**CHAPTER 1**

**INTRODUCTION**

The field of robotics has its origins in science fictions. The word robot comes from the Czech word "robota" means forced labor in 1920. It took another 40 years before the modern technology of industrial robotics began. Today, robots are highly automated mechanical manipulators controlled by computers. A robot may appear like a human being or an animal or a simple electro-mechanical device. A robot may act under the direct control of a human (e.g. the robotic arm of the space shuttle) or autonomously under the control of a programmed computer. Robots may be used to perform tasks that are too dangerous or difficult for humans to implement directly (e.g. nuclear waste cleanup) or may be used to automate repetitive tasks that can be performed more cheaply by a robot than by the employment of a human (e.g. automobile production) or may be used to automate mindless repetitive tasks that should be performed with more precision by a robot than by a human (material handling, material transfer applications, machine loading and unloading, processing operations, assembly and inspection).

The last two decades have witnessed a significant advance in the field of robots application. Many more applications are expected to appear in space exploration, battlefield and in various actives of daily life in the coming years. A robot is a mechanical device that performs automated tasks and movements, according to either pre-defined program or a set of general guidelines and direct human supervision. These tasks either replace or enhance human work, such as in manufacturing, contraction or manipulation of heavy or hazardous material. Robot is an integral part in automating the flexible manufacturing system that one greatly in demand these days. Robots are now more than a machine, as robots have become the solution of the future as cost labor wages and customers demand. Even though the cost of acquiring robotic system is quite expensive but as today's rapid development and a very high demand in quality with IS0 standards,human are no longer capable of such demands. Research and development of future robots is moving at a very rapid pace due to the constantly improving and upgrading of the quality standards of products.

The most apparent reasons that are associated in installing of robotic systems in food industry are;

1) Saving of manpower.

2) Improved quality & efficiency.

3) Ability to work in any hostile environment.

4) Increased consistency & flexibility.

5) Increased yields and reduced wastage.

* 1. **.PROBLEM DEFINITION**

To solve the problems of pick and place the cylindrical and square object having 1000 gram weight from one conveyor or place to another place at room temperature in order to reduce human efforts and efficient material handling.

**CHAPTER 2**

**LITERATURE SURVEY**

**2.1. INTRODUCTION**

While there is no generally-accepted definition for the term “mobile robot,” it is often taken to mean a device that can move autonomously from place to place to achieve a set of goals (see, e.g., Tzafestas). Mobile robots are used in a wide range of applications including in factories (e.g., automated guided vehicles or AGVs), for military operations (e.g., unmanned ground reconnaissance vehicles), in healthcare (e.g., pharmaceutical delivery), for search and rescue, as security guards, and in homes (e.g., floor cleaning and lawn mowing). Automated guided vehicles or automatic guided vehicles (AGVs) were invented in 1953 . AGVs are most often used in industrial applications to move materials around a manufacturing facility or a warehouse, tuggers (AGVs that pull carts), unit loaders (AGVs with onboard roller tables for parts-tray transfers), and fork trucks (robots similar to manual fork trucks). Use of mobile robots, and AGVs in particular, is growing as the range of robot applications in factories, hospitals, office buildings, etc. increases. While mobile robots can use a range of locomotion techniques such as flying, swimming, crawling, walking, or rolling, this paper focuses mainly on rolling or wheeled mobile robots. More advanced mobile robots are briefly discussed and referenced in the sections on Localizing the Mobile Robot and Advanced Applications. tugger unit load fork lift Typical AGV types (tugger and unit load AGV photos courtesy of America In Motion) This paper reviews research and applications on a range of topics of importance for implementing mobile robots and AGVs in manufacturing. These include planning, navigation, vehicle localization, and interactions between mobile robots and humans and between groups of mobile robots. It also covers a sampling of applications in real-world factories and provides a brief discussion of some advanced mobile robot concepts. Mobile robots address the demand for flexible material handling, the desire for robots to be able to operate on large structures, and the need for rapid reconfiguration of work areas. Much of the earlier work on outdoor vehicles for defense, search and rescue, and bomb disposal is relevant to the manufacturing domain, as is work that has been done on personal care robots and robots for household and hospital applications. When a robot arm is added to the mobile robot, we term this a “mobile manipulator,” discussed briefly in the Advanced Applications section. Two roadmaps provide predictions for future mobile robot systems. For material handling, the Material Handling Institute expects new capabilities in autonomous control, artificial intelligence, and robotics, along with motion- and gesture-sensitive technologies that could lead to systems in which humans, machines, and computers interact freely and effectively in completely new ways. By 2025, it is expected that economical, high-speed automation for loading and unloading trucks should be available, both at the carton and pallet level. For mobile robots used in manufacturing, a recent roadmap for U.S. robotics predicts that by 2030, autonomous vehicles will be capable of driving in any environment in which humans can drive, and furthermore be safer and more predictable than a human driver. Vehicles will be able to learn on their own how to drive in previously unseen scenarios.

**2.2. HUMAN-ROBOT INTERACTION**

It is becoming more common for humans and robots to share a workspace. This has led to the need for improved human-robot communication and for awareness by the robot of what can be expected of the people around it and, similarly, by the people of what can be expected of the robots. An aspect of interaction with robots that is not unique to mobile robots is teaching them the tasks they are expected to accomplish. In Argall a method of teaching by demonstration is described, in which primitive components of motions are learned by a robot through teleoperation. The method is able to extrapolate from a set of basic motions to the development of a complete task without the user having to demonstrate all aspects of the task. Another approach is to use gestures to show the mobile manipulator what it should pick up or where it should go (Pedersen).This requires the definition of gestures that are both easily communicated by humans and easily recognized and disambiguated by the sensors on the mobile robot. Some researchers have also investigated ways in which a robot can ask for help. Rosenthal and Veloso describe a mobile robot that can navigate around an office environment but has no manipulator, so, for example, cannot push the elevator button. The robot has algorithms to enable it to find people and ask them for help, taking into account the imposition on the people it asks (the travel distance to the help location) and the robot’s own need for a short task completion time. Another issue to consider when people are in the environment is addressed by Sisbot et al. Here a planner is developed that computes paths that take into account the comfort and expectations of people that may be near the robot. The plan assures that the robot both keeps a safe distance from all people and tries to keep the robot in the field of view of the people to prevent surprise appearances.

Personal care robots have developed into advanced human-robot interactive systems. For example, Care-O-bot (Graf et al.) is now in its third generation with characteristics that are potentially very useful to the industrial mobile robot community. Navigation is via odometry (measurements of vehicle motion) improved by simultaneous localization and mapping (SLAM) based on front and rear laser scan data that is compared with a global map. A three-level hierarchical controller includes single wheel control, four wheel control, and a trajectory planner to enable path planning around obstacles and through narrow passageways. The omni-directional mobile manipulator includes a tray and robot arm and can compute collision-free manipulation paths based on data from a color camera and light detection and ranging (LIDAR) sensors. The system also implements spatial segmentation for obstacle learning and interpretation of the three-dimensional cloud of points detected by the LIDAR sensors for object recognition.

**2.3. STANDARDS FOR INDUSTRIAL MOBILE ROBOTS**

Currently, there no performance standards for automatic industrial vehicles anywhere in the world . There are, however, safety standards both national and international. A key US standard for AGVs is the American National Standards Institute/ Industrial Truck Standards Development Foundation (ANSI/ITSDF) B56.5 for AGVs and manned vehicles with automated functions . The scope of B56.5 is to define the safety requirements relating to the elements of design, operation, and maintenance of powered, not mechanically restrained, unmanned automatic guided industrial vehicles and the systems of which the vehicles are a part. It also applies to vehicles originally designed to operate exclusively in a manned mode but which are subsequently modified to operate in an unmanned, automatic mode, or in semiautomatic, manual, or maintenance modes.

A new ASTM International AGV performance standards-development task was formally approved by the ASTM main committee on May1, 2014. The effort forms a new committee, entitled: “Driverless Automatic Guided Industrial Vehicles,” with the scope being to develop “standardized nomenclature and definitions of terms, recommended practices, guides, test methods, specifications, and performance standards for Driverless Automatic Guided Industrial Vehicles” while encouraging research and sponsoring symposia, workshops, and publications to facilitate the standards development in coordination with other ASTM technical committees. Five associated sub-committees will be structured to address Environmental Effects, Docking & Navigation, Object Detection & Protection, Communications & Integration, and Terminology. Beyond industrial automatic guided vehicles, yet relevant to humans working close to mobile robots, ISO 13482 crosses over from personal care mobile robots to industrial mobile robot risk assessment and mitigation. This standard includes safety of personal care robots designed to improve quality of life for people. Most of these robots are mobile and intended to directly interact with humans and obstacles in their environments.

**2.4. ADVANCED APPLICATIONS AND AGVS**

The range of applications that lend themselves to the use of mobile robots is growing as the capabilities of the robots and related sensors improve. This growth is also spurred by the overall demand by industry for greater automation and the development of safety standards that let humans and robots share a work area. As a result, a number of prototype mobile robots have been built for manufacturing applications. These range from vehicles that track materials, to mobile manipulators, to aerial drones used for material handling. A novel application of AGVs to keep track of and optimize the locations of items in a warehouse is provided in Hildebrandt. The authors assume that stock items are equipped with radio frequency identification (RFID) tags and that a set of mobile robots can both localize their own positions in the facility and determine the locations of stock items using the RFID tags. The robots move about in the facility and, by tracking the movement of items, robots can identify preferred paths, find opportunities for optimizing storage locations and vehicle trajectories, and keep track of inventory. The Kiva Mobile Fulfillment System [47] is one of several examples where a beneficial side-effect of the way that items are delivered from and returned to storage is the optimization of the placement of items in a warehouse. For example, items that are required frequently will, over time, be stored closer to the delivery area because the robots find convenient places to store them rather than relying on fixed locations.

Throughout this document, discussion of mobile manipulator research has been interspersed with mobile robot research as an extension of mobility. Mobile manipulators are being discussed in standards committees due to the gaps between AGV and robot arm safety standards and the possibility of opening new areas of research (Marvel and Bostelman). Small drone multi-rotor copters are beginning to be explored for use in material handling with the recent concept of drone delivery of small packages weighing up to approximately 2.2 kg (5 lb). This concept from the Netherlands requires minimal infrastructure to install, enables rapid deployment, and is expected to maintain a relatively high sustained throughput. Interest from companies like Amazon will continue to drive battery and control development.The robot is designed to inspect the outer surfaces of large oil ship hulls and floating production storage and offloading platforms. Locomotion over the hull is provided through magnetic tracks, and the system is controlled by two networked PCs and a set of custom hardware devices to drive motors, video cameras, ultrasound, inertial platform, and other devices. The navigation algorithm uses an extended-Kalman-filter (EKF) sensor-fusion formulation, integrating odometry and inertial sensors. When the work area is cluttered or the floor is not level, combinations of mobility methods may be needed. For example, Michaud discuss a robot with legs for climbing over obstacles or changing robot height combined with tracks for mobility on hard or soft surfaces. Autonomous control for this type of tracked mobile robot is discussed by Mihankhah for navigating and traversing obstacles (e.g., stairs). These types of robots could provide material handling or mobile manipulation in highly unstructured environments, such as shipyard dry docks, aircraft manufacturing, or other large,small-batch manufacturing projects. An alternative to a traditional mobile robot with onboard manipulator is described by Yang .It consists of a four-legged, parallel robot with clamping devices at the end of each leg. A set of supporting pins is placed in the work environment at known locations. The robot moves by detaching a leg from one pin and attaching it to another, thus always accurately knowing its position. It is able to climb, so does not have to remain on a flat surface. The legs do not all have to be clamped and the platform mounted on the legs can carry and manipulate tools to perform work when reaching its destination.

**2.5. Summary and Conclusions**

The field of mobile robotics is much larger than what has been described in this document. It covers autonomous driving on roads and across country, flying and water-based mobile robots, and a range of indoor applications that are not related to manufacturing. Historically, research in the United States has focused largely on areas of interest to the military and emergency services because that is where funding for research has been available. More recently, interest has been growing in mobile robots to assist people or provide services because there is a perception that robotic solutions might be commercially viable. Research in Europe has been more varied and has addressed more of the manufacturing needs, while Japan has focused, until recently, on humanoid robots and Australia has conducted substantial work in mining and agriculture. All of these strands of research are starting to be combined into systems with greater capabilities both for movement and autonomous action. As a result, it can be expected that the number of mobile robots in manufacturing will increase and the tasks that they will be expected to accomplish will become more complex. With a parallel increase in sensor processing capabilities and hardware robustness, it will become more common for people and robots to interact in a common workspace. A range of manufacturing applications will be made possible that are currently very difficult or expensive to achieve. For example, instead of requiring large, custom machine tools to fabricate large components, it will be possible to move smaller, general-purpose tools around the component and fabricate it in a new way. This will require highly-accurate position measurements, but such tools already exist and have started to be applied in robotics applications. Another advantage of not requiring large “monument” machine tools is more flexibility in arranging the assembly line and, ultimately, enabling dynamic reconfiguration as the product mix changes. Other advantages of using mobile robots include the ability to offload dangerous or ergonomically-challenging tasks from people and to automate tedious tasks such as kitting and palletizing. Before these capabilities can reach the marketplace, however, vendors will have to be able to guarantee the specifications and range of application areas of their products, and purchasers will want ways of comparing products and determining which are most suited to their needs. This will require performance metrics and procedures that are currently in their infancy. There will also be the need to program the tasks the robots will carry out in an easy and flexible manner, to be able to change tasks rapidly as the product mix changes, and to deal with the much less constrained work environments that inevitably accompany people working alongside robots. Standards will also have to be enhanced and harmonized, especially when mobile robots incorporate manipulators and dexterous end-effectors.

**CHAPTER 3**

# COMPONENTS AND DESCRIPTION

1. CHASSIS
2. DC MOTOR
3. LINKS
4. ARDUINO MICROCONTROLLER
5. WEMOS D1
6. RELAY SWITCHES
7. GRIPPER
8. FASTENERS
9. MOBILE APPLICATION
10. CONNECTING WIRES
11. SOFTWARE REQUIRED

**3.1.CHASSIS**

Chassis is the underpart of a motor vehicle, consisting of the frame (on which the body is mounted). If the running gear such as wheels and transmission, and sometimes even the driver's seat, are included.

**3.2.DC MOTOR**

A **DC motor** is any of a class of rotary electrical machines that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current flow in part of the motor.

DC motors were the first type widely used, since they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances. The universal motor can operate on direct current but is a lightweight motor used for portable power tools and appliances. Larger DC motors are used in propulsion of electric vehicles, elevator and hoists, or in drives for steel rolling mills. The advent of power electronics has made replacement of DC motors with AC motors possible in many applications.



Fig.3.1.DC Stepper motor

**3.2.1.BRUSHED**

The [brushed DC electric motor](https://en.wikipedia.org/wiki/Brushed_DC_electric_motor) generates torque directly from DC power supplied to the motor by using internal commutation, stationary magnets  ([permanent](https://en.wikipedia.org/wiki/Magnet) or [electromagnets](https://en.wikipedia.org/wiki/Electromagnet)), and rotating electromagnets.

Advantages of a brushed DC motor include low initial cost, high reliability, and simple control of motor speed. Disadvantages are high maintenance and low life-span for high intensity uses. Maintenance involves regularly replacing the carbon brushes and springs which carry the electric current, as well as cleaning or replacing the [commutator](https://en.wikipedia.org/wiki/Commutator_(electric)). These components are necessary for transferring electrical power from outside the motor to the spinning wire windings of the rotor inside the motor.

Brushes are usually made of graphite or carbon, sometimes with added dispersed copper to improve conductivity. In use, the soft brush material wears to fit the diameter of the commutator, and continues to wear. A brush holder has a spring to maintain pressure on the brush as it shortens. For brushes intended to carry more than an ampere or two, a flying lead will be molded into the brush and connected to the motor terminals. Very small brushes may rely on sliding contact with a metal brush holder to carry current into the brush, or may rely on a contact spring pressing on the end of the brush. The brushes in very small, short-lived motors, such as are used in toys, may be made of a folded strip of metal that contacts the commutator.

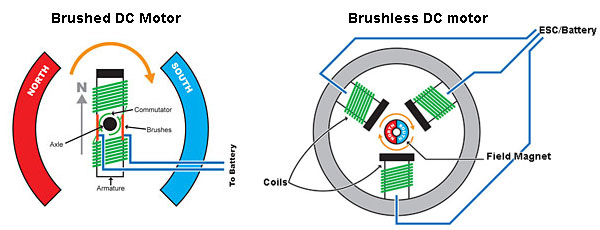


Fig.3.2.Brushed and Brushless DC Motor

**3.2.2.BRUSHLESS**

Typical brushless DC motors use one or more permanent magnets in the rotor and [electromagnets](https://en.wikipedia.org/wiki/Electromagnet) on the motor housing for the stator. A motor controller converts DC to [AC](https://en.wikipedia.org/wiki/Alternating_current). This design is mechanically simpler than that of brushed motors because it eliminates the complication of transferring power from outside the motor to the spinning rotor. The motor controller can sense the rotor's position via [Hall effect](https://en.wikipedia.org/wiki/Hall_effect) sensors or similar devices and can precisely control the timing, phase, etc., of the current in the rotor coils to optimize torque, conserve power, regulate speed, and even apply some braking. Advantages of brushless motors include long life span, little or no maintenance, and high efficiency. Disadvantages include high initial cost, and more complicated motor speed controllers. Some such brushless motors are sometimes referred to as "synchronous motors" although they have no external power supply to be synchronized with, as would be the case with normal AC synchronous motors.

**3.2.3.UNCOMMUTATED**

* [Homopolar motor](https://en.wikipedia.org/wiki/Homopolar_motor) – A homopolar motor has a magnetic field along the axis of rotation and an electric current that at some point is not parallel to the magnetic field. The name homopolar refers to the absence of polarity change. Homopolar motors necessarily have a single-turn coil, which limits them to very low voltages. This has restricted the practical application of this type of motor.
* [Ball bearing motor](https://en.wikipedia.org/wiki/Ball_bearing_motor) – A ball bearing motor is an unusual electric motor that consists of two [ball bearing](https://en.wikipedia.org/wiki/Ball_bearing)-type bearings, with the inner races mounted on a common conductive shaft, and the outer races connected to a high current, low voltage power supply. An alternative construction fits the outer races inside a metal tube, while the inner races are mounted on a shaft with a non-conductive section (e.g. two sleeves on an insulating rod). This method has the advantage that the tube will act as a flywheel. The direction of rotation is determined by the initial spin which is usually required to get it going.

**3.2.4.PERMANENT MAGNET STARTOR**

A PM motor does not have a field winding on the stator frame, instead relying on PMs to provide the magnetic field against which the rotor field interacts to produce torque. Compensating windings in series with the armature may be used on large motors to improve commutation under load. Because this field is fixed, it cannot be adjusted for speed control. PM fields (stators) are convenient in miniature motors to eliminate the power consumption of the field winding. Most larger DC motors are of the "dynamo" type, which have stator windings. Historically, PMs could not be made to retain high flux if they were disassembled; field windings were more practical to obtain the needed amount of flux. However, large PMs are costly, as well as dangerous and difficult to assemble; this favors wound fields for large machines.

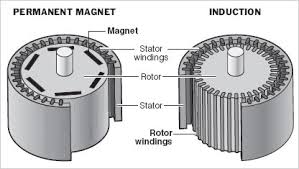
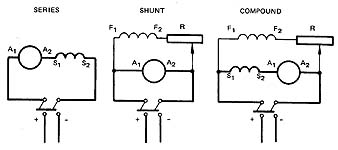


Fig.3.3.Permanent magnet and induction motor

To minimize overall weight and size, miniature PM motors may use high energy magnets made with [neodymium](https://en.wikipedia.org/wiki/Neodymium) or other strategic elements; most such are neodymium-iron-boron alloy. With their higher flux density, electric machines with high-energy PMs are at least competitive with all optimally designed [singly fed](https://en.wikipedia.org/wiki/DC_motor#Singly_fed_electric_motor) synchronous and induction electric machines. Miniature motors resemble the structure in the illustration, except that they have at least three rotor poles (to ensure starting, regardless of rotor position) and their outer housing is a steel tube that magnetically links the exteriors of the curved field magnets.

**3.2.5.WOUND STARTER**

Wound (various blends of series and shunt/parallel) and each has unique speed/torque characteristics appropriate for different loading torque profiles/signatures. 

### Fig.3.4.Series, Shunt, Compound connection

### 3.2.5.1.SERIES CONNECTION

A series DC motor connects the [armature](https://en.wikipedia.org/wiki/Armature_(electrical_engineering)) and [field windings](https://en.wikipedia.org/wiki/Field_coil) in [series](https://en.wikipedia.org/wiki/Series_circuits) with a [common](https://en.wikipedia.org/wiki/Battery_(electricity)) D.C. power source. The motor speed varies as a non-linear function of load torque and armature current; current is common to both the stator and rotor yielding current squared (I^2) behavior[[citation needed](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]. A series motor has very high starting torque and is commonly used for starting high inertia loads, such as trains, elevators or hoists. This speed/torque characteristic is useful in applications such as[dragline excavators](https://en.wikipedia.org/wiki/Dragline_excavator), where the digging tool moves rapidly when unloaded but slowly when carrying a heavy load.

A series motor should never be started at no load. With no mechanical load on the series motor, the current is low, the counter-Electro motive force produced by the field winding is weak, and so the armature must turn faster to produce sufficient counter-EMF to balance the supply voltage. The motor can be damaged by over speed. This is called a runaway condition.

Series motors called [universal motors](https://en.wikipedia.org/wiki/Universal_motor) can be used on [alternating current](https://en.wikipedia.org/wiki/Alternating_current). Since the armature voltage and the field direction reverse at the same time, torque continues to be produced in the same direction. However they run at a lower speed with lower torque on AC supply when compared to DC due to [reactance](https://en.wikipedia.org/wiki/Electrical_reactance) voltage drop in AC which is not present in DC. Since the speed is not related to the line frequency, universal motors can develop higher-than-synchronous speeds, making them lighter than induction motors of the same rated mechanical output. This is a valuable characteristic for hand-held power tools. Universal motors for commercial [utility](https://en.wikipedia.org/wiki/Utility_frequency) are usually of small capacity, not more than about 1 kW output. However, much larger universal motors were used for electric locomotives, fed by special low-frequency [traction power networks](https://en.wikipedia.org/wiki/Traction_power_network) to avoid problems with commutation under heavy and varying loads.

### 3.2.5.2.SHUNT CONNECTION

A shunt DC motor connects the armature and field windings in parallel or shunt with a common D.C. power source. This type of motor has good speed regulation even as the load varies, but does not have the starting torque of a series DC motor. It is typically used for industrial, adjustable speed applications, such as machine tools, winding/unwinding machines and tensioners.

### 3.2.5.3. COMPOUND CONNECTION

A compound DC motor connects the armature and fields windings in a shunt and a series combination to give it characteristics of both a shunt and a series DC motor. This motor is used when both a high starting torque and good speed regulation is needed. The motor can be connected in two arrangements: cumulatively or differentially. Cumulative compound motors connect the series field to aid the shunt field, which provides higher starting torque but less speed regulation. Differential compound DC motors have good speed regulation and are typically operated at constant speed.

****

Fig.3.5.DC Motor

**3.3.LINKS & JOINTS**

### Different types of robot joints:

### The Robot Joints is the important element in a robot which helps the links to travel in different kind of movements. There are five major types of joints such as:

* Rotational joint
* Linear joint
* Twisting joint
* Orthogonal joint
* Revolving joint

### mechatronics-4-8-638.jpg

### Fig.3.6.Various types of joints

### 3.3.1. ROTATIONAL JOINT

Rotational joint can also be represented as R – Joint. This type will allow the joints to move in arotary motion along the axis, which is vertical to the arm axes.

### 3.3.2. LINEAR JOINT

Linear joint can be indicated by the letter L – Joint. This type of joints can perform both translational and sliding movements. These motions will be attained by several ways such as telescoping mechanism and piston. The two links should be in parallel axes for achieving the linear movement.

### IMG_20180329_121512.jpg

### Fig.3.7.Robotic arm link

### 3.3.3. TWISTING JOINT

Twisting joint will be referred as V – Joint. This joint makes twisting motion among the output and input link. During this process, the output link axis will be vertical to the rotational axis. The output link rotates in relation to the input link.

### 3.3.4. ORTHOGONAL JOINT

The O – joint is a symbol that is denoted for the orthogonal joint. This joint is somewhat similar to the linear joint. The only difference is that the output and input links will be moving at the right angles.

### 3.3.5. REVOLVING JOINT

Revolving joint is generally known as V – Joint. Here, the output link axis is perpendicular to the rotational axis, and the input link is parallel to the rotational axes. As like twisting joint, the output link spins about the input link.

**4.ARDUINO MICROCONTROLLER**

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 (datasheet). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

**3.4.1.Microcontroller ATmega2560**

* Operating Voltage 5V
* Input Voltage (recommended) 7-12V
* Input Voltage (limits) 6-20V
* Digital I/O Pins 54 (of which 14 provide PWM output)
* Analog Input Pins 16
* DC Current per I/O Pin 40 mA
* DC Current for 3.3V Pin 50 mA
* Flash Memory 256 KB of which 8 KB used by bootloader
* SRAM 8 KB
* EEPROM 4 KB
* Clock Speed 16 MHz

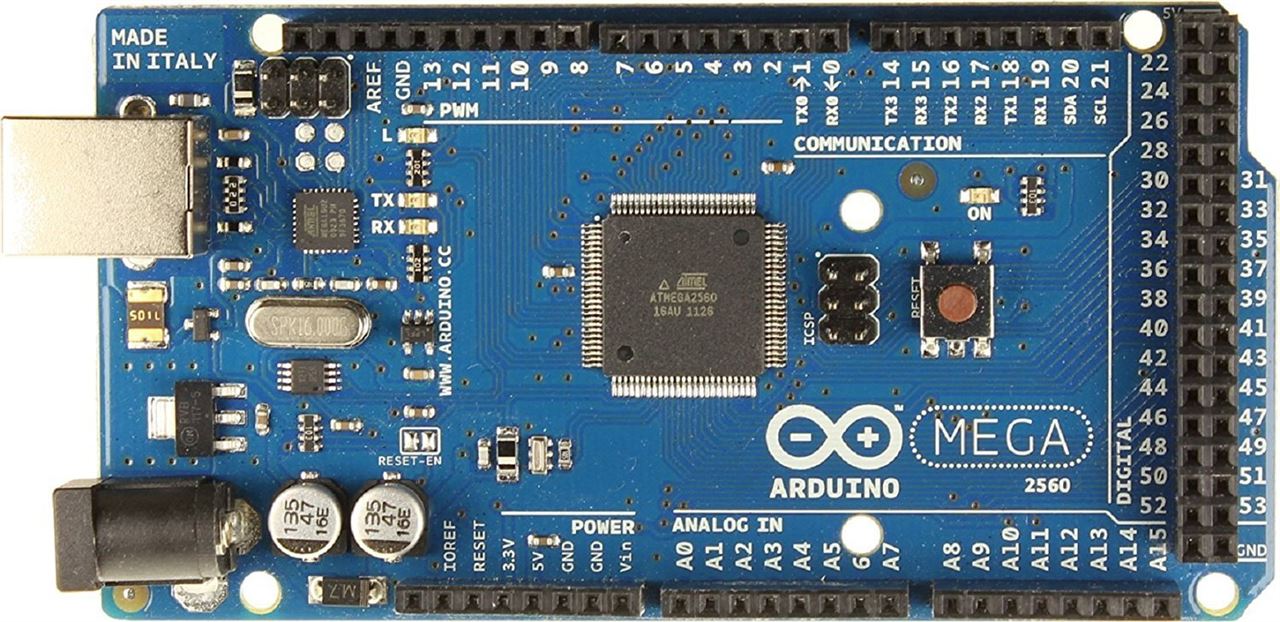
****

Fig.3.8.Arduino Mega2560 Microcontroller

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog [input/output](https://en.wikipedia.org/wiki/Input/output) (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The boards feature serial communications interfaces, including [Universal Serial Bus](https://en.wikipedia.org/wiki/Universal_Serial_Bus) (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers are typically programmed using a dialect of features from the programming languages [C](https://en.wikipedia.org/wiki/C_%28programming_language%29) and [C++](https://en.wikipedia.org/wiki/C%2B%2B). In addition to using traditional compiler toolchains, the Arduino project provides an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) based on the [Processing](https://en.wikipedia.org/wiki/Processing_%28programming_language%29) language project.

**3.5.WEMOS WIFI SHIELD**

It is similar to Arduino microcontroller

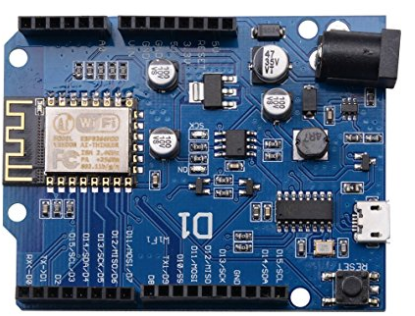


Fig.3.9.Wemos Wifi shield

* Based on the ESP-8266EX.
* Arduino Compatible, you can use it on Arduino IDE.
* 11 Digital I/O pins
* 1 Analog Input pin
* OTA -- Wireless Upload（Program）
* On board switching power supply -- Max 24V input, 5V 1A output

**3.6. RELAY SWITCHES**

A relay is an [electromagnetic](http://www.explainthatstuff.com/magnetism.html) switch operated by a relatively small [electric](http://www.explainthatstuff.com/electricity.html) current that can turn on or off a much larger electric current. The heart of a relay is an electromagnet (a coil of wire that becomes a temporary [magnet](http://www.explainthatstuff.com/magnetism.html) when electricity flows through it).

You can think of a relay as a kind of electric [lever](http://www.explainthatstuff.com/toolsmachines.html): switch it on with a tiny current and it switches on ("leverages") another appliance using a much bigger current. Why is that useful? As the name suggests, many sensors are incredibly *sensitive* pieces of [electronic](http://www.explainthatstuff.com/electronics.html) equipment and produce only small electric currents. But often we need them to drive bigger pieces of apparatus that use bigger currents.

Relays bridge the gap, making it possible for small currents to activate larger ones. That means relays can work either as switches (turning things on and off) or as amplifiers (converting small currents into larger ones).

A relay may be used in circuits where it is not possible to have a direct electrical connection between the control circuit and the output device: eg due to the risk of damage to the circuit from back electromotive force (EMF).

A relay is a special type of switch turned on and off by an electromagnet (see the diagram of a simple relay). When a current flows through the coil an electro-magnetic field is set up.

The field attracts an iron armature, whose other end pushes the contacts together, completing the circuit. When the current is switched off, the contacts open again, switching the circuit off.

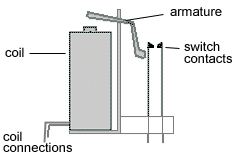
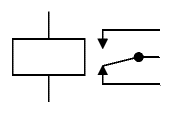
 

Fig.3.10.Circuit of Relay

A useful property of relays is that the circuit powering the coil is completely separate from the circuit switched on by the relay. For this reason relays are used where a safe low-voltage circuit controls a high-voltage circuit.

The symbol for a relay makes the separation of the two circuits clear by separating the coil symbol from the switch symbol.

The relays used in schools generally have a voltage between 6V and 15V. When using a diode with a relay, be sure that it is connected across the poles (terminals) of the relay to prevent damage to the transistor from back EMF, generated when the relay coil switches off.

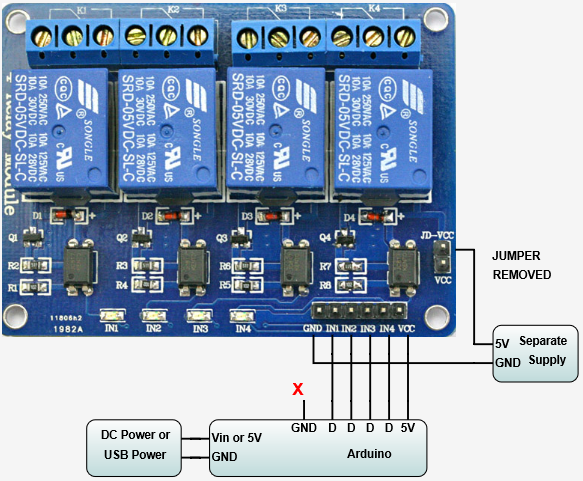


Fig.3.11.Relay Switches

**7.GRIPPER**

In robotics, an end effector is the device at the end of a [robotic](https://en.wikipedia.org/wiki/Robotic) arm, designed to interact with the environment. The exact nature of this device depends on the application of the robot.

In the strict definition, which originates from serial robotic [manipulators](https://en.wikipedia.org/wiki/Manipulator_(device)), the end effector means the last link (or end) of the robot. At this endpoint the [tools](https://en.wikipedia.org/wiki/Tool) are attached. In a wider sense, an end effector can be seen as the part of a robot that interacts with the work environment. This does not refer to the wheels of a [mobile robot](https://en.wikipedia.org/wiki/Mobile_robot) or the feet of a [humanoid robot](https://en.wikipedia.org/wiki/Humanoid_robot) which are also not end effectors—they are part of the robot's mobility.

End effectors may consist of a gripper or a tool. When referring to robotic prehension there are four general categories of robot grippers, these are:

1. Impactive – jaws or claws which physically grasp by direct impact upon the object.
2. Ingressive – pins, needles or hackles which physically penetrate the surface of the object (used in textile, carbon and glass fibre handling).
3. Astrictive – suction forces applied to the objects surface (whether by vacuum, magneto- or [electroadhesion](https://en.wikipedia.org/wiki/Electroadhesion)).
4. Contigutive – requiring direct contact for adhesion to take place (such as glue, [surface tension](https://en.wikipedia.org/wiki/Surface_tension) or freezing).

They are based on different physical effects used to guarantee a stable grasping between a gripper and the object to be grasped. Industrial grippers can be mechanical, the most diffused in industry, but also based on suction or on the magnetic force. Vacuum cups and electromagnets dominate the automotive field and in particular metal sheet handling. Bernoulli grippers exploit the airflow between the gripper and the part that causes a lifting force which brings the gripper and part close each other (i.e. the [Bernoulli's principle](https://en.wikipedia.org/wiki/Bernoulli%27s_principle)). Bernoulli grippers are a type of contactless grippers, namely the object remains confined in the force field generated by the gripper without coming into direct contact with it. Bernoulli grippers have been adopted in photovoltaic cell handling, [silicon wafer](https://en.wikipedia.org/wiki/Silicon_wafer) handling, and also in the textile and leather industries. Other principles are less used at the macro scale (part size >5mm), but in the last ten years they demonstrated interesting applications in micro-handling. Some of them are ready of spreading out their original field. The other adopted principles are: Electrostatic grippers and van der Waals grippers based on electrostatic charges (i.e. [van der Waals' force](https://en.wikipedia.org/wiki/Van_der_Waals%27_force)), capillary grippers and cryogenic grippers, based on liquid medium, and ultrasonic grippers and laser grippers, two contactless grasping principles. Electrostatic grippers are based on charge difference between the gripper and the part (i.e. [electrostatic force](https://en.wikipedia.org/wiki/Electrostatic_force)) often activated by the gripper itself, while van der Waals grippers are based on the low force (still electrostatic) due to the atomic attraction between the molecules of the gripper and those of the object. Capillary grippers use the surface tension of a liquid meniscus between the gripper and the part to center, align and grasp the part, cryogenic grippers freeze a small amount of liquid and the resulting ice guarantees the necessary force to lift and handle the object (this principle is used also in food handling and in textile grasping). Even more complex are [ultrasonic](https://en.wikipedia.org/wiki/Ultrasound) based grippers, where pressure [standing waves](https://en.wikipedia.org/wiki/Standing_wave) are used to lift up a part and trap it at a certain level (example of levitation are both at the micro level, in screw and gasket handling, and at the macro scale, in solar cell or silicon wafer handling), and laser source that produces a pressure able to trap and move microparts in a liquid medium (mainly cells). The laser gripper are known also as [laser tweezers](https://en.wikipedia.org/wiki/Laser_tweezers).

A particular category of friction/jaw gripper are the needle grippers: they are called intrusive grippers and exploits both friction and form closure as standard mechanical grippers.

The most known mechanical gripper can be of two, three or even five fingers.

The end effectors that can be used as tools serve various purposes, such as spot welding in an assembly, spray painting where uniformity of painting is necessary, and for other purposes where the working conditions are dangerous for human beings. Surgical robots have end effectors that are specifically manufactured for the purpose. They are very helpful according to many sources

****

Fig.3.12.Gripper

**8.FASTENER**

It is a [hardware](https://en.wikipedia.org/wiki/Household_hardware) device that mechanically joins or affixes two or more objects together. In general, fasteners are used to create non-permanent joints; that is, joints that can be removed or dismantled without damaging the joining components. [Welding](https://en.wikipedia.org/wiki/Welding) is an example of creating permanent joints.

Other alternative methods of joining materials include: [crimping](https://en.wikipedia.org/wiki/Crimp_(joining)), [welding](https://en.wikipedia.org/wiki/Welding), [soldering](https://en.wikipedia.org/wiki/Soldering), [brazing](https://en.wikipedia.org/wiki/Brazing), [taping](https://en.wikipedia.org/wiki/Adhesive_tape), [gluing](https://en.wikipedia.org/wiki/Adhesive), [cementing](https://en.wikipedia.org/wiki/Cement), or the use of other adhesives. The use of [force](https://en.wikipedia.org/wiki/Force) may also be used, such as with [magnets](https://en.wikipedia.org/wiki/Magnet), [vacuum](https://en.wikipedia.org/wiki/Vacuum) (like [suction cups](https://en.wikipedia.org/wiki/Suction_cups)), or even [friction](https://en.wikipedia.org/wiki/Friction) (like[sticky pads](https://en.wikipedia.org/wiki/Sticky_pad)). Some types of [woodworking joints](https://en.wikipedia.org/wiki/Woodworking_joints) make use of separate internal reinforcements, such as [dowels](https://en.wikipedia.org/wiki/Dowel) or [biscuits](https://en.wikipedia.org/wiki/Biscuit_joiner), which in a sense can be considered fasteners within the scope of the joint system, although on their own they are not general purpose fasteners.

Furniture supplied in [flat-pack](https://en.wikipedia.org/wiki/Flat-pack) form often uses [cam dowels](https://en.wikipedia.org/wiki/Dowel) locked by [cam locks](https://en.wikipedia.org/wiki/Cam_lock_(latch)), also known as conformat fasteners. Fasteners can also be used to close a container such as a bag, a box, or an envelope; or they may involve keeping together the sides of an opening of flexible material, attaching a [lid](https://en.wikipedia.org/wiki/Lid_(container)) to a container, etc. There are also special-purpose closing devices, e.g. a [bread clip](https://en.wikipedia.org/wiki/Bread_clip).

Items like a [rope](https://en.wikipedia.org/wiki/Rope), string, [wire](https://en.wikipedia.org/wiki/Wire) (e.g. metal wire, possibly coated with plastic, or multiple parallel wires kept together by a plastic strip coating), [cable](https://en.wikipedia.org/wiki/Wire_rope), [chain](https://en.wikipedia.org/wiki/Chain), or [plastic wrap](https://en.wikipedia.org/wiki/Plastic_wrap) may be used to mechanically join objects; but are not generally categorized as fasteners because they have additional common uses. Likewise, [hinges](https://en.wikipedia.org/wiki/Hinge) and [springs](https://en.wikipedia.org/wiki/Spring_(device)) may join objects together, but are ordinarily not considered fasteners because their primary purpose is to allow articulation rather than rigid affixment.

****

Fig.3.13.Fasteners

**9.MOBILE APPLICATION**

Blynk is a Platform with iOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet .It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets.It's really simple to set everything up and you'll [start tinkering](http://www.blynk.cc/getting-started) in less than 5 mins. Blynk  is not tied to some specific board or shield. Instead, it's supporting hardware of your choice.  Arduino linked to the Internet over Wi-Fi, Ethernet or this new ESP8266 chip, Blynk will get you online and ready for the **Internet Of Things.**

******

Fig.3.14.Blynk Application

**10.CONNECTING WIRES**

A wire is a single, usually [cylindrical](https://en.wikipedia.org/wiki/Cylinder_(geometry)), flexible strand or rod of metal. Wires are used to bear mechanical [loads](https://en.wikipedia.org/wiki/Structural_load) or [electricity](https://en.wikipedia.org/wiki/Electricity) and [telecommunications signals](https://en.wikipedia.org/wiki/Signal_(electronics)). Wire is commonly formed by [drawing](https://en.wikipedia.org/wiki/Drawing_(manufacturing)) the metal through a hole in a [die](https://en.wikipedia.org/wiki/Die_(manufacturing)) or [draw plate](https://en.wikipedia.org/wiki/Draw_plate). [Wire gauges](https://en.wikipedia.org/wiki/Wire_gauge) come in various [standard](https://en.wikipedia.org/wiki/Standardisation) sizes, as expressed in terms of a [gauge number](https://en.wikipedia.org/wiki/American_wire_gauge). The term wire is also used more loosely to refer to a bundle of such strands, as in "multi stranded wire", which is more correctly termed a [wire rope](https://en.wikipedia.org/wiki/Wire_rope) in mechanics, or a [cable](https://en.wikipedia.org/wiki/Electrical_cable) in electricity.

Wire comes in solid core, stranded, or braided forms. Although usually circular in cross-section, wire can be made in square, hexagonal, flattened rectangular or other cross-sections, either for decorative purposes, or for technical purposes such as high-efficiency collision [loudspeakers](https://en.wikipedia.org/wiki/Loudspeaker). Edge-wound [coil springs](https://en.wikipedia.org/wiki/Coil_spring), such as the [Slinky](https://en.wikipedia.org/wiki/Slinky) toy, are made of special flattened wire.



Fig.3.15.Connecting Wires

**11.SOFTWARE REQUIRED**

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

### ARDUINO-Windows-Store.png

### Fig.3.16.Arduino IDE

### 11.1. WRITING SKETCHES

Programs written using Arduino Software (IDE) are called **sketches**. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

|  |  |
| --- | --- |
| https://www.arduino.cc/en/uploads/Guide/IDE_VERIFY_File.jpg | Verify  Checks your code for errors compiling it. |
| https://www.arduino.cc/en/uploads/Guide/IDE_UPLOAD_File.jpg | Upload  Compiles your code and uploads it to the configured board. See [uploading](https://www.arduino.cc/en/Guide/Environment#uploading) below for details.  Note: If you are using an external programmer with your board, you can hold down the "shift" key on your computer when using this icon. The text will change to "Upload using Programmer" |
| https://www.arduino.cc/en/uploads/Guide/IDE_NEW_File.jpg | New  Creates a new sketch. |
| https://www.arduino.cc/en/uploads/Guide/IDE_OPEN_File.jpg | Open  Presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.  Note: due to a bug in Java, this menu doesn't scroll; if you need to open a sketch late in the list, use the **File | Sketchbook** menu instead. |
| https://www.arduino.cc/en/uploads/Guide/IDE_SAVE_File.jpg | Save  Saves your sketch. |
| https://www.arduino.cc/en/uploads/Guide/IDE_SERMON_File.jpg | SerialMonitor  Opens the [serial monitor](https://www.arduino.cc/en/Guide/Environment#serialmonitor). |

Additional commands are found within the five menus: **File**, **Edit**, **Sketch**, **Tools**, **Help**. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

* **New**  
  Creates a new instance of the editor, with the bare minimum structure of a sketch already in place.
* **Open**  
  Allows to load a sketch file browsing through the computer drives and folders.
* **Open Recent**   
  Provides a short list of the most recent sketches, ready to be opened.
* **Sketchbook**  
  Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.
* **Examples**  
  Any example provided by the Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.
* **Close**  
  Closes the instance of the Arduino Software from which it is clicked.
* **Save**  
  Saves the sketch with the current name. If the file hasn't been named before, a name will be provided in a "Save as.." window.
* **Save as...**   
  Allows to save the current sketch with a different name.
* **Page Setup**   
  It shows the Page Setup window for printing.
* **Print**  
  Sends the current sketch to the printer according to the settings defined in Page Setup.
* **Preferences**   
  Opens the Preferences window where some settings of the IDE may be customized, as the language of the IDE interface.
* **Quit**Closes all IDE windows. The same sketches open when Quit was chosen will be automatically reopened the next time you start the IDE.

**CHAPTER 4**

**SYSTEM STUDY**

**4.1. FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the industry. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* Economical Feasibility
* Technical Feasibility
* Social Feasibility

**4.1.1. ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system and the small scale industry also. The amount of fund that the user can pour into this project to implement in his field and development of the system is limited. The expenditures must be justified. Thus the system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

**4.1.2 .TECHNICAL FEASIBILITY**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**4.1.3. SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system .

**CHAPTER 5**

**DESIGN AND IMPLEMENTATION:**

The mechanical construction in this project is to build and assemble the robotic arm body. After giving a thorough consideration of all the preceding works in this field, a four degree of freedom manipulator having variable programmed motions to carry out variety of tasks in diverse environments is chosen. This is a four axis articulate manipulator designed to move material like machine parts, tools, specialized devices, etc It is driven by four dc motors and has a gripper as end-effectors. The gripper has fingers grasping and manipulation of objects as big as a 70 mm dia cylindrical bottle and having a weight of about 1000 gm throughout the arm’s workspace.



Fig.5.1.Pick and place robot

**5.1.DESIGN CONSIDERATION:**

The following were put into consideration in the design process.

i. Electrical actuators DC motor are chosen instead of hydraulic and pneumatic actuators because of the little power requirement and its light weight which is suitable for this design.

ii. Materials used for the fabrication were locally sourced from available materials.

iii. The materials which will be used for the design will be light in weight so as to reduce the weight concentration on the base and the shoulder.

iv. Cylindrical rods instead of blocks are chosen for the links because of their light weight and stability and to reduce the weight of the arm.

v. A Arduino microcontroller was chosen (AT Mega2560 microcontroller was used).

vi. The torque is fully balanced by the inertia of the electric motors.

**5.2.SELECTION OF DEGREE OF FREEDOM (DOF):**

The number of independents movements that an object can perform in 3D space is termed as the number of degree of freedom (DOF). Thus, a rigid body free in space has six degree of freedom – three for positions and three for orientations.

For our application 4 DOF are suitable.

**5.3.MATERIAL SELECTION:**

The most suitable material to fabricate the structure of the arm has to be light and strong. Otherwise, the servo motor will not be able to pull up the arm and to perform the desired turning degree. Among the materials that can be considered to fabricate the structure are iron, steel, aluminum, Perspex, plastic polymer and carbon fiber. In choosing the fabrication materials, the aspect of availability of the materials, the overall cost and the flexibility to be shaped, should also be taken into consideration. Thus among the four materials considered, the steel rod is the most ideal material to be chosen as fabrication material.

**5.4.SPECIFICATIONS OF ROBOTIC ARM:**

The robot arm for this project is the revolute type that closely resembles the human arm. Shoulder that mounted on base can move the arm through 180 degrees, from horizontal to vertical on each side. The shoulder uses large -scale servo, provide the torque needed to lift the rest of the arm, as well as any object that it may be grasping. Attached to the shoulder piece is an elbow that can move through 90 degrees, also powered by a large -scale servo. The wrist is made up of one standard servo and can move through 90 degrees, in vertical direction. Attached to the wrist is a two-fingered gripper that utilizes a unique design built around a single standard servo.

**Specification Value:**

**Number of axes** : 4

**Horizontal reach** : 250 mm

**Vertical reach** : 200 mm

**Drives** : 2 DC motors +2 Stepper motor

**Configuration** : 4 Axes plus gripper.

All axes completely independent.

All axes can be controlled simultaneously.

**Work Envelope**  : (a) Shoulder Rotation -180 degrees

(b) Elbow Rotation -90 degrees

(c) Wrist Rotation -90 degrees

(d) Gripper Rotation -90 degrees

**5.5.PROGRAMMING**

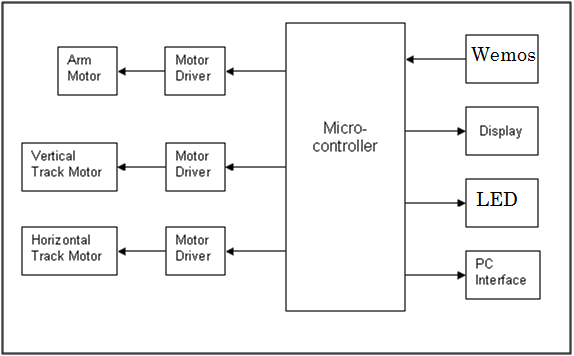
Programming of the Pick and place Robot is to be done using a suitable Programming Language. The Robot is to been interfaced with the computer by the programmed software, which will guide the robot to do its job for which it is been 

Fig.5.2.Basic circuit diagram

programmed .There are numbers of various programming languages available now a days in the market, so the appropriate programming language is to be selected for the programming purpose and the programming is to be done.

**5.6. INTERFACING WITH THE COMPUTER**

In the [industrial design](http://en.wikipedia.org/wiki/Industrial_design) field of [human-machine interaction](http://en.wikipedia.org/wiki/Human-computer_interaction), the user interface is where interaction between humans and machines occurs. The goal of interaction between a human and a machine at the user interface is effective operation and control of the machine, and feedback from the machine which aids the operator in making operational decisions.

A user interface is the system by which people ([users](http://en.wikipedia.org/wiki/User_%28computing%29)) [interact](http://en.wikipedia.org/wiki/Interaction) with a [machine](http://en.wikipedia.org/wiki/Machine). The user interface includes hardware (physical) and software (logical) components. User interfaces exist for various [systems](http://en.wikipedia.org/wiki/System), and provide a means of:

* Input, allowing the users to manipulate a system,
* Output, allowing the system to indicate the effects of the users' manipulation

After completion of the model of the pick and place robot and selection of programming language both should be interfaced. The interfacing of robot and computer using the software is the most important thing in the project. It should be interfaced using trial and error method, and then final movement should be set using the software’s. The movement of robot should be precisely managed causing no harm to the operator, and also the batteries which are to be moved from one station to another.

**5.7.PROGRAM FOR WEMOS WIFI SHIELD**

#define BLYNK\_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.

// Go to the Project Settings (nut icon).

char auth[] = "387534fe2c064d15929af879b71dd90e";

// Your WiFi credentials.

// Set password to "" for open networks.

char ssid[] = "madhu";

char pass[] = "madhusuthanan";

void setup()

{

Serial.begin(9600);

Blynk.begin(auth, ssid, pass);

}

void loop()

{

Blynk.run();

}

**5.8.PROGRAM FOR ARDUINO**

#include <LiquidCrystal.h>

const int rs = 30, en = 31, d4 = 32, d5 =33, d6 = 34, d7 = 35;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup()

{

pinMode(2,OUTPUT);

pinMode(12,OUTPUT);

pinMode(11,OUTPUT);

pinMode(10,OUTPUT);

pinMode(9,OUTPUT);

pinMode(4,OUTPUT);

pinMode(5,OUTPUT);

pinMode(6,OUTPUT);

pinMode(7,OUTPUT);

pinMode(22,OUTPUT);

pinMode(23,OUTPUT);

pinMode(24,OUTPUT);

pinMode(25,OUTPUT);

pinMode(26,OUTPUT);

pinMode(27,OUTPUT);

pinMode(28,OUTPUT);

pinMode(29,OUTPUT);

pinMode(38,OUTPUT);

pinMode(39,OUTPUT);

pinMode(40,OUTPUT);

pinMode(41,OUTPUT);

Serial.begin(9600);

lcd.begin(16, 2);

lcd.print("Welcome-Madhu'\_'");

}

void loop()

{

int a,b,c,d,e,f,g,h,i,j,k;

int sensorValue = analogRead(A0);

int sensorValue1 = analogRead(A1);

int sensorValue2 = analogRead(A2);

int sensorValue3 = analogRead(A3);

int sensorValue4 = analogRead(A4);

int sensorValue5 = analogRead(A5);

int sensorValue6 = analogRead(A6);

int sensorValue7 = analogRead(A7);

int sensorValue8 = analogRead(A8);

int sensorValue9 = analogRead(A9);

int sensorValue10 = analogRead(A10);

Serial.println(sensorValue);

Serial.println(sensorValue1);

Serial.println(sensorValue2);

Serial.println(sensorValue3);

Serial.println(sensorValue4);

Serial.println(sensorValue5);

Serial.println(sensorValue6);

Serial.println(sensorValue7);

Serial.println(sensorValue8);

Serial.println(sensorValue9);

Serial.println(sensorValue10);

delay(1);

a=sensorValue;

b=sensorValue1;

c=sensorValue2;

d=sensorValue3;

e=sensorValue4;

f=sensorValue5;

g=sensorValue6;

h=sensorValue7;

k=sensorValue8;

j=sensorValue9;

i=sensorValue10;

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

if(k<100)

{

lcd.setCursor(0, 1);

lcd.print("READY...............");

}

if(k<100&&h<100)

{

lcd.setCursor(0, 1);

lcd.print("Gripper...............");

}

if(k>100)

{

lcd.begin(16, 2);

lcd.print("Welcome-Madhu'\_'");

lcd.setCursor(0, 1);

lcd.print("Press key...............");

}

if(a<100&&k<100) //RIGHT TURN

{

digitalWrite(12,LOW); digitalWrite(4,HIGH);

digitalWrite(11,LOW); digitalWrite(5,HIGH);

digitalWrite(10,HIGH); digitalWrite(6,LOW);

digitalWrite(9,HIGH); digitalWrite(7,LOW);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

}

if(b<100&&k<100)//LEFT

{

digitalWrite(12,HIGH); digitalWrite(4,LOW);

digitalWrite(11,HIGH); digitalWrite(5,LOW);

digitalWrite(10,LOW); digitalWrite(6,HIGH);

digitalWrite(9,LOW); digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

}

if(a<100&&b<100&&k<100)//LEFT,RIGHT

{

digitalWrite(12,LOW); digitalWrite(4,LOW);

digitalWrite(11,LOW); digitalWrite(5,LOW);

digitalWrite(10,HIGH); digitalWrite(6,HIGH);

digitalWrite(9,HIGH); digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

}

if(c<100&&k<100)//REVERSE

{

digitalWrite(12,HIGH); digitalWrite(4,HIGH);

digitalWrite(11,HIGH); digitalWrite(5,HIGH);

digitalWrite(10,LOW); digitalWrite(6,LOW);

digitalWrite(9,LOW); digitalWrite(7,LOW);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

}

if(d<100&&k<100&&h>100)//1 MOTION down

{

digitalWrite(22,LOW); digitalWrite(26,HIGH);

digitalWrite(23,LOW); digitalWrite(27,HIGH);

digitalWrite(24,HIGH); digitalWrite(28,HIGH);

digitalWrite(25,HIGH); digitalWrite(29,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

lcd.setCursor(0, 1);

lcd.print("Motion-DOWN.......");

delay(5);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

digitalWrite(38,LOW);

digitalWrite(39,LOW);

digitalWrite(40,LOW);

digitalWrite(41,LOW);

delay(100);

lcd.setCursor(0, 1);

lcd.print("..............");

}

if(e<100&&k<100&&h>100)//1 MOTION-upp

{

digitalWrite(22,HIGH); digitalWrite(26,HIGH);

digitalWrite(23,HIGH); digitalWrite(27,HIGH);

digitalWrite(24,LOW); digitalWrite(28,HIGH);

digitalWrite(25,LOW); digitalWrite(29,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

lcd.setCursor(0, 1);

lcd.print("Motion-UPP.......");

delay(7);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

delay(70);

lcd.setCursor(0, 1);

lcd.print("..............");

}

if(f<100&&k<100&&h>100)//2 MOTION-front

{

digitalWrite(22,HIGH); digitalWrite(26,LOW);

digitalWrite(23,HIGH); digitalWrite(27,LOW);

digitalWrite(24,HIGH); digitalWrite(28,HIGH);

digitalWrite(25,HIGH); digitalWrite(29,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

lcd.setCursor(0, 1);

lcd.print("Motion-FRONT.......");

delay(10);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

delay(70);

lcd.setCursor(0, 1);

lcd.print(" ..............");

}

if(g<100&&k<100&&h>100)//2 MOTION-back

{

digitalWrite(22,HIGH); digitalWrite(26,HIGH);

digitalWrite(23,HIGH); digitalWrite(27,HIGH);

digitalWrite(24,HIGH); digitalWrite(28,LOW);

digitalWrite(25,HIGH); digitalWrite(29,LOW);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

lcd.setCursor(0, 1);

lcd.print("Motion-BACK.......");

delay(15);

digitalWrite(12,HIGH);

digitalWrite(11,HIGH);

digitalWrite(10,HIGH);

digitalWrite(9,HIGH);

digitalWrite(4,HIGH);

digitalWrite(5,HIGH);

digitalWrite(6,HIGH);

digitalWrite(7,HIGH);

digitalWrite(22,HIGH);

digitalWrite(23,HIGH);

digitalWrite(24,HIGH);

digitalWrite(25,HIGH);

digitalWrite(26,HIGH);

digitalWrite(27,HIGH);

digitalWrite(28,HIGH);

digitalWrite(29,HIGH);

delay(50);

lcd.setCursor(0, 1);

lcd.print("..............");

}

if(f<100&&k<100&&h<100)//2 MOTION-front

{

digitalWrite(38,HIGH);

delay(100);

digitalWrite(38,LOW);

}

if(g<100&&k<100&&h<100)//2 MOTION-back

{

digitalWrite(39,HIGH);

delay(100);

digitalWrite(39,LOW);

}

if(e<100&&k<100&&h<100)//1 MOTION-down

{

digitalWrite(41,HIGH);

delay(100);

digitalWrite(41,LOW);

}

if(d<100&&k<100&&h<100)//1 MOTION upp

{

digitalWrite(40,HIGH);

delay(100);

digitalWrite(40,LOW);

}

}//exits

**5.9.Mobile Application**

****

Fig.5.3.Mobile application keys

The following keys are used to control the pick and place material handling robot

**CHAPTER 6**

**CONCLUSION**

In this project we are designing and fabricating of a 4-DOF manipulator has been successfully completed. With reference to many available manipulators and mobile platforms in market, a practical design for the manipulator has been perceived .The load capacity of the robot is 1000gms. The accuracy rate of the project is 70% due to mechanical links.

**6.1.FUTURE IMPLEMENTATIONS**

We have done our best efforts to make the project feasible, simple

and reliable for the local industrial usage. There can be modification in

this robotic system that can be more efficient and effective like;

* The robotic system can be modified by implementing vision system and artificial intelligence to avoid obstacles in between the path.
* The microcontroller programming can be replaced by PLC, as a new technology.
* Making magnetic gripper more powerful can increase the payload property of the robot.
* The gripper can be modified for different operations in different industries.
* Advancement can be made in the robot by implanting sensors.

**BILL OF MATERIAL**

|  |  |  |  |
| --- | --- | --- | --- |
| **SL.NO** | **NAME OF MATERIAL** | **QUANTITY** | **PRICE AMOUNT**  **(Rupees)** |
| 1 | Wheels | 04 | 400 |
| 2 | DC stepper motor | 04 | 800 |
| 3 | DC motor | 02 | 1500 |
| 4 | Relay switch | 04 | 1000 |
| 5 | Wemos D1 wifi shield | 01 | 500 |
| 6 | Battery (12V rechargeable) | 01 | 700 |
| 7 | 19v DC adaptor | 01 | 500 |
| 8 | Connecting wires | 30 | 200 |
| 9 | Arduino Mega2560 | 01 | 950 |
| 10 | Steel rods | 10 | 100 |
| 11 | Fasteners | 50 | 500 |
| 12 | Wood plate | 02 | 200 |
| 13 | Wood | 01 | 100 |
| 14 | Mobile Application | 01 | 40 |
| 15 | Gripper | 01 | 1500 |
| 16 | L293Driver | 01 | 350 |

TOTAL: 9340

**REFERENCES**

[1] B.O. Omijeh, R. Uhunmwangho, M. Ehikhamenle, “Design Analysis of a Remote Controlled Pick and Place Robotic Vehicle”, International Journal of Engineering Research and Development, Volume 10, PP.57-68, May 2014.

[2] S. Berman, E. Schechtman, and Y. Edan, "Evaluation of automatic guided vehicle systems," Robotics and Computer-Integrated Manufacturing, vol. 25, pp. 522-528, 2009.

[3] M. Hvilshøj, S. Bøgh, O. Madsen, and M. Kristiansen, "The mobile robot "Little Helper": Concepts, ideas and working principles," presented at the IEEE Conference on Emerging Technologies & Factory Automation, ETFA 2009, 2009

[4] Mohamed Naufal Bin Omar, “Pick and Place Robotic Arm Controlled By Computer”, Faculty of Manufacturing Engineering, April 2007.

[5] Ankit Gupta, Mridul Gupta, NeelakshiBajpai, Pooja Gupta, Prashant Singh, “Efficient Design and Implementation of 4-Degree of Freedom Robotic Arm”, International Journal of Engineering and Advanced Technology (IJEAT)

ISSN: 2249 – 8958, Volume-2, June 2013.

[6] M. Luimula, K. Sääskilahti, T. Partala, S. Pieskä, and J. Alaspää, "Remote navigation of a mobile robot in an RFID-augmented environment," Pers Ubiquit Comput, vol. 14, pp. 125-136, 2010