalWrite(enable2Pin, HIGH);
}
void Reverse(){
    digitalWrite(motor1Pin1, HIGH);
    digitalWrite(motor1Pin2, LOW);
    digitalWrite(motor2Pin1, LOW);
    digitalWrite(motor2Pin2, HIGH);
    digitalWrite(enable1Pin, HIGH);
    digitalWrite(enable2Pin, HIGH);
}
void Left(){
    digitalWrite(motor1Pin1, HIGH);
    digitalWrite(motor1Pin2, LOW);
    digitalWrite(motor2Pin1, HIGH);
    digitalWrite(motor2Pin2, LOW);
    digitalWrite(enable1Pin, HIGH);
    digitalWrite(enable2Pin, HIGH);
}
void Right(){
    digitalWrite(motor1Pin1, LOW);
    digitalWrite(motor1Pin2, HIGH);
    digitalWrite(motor2Pin1, LOW);
    digitalWrite(motor2Pin2, HIGH);
    digitalWrite(enable1Pin, HIGH);
    digitalWrite(enable2Pin, HIGH);
}
void Pick(){
    digitalWrite(motor3Pin1, LOW);
    digitalWrite(motor3Pin2, HIGH);
    digitalWrite(enable3Pin, HIGH);
}
void Place(){
    digitalWrite(motor3Pin1, HIGH);
    digitalWrite(motor3Pin2, LOW);
    digitalWrite(enable3Pin, HIGH);
}
void Hold(){
    digitalWrite(motor4Pin1, LOW);
    digitalWrite(motor4Pin2, HIGH);
    digitalWrite(enable4Pin, HIGH);
}
void Free(){
    digitalWrite(motor4Pin1, HIGH);
    digitalWrite(motor4Pin2, LOW);
    digitalWrite(enable4Pin, HIGH);
}
void Stop(){ // Stop the motors
    digitalWrite(enable1Pin, LOW);
    digitalWrite(enable2Pin, LOW);
    digitalWrite(enable3Pin, LOW);
    digitalWrite(enable4Pin, LOW);
    digitalWrite(motor1Pin1, LOW);
    digitalWrite(motor1Pin2, LOW);
    digitalWrite(motor2Pin1, LOW);
    digitalWrite(motor2Pin2, LOW);
    digitalWrite(motor3Pin1, LOW);
    digitalWrite(motor3Pin2, LOW);
    digitalWrite(motor4Pin1, LOW);

```

```
digitalWrite(motor4Pin2, LOW);  
}
```

