A Project Report on

RASPBERRY PI BASED AGRICULTURAL ROBOT

Submitted in partial fulfillment of the requirements

for the award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS & COMMUNICATION ENGINEERING

(2015-2019)

By

MK. AISHWARYA	**(15241A04F5)**	**L. SAI AKSHAYA SRI**	**(15241A04M7)**
G. SUSHMA	**(15241A04J5)**	**S. TANUJA**	**(15241A04N2)**
P. PRIYANKA	**(15241A04M0)**	**T. SAMHITA**	**(15241A04N7)**

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution)

Nizampet, Bachupally, Hyderabad-500090

(Affiliated To JNTUH, Hyderabad, 2018-2019)

A Project Report on

RASPBERRY PI BASED AGRICULTURAL ROBOT

Submitted in partial fulfillment of the requirements

for the award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS & COMMUNICATION ENGINEERING (2015-2019)

By

MK. AISHWARYA	**(15241A04F5)**	**L. SAI AKSHAYA SRI**	**(15241A04M7)**
G. SUSHMA	**(15241A04J5)**	**S. TANUJA**	**(15241A04N2)**
P. PRIYANKA	**(15241A04M0)**	**T. SAMHITA**	**(15241A04N7)**

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(An Autonomous Institution, Affiliated To JNTUH, Hyderabad, 2018-19)

Nizampet, Bachupally, Hyderabad-500090

DEPARTMENT OF ELECTRONICS & COMMUNICATIONS ENGINEERING

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING & TECHNOLOGY

(AUTONOMOUS)

CERTIFICATE

This is to certify that the project entitled **"RASPBERRY PI BASED AGRICULTURAL ROBOT"**, been carried out by the following students:

MK. AISHWARYA	**(15241A04F5)**	**L. SAI AKSHAYA SRI**	**(15241A04M7)**
G. SUSHMA	**(15241A04J5)**	**S. TANUJA**	**(15241A04N2)**
P. PRIYANKA	**(15241A04M0)**	**T. SAMHITA**	**(15241A04N7)**

of IV year, II semester **B.Tech (Electronics & Communications Engineering)** under our supervision during the year 2018-2019 in partial fulfilment of the requirement for the award of Bachelor of Technology degree in Electronics & Communications Engineering, Jawaharlal Nehru Technological University, Hyderabad.

G.L.Sumalatha	**Dr. N. Swetha**
(Internal guide & Associate Professor)	**(Professor and HOD)**
Electronics & Communications Engg Dept.	**Electronics & Communications Engg Dept.**

ACKNOWLEDGEMENT

We wish to express our deep sense of gratitude to our project coordinator **G.L. Sumalatha** Associate Professor, for her able guidance and useful suggestions, which helped us in completing the project work in time.

We express our heartiest gratitude and respectful regards to **Dr. N. Swetha,** Professor and Head of Department of ECE, Gokaraju Rangaraju Institute of Engineering and Technology, for her encouragement in doing the project.

Last but not the least our special thanks to our parents and friends for their support and constant encouragement during the project work.

MK. Aishwarya

G. Sushma

P. Priyanka

L. Sai Akshaya Sri

S. Tanuja

T. Samhita

ABSTRACT

The lag between what is known and what is done by most of the farmers has been the focus of considerable research in recent years. It has been convinced that individuals that are involved in the farming industry are some of the least susceptible to change. They are very set in the ways of those came before them. When we take a look at the farming industry now, we can see that this is rapidly changing. Farmers are looking for new ways to implement technology which could be a replacement to human cultivation and also to cut costs and reduce labor hours.

As a part of this development, agricultural robots are the fastest growing technology which could help a farmer to perform various complex tasks that are difficult for humans to achieve. One of the ways that farmers are beginning to explore new technologies come from the autonomous agri-bot. In this regard, this project work is taken up, which is aimed to provide a farmer-friendly autonomous agri-bot.

The main objective of autonomous agri-bot is efficient utilization of resources and to reduce labor work. It can perform various tasks like soil testing, sowing of seeds, spraying of fertilizers and soil drilling. It can dig a hole in soil by drilling mechanisms and plants seed and cover hole by soil again. It can spray the pesticides using spraying mechanisms.

All the above operations are performed by using the Raspberry PI controller which is master and others are slaves which perform a specific operation. By using Wi-Fi, the robot can be controlled over large distance i.e. from anywhere in the world by using Raspberry. The robot takes advantage of sensors such as humidity & temperature sensor, soil moisture sensor attached to it.

INDEX

CHAPTER 1: INTRODUCTION

1.1 Introduction:

Increasing population requires the food production to be increased which requires better cultivation in the form of proper utilization of seeds and fertilizers with minimum labor work. The main objective of autonomous agri-bot is efficient utilization of resources and to reduce labor work. It can perform various tasks like soil testing, sowing of seeds, spraying of fertilizers and soil drilling. It can dig a hole in soil by drilling mechanisms and plants seed and cover hole by soil again. It can spray the pesticides using spraying mechanisms.

All above operations are performed by using Raspberry PI controller which is master and others are slaves which performs specific operation. By using Wi-Fi, the robot can be controlled over large distance i.e. from anywhere in the world by using Raspberry. The robot takes advantage of sensors such as humidity & temperature sensor, soil moisture sensor attached to it.

When the switch is on the robot will be connected to network to get inputs from the phone over which in turn provides the commands to the robot for different functionalities i.e. movement of robot, sowing of seeds, ploughing and for sprinkling the data will be displayed on LCD attached to it.

1.2 Project Overview:

An embedded system is a combination of software and hardware to perform a dedicated task. Some of the main devices used in embedded products are Microprocessors and Microcontrollers.

Microprocessors are commonly referred to as general purpose processors as they simply accept the inputs, process it and give the output. In contrast, a microcontroller not only accepts the data as inputs but also manipulates it, interfaces the data with various devices, controls the data and thus finally gives the result.

Agribot integrated system which uses Wi-Fi to communicate between two robots is presented in ,which perform activities like seeding, weeding, spraying of fertilizers and insecticides. It is controlled using powerful Raspberry pi minicomputer to control and monitor working of robot. It can move in any direction as per required. It has soil moisture sensor and humidity sensor integrated to it. It can spray fertilizer and dig a pit to sow a seed, plough the land and move on along with communicating with other robot near to it using Wi-Fi.

The controlling device of the whole system is a Microcontroller. After starting the robot the robot takes input from the mobile phone over internet the device then can be used to perform the operations of ploughing, sowing seeds and spraying fertilizers with the help of Raspberry Pi controller.

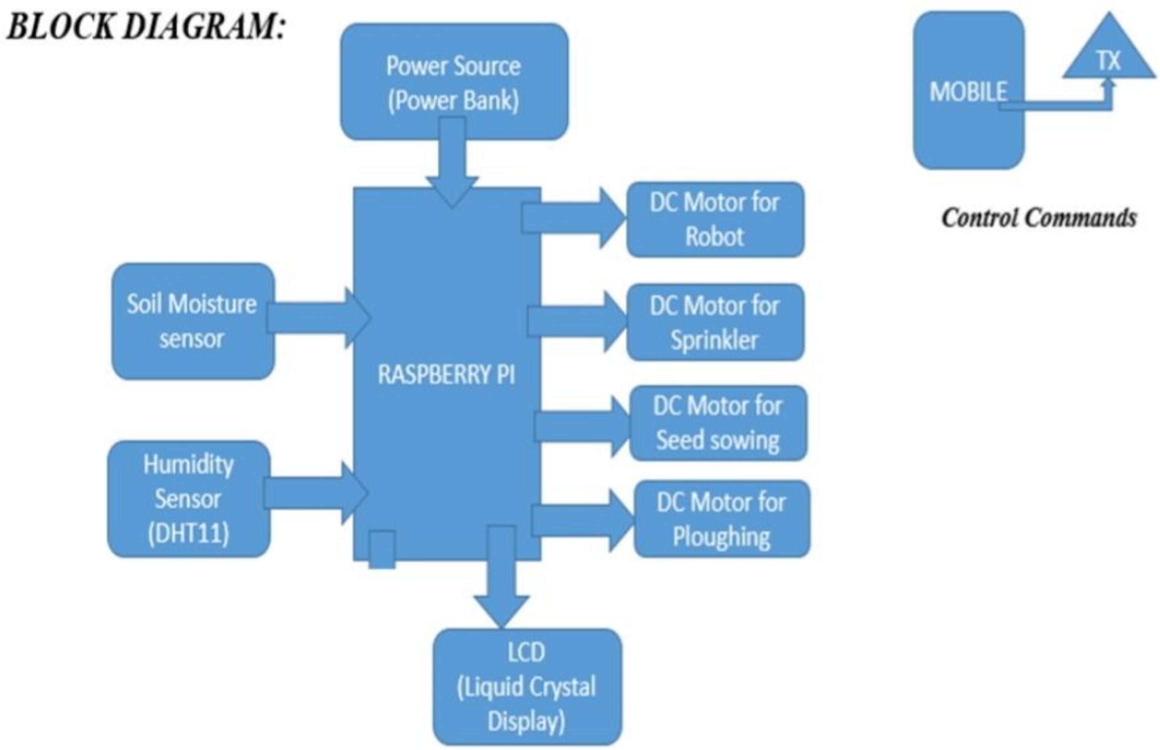


Fig 1.1: Block Diagram

1.3 Schematic diagram:

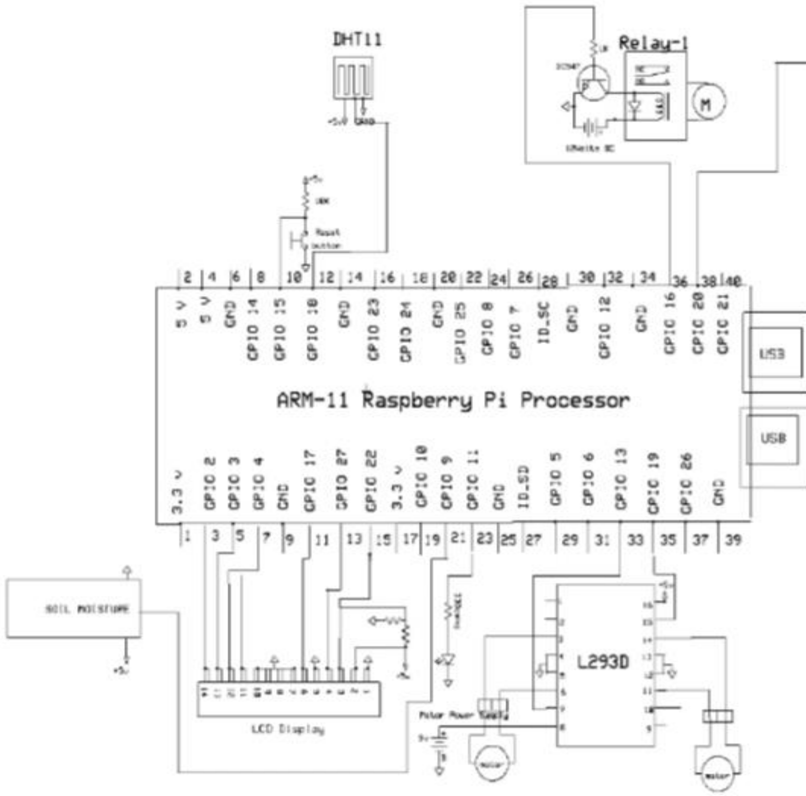


Fig 1.2: Schematic Diagram

The above schematic diagram of **Smart Irrigation System** interfacing section of each component with RASPBERRY PI3 PROCESSOR AND SENSORS.

CHAPTER 2: EMBEDDED SYSTEMS

2.1 Introduction of Embedded System:

An Embedded System is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function. A good example is the microwave oven. Almost every household has one, and tens of millions of them are used everyday, but very few people realize that a processor and software are involved in the preparation of their lunch or dinner.

2.2 Block Diagram Of EMBEDDED SYSTEM:

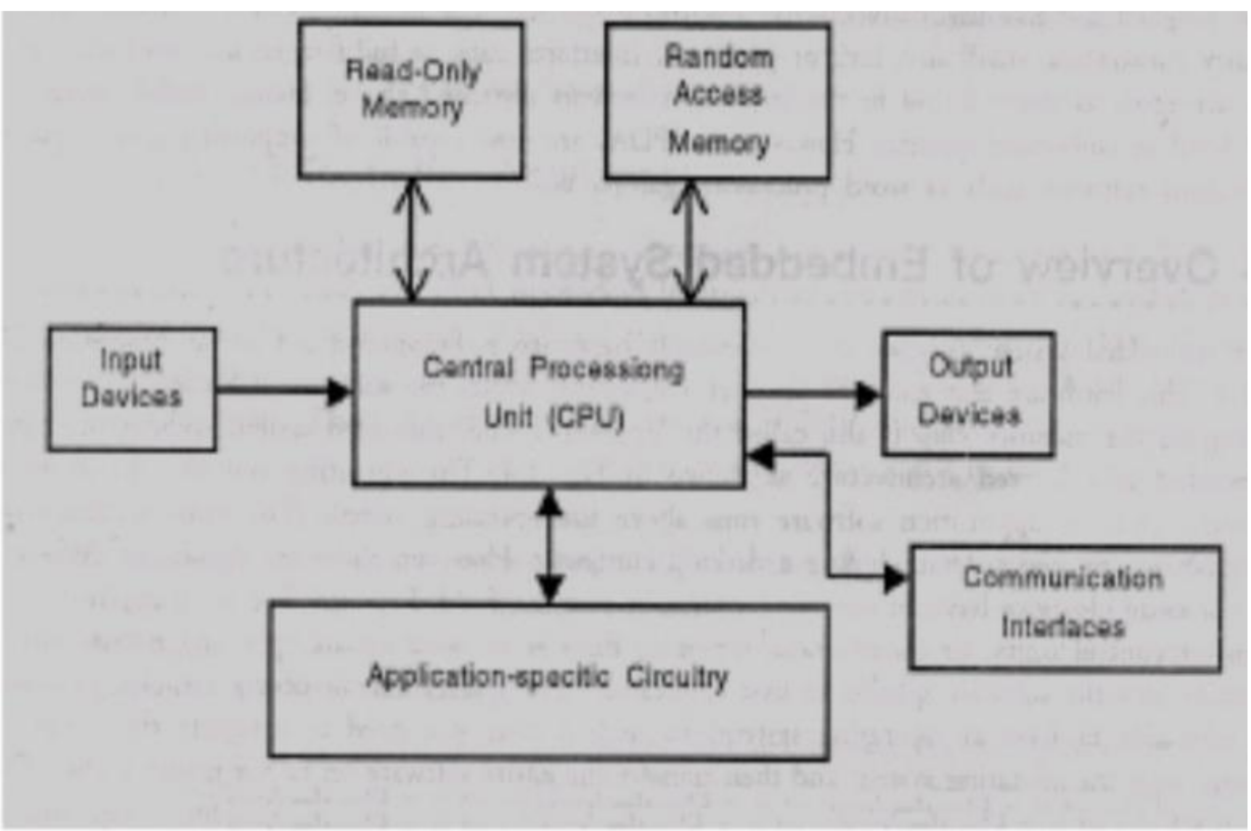


Fig.2: Block diagram of Embedded system

This is in direct contrast to the personal computer in the family room. It too is comprised of computer hardware and software and mechanical components (disk drives, for example). However, a personal computer is not designed to perform a specific function rather; it is able to do many different things. Many people use the term general-purpose computer to make this distinction clear. As shipped, a general-purpose computer is a blank slate; the manufacturer does not know what the customer will do wish it. One customer may use it for a network file server another may use it exclusively for playing games, and a third may use it to write the next great American novel.

Frequently, an embedded system is a component within some larger system. For example, modern cars and trucks contain many embedded systems. One embedded system controls the anti-lock brakes, other monitors and controls the vehicles emissions, and a third displays information on the dashboard. In some cases, these embedded systems are connected by some sort of a communication network, but that is certainly not a requirement. At the possible risk of confusing you, it is important to point out that a general-purpose computer is itself made up of numerous embedded systems. For example, computer consists of a keyboard, mouse, video card, modem, hard drive, floppy drive, and sound card-each of which is an embedded system. Each of these devices contains a processor and software and is designed to perform a specific function. For example, the modem is designed to send and receive digital data over analog telephone line. Thats it and all of the other devices can be summarized in a single sentence as well.

If an embedded system is designed well, the existence of the processor and software could be completely unnoticed by the user of the device. Such is the case for a microwave oven, VCR, or alarm clock. In some cases, it would even be possible to build an equivalent device that does not contain the processor and software. This could be done by replacing the combination with a custom integrated circuit that performs the same functions in hardware. However, a lot of flexibility is lost when a design is hard-cooled in this way. It is mush easier, and cheaper, to change a few lines of software than to redesign a piece of custom hardware.

2.3 History and Future:

Given the definition of embedded systems earlier is this chapter; the first such systems could not possibly have appeared before 1971. That was the year Intel introduced the worlds first microprocessor. This chip, the 4004, was designed for use in a line of business calculators produced by the Japanese Company Busicom. In 1969, Busicom asked Intel to design a set of custom integrated circuits-one for each of their new calculator models. The 4004 was Intel's response rather than design custom hardware for each calculator, Intel proposed a general-purpose circuit that could be used throughout the entire line of calculators. Intel's idea was that the software would give each calculator its unique set of features.

The microcontroller was an overnight success, and its use increased steadily over the next decade. Early embedded applications included unmanned space probes, computerized traffic lights, and aircraft flight control systems. In the 1980s, embedded systems quietly rode the waves of the microcomputer age and brought microprocessors into every part of our kitchens (bread machines, food processors, and microwave ovens), living rooms (televisions, stereos, and remote controls), and workplaces (fax machines, pagers, laser printers, cash registers, and credit card readers).

It seems inevitable hat the number of embedded systems will continue to increase rapidly. Already there are promising new embedded devices that have enormous market potential; light switches and thermostats that can be central computer, intelligent air-bag systems that don't inflate when children or small adults are present, pal-sized electronic organizers and personal digital assistants (PDAs), digital cameras, and dashboard navigation systems. Clearly, individuals who possess the skills and desire to design the next generation of embedded systems will be in demand for quite some time.

2.4 Real Time Systems:

One subclass of embedded is worthy of an introduction at this point. As commonly defined, a real-time system is a computer system that has timing constraints. In other words, a real-time system is partly specified in terms of its ability to make certain calculations or decisions in a timely manner. These important calculations are said to have deadlines for completion. And, for all practical purposes, a missed deadline is just as bad as a wrong answer.

The issue of what if a deadline is missed is a crucial one. For example, if the real-time system is part of an airplane’s flight control system, it is possible for the lives of the passengers and crew to be endangered by a single missed deadline. However, if instead the system is involved in satellite communication, the damage could be limited to a single corrupt data packet.

The more severe the consequences, the more likely it will be said that the deadline is "hard" and thus, the system is a hard real-time system. Real-time systems at the other end of this discussion are said to have "soft" deadlines.

All of the topics and examples presented in this book are applicable to the designers of real-time system who is more delight in his work. He must guarantee reliable operation of the software and hardware under all the possible conditions and to the degree that human lives depend upon three system’s proper execution, engineering calculations and descriptive paperwork.

2.5 Applications of Embedded systems:

Nearly 99 per cent of the processors manufactured end up in embedded systems. The embedded system market is one of the highest growth areas as these systems are used in very market segment- consumer electronics, office automation, industrial automation, biomedical engineering, wireless communication, data communication, telecommunications, transportation, military and so on.

Consumer appliances:

At home we use a number of embedded systems which include digital camera, digital diary, DVD player, electronic toys, microwave oven, remote controls for TV and air-conditioner, VCO player, video game consoles, video recorders etc. Today’s high-tech car has about 20 embedded systems for transmission control, engine spark control, air-conditioning, navigation etc. Even wristwatches are now becoming embedded systems.

The palmtops are powerful embedded systems using which we can carry out many general-purpose tasks such as playing games and word processing.

Office automation:

The office automation products using em embedded systems are copying machine, fax machine, key telephone, modem, printer, scanner etc.

Industrial automation:

Today a lot of industries use embedded systems for process control. These include pharmaceutical, cement, sugar, oil exploration, nuclear energy, electricity generation and transmission. The embedded systems for industrial use are designed to carry out specific tasks such as monitoring the temperature, pressure, humidity, voltage, current etc. and then take appropriate action based on the monitored levels to control other devices or to send information to a centralized monitoring station. In hazardous industrial environment, where human presence has to be avoided, robots are used, which are programmed to do specific jobs. The robots are now becoming very powerful and carry out many interesting and complicated tasks such as hardware assembly.

Medical electronics:

Almost every medical equipment in the hospital is an embedded system. These equipments include diagnostic aids such as ECG, EEG, blood pressure measuring devices, X-ray scanners, equipment used in blood analysis, radiation, colonoscopy, endoscopy etc. Developments in medical electronics have paved way for more accurate diagnosis of diseases.

Computer networking:

Computer networking products such as bridges, routers, Integrated Services Digital Networks (ISDN), Asynchronous Transfer Mode (ATM), X.25 and frame relay switches are embedded systems which implement the necessary data communication protocols. For example, a router interconnects two networks. The two networks may be running different protocol stacks.

The router's function is to obtain the data packets from incoming pores, analyze the packets and send them towards the destination after doing necessary protocol conversion. Most networking equipments, other than the end systems (desktop computers) we use to access the networks, are embedded systems.

Telecommunications:

In the field of telecommunications, the embedded systems can be categorized as subscriber terminals and network equipment. The subscriber terminals such as key telephones, ISDN phones, terminal adapters, web cameras are embedded systems. The network equipment includes multiplexers, multiple access systems, Packet Assemblers Dissemblers (PADs), satellite modems etc. IP phone, IP gateway, IP gatekeeper etc. are the latest embedded systems that provide very low-cost voice communication over the Internet.

Wireless technologies:

Advances in mobile communications are paving way for many interesting applications using embedded systems. The mobile phone is one of the marvels of the last decade of the 20th century. It is a very powerful embedded system that provides voice communication while we are on the move. The Personal Digital Assistants and the palmtops can now be used to access multimedia services over the Internet. Mobile communication infrastructure such as base station controllers, mobile switching centers are also powerful embedded systems.

Insemination:

Testing and measurement are the fundamental requirements in all scientific and engineering activities. The measuring equipment we use in laboratories to measure parameters such as weight, temperature, pressure, humidity, voltage, current etc. are all embedded systems. Test equipment such as oscilloscope, spectrum analyzer, logic analyzer, protocol analyzer, radio communication test set etc. are embedded systems built around powerful processors. Thank to miniaturization, the test and measuring equipment are now becoming portable facilitating easy testing and measurement in the field by field-personnel.

Security:

Security of persons and information has always been a major issue. We need to protect our homes and offices, and also the information we transmit and store. Developing embedded systems for security applications is one of the most lucrative businesses nowadays. Security devices at homes, offices, airports etc. for authentication and verification are embedded systems. Encryption devices are nearly 99 per cent of the processors that are manufactured end up in embedded systems.

Embedded systems find applications in every industrial segment- consumer electronics, transportation, avionics, biomedical engineering, manufacturing, process control and industrial automation, data communication, telecommunication, defense, security etc. Used to encrypt the data/voice being transmitted on communication links such as telephone lines.Biometric systems using fingerprint and face recognition are now being extensively used for user authentication in banking applications as well as for access control in high security buildings.

Finance:

Financial dealing through cash and cheques are now slowly paving way for transactions using smart cards and ATM (Automatic Teller Machine, also expanded as Any Time Money) machines. Smart card, of the size of a credit card, has a small micro-controller and memory; and it interacts with the smart card reader. ATM machine acts as an electronic wallet.

CHAPTER 3: COMPONENT DESCRIPTION

3.1 RASPBERRY PI

3.1.1 Raspberry Pi:

The Raspberry Pi is a credit-card-sized single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools.

The Raspberry Pi is manufactured through licensed manufacturing deals with Newark element14 (Premier Farnell), RS Components and Egoman. All of these companies sell the Raspberry Pi online. Egoman produces a version for distribution solely in China and Taiwan, which can be distinguished from other Pi by their red coloring and lack of FCC/CE marks. The hardware is the same across all manufacturers.

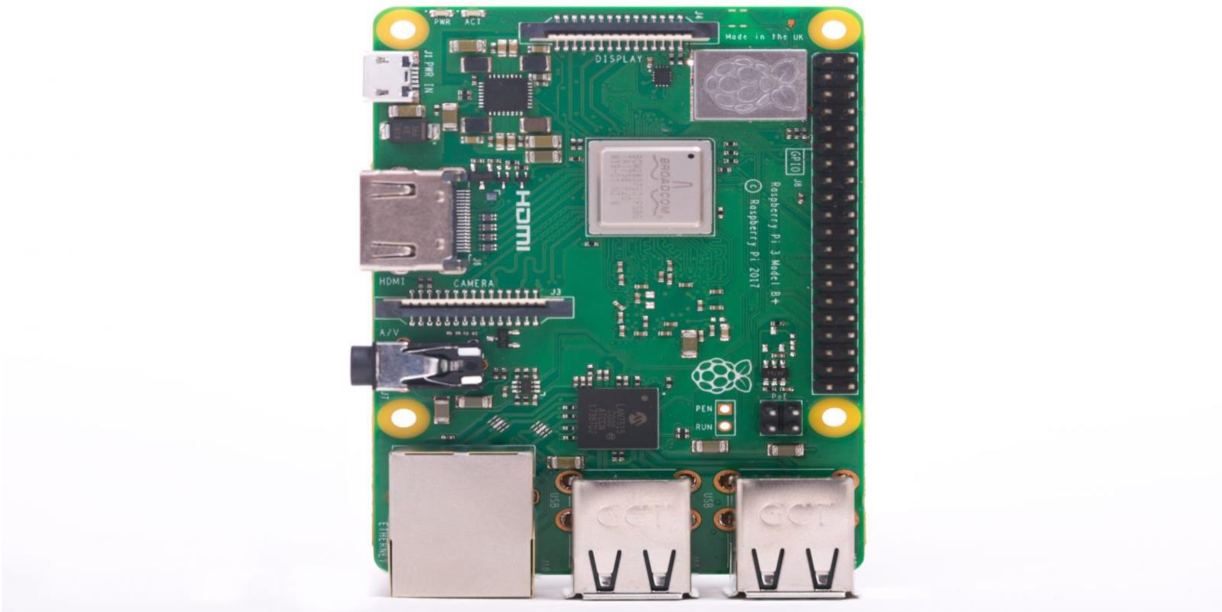


Fig 3.1.1: Raspberry Pi 3

Product Information:

- RASPBERRY PI 3 BOARD
- Silicon Manufacturer: Broadcom
- Core Architecture: ARM
- Core Sub-Architecture: ARM1176JZF-S
- Silicon Core Number: BCM2837
- Silicon Family Name: BCM2xxx
- No. of Bits: 64 bit
- Kit Contents: Raspberry Pi 3

- Features: Credit card size computer, HDMI, Ethernet & 4 USB ports, 1GB RAM, Micro SD Socket, 40 GPIO

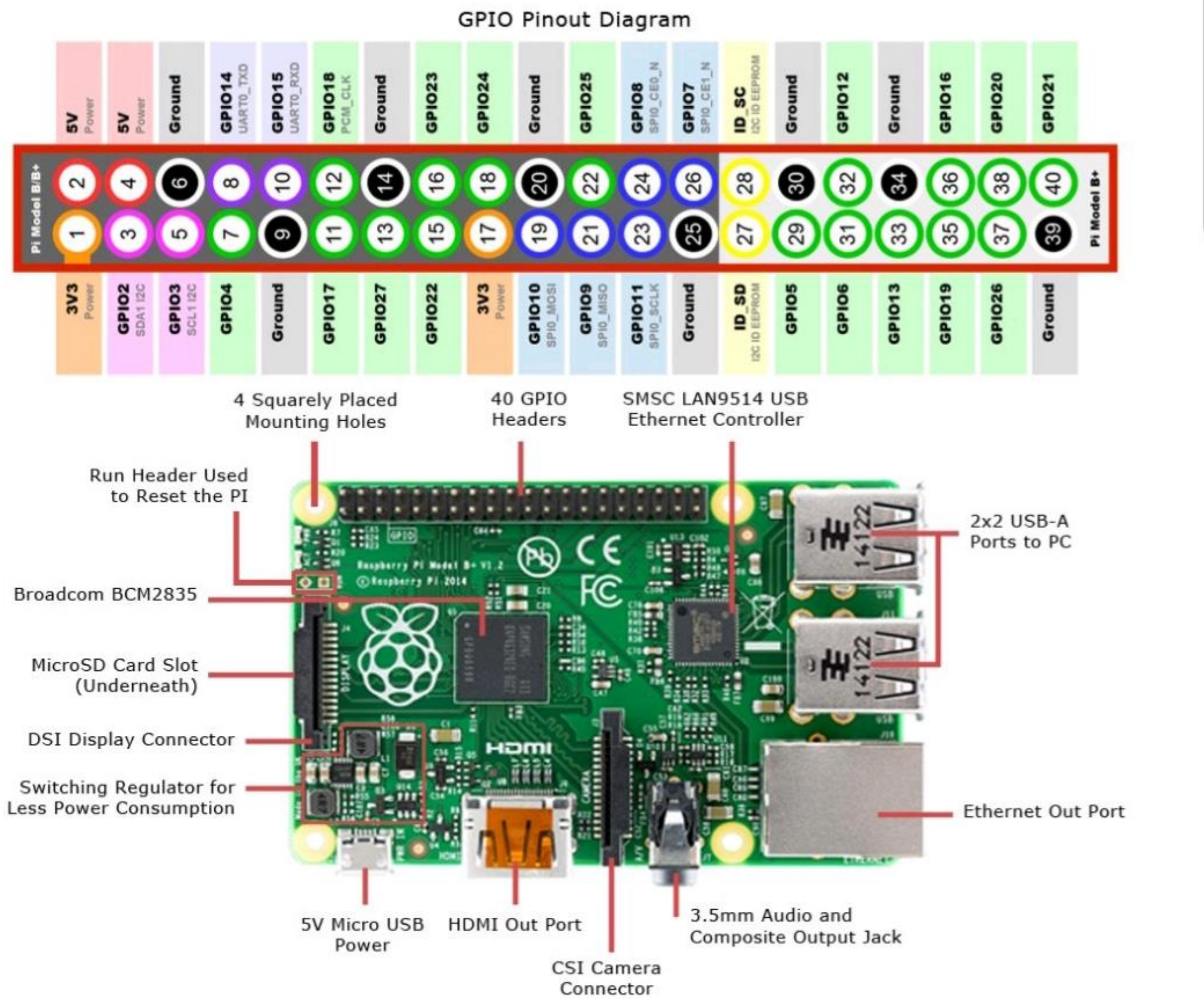


Fig 3.1.2: Pin Diagram Of Raspberry Pi 3

Raspberry Pi 3 Featuring the ARM1176JZF-S Running at 1.2 Ghz, with 1 GB of RAM

The RASPBERRY Pi 3 is a credit card sized computer that plugs into your TV and a keyboard, its like a little PC which can be used for many of the things that your desktop PC does, like spreadsheets, word processing and games. It also plays high definition video. The design is based around a Broadcom BCM2837 SoC, which includes an ARM1176JZF-S 1.2 Ghz processor, VideoCore IV GPU and 1 GB of RAM. The design does not include a built in hard disk or solid state drive, instead relying on a microSD card for booting and long term storage.

This board is intended to run Linux kernel based operating systems.

- Integrated videocore 4 GPU capable of playing full 1080p high definition Blu-Ray quality
- video
- Free, versatile, and highly developer friendly Debian GNU/Linux operating system
- HDMI video output and RCA video output
- Four USB ports
- 4-pole 3.5mm stereo audio jack with composite video output
- 5V micro USB power input jack
- microSD, MMC, SDIO flash memory card slot
- 40 pin 2.54mm header expansion slot
- Hobby & Hardware Hacking
- Imaging, Video & Vision
- Consumer Electronics
- Communications & Networking

3.1.2 Advantages:

Advantages

The Raspberry Pi has found its way in to the hobbyist market for computing, but it is also very capable for other business and personal use as well. An extremely low power draw, small form factor, no noise, solid state storage, and other features make it an attractive solution for a small and lightweight server.

Although Raspberry Pi is as small as the size of a credit card, it works as if a normal computer at a Relatively low price. it is possible to work as a low-cost server to handle light internal or web traffic.

Grouping a set of Raspberry Pi to work as a server is more cost-effective than a normal server. If all light traffic servers are changed into Raspberry Pi, it can certainly minimize an enterprise’s budget.

1. Power consumption - The Pi draws about five to seven watts of electricity. This is about one tenth of what a comparable full-size box can use. Since servers are running constantly night and day, the electrical savings can really add up.

2. No moving parts - The Pi uses an SD card for storage, which is fast and has no moving parts. There are also no fans and other things to worry about. A Class 10 SD card is usually the best performing compared to lower class cards, but this will mainly only affect boot time where there is the most I/O. There is a compatibility chart for SD cards here, results may vary.

3. Small form factor - The Pi (with a case) can be held in your hand. A comparable full-size box cannot. This means the Pi can be integrated inside of devices, too.

4. No noise - The Pi is completely silent.

5. Status lights - There are several status lights on the Pi's motherboard. With a clear case you can see NIC activity, disk I/O, power status, etc.

6. Expansion capabilities - There are numerous devices available for the Pi, all at very affordable prices. Everything from an I/O board (GPIO) to a camera. The Pi has two USB ports, however by hooking up a powered USB hub, more devices can be added.

7. Built-in HDMI capable graphics - The display port on the Pi is HDMI and can handle resolutions up to 1920×1200, which is nice for making the Pi in to a video player box for example. There are some converters that can convert to VGA for backwards compatibility. A list of HDMI to VGA converters can be found here.

8. Affordable - compared to other similar alternatives, the Pi (revision B) offers the best specs for the price. It is one of the few devices in its class that offers 512 MB of RAM.The Pi has come down in price since it first arrived, and is finally affordable as a hobby, business use, or whatever need there is.

9. Huge community support - The Pi has phenomenal community support. Support can be obtained quite easily for the hardware and/or GNU/Linux software that runs on the Pi mainly in user forums, depending on the GNU/Linux distribution used. A good list of distributions can be found here.

10. Over clocking capability - The Pi can be over clocked if there are performance problems with the application used, but it is at the user's risk to do this.

11. Multiple uses - Having the storage on an SD card makes it easy to swap with other SD cards running other GNU/Linux distributions to quickly and easily change the functionality of the Pi. If you want to set up the Pi to run as a server to test it out, then later try something else, just swap the SD card and you're done. Using the "dd" command on a GNU/Linux computer, a backup of the SD card can be created and later restored if needed.

3.1.3 Software for the Pi:

The Pi runs GNU/Linux and variants of similar operating systems.

A single or multi-purpose server of these types or others will probably not be an issue with memory, with light use. Memory intensive applications generally become an issue with the X11 desktop and running end user applications.

Today virtualization is very popular so some may say that the cost of spinning up a virtual machine is less than running a Raspberry Pi. But, calculate the power consumption for your hypervisor, and weigh out the differences to see which method in fact costs less overall.

Sometimes, a physical box or physical segmentation is needed, or avoiding high costs of running a full hypervisor is a factor, and this is where the Pi can step in.

The Raspberry Pi will run a range of OS Distributions and run a variety of software.

- See Software for an overview and OS Distributions for supported operating system and pre-configured images.
- Officially supported OS distributions include Raspbian, Arch Linux and RISC OS Open.
- Many unofficial distributions are available on the Distributions page.

- Advice is also available if you want to compile a kernel, boot from the network using U-Boot, or test the Pi's performance.
- The Raspberry Pi supports a wide range of programming languages, with many tutorials available.
- Information about installing. A Specific application is available through the link.

- Extensive (boot) configuration info (config.txt) is available here
  .
- Information about various utilities that can be used with your Raspberry Pi can be found here.

3.1.4 Future Use of Raspberry Pi:

Raspberry Pi was designed by the Raspberry Pi Foundation in UK, initially designed to help kids in the school learn basic computer science knowledge, but the very first idea is that everyone should have a personal computer. To make this idea come true, they make this project start from children at school. The education version of this device quickly showed its pros, as a result of applying this technology in the school, a lot of children develop their knowledge and capability about computer science. So, in the future they should continue develop in this industry, and create different version for teachers and students, for example, make the outfits seems more cute for children so that it can attract more kids' attention.

The Raspberry Pi is using the Linux as basic programming language, and they attempt to come up with their own language that fits this technology better sometime in the future.

The Raspberry Pi is already pre-order out all, and there is a lot individuals order it because it can be built by the customer's own needs. So there may be a possibility it could be used in some special industry like the spy industry, authorized by the government, it credit card size is a big advantage and it can be transferred to any kind of computer that may just fit the government's needs. But the problem is that this technology is so common that may be traced very easily by the enemy.

The current version of Raspberry Pi is only 255MB, in the future, this is a main point to break, however, if the enhanced has a RAM boost, its cost will essentially increase, and the sell price may not be able to cover its cost, but if they increase the sell price they

may not get many customer like before. So they may want to keep the cost down, if so, they must cut some budget in other features of model B, such as Bluetooth, inbuilt analogy-to-digital audio converters, the whole technology is will not be very valuable. Since this technology is so good, will it be prevalent? The answer is no. As the technology is developing everyday, we got a lot high technology device that function likely as Raspberry Pi, like Smartphone and tablet, the very different between Raspberry Pi and other devices is that you can build your own small personal computer using Raspberry Pi, it may seems attractive for a few people, but others as we know are more lazy and less interested in developing IT knowledge, they like to enjoy the high technology that others created for them. In a word, high technology doesn't always become prevalent.

3.2 DC MOTORS

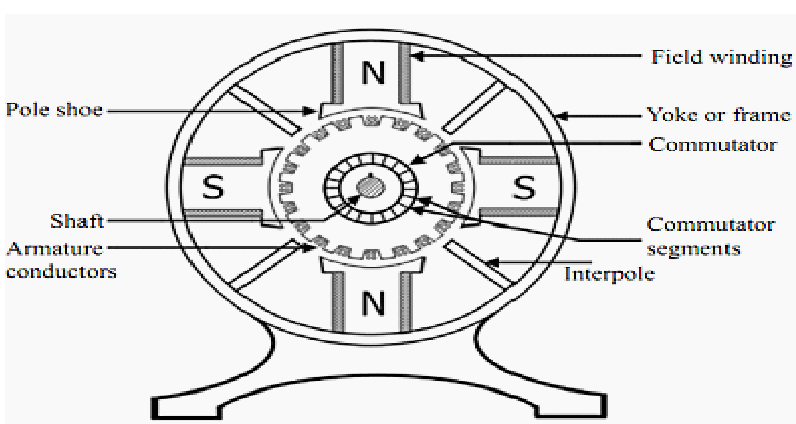
3.2.1 Introduction:

Permanent magnet DC motor responds to both voltage and current. The steady state voltage across the motor determines the motor's running speed, and the current through its armature windings determines the torque. Apply a voltage and the motor will start running in one direction; reverse the polarity and the direction will be reversed. If you apply a load to the motor shaft, it will draw more current, if the power supply does not able to provide enough current, the voltage will drop the speed of the motor will be reduced. However, if the power supply can maintain voltage while supplying the current, the motor will run at the same speed. In general, you can control the speed by applying the appropriate voltage, while current controls torque. In most cases, DC motors are powered up by using fixed DC power supply, therefore; it is more efficient to use a chopping circuit.

Consider what happens when a voltage applied to a motor's winding is rapidly turned ON and OFF in such a way that the frequency of the pulses produced remains constant, but the width of the ON pulse is varied. This is known as Pulse Width Modulation (PWM). Current only flows through the motor during the ON portion of the PWM waveform. If the frequency of the PWM input is high enough, the mechanical inertia of the motor cannot react to the ripple wave; instead, the motor behaves as if the current were the DC average of the ripple wave. Therefore, by changing the width of pulse, we can control the motor speed. At the most basic level, electric motors exist to convert electrical energy into mechanical energy. This is done by way of two interacting magnetic fields – one stationery, and another attached to a part that can move. A number of types of electric motors exist, but most BEAM bots used DC motors in some form or another. DC motors have the potential for very high torque capabilities (all though this is generally a function of the physical size of the motor), are easy to miniaturize, and can be "throttled" be a adjusting their supply voltage DC motors are also not only the simplest, but the oldest electric motors.

3.2.2 Principles of operation :

In any electric motor, operation is based on simple electromagnetism. A current-carrying conductor generates a magnetic field; when this is then placed in and external magnetic field, it will experience force proportional to the current in the conductor, and to the strength of the external magnetic field. As you are well aware of from playing with magnets as a kid, opposite (north and south) polarities attract, while like polarities (north and north, south and south) repel. The internal configuration of a DC motor is designed to harness the magnetic interaction between a current-carrying conductor and an external magnetic field to generate rotational motion. Let's start by looking at a simple 2-pole DC electric motor.



.

Fig 3.2.1: DC motor

Every DC motor has six basic parts 2- axle, rotor(a.k.a., armature), stator, commutator, field magnet's, and brushes. In most common DC motors the external magnetic field is produced by high-strength permanent magnets. The stator is the stationery part of the motor – this includes the motor casing, as well as two or more permanent magnet pieces. The rotor (togeather with the axle and attached commutator) rotates with respect to the stator. The rotor consists of windings(generally on a core), the windings being electrically connected to the commutator.

The geometry of the brushes, commutator contacts, and rotor windings are such that when power is applied, the polarities of the energized winding and stator magnet(s) are misaligned, and the rotor will rotate until it is almost aligned with the stator's field magnets. As the rotor reaches alignment, the brushes move to the next commutator contacts, and energize the next winding. Given are example two-pole motor, the rotation reverses the direction of current through the rotor winding, leading to a "flip" of the rotors magnetic field, and driving it to continue rotating.

In real life, though, DC motors will always have more than two poles(three is a very common number).in particular, this avoids "dead spots" in the commutator. You can imagine how with our example two-pole motor, if the rotor is exactly at the middle of its rotation (perfectly aligned with the field magnets), it will get "struck there". Mean while, with a two-pole motor, there is a moment where the commutator shorts out the power supply(i.e., both brushes touch both commutator contacts simultaneously). This would be bad for the power supply, waste energy, and damage motor components as well. Yet another disadvantage of such a simple motor is that It would exhibit a high amount if torque "ripple" (the amount of torque it could produce is cyclic with th position of the rotor).

3.3 H-BRIDGE

3.3.1 Introduction:

An H-bridge is an electronic circuit which enables DC electric motors to run forward and backward. These circuits are often used in robotics. H-bridges are available as integrated circuits, or can be built from discrete components.

The two basic states of an H-bridge. The term "H-bridge" is derived from the typical graphical representation of such a circuit. An H-bridge is built with four switches (solid-state or mechanical). When the switches S1 and S4 (according to the first figure) are closed (and S2 and S3 are open) a positive voltage will be applied across the motor. By opening S1 and S4 switches and closing S2 and S3 switches, this voltage is reversed, allowing reverse operation of the motor.

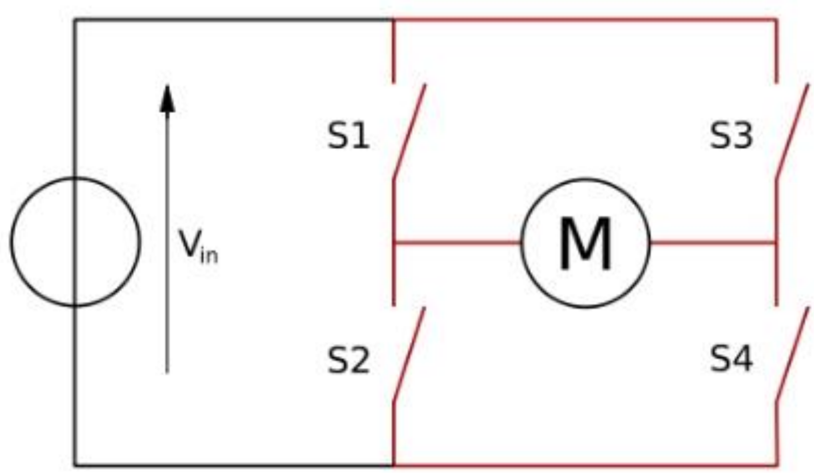


Fig 3.3.1: Circuit for H-BRIDGE

Using the nomenclature above, the switches S1 and S2 should never be closed at the same time, as this would cause a short circuit on the input voltage source. The same applies to the switches S3 and S4. This condition is known as shoot-through.

3.3.2 Operation:

The H-Bridge arrangement is generally used to reverse the polarity of the motor, but can also be used to 'break' the motor, where the motor comes to a sudden stop, as the terminals of the motor are shorted, or to let the motor 'free run' to a stop, as the motor is effectively disconnected from the circuit. The following table summarizes operation.

S1	S2	S3	S4	Result
1	0	0	1	Motor moves right
0	1	1	0	Motor moves left
0	0	0	0	Motor free runs
0	1	0	1	Motor brakes

Table 3.3.1:operation

3.3.3 H-Bridge Driver:

The switching property of this H-Bridge can be replaced by a Transistor or a Relay or a Mosfet or even by an IC. Here we are replacing this with an IC named L293D as the driver whose description is as given below.

Features:

- 600mA OUTPUT CURRENT CAPABILITY
- PER CHANNEL
- 1.2A PEAK OUTPUT CURRENT (non repetitive)
- PER CHANNEL
- ENABLE FACILITY
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V
- (HIGH NOISE IMMUNITY)
- INTERNAL CLAMP DIODES

3.3.4 Description:

The Device is a monolithic integrated high voltage, high current four channel driver designed to accept standard DTL or TTL logic levels and drive inductive loads (such as relays solenoids, DC and stepping motors) and switching power transistors. To simplify use as two bridges each pair of channels is equipped with an enable input. A separate supply input is provided for the logic, allowing operation at a lower voltage and internal clamp diodes are included. This device is suitable for use in switching applications at frequencies up to 5 kHz. The L293D is assembled in a 16 lead plastic package which has 4 center pins connected together and used for heatsinking The L293DD is assembled in a 20 lead surface mount which has 8 center pins connected together and used for heatsinking.

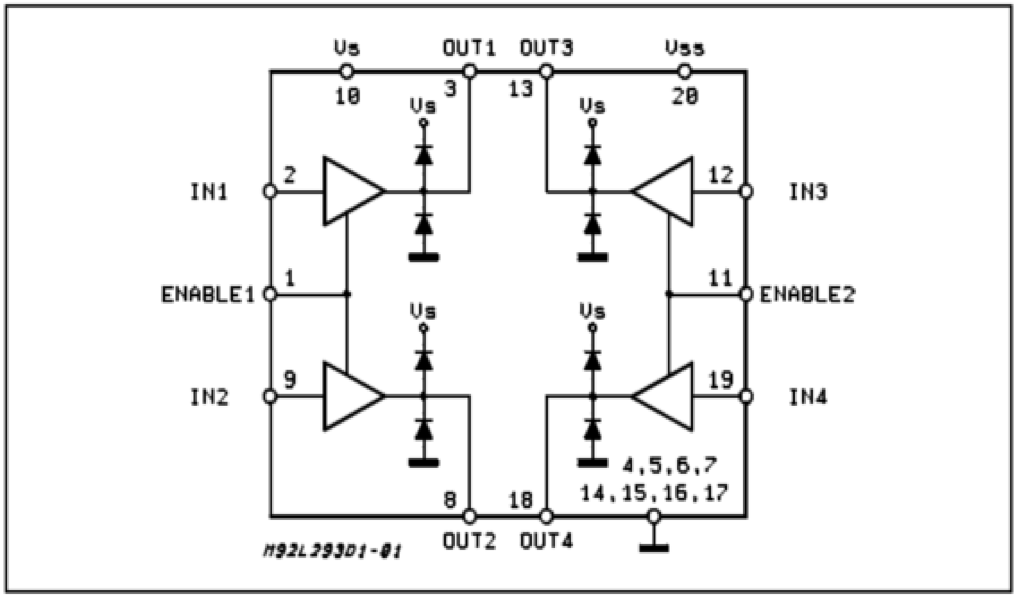


Fig 3.3.2: Circuit Diagram of H-BRIDGE

Symbol	Parameter	Value	Unit
V_S	Supply Voltage	36	V
V_{SS}	Logic Supply Voltage	36	V
V_i	Input Voltage	7	V
V_{en}	Enable Voltage	7	V
I_o	Peak Output Current (100 µs non repetitive)	1.2	A
P_{tot}	Total Power Dissipation at T_{pins} = 90 °C	4	W
T_{stg}, T_j	Storage and Junction Temperature	– 40 to 150	°C

Table 3.3.2: Absolute Maximum Ratings

3.4 SOIL MOISTURE SENSOR

3.4.1 Definition:

Soil moisture sensors measure the volumetric water content in soil. Since the direct gravimetric measurement of free soil moisture requires removing, drying, and weighting of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with neutrons, as a proxy for the moisture content.

The relation between the measured property and soil moisture must be calibrated and may vary depending on environmental factors such as soil type, temperature, or electric conductivity. Reflected microwave radiation is affected by the soil moisture and is used for remote sensing in hydrology and agriculture. Portable probe instruments can be used by farmers or gardeners.

Soil moisture sensors typically refer to sensors that estimate volumetric water content. Another class of sensors measure another property of moisture in soils called water potential; these sensors are usually referred to as soil water potential sensors and include tensiometers and gypsum blocks.

3.4.2 Application in the project:

We will insert this sensor in to the soil. It always checks the presence of moisture. If the soil is dry then it will give information to Microcontroller that the soil is dry so need to switch on the motor. By this Microcontroller will perform the specified action depending up on our requirement. After getting sufficient moisture the sensor again senses the presence of moisture.

The same data will be sending to the Microcontroller. After this information from moisture sensor it switches off the motor or it will perform the specified action. The moisture data is transmitted to the irrigation controller on the remote control valve wiring. Special wire runs between the irrigation controller and the sensor are not necessary. The total maximum wire run between moisture sensor and the irrigation controller is 3,000 feet.In this way moisture sensor will senses the moisture and it will send data to Microcontroller.

Fig 3.4.1 : Soil Moisture Sensor

3.4.3 Specifications:

Working Voltage: **5V**

Working Current :< **20mA**

Interface type: **Analog**

Working Temperature: **10°C~30°C**

3.4.4 Working Principle of Moisture Sensor:

The Soil Moisture Sensor uses capacitance to measure dielectric permittivity of the surrounding medium. In soil, dielectric permittivity is a function of the water content. The sensor creates a voltage proportional to the dielectric permittivity, and therefore the water content of the soil. The sensor averages the water content over the entire length of the sensor. There is a 2 cm zone of influence with respect to the flat surface of the sensor, but it has little or no sensitivity at the extreme edges. The Soil Moisture Sensor is used to measure the loss of moisture over time due to evaporation and plant uptake, evaluate optimum soil moisture contents for various species of plants, monitor soil moisture content to control irrigation in greenhouses and enhance bottle biology experiments.

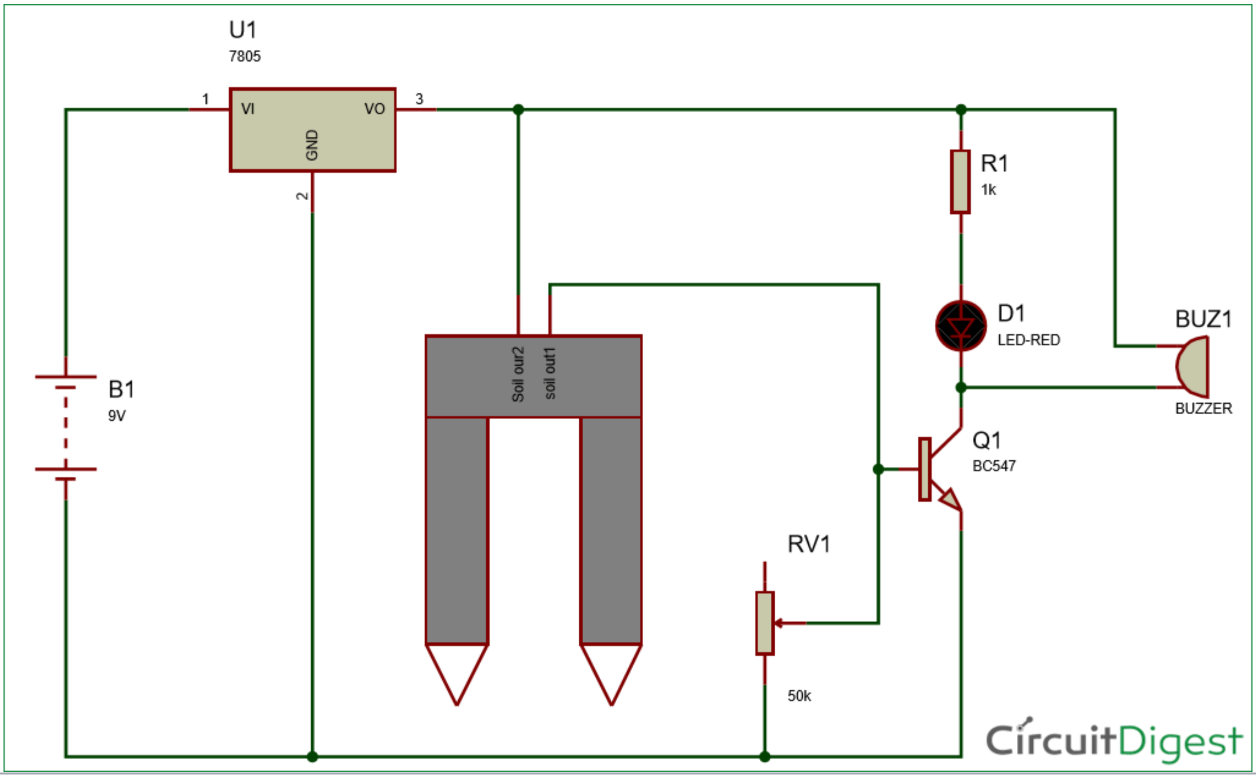


Fig 3.4.2 :Circuit Diagram of Soil Moisture Sensor

3.5 DHT11 SENSOR

3.5.1 Definition:

The digital temperature and humidity sensor DHT11 is a composite sensor that contains a calibrated digital signal output of temperature and humidity. The technology of a dedicated digital modules collection and the temperature and humidity sensing technology are applied to ensure that the product has high reliability and excellent long-term stability.

The sensor includes a resistive sense of wet component and an NTC temperature measurement device, and is connected with a high-performance 8-bit microcontroller. The schematic diagram of the Humiture Sensor Module is as shown following

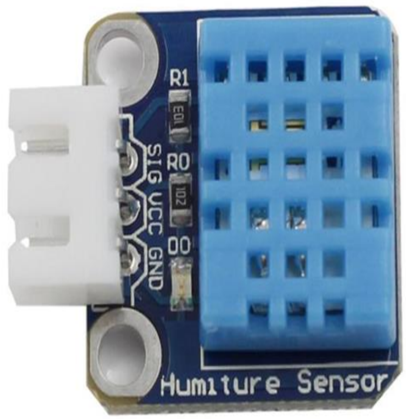


Fig 3.5.1 : DHT11 Sensor

Only three pins are available for use: VCC, GND, and DATA. The communication process begins with the DATA line sending start signals to DHT11, and DHT11 receives the signals and returns an answer signal. Then the host receives the answer signal and begins to receive 40-bit humiture data (8-bit humidity integer + 8-bit humidity decimal + 8-bit temperature integer + 8-bit temperature decimal + 8-bit checksum).

3.5.2 Features:

1)A humiture sensor module to test temperature and humidity, which uses the sensor DHT11.
2) Humidity measurement range: 20 - 90%RH
3) Temperature measurement range: 0 - 60℃
4) Output digital signals indicating temperature and humidity
5) Working voltage: DC 5V; PCB size: 2.0 x 2.0 cm
6) Humidity measurement accuracy: ±5%RH
7) Temperature measurement accuracy: ±2℃

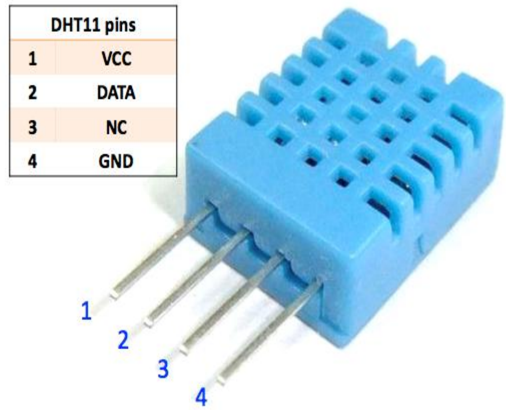


Fig 3.5.1 : Pin Diagram of DHT11 Sensor

3.6 LIQUID CRYSTAL DISPLAY (LCD)

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multisegment LEDs. The reasons being: LCDs are economical, easily programmable, have no limitation of displaying special &amp, even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data. The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

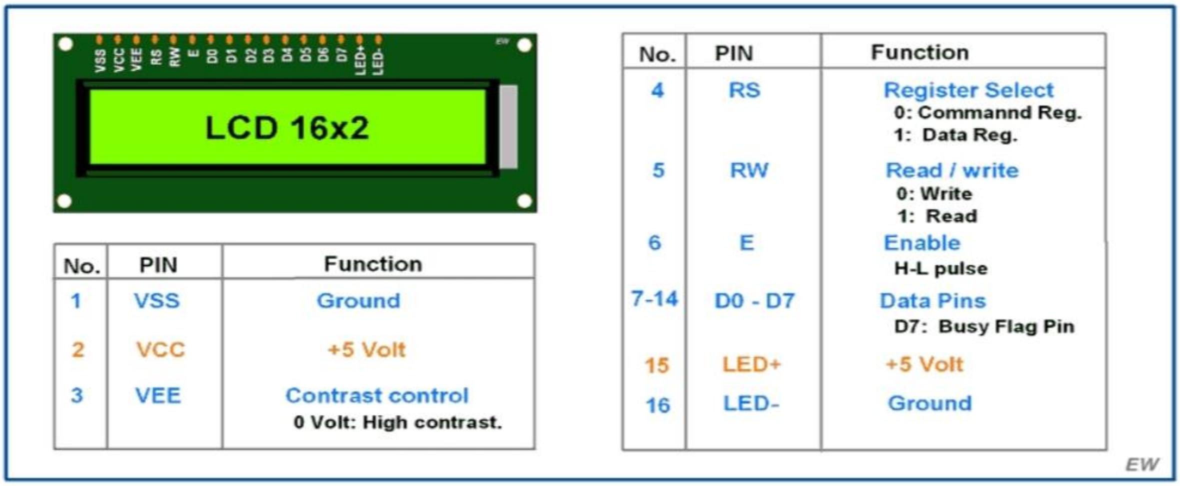


Fig 3.6 :Pin Diagram of LCD

3.7 LIGHT EMITTING DIODES (LED)

An LED is an electronic device that emits light when an electrical current is passed through it. Early LEDs produced only red light, but modern. LEDs can produce several different colors, including red, green, and blue (RGB) light. Recent advances in LED technology have made it possible for LEDs to produce white light as well.

LEDs are commonly used for indicator lights (such as power on/off lights) on electronic devices. They also have several other applications, including electronic signs, clock displays, and flashlights. Since LEDs are energy efficient and have a long lifespan (often more than 100,000 hours), they have begun to replace traditional light bulbs in several areas.

Some examples include street lights, the red lights on cars, and various types of decorative lighting. We can typically identify LEDs by a series of small lights that make up a larger display. For example, if we look closely at a street light, we can tell it is an LED light if each circle is comprised of a series of dots.

The energy efficient nature of LEDs allows them to produce brighter light than other types of bulbs while using less energy. For this reason, traditional flat screen LCD displays have started to be replaced by LED displays, which use LEDs for the backlight. LED TVs and computer monitors are typically brighter and thinner than their LCD counterparts.

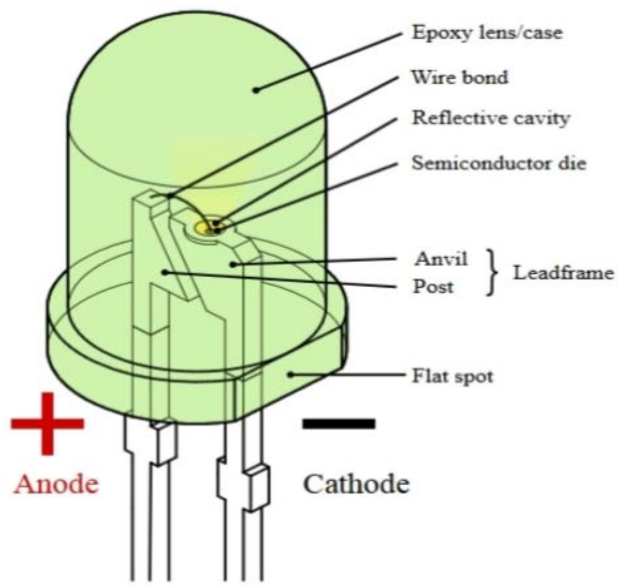


Fig 3.7 : LED

3.8 RELAY MODULE

3.8.1 Definition:

The relay module is a separate hardware device used for remote device switching. With it you can remotely control devices over a network or the Internet. Devices can be remotely powered on or off with commands coming from ClockWatch Enterprise delivered over a local or wide area network. You can control computers, peripherals or other powered devices from across the office or across the world. The Relay module can be used to sense external On/Off conditions and to control a variety of external devices.

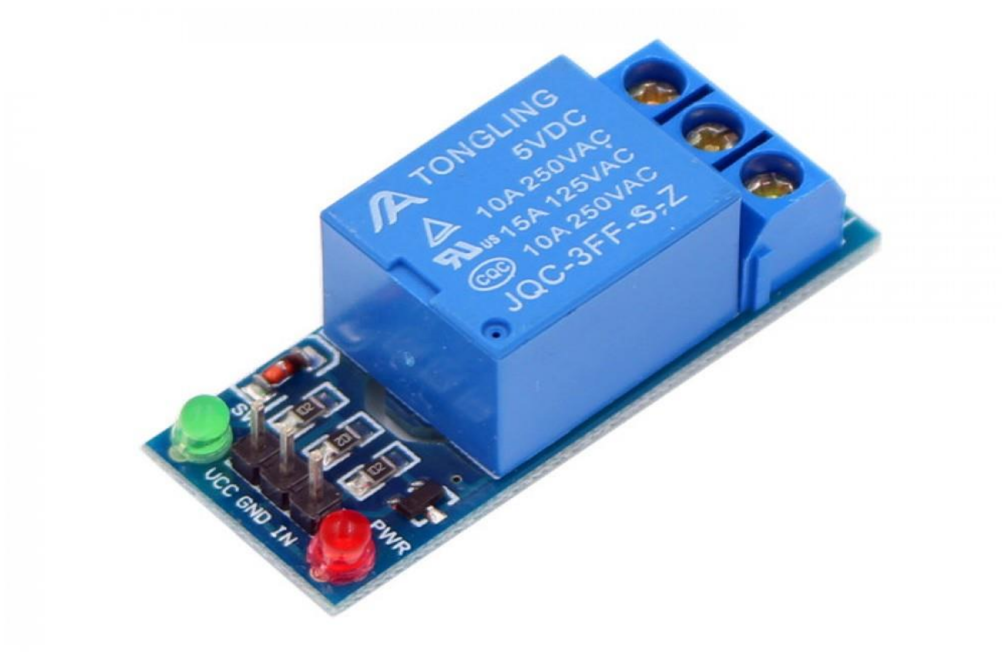
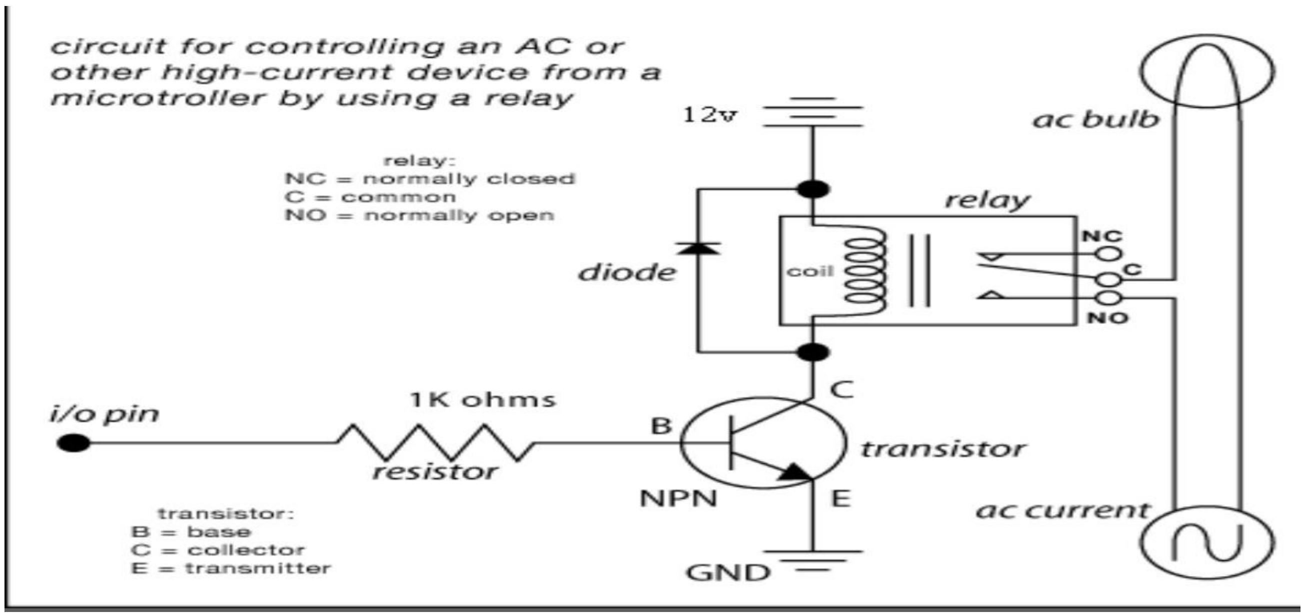


Fig 3.8.1 : Relay Module

3.8.2 Description:

Relay is an electromagnetic switch. It consists of a coil of wire surrounding a soft iron core, an iron yoke, which provides a low reluctance path for magnetic flux, a
movable iron armature, and a set, or sets, of contacts; two in the relay pictured. The armature is hinged to the yoke and mechanically linked to a moving contact or contacts.

When an electric current is passed through the coil, the resulting magnetic field attracts the armature and the consequent movement of the movable contact or contacts either makes or breaks a connection with a fixed contact.



<u>Fig 3.8.2 : Circuit Diagram of Relay Module</u>

3.8.3 Control Relay switch via GPIO:

Often you want to control modules with a higher voltage with the Raspberry Pi. For this purpose, relays can be used on the Raspberry Pi: The relay "switch" is utilized by means of a low-voltage pulse. Since the Pi only tolerates a maximum of 5V (the GPIOs even only 3.3V) without relays, there is the risk that the Pi could burn out. However, if you have two separate circuits this cannot happen.

3.8.4 Applications:

- Control a high-voltage circuit with a low-voltage signal, as in some types of modems or audio amplifiers.
- Control a high-current circuit with a low-current signal, as in the starter solenoid of an automobile.
- Detect and isolate faults on transmission and distribution lines by opening and closing circuit breakers (protection relays).

3.9 POWER SUPPLY DESIGN

3.9.1 POWER SUPPLY:

The input to the circuit is applied from the regulated power supply. The a.c. input i.e., 230V from the mains supply is step down by the transformer to 12V and is fed to a rectifier. The output obtained from the rectifier is a pulsating d.c voltage. So in order to get a pure d.c voltage, the output voltage from the rectifier is fed to a filter to remove any a.c components present even after rectification. Now, this voltage is given to a voltage regulator to obtain a pure constant dc voltage.

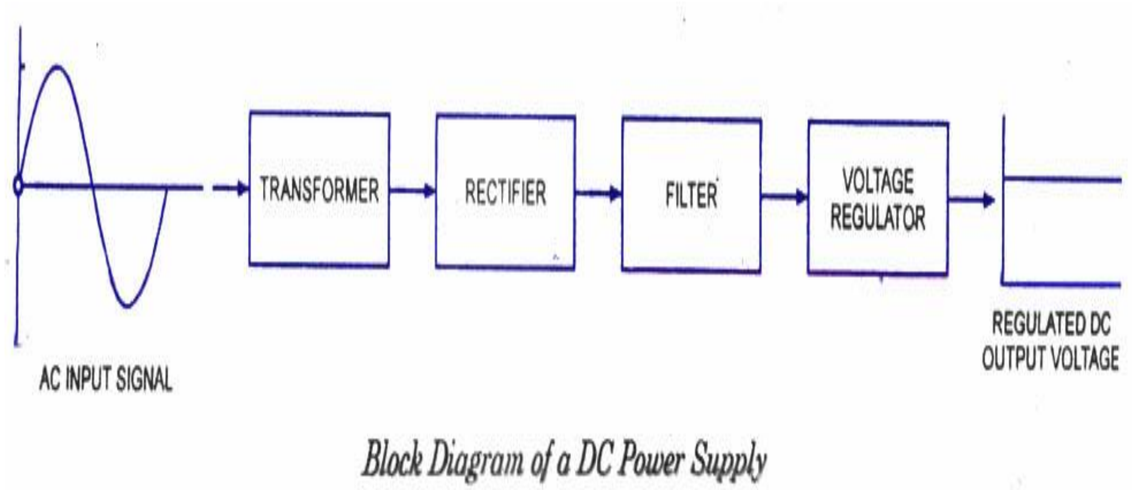


Fig 3.9.1: Block diagram of Power supply

3.9.2 Transformer:

Usually, DC voltages are required to operate various electronic equipment and these voltages are 5V, 9V or 12V. But these voltages cannot be obtained directly.

Thus the a.c input available at the mains supply i.e., 230V is to be brought down to the required voltage level. This is done by a transformer. Thus, a step down transformer is employed to decrease the voltage to a required level.

3.9.3 Bridge Rectifier:

The output from the transformer is fed to the rectifier. It converts A.C. into pulsating D.C. The rectifier may be a half wave or a full wave rectifier. In this project, a bridge rectifier is used because of its merits like good stability and full wave rectification.

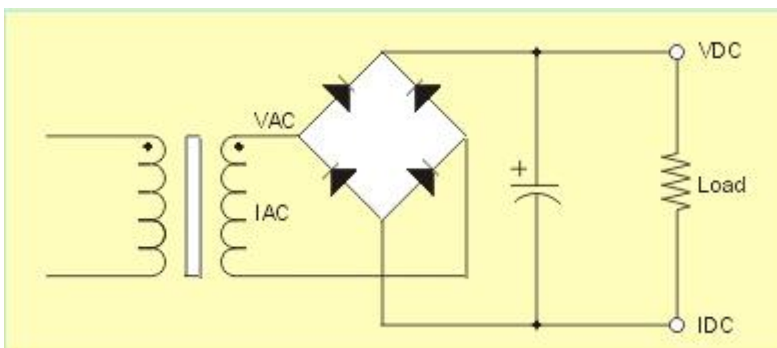


Fig 3.9.2: Rectifier

The Bridge rectifier is a circuit, which converts an ac voltage to dc voltage using both half cycles of the input ac voltage. The Bridge rectifier circuit is shown in the figure. The circuit has four diodes connected to form a bridge. The ac input voltage is applied to the diagonally opposite ends of the bridge. The load resistance is connected between the other two ends of the bridge.

For the positive half cycle of the input ac voltage, diodes D1 and D3 conduct, whereas diodes D2 and D4 remain in the OFF state.

The conducting diodes will be in series with the load resistance R_L and hence the load current flows through R_L. For the negative half cycle of the input ac voltage, diodes D2 and D4 conduct whereas, D1 and D3 remain OFF. The conducting diodes D2 and D4 will be in series with the load resistance R_L and hence the current flows through R_L in the same direction as in the previous half cycle. Thus a bi-directional wave is converted into a unidirectional wave.

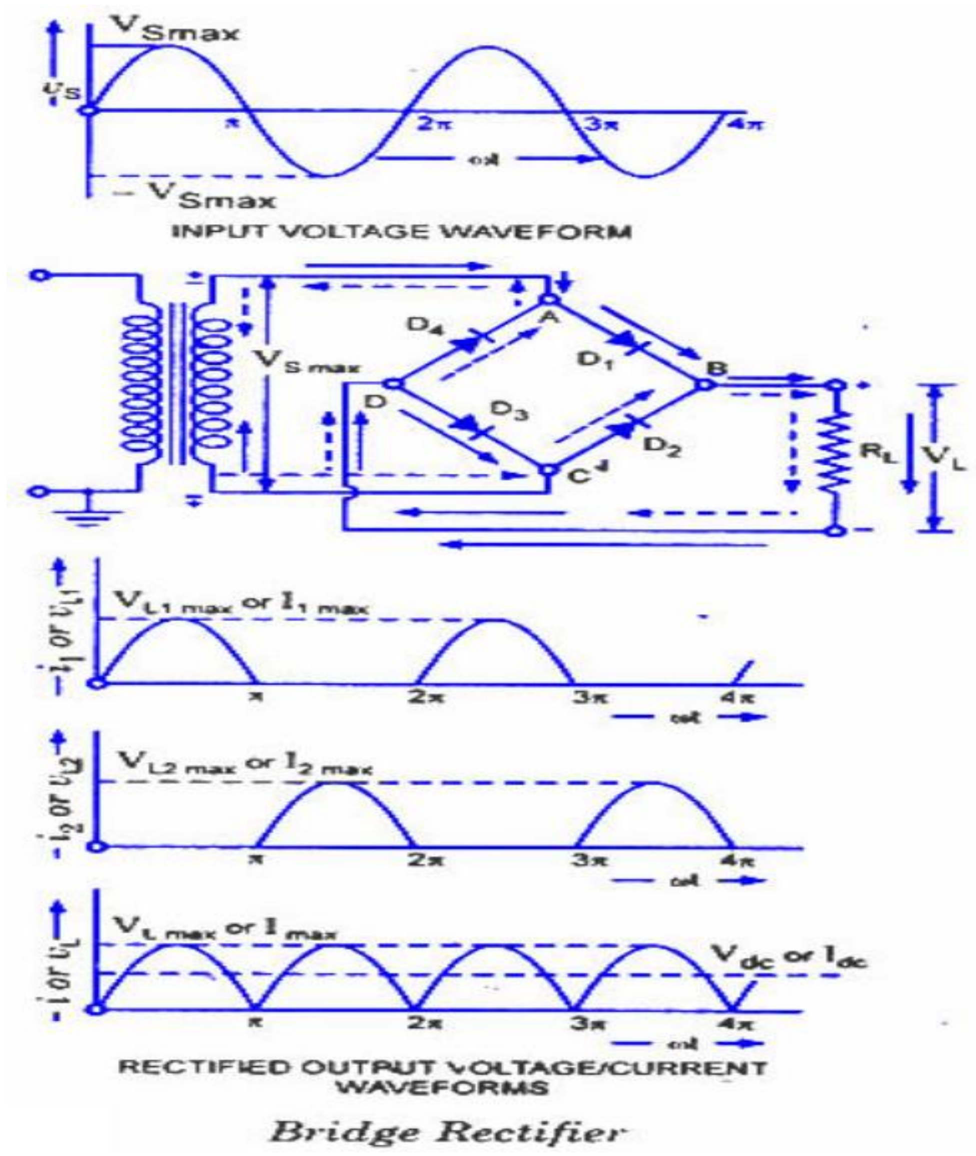


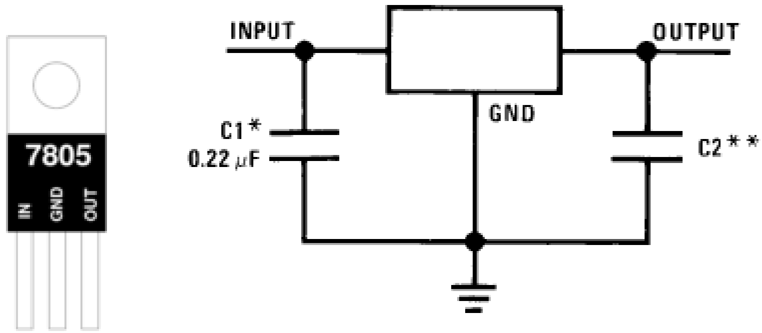
Fig 3.9.3: Bridge Rectifier Waveforms

3.9.4 Filter:

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.

3.9.5 Voltage regulator:

As the name itself implies, it regulates the input applied to it. A voltage regulator is an electrical regulator designed to automatically maintain a constant voltage level. In this project, power supply of 5V and 12V are required. In order to obtain these voltage levels, 7805 and 7812 voltage regulators are to be used. The first number 78 represents positive supply and the numbers 05, 12 represent the required output voltage levels. The L78xx series of three-terminal positive regulators is available in TO-220, TO-220FP, TO-3, D2PAK and DPAK packages and several fixed output voltages, making it useful in a wide range of applications. These regulators can provide local on-card regulation, eliminating the distribution problems associated with single point regulation. Each type employs internal current limiting, thermal shut-down and safe area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1 A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltage and currents.



<u>Fig 3.9.4: Bridge Rectifier</u>

CHAPTER 4: SOFTWARE DESCRIPTION

This project is implemented using following software's and Linux Operating System:

- PYTHON
- HTML

4.1 PYTHON

4.1.1 Introduction:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding; make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace.

A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

Python is a **programming language**, as are C, FORTRAN, BASIC, PHP, etc. Some specific features of Python are as follows:

- An *interpreted* (as opposed *to compiled*) language. Contrary to e.g. C or FORTRAN, one does not compile Python code before executing it. In addition, Python can be used **interactively**: many Python interpreters are available, from which commands and scripts can be executed.
- a free software released under an **open-source** license: Python can be used and distributed free of charge, even for building commercial software.
- **multi-platform**: Python is available for all major operating systems, Windows, Linux/Unix, MacOS X, most likely your mobile phone OS, etc.
- a very readable language with clear non-verbose syntax
- a language for which a large variety of high-quality packages are available for various applications, from web frameworks to scientific computing.
- a language very easy to interface with other languages, in particular C and C++

4.1.2 Basic Principles of Python:

Python has many features that usually are found only in languages which are much more complex to learn and use. These features were designed into python from its very first beginnings, rather than being accumulated into an end result, as is the case with many other scripting languages.

If you're new to programming, even the basic descriptions which follow may seem intimidating. But don't worry – all of these ideas will be made clearer in the chapters which follow. The idea of presenting these concepts now is to make you aware of how python works, and the general philosophy behind python programming. If some of the concepts that are introduced here seem abstract or overly complex, just try to get a general feel for the idea, and the details will be fleshed out later

Object Oriented Programming

Python is a true object-oriented language. The term "object oriented" has become quite a popular buzzword; such high profile languages as C++ and Java are both object oriented by design. Many other languages add some object-oriented capabilities, but were not designed to be object oriented from the ground up as python was.

4.1.3 Python programming on Desktop:

There is an application called IDLE (Integrated Development Environment) on the desktop which provides a slightly easier means to edit and test your python code.

Double click on the IDLE icon on the desktop to start up the application.(this takes you into the python shell where you can test commands)

Create a new window to write your program, click on File, then New Window
(this is where you write your python program)

Type:
Print “hello world”
Save the file to the desktop, click on File, then Save.

Run the code, click on Run, then Run Module. (You should see its output in the python shell)

You can a see and run other people’s source code’ by opening .py files in the ‘python_games’ folder. Or there are millions of examples on the internet!

Switch an LED off and on

Raspberry Pi can be used to read sensors (to measure light, temperature, pressure, distance, etc) and to drive actuators (for instance, switch ON and OFF LED lights and small motors)

Open a terminal window.
Then ‘move’ to the ‘Desktop’ directory:
$ cd ~/Desktop
Create and open a new file to edit using the nano text editor:
$ nano gpio.py

In nano write:

```
import RPi.GPIO as GPIO
pin = 14



GPIO.setmode(GPIO.BCM)
GPIO.setup(pin, GPIO.OUT)
state = False
while 1:
GPIO.output(pin, state)


command = raw_input(“wait for key”)
state = not state
```

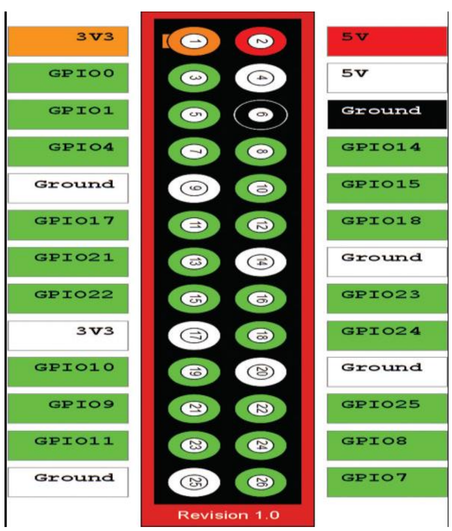
Then save and exit nano by clicking ‘Ctrl+o’, ‘return’, then ‘Ctrl+x’.

Run the program (as the root user) from the terminal:

$ sudo python gpio.py

Attach an LED to pins 6 (ground, short leg) and 8 (GPIO14, long leg).

Press return.



Read a Sensor

Learn how to find out whether a switch is ON or OFF using python.

Open a terminal window.

Then ‘move’ to the ‘Desktop’ directory:

$ cd ~/Desktop

Create and open a new file to edit using the nano text editor:

$ nano sensor.py

In nano write:

```
import RPi.GPIO as GPIO
import time
pin = 24
GPIO.setmode(GPIO.BCM)
GPIO.setup(pin, GPIO.IN)
while 1:
sensorValue = GPIO.input(pin)
print sensorValue
time.sleep(0.1)
```

Then save and exit nano by clicking ‘Ctrl+o’, ‘return’, then ‘Ctrl+x’.

Attach the sensor to 5V, pin 18 (GPIO24) and ground.

Run the program (as the root user) from the terminal:

$ sudo python sensor.py

Press return.

To stop the programme press “cntr+c”.

4.2 HTML

4.2.1 Introduction:

HTML stands for **H**yper**t**ext **M**arkup **L**anguage, and it is the most widely used language to write Web Pages.

- **Hypertext** refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.
- As its name suggests, HTML is a **Markup Language** which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

Now, HTML is being widely used to format web pages with the help of different tags available in HTML language

4.2.2 HTML Document:

Following is an example of an HTML document

```
<!DOCTYPE html>
<html>
```

```
<body>
<table style="width:100%">
<tr>
<td>
<form action="/1" method="POST">
<input type="image" name="direction" src="/static/img/left.png" >
</br>
</form>
</td>
<td>

<form action="/2" method="POST">
<input type="image" name="direction" src="/static/img/up.png" >
</br>
</form>
</td>
<td>
<form action="/3" method="POST">
<input type="image" name="direction" src="/static/img/right.png">
</br>
</form>
</td>
<td>
<form action="/6" method="POST">
<input type="image" name="direction" src="/static/img/A.png" >
</br>
</form>
</td>
<td>
<form action="/8" method="POST">
input type="image" name="direction" src="/static/img/D.png">
</br>
</form>
</td>
<td>
<form action="/7" method="POST">
<input type="image" name="direction" src="/static/img/B.png">
</br>
</form>
</td>
</tr>
<tr>
```

```
<td>
</td>
<td>
<form action="/4" method="POST">
<input type="image" name="direction" src="/static/img/down.png" >
</br>
</form>
</td>
<td>
<form action="/5" method="POST">
<input type="image" name="direction" src="/static/img/stop.png">

</br>
</form>
</td>
<td>
</td>
<td>
<form action="/9" method="POST">
<input type="image" name="direction" src="/static/img/C.png">
</br>
</form>
</td>
</tr>
</table>
</body>
</html>
```

Python language plays a keyrole in building the Agriculture robot. Python is a good platform to automate, teach, and post-process robot programs. Many choose this programming language as we can write script that calculates, records, and simulates an entire robot program instead of manually teaching every statement to a robot. This helps to quickly test and visualize solution in simulation as well as refine the program and its logic. HTML is used to design the webpage for controlling the robot from anywhere you want to.

CHAPTER 5: WORKING PROCEDURE

7.1 Main Objectives in the Working procedure:

7.1.1 Ploughing:

There are two objectives of this project. First one is to develop a model of an automated ploughing machine controlled by wireless devices. This automated ploughing machine will be developed by an artificial intelligent system which can move in the field by remote sensing through IoT. This will improve the farmers' efficiency and they will be able to farm more lands with a short time.

Another objective is to develop a remote monitoring irrigation system. One of the crucial parts of agriculture is irrigation. The necessary condition for irrigation includes the knowledge of moisture content of the soil and the climatic conditions. So, the farmers need to monitor the plantation field at regular intervals, which is a very hectic work. So, to reduce the workload of a farmer by giving an opportunity to monitor irrigation through Internet is another part of the project. We want to develop a remote irrigation system through Internet of Things (IoT). This part can be controlled by SMART phone of the user. It will greatly improve the water usage problems.

7.1.2 Seed dispenser:

In the field of agriculture, plantation begins with ploughing the land and sowing seeds. The old traditional method plough attached to an OX and tractors needs human involvement to carry the process. The driving force behind this work is to reduce the human interference in the field of agriculture and to make it cost effective. In this work, apart of the land is taken into consideration and the robot introduced localizes the path and can navigate itself with the human action. The sowing mechanism initiates with long toothed gears actuated with motors.

The complete body is divided into two parts the tail part acts as a container for seeds. The successor holds on all the electronics used for automating and actuation. The locomotion is provided with wheels covered under conveyor belts. The robot rotates in equal speed with respect to each other with the blades. For each rotation every tooth on gear will take seeds and will drop them on field. The front end tracks the path for every fixed distance and at the minimum distance it takes steps according to the instruction provided. In the agricultural robot, the consists the funnel shaped connected to the motors for the sowing of the seeds.

7.1.3 Water Sprinkler :

The water sprinkler in the agricultural robot plays an important role in the field. It automatically detects the moisture content in the soil with the help of the moisture sensor. When the soil is detected dry, the water sprinkling process is done automatically**.** Since water is essential to grow food ,a drought can pose major problem for agriculture. Drip and Water Sprinkler irrigation system are the most efficient water irrigation systems. They deliver the water directly to the roots of the plants and henceforth reducing the evaporation.

The life expectancy of the water head sprinkler can be expected up to 5-10 years. Timers can be used to schedule watering during the cooler parts of the day which further reduces water loss.The moisture sensor interface with IOT and the raspberry pi helps to send the information about the humidity , temperature to the user via WiFi module.On the Smartphone , displays the webpage that have controls ie ON,OFF.

7.2 WORKING:

The main objective of the agricultural robot is efficient utilization of resources and to reduce labor work. It can perform various tasks like soil testing, sowing of seeds, spraying of fertilizers and soil drilling. It can dig a hole in soil by drilling mechanisms and plants seed and cover hole by soil again. It can spray the pesticides using spraying mechanisms.

The robot is connected to the batteries and the respective power supply. Due to IOT and the working of raspberry pi , the robot gets switched on. All above operations are performed by using Raspberry PI controller which is master and others are slaves which performs specific operation. By using Wi-Fi, the robot can be controlled over large distance i.e. from anywhere in the world by using Raspberry. The robot takes advantage of sensors such as humidity & temperature sensor, soil moisture sensor attached to it.

The Robot consists of the 4 DC motors. The first DC motor is used for the controlling of the robot ie moving forward, backward, towards left ,towards right. The second DC motor is used to the control of the water sprinkler. The third DC motor is used to control the seed sowing. The Fourth Dc motor is used to control the ploughing tenchique.

When the switch is on the robot will be connected to network to get inputs from the phone over which in turn provides the commands to the robot for different functionalities i.e. movement of robot, sowing of seeds, ploughing and for sprinkling the data will be displayed on lcd attached to it.

CHAPTER 6: RESULTS

6.1 Result:

The projects “**RASPBERRY PI BASED AGRICULTURAL ROBOT**”, when the user switch is on the robot will be connected to network to get inputs from the phone over which in turn provides the commands to the robot for different functionalities i.e. movement of robot, sowing of seeds, ploughing and for sprinkling the data will be displayed on lcd attached to it.

When the robot is moving on a surface, it is controlled by a web page using android mobile . This can be moved forward ,backward,left and right directions using motors.

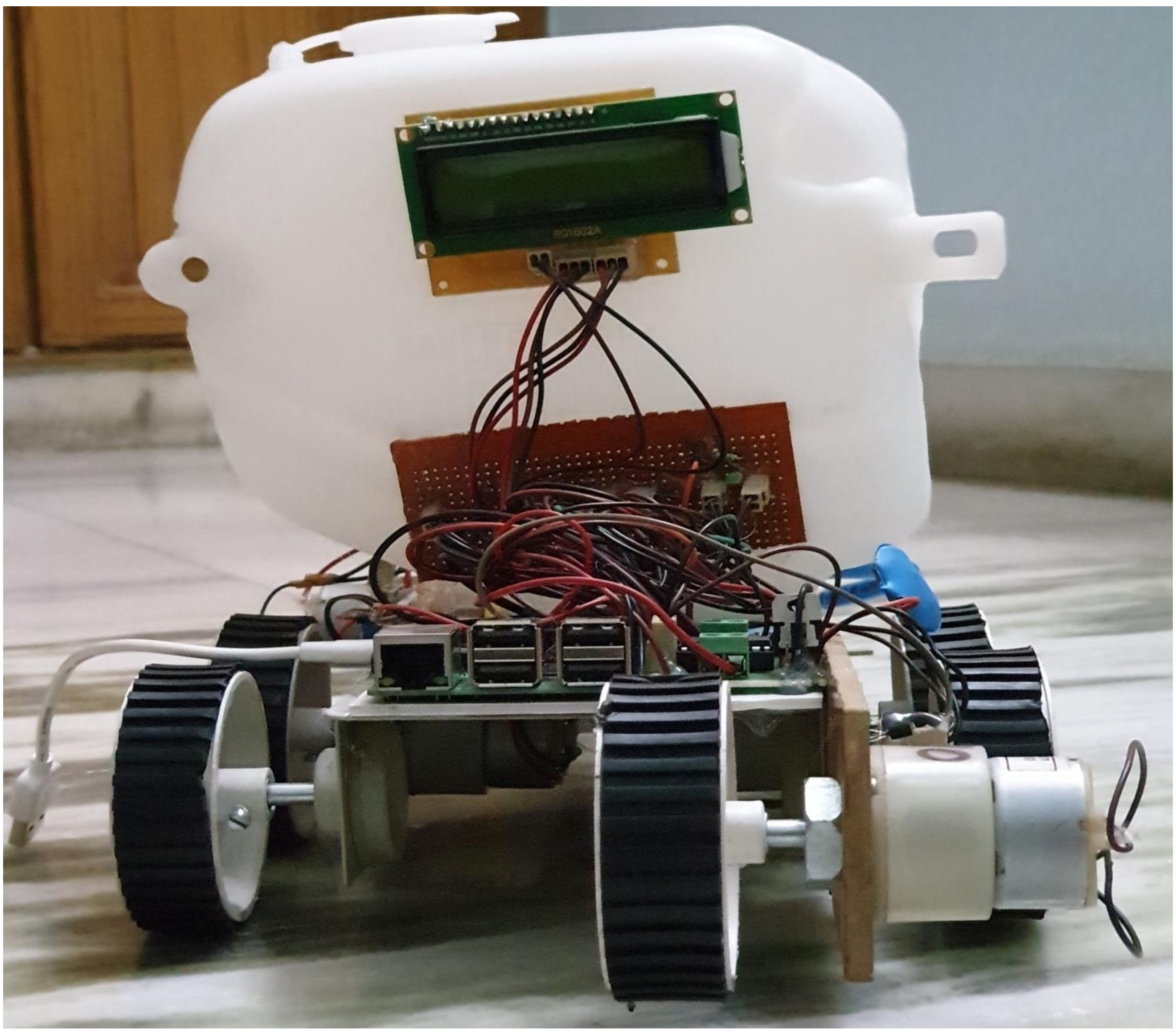


Fig.6.1: Agricultural Robot kit

The Humidity and Temperature values are displayed on the LCD screen.

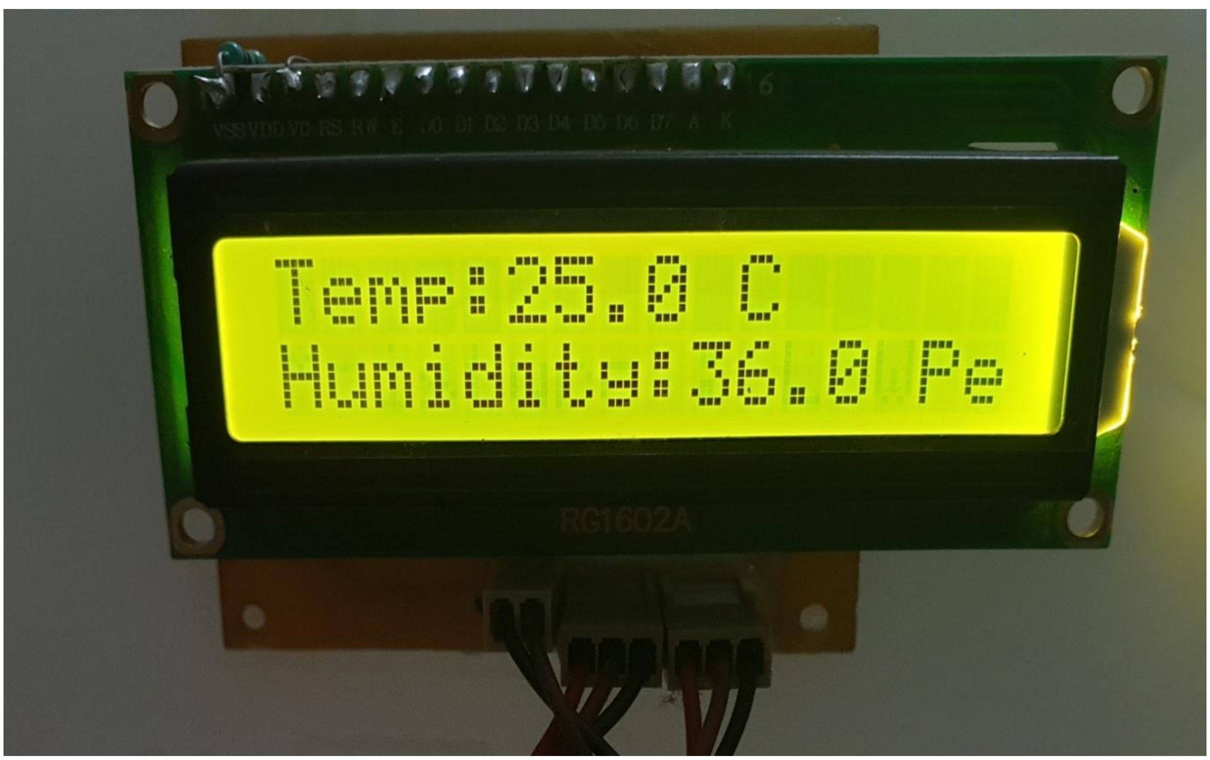


Fig.6.2: Temperature and Humidity values are displayed on the LCD

The robot takes the instructions from the android mobile which is communicating through Web page and act accordingly such as Ploughing, Seed Dispensing and Water Sprinkling.

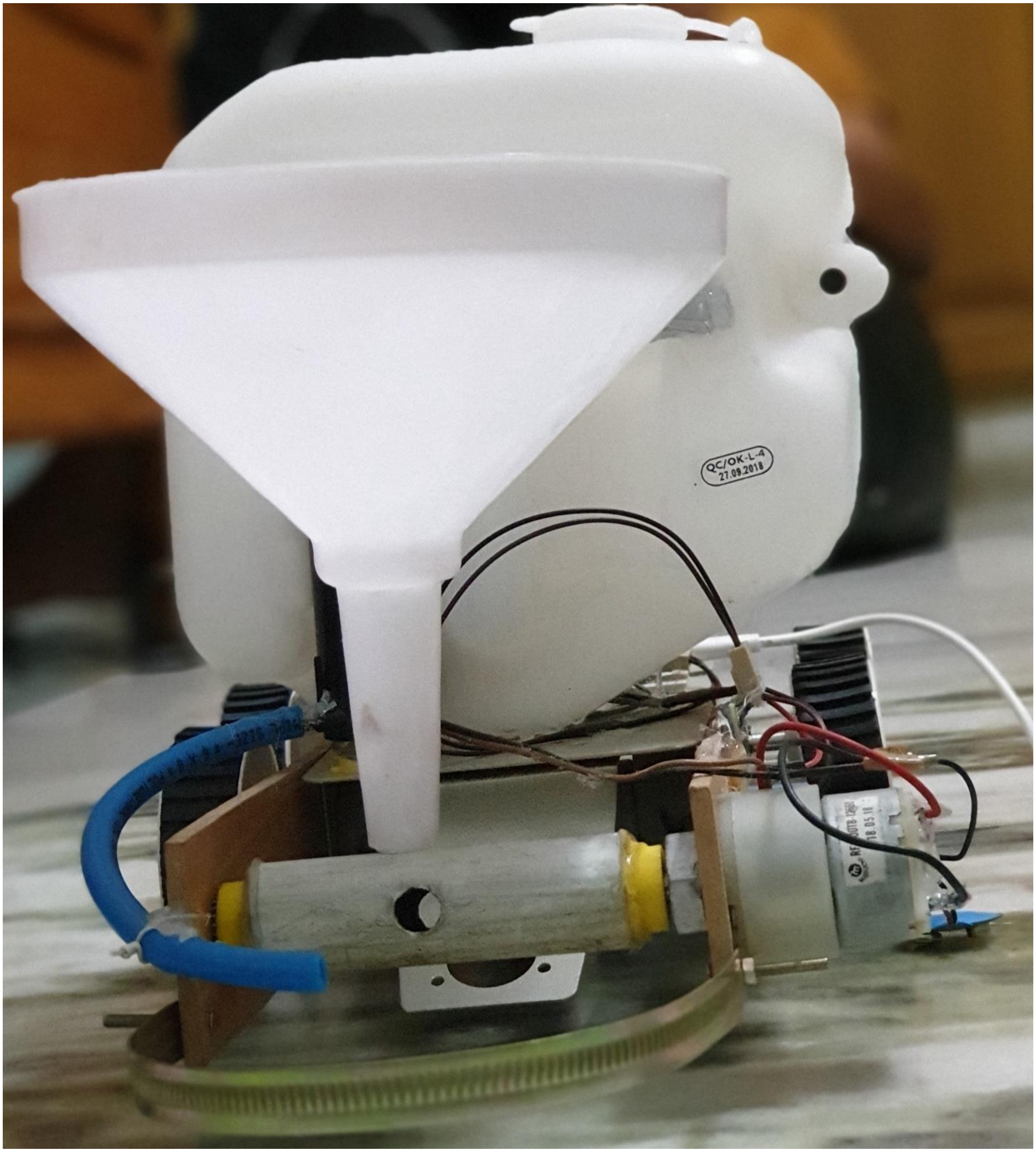


Fig.6.3: Plough, seed dispenser and water sprinkler

6.2 Conclusion:

Integrating features of all the hardware components used have been developed in it. Presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Secondly, using highly advanced IC's with the help of growing technology, the project has been successfully implemented. Thus the project has been successfully designed and tested.

6.3 Future Scope:

We can extend it by using more efficient wifi module and use of video camera to operate it remotely & increase the distance of communication so that we can access from anywhere in the world.

REFERENCES:

The sites which were used while doing this project:

1. www.wikipedia.com

2. www.allaboutcircuits.com

3.www.microchip.com

4.www.howstuffworks.com

Books referred:

1. Raj kamal –Microcontrollers Architecture, Programming, Interfacing and System Design.
2. Mazidi and Mazidi –Embedded Systems. PCB Design Tutorial
3. Embedded C –Michael.J.Pont.

APPENDICES:

CODE:

WEB PAGE CODE:

```
<!DOCTYPE html>
<html>
<body>
<table style="width:100%">
<tr>
<td>
<form action="/1" method="POST">
<input type="image" name="direction" src="/static/img/left.png" >
</br>
</form>
</td>
<td>
<form action="/2" method="POST">
<input type="image" name="direction" src="/static/img/up.png" >
</br>
</form>
</td>
<td>
<form action="/3" method="POST">
<input type="image" name="direction" src="/static/img/right.png">
</br>
</form>
</td>
<td>
<form action="/6" method="POST">
<input type="image" name="direction" src="/static/img/A.png" >
</br>
</form>
</td>
```

```
<td>
<form action="/8" method="POST">


input type="image" name="direction" src="/static/img/D.png">
</br>
</form>
</td>
<td>
<form action="/7" method="POST">
<input type="image" name="direction" src="/static/img/B.png">
</br>
</form>
</td>
</tr>
<tr>
<td>
</td>
<td>
<form action="/4" method="POST">
<input type="image" name="direction" src="/static/img/down.png" >
</br>
</form>
</td>
<td>
<form action="/5" method="POST">
<input type="image" name="direction" src="/static/img/stop.png">
</br>
</form>
</td>
<td>
</td>
<td>
<form action="/9" method="POST">
<input type="image" name="direction" src="/static/img/C.png">
</br>
</form>
</td>
</tr>
</table>
</body>
</html>
```

RASPBERRY PI CODE:

```
importRPi.GPIO as GPIO
import time
from time import sleep
import serial
import sys
importos
importAdafruit_DHT
importdatetime

LCD_RS = 2
LCD_E  = 3
LCD_D4 = 4
LCD_D5 = 17
LCD_D6 = 27
LCD_D7 = 22

soil = 11

led = 9

print "init Done - HVS"

def main():

lcd_init()

  # Send some test
lcd_string("Smart Irrigation",LCD_LINE_1)
lcd_string("     Robot  ",LCD_LINE_2)
  #time.sleep(3) # 3 second delay
```

```
GPIO.output(led, True) # voicetime.sleep(1)
GPIO.output(led, False) # voicetime.sleep(1)
GPIO.output(led, True) # voicetime.sleep(1)

GPIO.output(led, False) # voice


while True:

GPIO.output(led, True) # voice
time.sleep(0.1)
GPIO.output(led, False) # voice

        humidity, temperature = Adafruit_DHT.read_retry(18)
        buf = "Temp:%0.1f C"%(temperature)
        buf1 = "Humidity:%0.1f Perc"%(humidity)
        print "Temp:%0.1f C"%(temperature)
        print "Humidity:%0.1f Perc"%(humidity)

        lcd_string(buf,LCD_LINE_1)
        lcd_string(buf1,LCD_LINE_2)
        time.sleep(1) # 3 second delay

ifGPIO.input(soil) == 0:
          buf1 = "Moisture : HIGH"
lcd_string(buf1,LCD_LINE_2)
else:
          buf1 = "Moisture : LOW"
lcd_string(buf1,LCD_LINE_2)
```

MOTORS CODE:

```
importRPi.GPIO as GPIO
```

```
from time import sleep
import serial
#bt.write("Welcome to the project")


Motor1A = 5
Motor1B = 6

Motor2A = 13
Motor2B = 9

Motor3 = 16
Motor4 = 20

def forward():
	GPIO.output(Motor1A,GPIO.LOW)
	GPIO.output(Motor1B,GPIO.HIGH)

	GPIO.output(Motor2A,GPIO.LOW)
	GPIO.output(Motor2B,GPIO.HIGH)
	sleep(0.3)

def backward():
	GPIO.output(Motor1B,GPIO.LOW)
	GPIO.output(Motor1A,GPIO.HIGH)

	GPIO.output(Motor2B,GPIO.LOW)
	GPIO.output(Motor2A,GPIO.HIGH)
	sleep(0.1)

defturnRight():

	GPIO.output(Motor1A,GPIO.LOW)
	GPIO.output(Motor1B,GPIO.HIGH)

	GPIO.output(Motor2A,GPIO.HIGH)
	GPIO.output(Motor2B,GPIO.low)
```

```
        sleep(0.1)

defturnLeft():

        GPIO.output(Motor1A,GPIO.HIGH)
        GPIO.output(Motor1B,GPIO.LOW)

        GPIO.output(Motor2A,GPIO.LOW)
        GPIO.output(Motor2B,GPIO.HIGH)
        sleep(0.1)

def stop():

        GPIO.output(Motor1A,GPIO.LOW)
        GPIO.output(Motor1B,GPIO.LOW)

        GPIO.output(Motor2A,GPIO.LOW)
        GPIO.output(Motor2B,GPIO.LOW)
        sleep(0.1)

def m3on():
        GPIO.output(Motor3,GPIO.HIGH)

def m3off():
        GPIO.output(Motor3,GPIO.LOW)

def m4on():
        GPIO.output(Motor4,GPIO.HIGH)

def m4off():

        GPIO.output(Motor4,GPIO.LOW)
```

ROBOT CODE:

```
from flask import Flask, render_template, request, redirect, url_for,


make_response
import motors
importRPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)      # Use BCM GPIO numbers
app = Flask(__name__) #set up flask server

#when the root IP is selected, return index.html page
@app.route('/')
def index():

    returnrender_template('index.html')

@app.route('/<changepin>', methods=['POST'])
def reroute(changepin):

    changePin = int(changepin) #cast changepin to an int

    ifchangePin == 1:
        motors.turnLeft()
    elifchangePin == 2:
        motors.forward()
    elifchangePin == 3:
        motors.turnRight()
    elifchangePin == 4:
        motors.backward()

    elifchangePin == 6:
        motors.m3on()
    elifchangePin == 7:
```

```
            motors.m3off()

        elifchangePin == 8:
            motors.m4on()
        elifchangePin == 9:
            motors.m4off()

        else:
            motors.stop()


        response = make_response(redirect(url_for('index')))
        return(response)

app.run(debug=True, host='0.0.0.0', port=5000) #set up the server in debug mode to the port 8000
```