Abstract

As water supply is becoming scarce in today's world there is an urgency of adopting smart ways of irrigation. The project describes how irrigation can be handled smartly using IOT. This project aims at saving time and avoiding problems like constant vigilance. It also helps in conserving water by automatically providing water to the plants/field depending on the water requirements. This system can also prove to be helpful in agriculture, parks and lawns. The objective of this system is to detect the moisture content of the soil and depending on it sprinkle water .This entire information will be sent to the user's mobile phone.

The Internet of Things(IoT) is transforming the agriculture industry and enabling farmers to content with enormous challenges they face. Livestock monitoring, conservation monitoring and plant & soil monitoring are the challenges where IoT can be a solution. The innovative IoT applications address the issues in agriculture and increase the quality, quantity, sustainability and cost effectiveness of agricultural production. Today's large and local farms can leverage IoT to remotely monitor sensors that can detect soil moisture, crop growth and detect pest and control their smart connected harvesters and irrigation equipments.This project aims at monitoring the soil parameters like soil moisture, temperature and electrical conductivity and automates the irrigation process. Decision making is done through microcontroller. User is acknowledged about the field when there is any deviation from the expected values via text message. Along with soil parameters, plant pest detection is also included in this project. This ensures the complete system health.

CHAPTER 1

INTRODUCTION TO CONCEPT

1.1 INTRODUCTION:

Agriculture is considered as the basis of life for us as it is the main source of food and other raw materials. It plays vital role in the growth of country's economy. Growth in agricultural sector is necessary for the development of economic condition of the country. Unfortunately, many farmers still use the traditional methods of farming. In India most of the irrigation system are manually operated one's. These outdated techniques are replaced with automated techniques. This paper focuses primarily on reducing the wastage of water and minimizing the manual labor on field for irrigation . Recent advances in soil water monitoring combined with the growing popularity of Wireless Sensor Networks make the commercial use of such systems applicable for agriculture and Gardening. The system designed is programmed to irrigate at regular time intervals for predefined periods of time. In this technique, soil moisture sensors are placed root zone of plant and near the module and gateway unit handles the sensor information and transmit data to the controller which in turns operates the of control the flow of water through the valves. To give proper attention to the land located far away from the human settlement, supervisory automatic control systems like multi-terminal control systems are used since in many processes, factors like soil, salinity, irrigation, temperature, light intensity, etc. needs repeated tasks and have to work in abnormal environmental conditions of the soil and to overcome the flaws in the existing system here we are irrigating the land based on the soil humidity and at the same time the status of the irrigation is updated wirelessly to the based Android App .The proposed system will allow farmers to continuously monitor the moisture level in the field, controlling the supply remotely over the internet. When moisture goes below a certain level, sprinklers would be turned on automatically, thus achieving optimal irrigation using Internet of Things.

Internet of Things represents a general concept for the ability of network devices to sense and collect data from the world around us, and then share that data across the Internet where it can be processed and utilized for various interesting purposes. Internet of Things is very quickly becoming a reality. We can see the proof of it around us. Our devices are getting smarter each day from smartphones to smart TV to smart car to Smart kitchen. Everything is now getting connected to Internet. Internet of Things (IoT) describes a network of physical objects that connect to each other through the internet. Objects, or 'things' can transfer information wirelessly without requiring human interaction. A 'thing' can be any object that can be assigned an IP address and provided with the ability to transfer data over a network. A Thing, in the Internet of Things, can be a person with a heart monitor implant, a farm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low -- or any other natural or man-made object that can be assigned an IP address and provided with the ability to transfer data over a network. These devices collect useful data with the help of various existing technologies and then

autonomously flow the data between other devices. Current market examples include smart thermostat systems and washer/dryers that utilize Wi-Fi for remote monitoring.

Internet of Things or IoT is an architecture that comprises specialized hardware boards, Software systems, web APIs, protocols which together creates a seamless environment which allows smart embedded devices to be connected to internet such that sensory data can be accessed and control system can be triggered over internet. Also devices could be connected to internet using various means like WiFi, Ethernet and so on. Furthermore devices may not needed to be connected to internet independently.

1.1.1 PROCESSING UNIT

Arduino , Raspberry Pi, Intel Edison, Intel Galileo , Netduino are the best IoT boards on which several applications can be executed. Every board has its own features, advantages and disadvantages over one another.

1.1.2 SENSING UNIT

Usually composed two sub units: sensors and analog-to-digital Converters (ADCs). The analog signal produced by the sensors are converted to digital signals by the ADC, and fed into the processing unit.

1.1.3 IOT IN AGRICULTURE

Agriculture IoT helps in increasing crop productivity by way of managing and controlling the activities like Crop water management - Adequate water supply is an essence for agriculture and the crops can be damaged in either of situation of excess of water supply or in shortage of water supply. In areas of drought condition, IoT can prove to be a great value as it manages the limited water supply smartly with least wastage of water resource. Precision agriculture - The level of accuracy of temperature, moisture, pH of the soil affects the crop productivity to a greater extent. Higher the level of accuracy, lower would be the chances of crops being damaged.

CHAPTER II

LITERATURE SURVEY

2.1 LITERATURE SURVEY

A Remote Measurement and Control System for Greenhouse Based on GSM-SMS [4] the proposed system introduced a GSM-SMS remote measurement and control system for greenhouse based on PC-based database system connected with base station. Base station is developed by using a microcontroller, GSM module, sensors and actuators. In practical operation, the central station receives and sends messages through GSM module. Criterion value of parameters to be measured in every base station is set by central station, and then in base stations parameters including the air temperature, the air humidity. Indu et al. (2013) [5] mainly focuses on reviews in the field of remote monitoring and control, the technology used and their potential advantages. The paper proposes an innovative GSM/Bluetooth based remote controlled embedded system for irrigation. The system sets the irrigation time depending on the temperature and humidity reading from sensors and type of crop and can automatically irrigate the field when unattended. Information is exchanged between far end and designed system via SMS on GSM network. A WIFI module is also interfaced with the main microcontroller chip which eliminates the SMS charges when the user is within the limited range of few meters to the designated system. The system informs users about many conditions like status of electricity, dry running motor, increased temperature, water content in soil and smoke via SMS on GSM network or by Bluetooth. In [6], R.Suresh et al. (2014) mentioned about using automatic microcontroller based rain gun irrigation system in which the irrigation will take place only when there will be intense requirement of water that save a large quantity of water. These systems bring a change to management of field resource where they developed a software stack called Android is used for devices that include an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language. Mobile phones have almost become an integral part of us serving multiple needs of humans. This application makes use of the GPRS feature of mobile phone as a solution for irrigation control system. These system covered lower range of agriculture land and not economically affordable.IOT SMS alarm system based on SIM900A [7], an IOT alarm system based on SIM900A module of SIMCOM Company was designed for greenhouse. The system can gather environmental parameters such as air temperature and air humidity. Meanwhile, with the use of AT command, this system can also realize SMS automatic sending and receiving, environmental parameters overrun alarm and insufficient balance alarm. Through the system setting, the alarm message can be sent to the user-specified mobile phone automatically no matter what the users' location is. This system as a typical application of IOT in the agriculture has got some satisfactory results in the actual operation.

CHAPTER III

HARDWAE OVERVIEW

3.1 Hardware Parts:

The machines, wiring, and other physical components of a computer or other electronic system.

1. Node Mcu	2. Soil moisture sensor	3. Rain sensor
4. Humidity & Temperature	5.Electromagantic relay	6.pump
7.Power supply	8.Capcitive Touch sensor	9.Pvc lamp holder

3.2 Node MCU:

ESP8266NodeMCUis an open source IOT platform. It includes firmware which runs on the low cost Wi-Fi enabled ESP8266 Wi-Fi SOC from Espress if Systems, and hardware which is based on the ESP-12 module. It has GPIO, SPI, I2C, ADC, PWM AND UART pins for communication and controlling other peripherals attached to it. On board NodeMCU has CP2102 IC which provides USB to TTL functionality. In this IOT Fire Alarm, we are using one GPIO pin to get the digital data from the infrared sensor.The NodeMCU ESP8266 collect the information from the fire sensor and send it to Blynk App every second. NodeMCU is best known for IoT open-source platforms which added firmware that easily runs on the low cost as well. Moreover, this NodeMCU ESP8266 has even included the hardware based on the ERP-12 module. It is an open-source Lua based firmware and development board specially targeted for IoT based applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems and hardware which is based on the ESP-12 module, and like this, it can also be programmed using Arduino IDE and can act as both WiFi Hotspot or can connect to one. It has one Analog Input Pin, 16 Digital I/O pins along with the capability to connect with serial communication protocols like SPI, UART, and I2C. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects. Its applications include prototyping for IoT devices, low powered battery-operated applications, and projects requiring I/O interface with Bluetooth and WiFi capabilities.

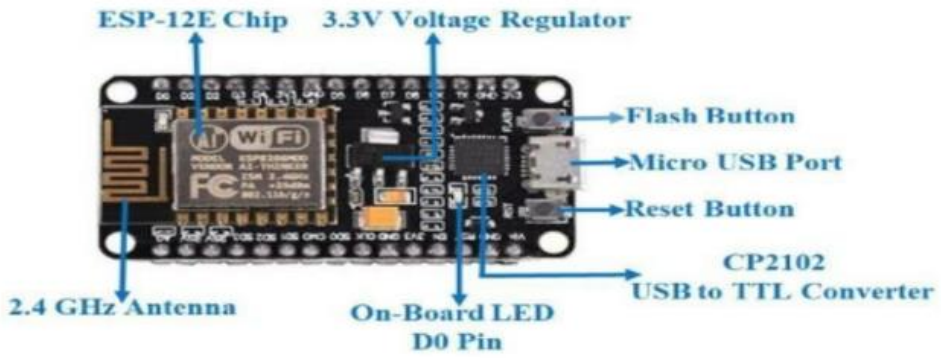


Figure 1 Node MCU

Specifications

Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106

Operating Voltage: 3.3V

Input Voltage: 7-12V

Digital I/O Pins (DIO): 16

Analog Input Pins (ADC): 1

UARTs: 1

SPIs: 1

I2Cs: 1

Flash Memory: 4 MB

SRAM: 64 KB

Clock Speed: 80 MHz

Description

The NodeMCU development board can be easily programmed with Arduino IDE all you need is the micro USB cable to connect to your computer. Check this NodeMCU tutorial to learn how to program your NodeMCU Development board and get going. Programming NodeMCU ESP8266 with Arduino IDE The NodeMCU Development Board can be easily programmed with Arduino IDE since it is easy to use .Programming NodeMCU with the Arduino IDE will hardly take 5-10 minutes. All you need is the Arduino IDE, a USB cable and the NodeMCU board itself. You can check this Getting Started Tutorial for NodeMCU to prepare your Arduino IDE for NodeMCU. Uploading your first program Once Arduino IDE is installed on the computer, connect the board with the computer usingthe USB cable. Now open the Arduino IDE and choose the correct board by selecting Tools>Boards>NodeMCU1.0 (ESP-12E Module),

and choose the correct Port by selecting Tools>Port. To get it started with the NodeMCU board and blink the built-in LED, load the example code by selecting Files>Examples>Basics>Blink. Once the example code is loaded into your IDE, click on the ‘upload’ button given on thetop bar. Once the upload is finished, you should see the built-in LED of the board blinking.

3.3 Soil moisture sensor:

Soil moisture sensors measure the volumetric water content in soil. Since the direct gravimetric measurement of free soil moisture requires removing, drying, and weighing 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.

Technology

Technologies commonly used to indirectly measure volumetric water content (soil moisture) include:

- Frequency Domain Reflectometry (FDR): The dielectric constant of a certain volume element around the sensor is obtained by measuring the operating frequency of an oscillating circuit.
- Time Domain Transmission (TDT) and Time Domain Reflectometry (TDR): The dielectric constant of a certain volume element around the sensor is obtained by measuring the speed of propagation along a buried transmission line; (see also: TDR moisture sensor)
- Neutron moisture gauges: The moderator properties of water for neutrons are utilized to estimate soil moisture content between a source and detector probe.
- Soil resistivity: Measuring how strongly the soil resists the flow of electricity between two electrodes can be used to determine the soil moisture content.

- Galvanic cell: The amount of water present can be determined based on the voltage the soil produces because water acts as an electrolyte and produces electricity. The technology behind this concept is the galvanic cell.

Application

Agriculture

Measuring soil moisture is important for agricultural applications to help farmers manage their irrigation systems more efficiently. Knowing the exact soil moisture conditions on their fields, not only are farmers able to generally use less water to grow a crop, they are also able to increase yields and the quality of the crop by improved management of soil moisture during critical plant growth stages.

Landscape irrigation

In urban and suburban areas, landscapes and residential lawns are using soil moisture sensors to interface with an irrigation controller. Connecting a soil moisture sensor to a simple irrigation clock will convert it into a "smart" irrigation controller that prevents irrigation cycles when the soil is already wet, e.g. following a recent rainfall event.

Golf courses are using soil moisture sensors to increase the efficiency of their irrigation systems to prevent over-watering and leaching of fertilizers and other chemicals into the ground.

Research

Soil moisture sensors are used in numerous research applications, e.g. in agricultural science and horticulture including irrigation planning, climate research, or environmental science including solute transport studies and as auxiliary sensors for soil respiration measurements.

Simple sensors for gardeners

Relatively cheap and simple devices that do not require a power source are available for checking whether plants have sufficient moisture to thrive. After inserting a probe into the soil for approximately 60 seconds, a meter indicates if the soil is too dry, moist or wet for plants.

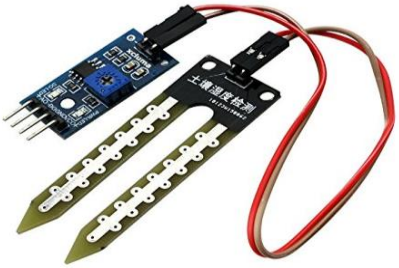


Figure 2 soil moisture sensor

3.4 Rain Sensor:

A **rain sensor** or *rain switch* is a switching device activated by rainfall. There are two main applications for rain sensors. The first is a water conservation device connected to an automatic irrigation system that causes the system to shut down in the event of rainfall.

Types of Rains Sensors

Depending on the type of yard you have, where you live, and how often it rains, you will want to determine the type of sensor to use. Here are some of the different rain sensors:

- Rainfall collection cup - This sensor stops the sprinklers from turning on when the rainfall cup fills to a certain level. An issue to be aware of with this kind of sensor is a false fill up. Leaves, debris, and even bugs or pests can fill up the collection cup causing the sprinklers to shut off. They often work through wireless communication.
- Expansion disk - This type of rain sensor is becoming more and more popular all the time. This particular sensor uses a cork disk that expands when there is rainfall to measure the level of precipitation. A pressure switch is then used to break the electrical connection which causes the sprinklers to turn on and off.

The second is a device used to protect the interior of an automobile from rain and to support the automatic mode of windscreen wipers.

How does this operate?

The rain sensor works on the principle of total internal reflection. ... An infrared light beams at a 45-degree angle on a clear area of the windshield from the sensor-inside the car. When it rains, the wet glass causes the light to scatter and lesser amount of light gets reflected back to the sensor

An additional application in professional satellite communications antennas is to trigger a rain blower on the aperture of the antenna feed, to remove water droplets from the my lar cover that keeps pressurized and dry air inside the wave-guides.

Figure 3 Rain Sensor

Pin Configuration

The pin configuration of this sensor is shown below. This sensor includes four pins which include the following.

- Pin1 (VCC): It is a 5V DC pin
- Pin2 (GND): it is a GND (ground) pin
- Pin3 (DO): It is a low/ high output pin
- Pin4 (AO): It is an analog output pin

Specifications

The specifications of the rain sensor include the following.

- This sensor module uses good quality of double-sided material.
- Anti-conductivity & oxidation with long time use
- The area of this sensor includes 5cm x 4cm and can be built with a nickel plate on the side
- The sensitivity can be adjusted by a potentiometer
- The required voltage is 5V
- The size of the small PCB is 3.2cm x 1.4cm
- For easy installation, it uses bolt holes
- It uses an LM393 comparator with wide voltage
- The output of the comparator is a clean waveform and driving capacity is above 15mA.

3.5 Humidity sensor:

A humidity sensor is an electronic device that measures the humidity in its environment and converts its findings into a corresponding electrical signal. Humidity sensors vary widely in size and functionality; some humidity sensors can be found in handheld devices (such as smartphones), while others are integrated into larger embedded systems (such as air quality monitoring systems). Humidity sensors are commonly used in the meteorology, medical, automobile, HVAC and manufacturing industries.

Humidity sensors can be divided into two groups, as each category uses a different method to calculate humidity: relative humidity (RH) sensors and absolute humidity (AH) sensors. Relative humidity is calculated by comparing the live humidity reading at a given temperature to the maximum amount of humidity for air at the same temperature. RH sensors must therefore measure temperature in order to determine relative humidity. In contrast, absolute humidity is measured without reference to temperature.

The two most common RH sensors are the capacitive and resistive humidity sensors. Capacitive sensors use two electrodes to monitor the capacitance (i.e. the ability to store an electric charge) of a thin metal strip placed between them. The metal's capacitance increases or decreases at a rate that is directly proportional to the change of humidity in the sensor's environment. The difference in charge (voltage) generated by an increase in humidity is then amplified and sent to the embedded computer for processing. Resistive humidity sensors operate on a different principle. These sensors utilize a small polymer comb that increases and decreases in size as the humidity changes, which directly affects the system's ability to store charge.

Thermal humidity sensors are used to measure absolute humidity. Unlike RH sensors, thermal humidity sensors utilize two probes, one to measure dry nitrogen and one to measure the air of its surrounding environment. When humidity is collected on the exposed probe, the difference in thermal conductivity is perceived by the sensor, and AH is calculated.

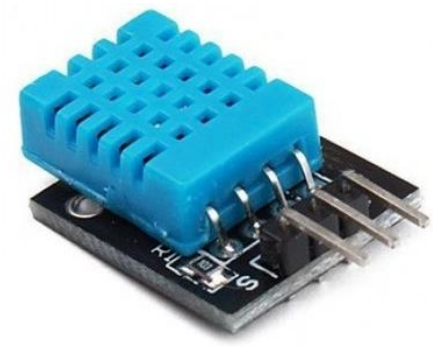


Figure 4 Humidity Sensor

Temperature sensor:

The thermistor is interfaced with Arduino to fetch the data of temperature in the room. The data is processed if the temperature is high fan speed is more relatively if the temperature is low fan speed is low. Moreover if the temperature is below the threshold set then fan will be in off condition. The data related to temperature, fan speed is displayed on the LCD for user interaction. The fan speed is controlled relatively with the temperature using PWM pins available on the Arduino. As duty cycle of PWM signal increases the fan speed increases and the same is true conversely.

LM35 is a device which converts the physical signal into electrical signal. That's why this is known as the transducer. It is calibrated with the environmental temperature and it is linearly varies with the temperature and its output is in volt. There is no need of external calibration to provide the accuracy of the LM35 at room temperature which is about ±¼°C. Minimum temperature that can be measured by the LM35 device is - 55°C. And maximum temperature that can be measured by LM35 is 150°C. Calibration of LM35 is done by trimming at the water level. To make the interfacing of control circuitry and readout circuitry very easy, low impedance at output side, output which is linear and precise inherent calibration of LM35 plays an important role. Temperature sensor takes a very low current of order 60 μA from the input supply. Heat loss in the LM35 is very less degree of around 0.1°C. LM35 can work in the range of -50°C to +150° which is the rated value. Another device which is also a temperature sensor of the family of LM35 known as LM35C which ranges from -40°C to +110°C. LM35 costs.

FEATURES:-

Figure 5 LM35 Sensor

- Low cost
- Accuracy is about ±¼°C
- Linearly varies with temperature (in centigrade)
- It can measure the temperature from -55°C to 150°C
- The current drawn from the supply is very less about 60 μA
- There is negligible heat in LM35
- Low impedance output; for a load of 1 mA about 0.1 ohm.17
- Calibrated linearly with Celsius. • Input voltage can vary from 4 volts to 30 volt

3.6 Electromagnetic relay:

The electromagnetic relay operates on the principle of a split-phase induction motor. The initial force is developed on the moving element that may be disc or another form of the rotor of the non-magnetic moving element. The force is developed by the interaction of electromagnetic fluxes with eddy current, that is induced in the rotor by these fluxes. The different type of structure has been used for obtaining the phase difference in the fluxes.

This coil is usually energised by current flowing in the single coil wound on a magnetic structure containing an air gap. The air-gap fluxes produce by the initializing current is split into two flux displace in time-space and by a shaded ring. The shaded ring is made up of the copper ring that encircles the part of the pole face of each pole. FIG:6 INDUCTION DISC RELAY The disc is made up of aluminum. The inertia of the aluminum disc is very less.. Hence they need less deflecting torque for its movement. The two rings have the current induced in them by the alternating flux of the electromagnetic. The magnetic field develops from the current produces the flux in the portion of the iron ring surrounded by the ring to lag in phase by 40° to 50° behind the flux in the un shaded portion of the pole.

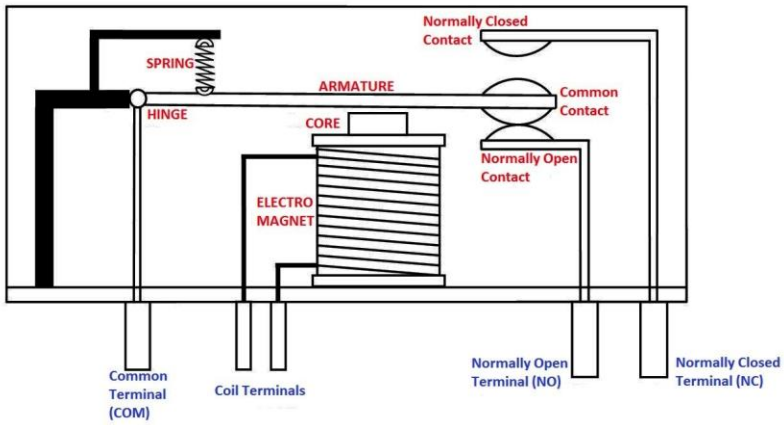


Figure 6 Electro Magnetic Relay Cross Section

Electromagnetic relays are those relays which are operated by electromagnetic action. Modern electrical protection relays are mainly micro processor based, but still electromagnetic relay holds its place. It will take much longer time to replace all electromagnetic relays by micro processor based static relays. So before going through detail of protection relay system we should review the various types of electromagnetic relays**.**

3.6.1 Electromagnetic Relay Working

Practically all the relaying device is based on either one or more of the following types of electromagnetic relays.

1. Magnitude measurement,
2. Comparison,
3. Ratio measurement.

Principle of electromagnetic relay working is on some basic principles. Depending upon working principle these can be divided into following types of electromagnetic relays.

1. Attracted Armature type relay,
2. Induction Disc type relay,
3. Induction Cup type relay,
4. Balanced Beam type relay,
5. Moving coil type relay,
6. Polarized Moving Iron type relay.

Attraction Armature Type Relay

Attraction armature type relay is the most simple both in construction as well as in its working principle. These types of electromagnetic relays can be utilized as either magnitude relay or ratio relay. These relays are employed as auxiliary relay, control relay, over current, under current, over voltage, under voltage and impedance measuring relays.

Figure 7 RELAY Single Channel

3.7 Pump:

This DC operated mini submersible water pump is ideal for small vending machines and other applications, in which a small amount of water has to be pumped. The operating voltage for this DC pump is between 3 to 12V and can be easily controlled with development boards like Arduino, Raspberry Pi, ESP, and other microcontrollers, so also frequently used in DIY electronics projects and hobby projects. The pumping height for this mini DC water pump.

A water pump is a part of a fire sprinkler system's water supply and powered by electric, diesel or steam. The pump intake is either connected to the public underground water supply piping, or a static water source (e.g., tank, reservoir, lake). The pump provides water flow at a higher pressure to the sprinkler system risers and hose standpipes. A fire pump is tested and listed for its use specifically for fire service by a third-party testing and listing agency, such as UL or FM Global. The main code that governs fire pump installations in North America is the National Fire Protection Association's NFPA 20 Standard for the Installation of Stationary Fire Pumps for Fire Protection.

The pumping of water is a basic and practical technique, far more practical than scooping it up with one's hands or lifting it in a hand-held bucket. This is true whether the water is drawn from a fresh source, moved to a needed location, purified, or used for irrigation, washing, or sewage treatment, or for

evacuating water from an undesirable location. Regardless of the outcome, the energy required to pump water is an extremely demanding component of water consumption. All other processes depend or benefit either from water descending from a higher elevation or some pressurized plumbing system.

The ancient concept of the aqueduct took simple and eloquent advantage of maintaining elevation of water for as long and far a distance as possible. Thus, as water moves over great distances, it retains a larger component of its potential energy by spending small portions of this energy flowing down a slight gradation. A useful aqueduct system ultimately depends on a fresh water source existing at a higher elevation than the location where the water can be of use. Gravity does all the work. In all other instances, pumps are necessary.

Figure 8 Pump

3.8 Power supply:

A power supply is an electrical device that supplies electric power to an electrical load. The primary function of a power supply is to convert electric current from a source to the correct voltage, current, and frequency to power the load. As a result, power supplies are sometimes referred to as electric power converters. Some power supplies are separate standalone pieces of equipment, while others are built into the load appliances that they power. Examples of the latter include power supplies found in desktop computers and consumer electronics devices. Other functions that power supplies may perform include limiting the current drawn by the load to safe levels, shutting off the current in the event of an electrical fault, power conditioning to prevent electronic noise or voltage surges on the input from reaching the load, power-factor correction, and storing energy so it can continue to power the load in the event of a temporary interruption in the source power (uninterruptible power supply). All power supplies have a power input connection, which receives energy in the form of electric current from a source, and one or more power output connections that deliver current to the load. The source power may come from the electric power grid, such as an electrical outlet, energy storage devices such as batteries or fuel cells, generators or alternators, solar power converters, or another power supply. The input and output are usually hardwired circuit connections, though some power supplies employ wireless energy transfer to

power their loads without wired connections. Some power supplies have other types of inputs and outputs as well, for functions such as external monitoring and control.

All power supplies have a *power input* connection, which receives energy in the form of electric current from a source, and one or more *power output* connections that deliver current to the load. The source power may come from the electric power grid, such as an electrical outlet, energy storage devices such as batteries or fuel cells, generators or alternators, solar power converters, or another power supply. The input and output are usually hardwired circuit connections, though some power supplies employ wireless energy transfer to power their loads without wired connections. Some power supplies have other types of inputs and outputs as well, for functions such as external monitoring and control.

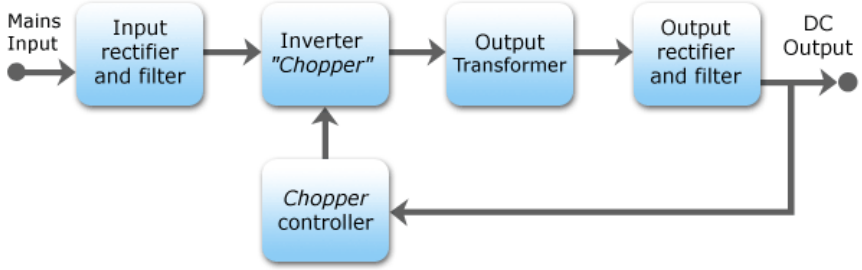


Figure 9 Rectifier Block Diagram

Things That You Will Need...

- Piece of Vero board
- Four 1N4001 diodes
- LM7812 regulator
- Transformer that has an output of 14v - 35v AC with an output current between 100mA to 1A, depending how much power you will need. (I found a 16v 200mA transformer in a broken alarm clock.)
- 1000uF - 4700uF capacitor
- 1uF capacitor
- Two 100nF capacitors
- Jumper wires (I used some plain wire as jumper wires)
- Heatsink (optional)

Also you will need the tools to make this power supply...

- Soldering iron
- Wire cutters
- Wire strippers
- A thing you can cut veroboard tracks.
- Hot glue (To hold components down and make the power supply physically strong and sturdy.)
- And some other tools that you might find helpful.

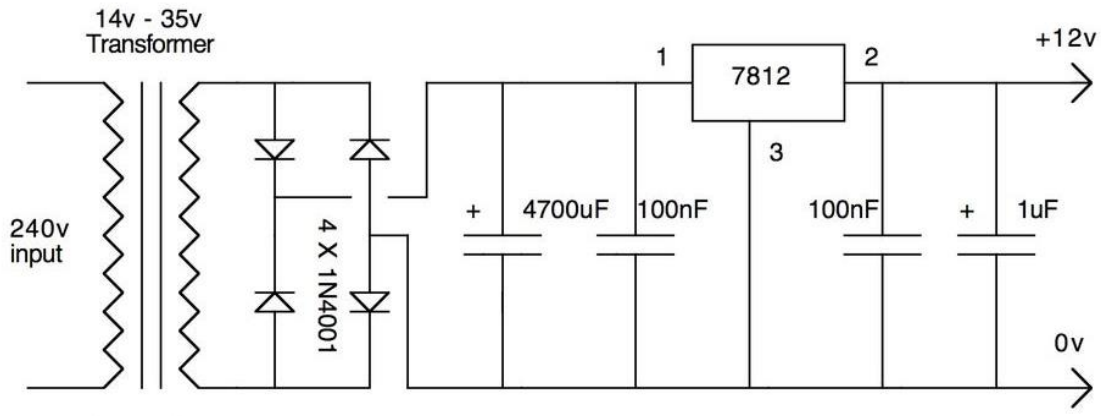


Figure 10 Circuit diagram

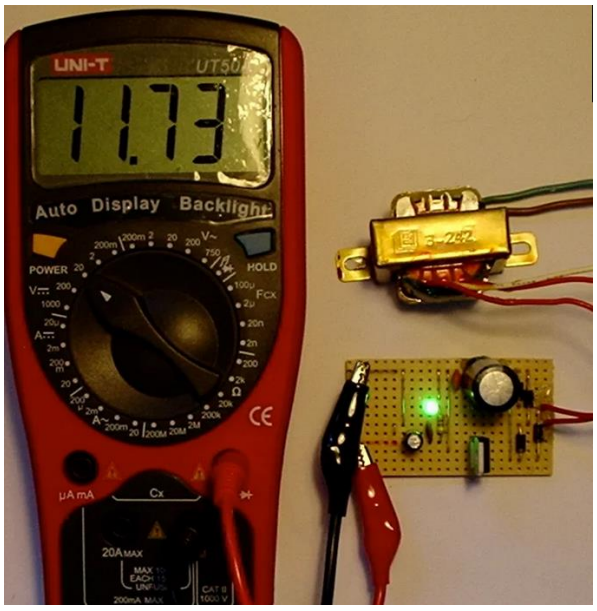


Figure 11 Power supply

3.10 9V battery:

The **nine-volt battery**, or **9-volt battery**, is a common size of battery that was introduced for the early transistor radios. It has a rectangular prism shape with rounded edges and a polarized snap connector at the top. This type is commonly used in smoke detectors, gas detectors, clocks, walkie-talkies, electric guitars and effects units.

Most nine-volt alkaline batteries are constructed of six individual 1.5 V LR61 cells enclosed in a wrapper.

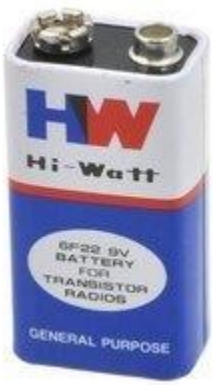


Figure 12 9v Battery

3.10 Touch sensor:

In electrical engineering, capacitive sensing (sometimes capacitance sensing) is a technology, based on capacitive coupling, that can detect and measure anything that is conductive or has a dielectric different from air. Many types of sensors use capacitive sensing, including sensors to detect and measure proximity, pressure, position and displacement, force, humidity, fluid level, and acceleration. Human interface devices based on capacitive sensing, such as touchpads, can replace the computer mouse. Digital audio players, mobile phones, and tablet computers use capacitive sensing touchscreens as input devices. Capacitive sensors can also replace mechanical buttons.

Design:

Capacitive sensors are constructed from many different media, such as copper, indium tin oxide (ITO) and printed ink. Copper capacitive sensors can be implemented on standard FR4 PCBs as

well as on flexible material. ITO allows the capacitive sensor to be up to 90% transparent (for one layer solutions, such as touch phone screens). Size and spacing of the capacitive sensor are both very important to the sensor's performance. In addition to the size of the sensor, and its spacing relative to the ground plane, the type of ground plane used is very important. Since the parasitic capacitance of the sensor is related to the electric field's (e-field) path to ground, it is important to choose a ground plane that limits the concentration of e-field lines with no conductive object present.

Designing a capacitance sensing system requires first picking the type of sensing material (FR4, Flex, ITO, etc.). One also needs to understand the environment the device will operate in, such as the full operating temperature range, what radio frequencies are present and how the user will interact with the interface.

There are two types of capacitive sensing system: mutual capacitance, where the object (finger, conductive stylus) alters the mutual coupling between row and column electrodes, which are scanned sequentially; and self- or absolute capacitance where the object (such as a finger) loads the sensor or increases the parasitic capacitance to ground. In both cases, the difference of a preceding absolute position from the present absolute position yields the relative motion of the object or finger during that time. The technologies are elaborated in the following section.

Surface capacitance

In this basic technology, only one side of the insulator is coated with conductive material. A small voltage is applied to this layer, resulting in a uniform electrostatic field. When a conductor, such as a human finger, touches the uncoated surface, a capacitor is dynamically formed. Because of the sheet resistance of the surface, each corner is measured to have a different effective capacitance. The sensor's controller can determine the location of the touch indirectly from the change in the capacitance as measured from the four corners of the panel: the larger the change in capacitance, the closer the touch is to that corner. With no moving parts, it is moderately durable, but has low resolution, is prone to false signals from parasitic capacitive coupling, and needs calibration during manufacture. Therefore, it is most often used in simple applications such as industrial controls and interactive kiosks.

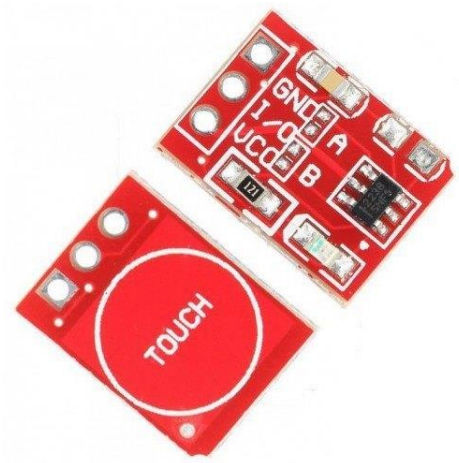


Figure 13 TP22 Touch sensor

The TTP223 is a touch pad detector IC replicating a single tactile button. This touch detection IC is designed for replacing traditional direct button key with diverse pad size.

Features

1. Input Voltage: 3.3V - 5.5V DC
2. Module size: 15mm x 11mm
3. Stable touching detection of human body for replacing traditional direct switch key.

Effect	B	A
Jog high output	No weld	No weld
Self lock high output	weld	No weld
Jog low loutput	No weld	weld
Low self lock output	weld	weld

3.11 Jumpers:

These set of 10 male to male connecting wires a.k.a jumper wires that can be used to make secure and fast connection for your prototypes. They can used on breadboard or on female berg sticks. Both the side of the wire has male pins. The color of all four wires will be different but the exact color might vary from that of the picture. These good quality male to male jumper wires can be used multiple times, and has an approximate length of 20cm.

Figure 14 Jumpers

3.12 5A lamp holder & lamp:

A lightbulb socket, light socket, lamp socket or lampholder is a device which mechanically supports and provides electrical connections for a compatible electric lamp. Sockets allow lamps to be safely and conveniently replaced (re-lamping). There are many different standards for lampholders, including early *de facto* standards and later standards created by various standards bodies.

Figure 15 Holder & Lamp

CHAPTER IV

SOFTWARE OVERVIEW

4.1 Softwares used:

- Arduino IDE
- Fritizing
- Portus
- Blynk as Cloud
- C-language Script

4.2 Arduino IDE:

Arduino IDE where IDE stands for Integrated Development Environment - An official software introduced by Arduino.cc, that is mainly used for writing, compiling and uploading the code in the Arduino Device. Almost all Arduino modules are compatible with this software that is an open source and is readily available to install and start compiling the code on the go.In this post, I'll take you through the brief Introduction of the Software, how you can install it, and make it ready for your required Arduino module. Let's dive in and get down to the nitty-gritty of this Software.

• Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.

• It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process .

• It is easily available for operating systems like MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.

• A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more.

• Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.

• The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.

• The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.

- This environment supports both C and C++ languages.

Features

- Sketch Editing Tools
- Libraries
- Serial Monitor
- Programmer Functions
- Burn Bootloader
- Sketches Management
- Sharing
- Auto Format
- User Preferences
- Fix Encoding & Reload
- Board Selection & Management
- Project Documentation
- Sketch Archive
- Port Menu
- Sketchbook
- Sketches Management

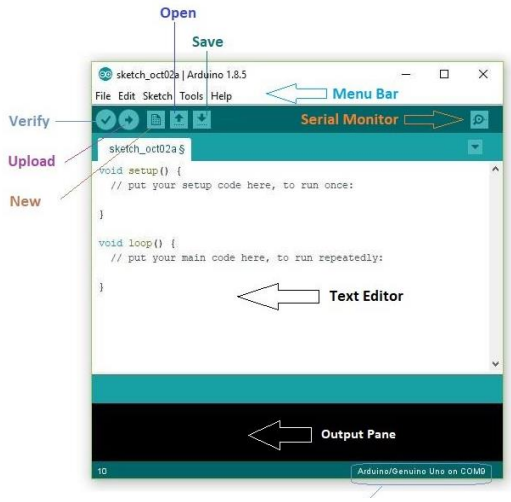


Figure 16 Ide Work Space

Advantages

The main benefits of Arduino IDE can be seen in its ability to function as an on-premise application and as an online editor, direct sketching, board module options, and integrated libraries. Specifically, here are the advantages users can expect from the system: Board Module Options The tool is armed with a board management module, wherein users can choose which board they want to use. If another board is needed, they can seamlessly select another option from the dropdown menu. PORT data is updated automatically whenever modifications are made on the board or if a new board is chosen. Direct Sketching Arduino IDE lets users can come up with sketches from within its text editor. The process simple and straightforward. What's more, the text editor has additional features that promote a more interactive experience. Documentation The tool gives users an option to have their projects documented. The feature makes it possible for them to track their progress and be aware of any changes made. In addition, documentation lets other programmers utilize the sketches on their very own boards. Sketch Sharing Arduino IDE allows users to share their sketches to other programmers. Each sketch comes with their own online link for users to share with their colleagues or friends. This feature is only available in the cloud version. Integrated Libraries The software has hundreds of integrated libraries. These libraries were made and openly shared by the Arduino community. Users can take advantage of this for their own projects without involving third-party installations. External Hardware Support41 While the tool itself is specifically intended for Arduino boards, it also has native connection support for third-party hardware. This ensures extensive use of Arduino IDE without being tied down to proprietary boards.

4.3 Fritizing

Fritzing is an open-source hardware initiative that makes electronics accessible as a creative material for anyone. We offer a software tool, a community website and services in the spirit of Processing and Arduino, fostering a creative ecosystem that allows users to document their prototypes, share them with others, teach electronics in a classroom, and layout and manufacture professional pcbs. Fritzing is an open-source initiative to develop amateur or hobby CAD software for the design of electronics hardware, to support designers and artists ready to move from experimenting with a prototype to building a more permanent circuit. It was developed at the University of Applied Sciences Potsdam. The source code is open-source, but download of executable files is paid. Fritzing software is an interesting open-source initiative to support designers, artists, researchers and hobbyists to work creatively with interactive electronics and develop electronic projects. Fritzing helps you learn more about electronic circuits, to document your projects and even let's you prepare them for production.The software is created in the spirit of the Processing programming language and the Arduino microcontroller and allows a designer, artist, researcher, or hobbyist to document their Arduino-based prototype and create a PCB layout for manufacturing. The associated website helps users share and discuss drafts and experiences as well as to reduce manufacturing costs. Fritzing can be seen as an electronic design automation (EDA) tool for non-engineers: the input metaphor is inspired by the environment of designers (the breadboard-based prototype), while the output is focused on accessible means of production. As of December 2, 2014 Fritzing has made a code view option, where one can modify code and upload it directly to an Arduino device.

Developer

The Fritzing source code is written in C++ using the Qt-framework. The source code can be downloaded and edited via the GitHub repositories. The source is split in two main repositories: Fritzing-App and Fritzing-Parts.

Figure 17 Fritzing logo

4.4 Proteus:

Generally we are listening the words **PCB's, PCB layout, PCB designing,** ect. But what is PCB? Why we are using this PCB? We want to know about all these things as a electronic engineer. PCB means Printed Circuit Board. This is a circuit board with printed copper layout connections. These PCB's are two types. One is dotted PCB and another one is layout PCB. The two examples are shown in below.

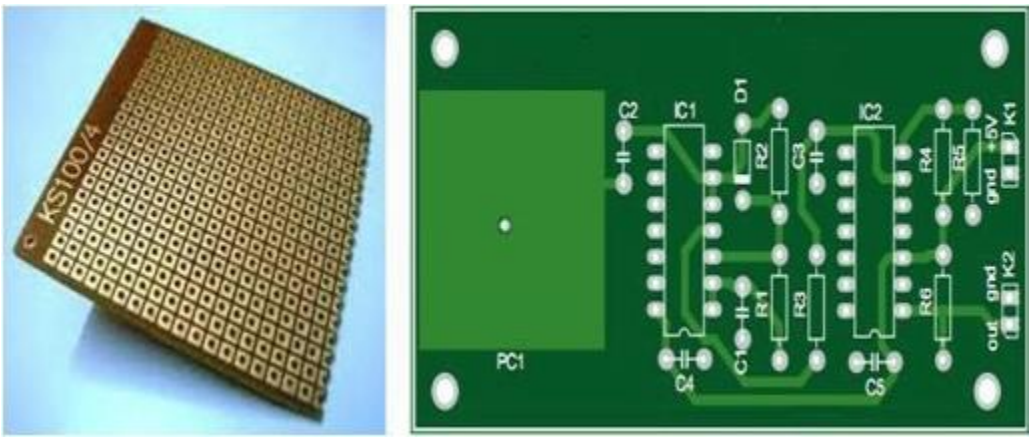


Figure 18 Proteus

What is the main difference between the dotted PCB and layout PCB? In dotted PCB board only dots are available. According to our requirement we can place or insert the components in those holes and attach the components with wires and soldering lid. In this dotted PCB we can make the circuit as out wish but it is very hard to design. There are so many difficulties are there. Those are connecting the proper pins,

avoiding shot connections and etc. Coming to the layout PCB this is simple to design. First we select the our circuit and by using different PCB designing software's, design the layout of the circuit and by itching process preparing the copper layout of our circuit and solder the components in the correct places. It is simple to design, take less time to design, no shortages, looking nice and perfect.

Up to now we have discussed about types of PCB's and difference between the types. Now we can discuss about PCB designing software. There are so many PCB designing softwares available. Some are Express PCB, eagle PCB, PCB Elegance, free PCB, open circuit design, zenith PCB andProteus etc. Apart from remaining Proteus is different. Proteus is design suit and PCB layout designing software. In Proteus we can design any circuit and simulate the circuit and make PCB layout for that circuit.

Introduction to Proteus: Proteus professional is a software combination of ISIS schematic capture program and ARES PCB layout program. This is a powerful and integrated development environment. Tools in this suit are very easy to use and these tools are very useful in education and professional PCB designing. As

professional PCB designing software with integrated space based auto router, it provides features such as fully featured schematic capture, highly configurable design rules, interactive SPICE circuit simulator, extensive support for power planes, industry standard CADCAM & ODB++ output and integrated 3D viewer. Up to know we have discussed about the basics and software description. Now we are entering into the designing section. Run the ISIS professional program by clicking the icon on the desktop, then this splash screen will appear.

Microcontroller Simulation

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a trainingor teaching tool. Support is available for co-simulation of:

- Microchip Technologies PIC10, PIC12, PIC16, PIC18, PIC24, dsPIC33 microcontrollers
- Atmel AVR (and Arduino), 8051 and ARM Cortex-M3 microcontrollers
- NXP 8051, ARM7, ARM Cortex-M0 and ARM Cortex-M3 microcontrollers
- Texas Instruments MSP430, PICCOLO DSP and ARM Cortex-M3 microcontrollers
- Parallax Basic Stamp, Freescale HC11, 8086 microcontroller

Next, a work space with interface buttons for designing circuit will appear as shown in figure below. Note that there is a blue rectangular line in the workspace; make sure that whole circuit is designed inside the rectangular space.

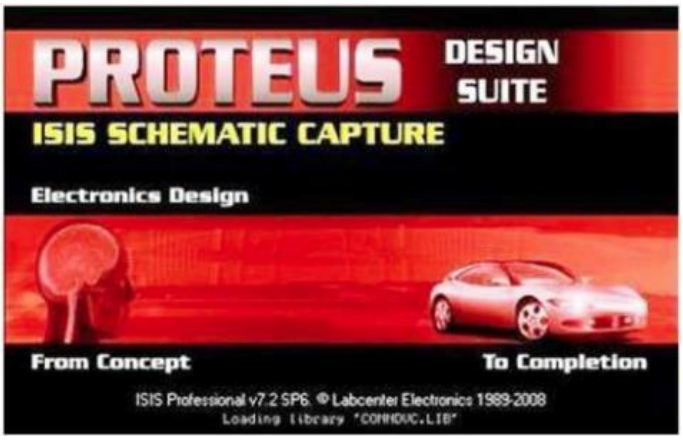


Figure 19 logo

4.5 Blynk as Cloud:

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, vizualize it and do many other cool things. There are three major components in the platform:

- Blynk App - allows to you create amazing interfaces for your projects using various widgets we provide.

- Blynk Server - responsible for all the communications between the smart phone and hardware. You can use our Blynk Cloud or run your private Blynk server locally. It's open source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.

- Blynk Libraries - for all the popular hardware platforms - enable communication with the server and process all the incoming and out coming commands.

Features

- Similar API & UI for all supported hardware & devices

- Connection to the cloud using: o WiFi o Bluetooth and BLE o Ethernet o USB (Serial) o GSM

- Set of easy-to-use Widgets

- Direct pin manipulation with no code writing

- Easy to integrate and add new functionality using virtual pins

- History data monitoring via SuperChart widget

- Device-to-Device communication using Bridge Widget

- Sending emails, tweets, push notifications, etc. new features are constantly added! You can find example sketches covering basic Blynk

Features. They are included in the library. All the sketches are designed to be easily combined with each other. At this point you might be thinking: "Ok, I want it. What do I need to get started?" –Just a couple of things, really.

1. Hardware. An Arduino, Raspberry Pi, or a similar development kit. Blynk works over the Internet. This means that the hardware you choose should be able to connect to the internet. Some of the boards, like Arduino Uno will need an Ethernet or Wi-Fi Shield to communicate, others are already Internet-enabled:

like the ESP8266, Raspberri Pi with WiFi dongle, Particle Photon or Spark Fun Blynk Board. But even if you don't have a shield, you can connect it over USB to your laptop or desktop (it's a bit more complicated for newbies, but we got you covered). What's cool, is that the list of hardware that works with Blynk is huge and will keep on growing.

2. A Smartphone. The Blynk App is a well designed interface builder. It works on both iOS and Android, so no holy wars here, ok. Getting Started with the Bylnk App Create a Bylnk account

- After you download the Blynk App, you'll need to create a New Blynk account. This account is separate from the accounts used for the Blynk Forums, in case you already have one.
- We recommend using a real email address because it will simplify things late.

Figure 20 Blynk

- An account is needed to save your projects and have access to them from multiple
- devices from anywhere. It's also a security measure.
- You can always set up your own Private Blynk Server and have full control.

4.5.1 Create a new project

- After you've successfully logged into your account, start by creating a new project. Choose your Hardware ¬ Select the hardware model you will use.

Auth Token

- Auth Token is a unique identifier which is needed to connect your hardware to your smartphone. Every new project you create will have its own Auth Token. You'll get Auth Token automatically on your email after project creation. You can also copy it manually. Click on devices section and selected required device : NOTE: Don't share your Auth Token with anyone, unless you want someone to have access to your hardware.

It’s very convenient to send it over e-mail. Press the e-mail button and the token will be sent to the e-mail address you used for registration. You can also tap on the Token line and it will be copied to the clipboard. Now press the “Create” button.

Add a widget

Your project canvas is empty, let’s add a button to control our LED.

Tap anywhere on the canvas to open the widget box. All the available widgets are located here. Now pick a button.

Widget Box

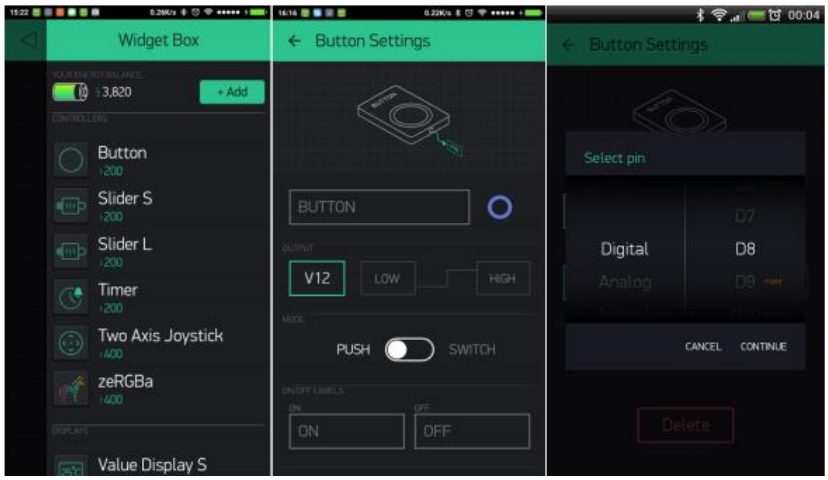


Figure21 Interface Of Blynk

Drag-n-Drop - Tap and hold the Widget to drag it to the new position.

Widget Settings - Each Widget has it’s own settings. Tap on the widget to get to them. The most important parameter to set is PIN . The list of pins reflects physical pins defined by your hardware. If your LED is connected to Digital Pin 8 - then select D8 (D - stands for Digital).

Run the project

When you are done with the Settings - press the PLAY button. This will switch you from EDIT mode to PLAY mode where you can interact with the hardware. While in PLAY mode, you won’t be able to drag or set up new widgets, press STOP and get back to EDIT mode.

4.6 C language script

C programming is a general-purpose, procedural, imperative computer programming language developed in 1972 by Dennis M. Ritchie at the Bell Telephone Laboratories to develop the UNIX operating system. C is the most widely used computer language. It keeps fluctuating at number one scale of popularity along with Java programming language, which is also equally popular and most widely used among modern software programmers.

A successor to the programming language B, C was originally developed at Bell Labs by Dennis Ritchie between 1972 and 1973 to construct utilities running on Unix. It was applied to re-implementing the kernel of the Unix operating system. During the 1980s, C gradually gained popularity. It has become one of the most widely used programming languages, with C compilers from various vendors available for the majority of existing computer architectures and operating systems. C has been standardized by the ANSI since 1989 (ANSI C) and by the International Organization for Standardization (ISO).

C is an imperative procedural language. It was designed to be compiled to provide lowlevel access to memory and language constructs that map efficiently to machine instructions, all with minimal runtime support. Despite its low-level capabilities, the language was designed to encourage cross-platform programming. A standards-compliant C program written with portability in mind can be compiled for a wide variety of computer platforms and operating systems with few changes to its source code36 C programming language is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Software Development Domain. I will list down some of the key advantages of learning C Programming:

- Easy to learn
- Structured language
- It produces efficient programs
- It can handle low-level activities
- It can be compiled on a variety of computer platforms

Facts about c

- C was invented to write an operating system called UNIX.
- C is a successor of B language which was introduced around the early 1970s.

- The language was formalized in 1988 by the American National Standard Institute (ANSI).
- The UNIX OS was totally written in C.
- Today C is the most widely used and popular System Programming Language.
- Most of the state-of-the-art software have been implemented using C.

Applications of c programming

C was initially used for system development work, particularly the programs that make-up the operating system. C was adopted as a system development language because it produces code that runs nearly as fast as the code written in assembly language. Some examples of the use of C are –

- Operating Systems
- Language Compilers
- Assemblers
- Text Editors37
- Print Spoolers
- Network Drivers
- Modern Programs
- Databases
- Language Interpreters

CHAPTER V

IMPLEMENTAION OF CIRCUIT

5.1 Steps in Implementation of Circuit:

- Block diagram
- Circuit diagram
- Flow diagram of code
- Code

5.2 Block diagram:

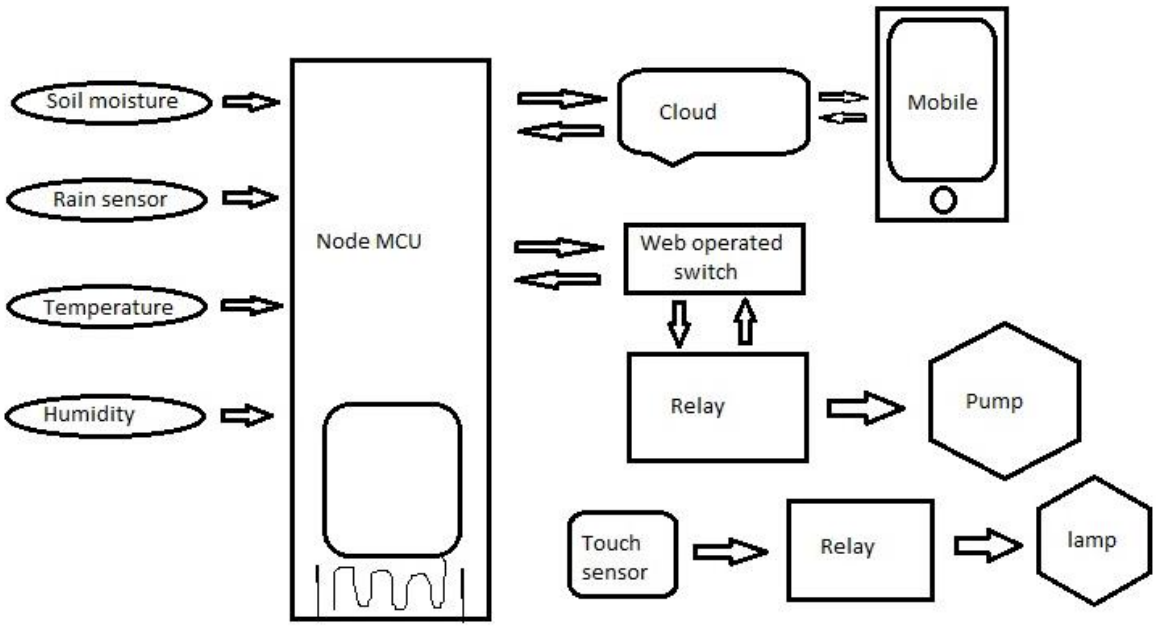


Figure 22 Block Diagram

5.3 Circuit Diagram:

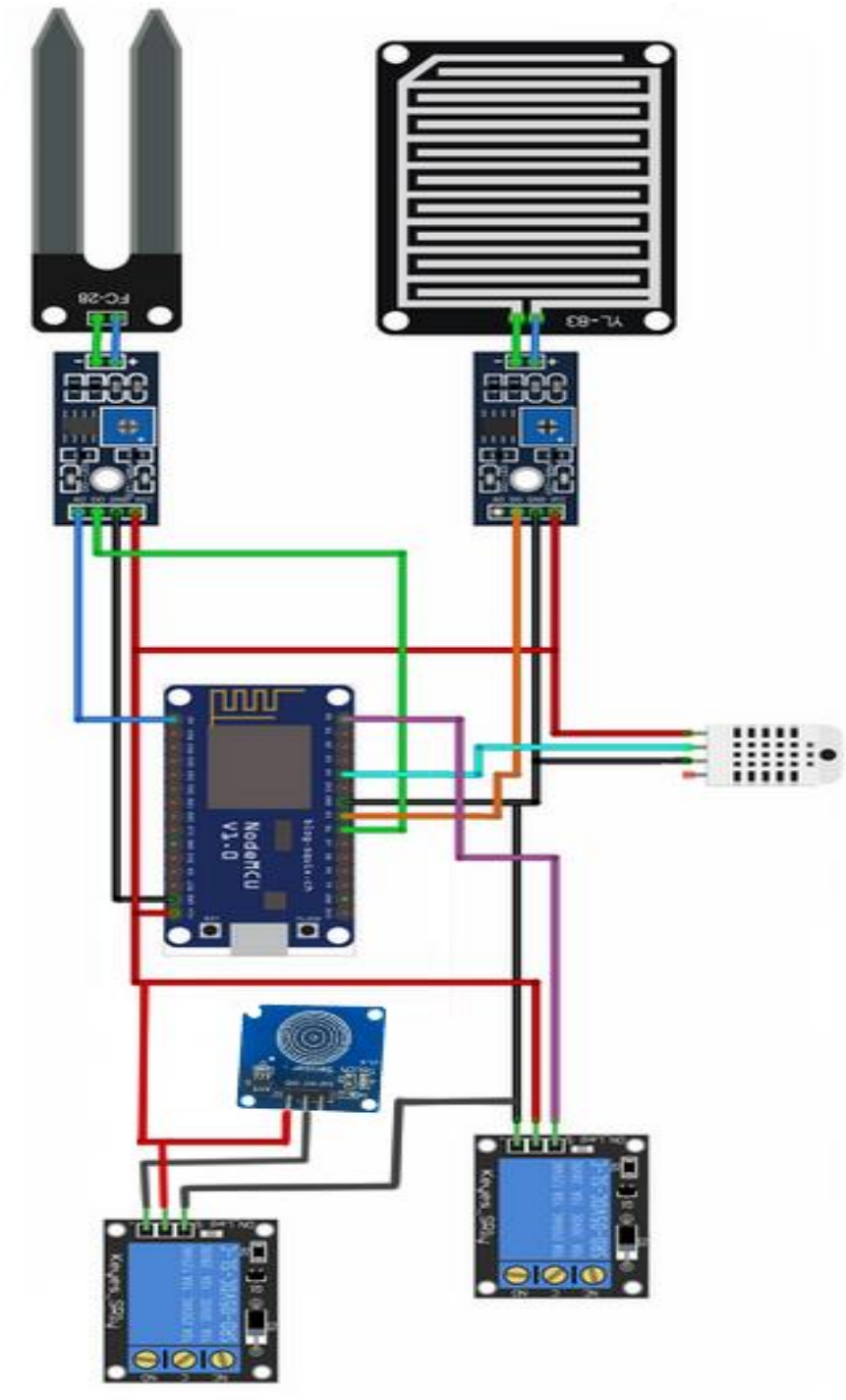


Figure 23 circuit diagram

5.4 Flow diagram of code:

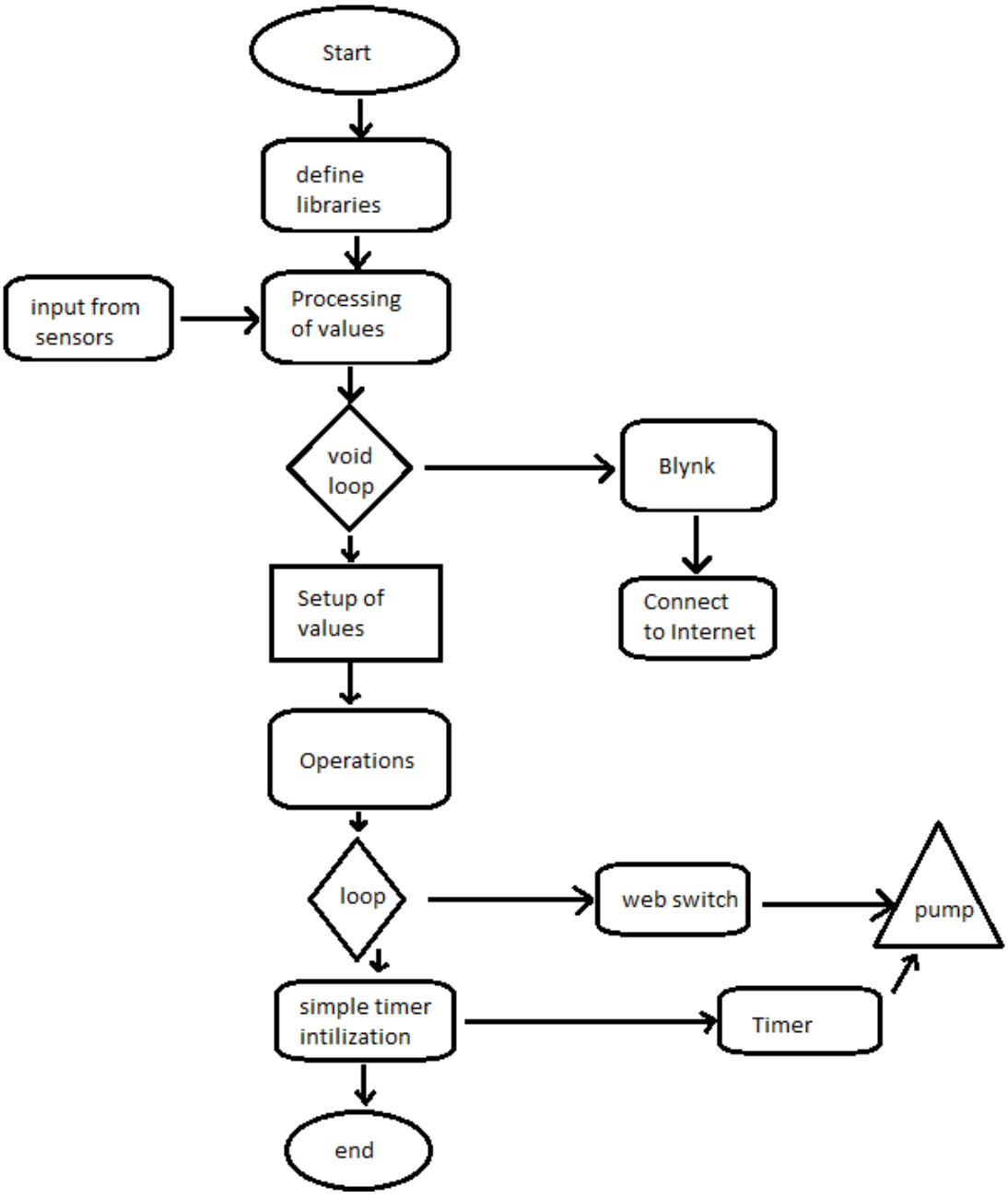


Figure 24 Flow Chart

5.5 Code:

```
#define BLYNK_PRINT Serial

#include <SPI.h>

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include <SimpleTimer.h>

#include <DHT.h>

#define BLYNK_PRINT Serial

#include <OneWire.h>

#include <DallasTemperature.h>

#define ONE_WIRE_BUS D2

OneWire oneWire(ONE_WIRE_BUS);

DallasTemperature sensors(&oneWire);

char auth[] = "----------------";              //Authentication code sent by Blynk

char ssid[] = "-------";                       //WiFi SSID

char pass[] = "-------";                       //WiFi Password

#define pirPin D1

int pirValue;

int pinValue;

#define sensorPin D6

#define rainPin D5

int sensorState = 0;
```

```
int rainState = 0;

int lastState = 0;

int lastRainState = 0;

#define DHTPIN 2

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

SimpleTimer timer;

BLYNK_WRITE(V0)

{

 pinValue = param.asInt();

}

void sendSensor()

{

  float h = dht.readHumidity();

  float t = dht.readTemperature();


  if (isnan(h) || isnan(t)) {

    Serial.println("Failed to read from DHT sensor!");

    return;

  }

  Blynk.virtualWrite(V5, h);  //V5 is for Humidity

  Blynk.virtualWrite(V6, t);  //V6 is for Temperature

}
```

```
void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
   pinMode(sensorPin, INPUT);
   pinMode(rainPin, INPUT);
   pinMode(pirPin, INPUT);
  dht.begin();
  timer.setInterval(1000L, sendSensor);
  Serial.begin(115200);
  Blynk.begin(auth, ssid, pass);
   sensors.begin();
}
int sensor=0;
void sendTemps()
{
sensor=analogRead(A0);
sensors.requestTemperatures();
float temp = sensors.getTempCByIndex(0);
Serial.println(temp);
Serial.println(sensor);
Blynk.virtualWrite(V1, temp);
Blynk.virtualWrite(V2,sensor);
```

```
delay(1000);
}
void getPirValue(void)        //Get PIR Data
  {
   pirValue = digitalRead(pirPin);
    if (pirValue)
     {
       Serial.println("Motion detected");
       Blynk.notify("Motion detected");
     }
  }
void loop()
{
  Blynk.run();
  timer.run();
  sendTemps();
sensorState = digitalRead(sensorPin);
Serial.println(sensorState);
if (sensorState == 1 && lastState == 0) {
  Serial.println("needs water, send notification");
  Blynk.notify("Water your plants");
  lastState = 1;
  delay(1000);
```

```
//send notification


  }
  else if (sensorState == 1 && lastState == 1) {
    //do nothing, has not been watered yet
  Serial.println("has not been watered yet");
  delay(1000);
  }
  else {
    //st
    Serial.println("does not need water");
    lastState = 0;
    delay(1000);
  }
rainState = digitalRead(rainPin);
Serial.println(rainState);


  if (rainState == 0 && lastRainState == 0) {
  Serial.println("Its Raining!");
  Blynk.notify("Its Raining!");
  lastRainState = 1;
  delay(1000);
//send notification
```

```
 }
 else if (rainState == 0 && lastRainState == 1) {
 delay(1000);
 }
 else {
  Serial.println("No Rains");
  lastRainState = 0;
  delay(1000);
 }
 if (pinValue == HIGH)
    {
      getPirValue();
    }
    Blynk.run();
 delay(100);
}
```

```
Executable segment sizes:
ICACHE : 32768           - flash instruction cache
IROM   : 251572          - code in flash         (default or ICACHE_FLASH_ATTR)
IRAM   : 28241   / 32768 - code in IRAM          (IRAM_ATTR, ISRs...)
DATA   : 1564  )         - initialized variables (global, static) in RAM/HEAP
RODATA : 2516  ) / 81920 - constants             (global, static) in RAM/HEAP
BSS    : 26424 )         - zeroed variables      (global, static) in RAM/HEAP
Sketch uses 283893 bytes (27%) of program storage space. Maximum is 1044464 bytes.
Global variables use 30504 bytes (37%) of dynamic memory, leaving 51416 bytes for local variables. Maximum is 81920 bytes.
```

Figure 25 Code Result

CHAPTER VI

RESULTS OF PROJECT

6.1 Result:

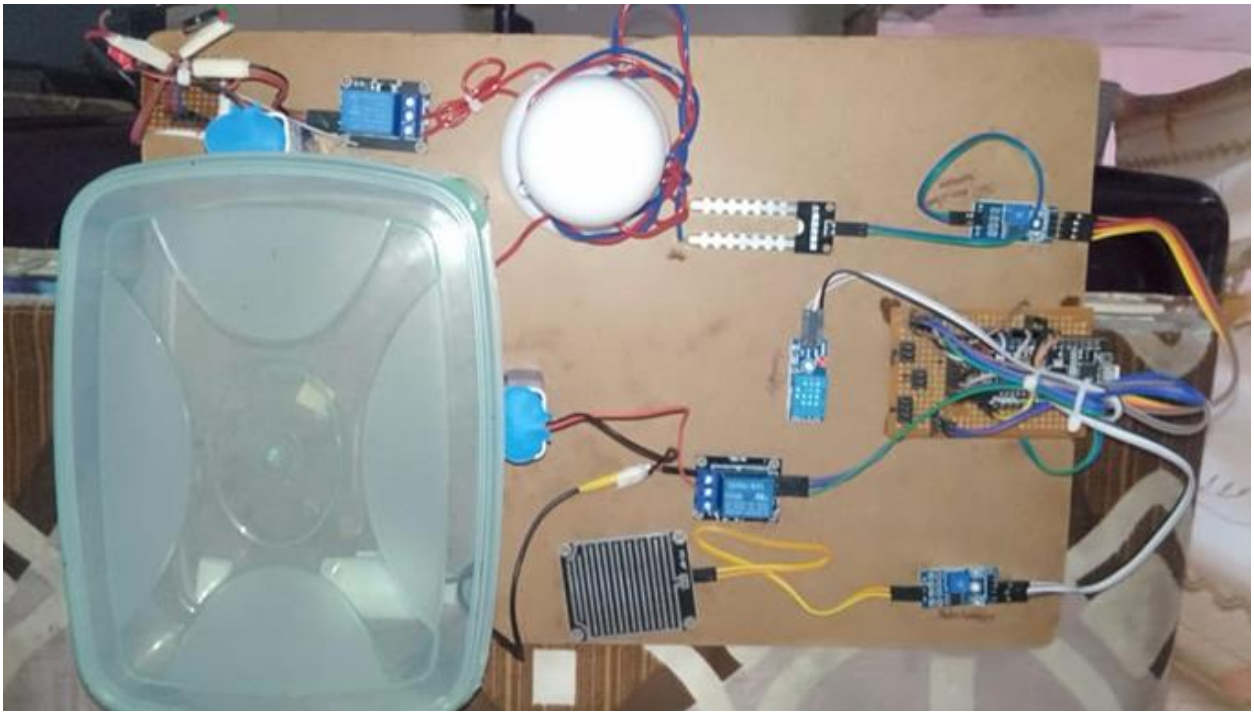


Figure 26 Result Kit

6.2 Cloud results:

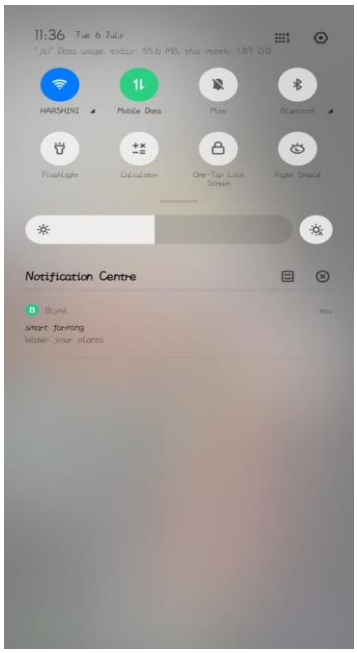
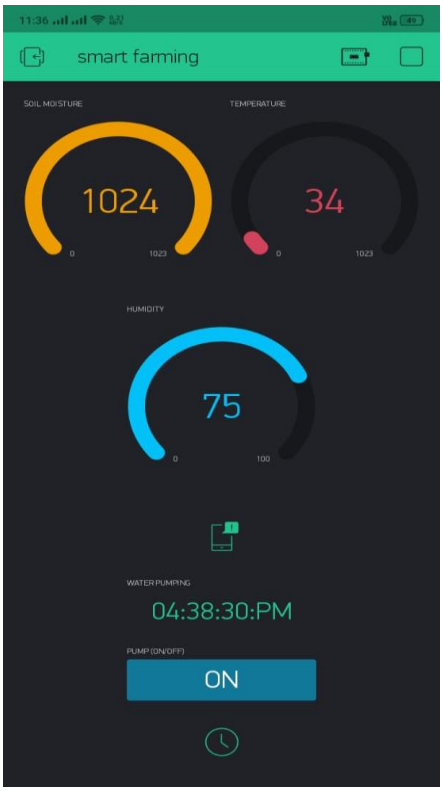
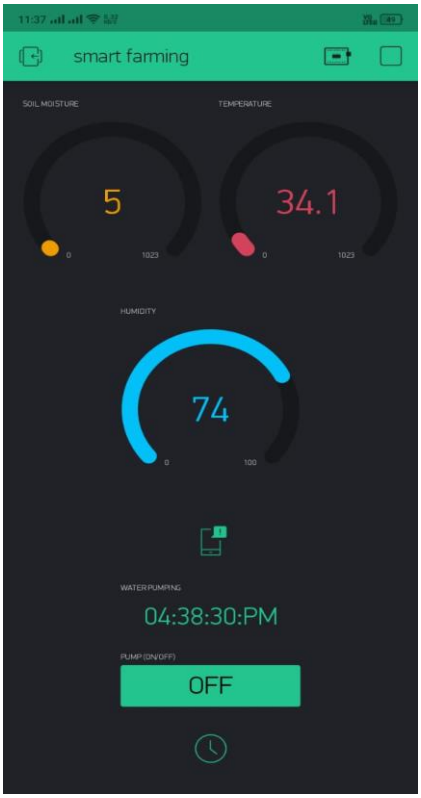
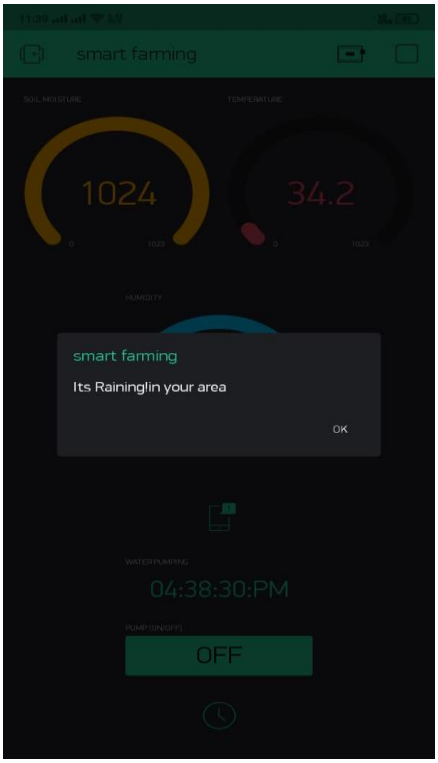


Figure 27 Notification to User







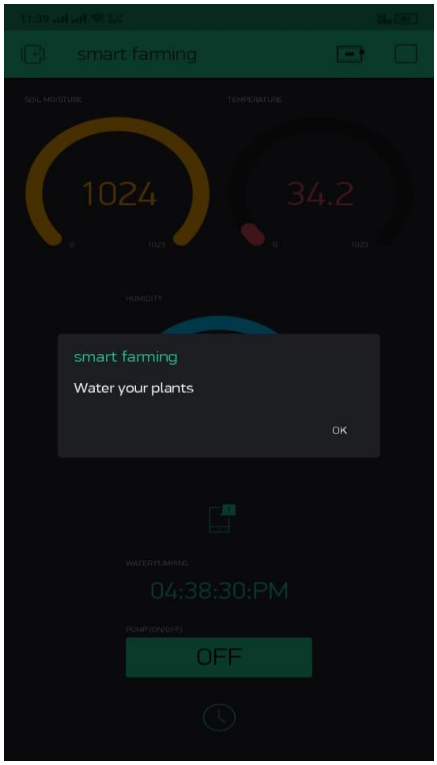


Figure 28 Result of cloud

CHAPTER VII

CONCLUSION & REFERENCES

Conclusion:

The smart irrigation system implemented is feasible and cost effective for optimizing water resources for agricultural production. This irrigation system allows cultivation in places with water scarcity thereby improving sustainability. The smart irrigation system developed proves that the use of water can be diminished for a given amount of fresh biomass production.

A system to monitor moisture levels in the soil was designed and the project provided an opportunity to study the existing systems, along with their features and drawbacks. The proposed system can be used to switch on/off the water sprinkler according to soil moisture levels thereby automating the process of irrigation which is one of the most time consuming activities in farming. Agriculture is one of the most water-consuming activities. The system uses information from soil moisture sensors to irrigate soil which helps to prevent over irrigation or under irrigation of soil
thereby avoiding crop damage. The farm owner can monitor the process online through a website. Through this project it can be concluded that there can be considerable development in farming with the use of IOT and automation. Thus, the system is a potential solution to the problems faced in the existing manual and cumbersome process of irrigation by enabling efficient utilization of water resources.

RELIABILITY: Irrigator cannot trust the automatic system to work correctly every time.Sometimes failures will occur.These errors are because of human errors in setting and maintaining the systems. A re-use system is good .

References:

- S.G.Manoj Guru, P.Naveen ,R.Vinodh Raja , V.SrirengaNachiyar, Special Issue -March 2017, SMART IRRIGATION SYSTEM USING ARDUINO

- Srishti Rawal Department of Computer Science, VIT University, International Journal of Computer Applications (0975 –8887) Volume 159 –No 8, February 2017, IOT based Smart Irrigation System.

- N.Usha and Dr.T.Menakadevi,Volume4,Issue7,March 2017,ISSN:2347-1697

- G.M.Pushpanjali, S.S.Sheetal, and A.D. Mhamane,International Journal on Emerging Technologies,Smart Irrigation System Using Microcontroller and ZigBee

- Ch. Apparao, G.Ravi Babu, A. Sambhaiah, L.Edukondalu, Vol Development of Low Cost Soil Moisture Sensor for Drip Irrigation System ume 5, Issue 9, September 2015,

- Chimata Suhasini , Diwakar R. Marur, Vol. 3, Issue 4, April 2015, GSM AND WIRELESS SENSOR NETWORK BASED SMART AUTOMATED IRRIGATION SYSTEM

- G. Parameswaran , K.Sivaprasath, Volume 6 IssueNo. 5, 2016 IJESC, Arduino Based Smart Drip Irrigation System Using Internet of Things.

- Priyanka,Sakshi Chaudhary,IJARSE,Vol.No.2,Issue No.3,March 2013,Smart Irrigation System.

- Richa Yadav, Manjari Sharma, Mohd Azharuddin, Mohd Monis, Kriti Sisodia, (IJSRMS) ISSN: 2349-3771 Volume 3 Issue 5, pg: 203-209, SMART IRRIGATION SYSTEM USING GSM AND SOLAR PANEL.

- Bhakti B. Bakle , Prof. Amol R. Wagh, e-ISSN No.:2349-9745, Date: 2-4 July, 2015, Smart Irrigation System Using Wireless Sensor Network and GPRSModule

- MirceaEremia,LucianToma,MihaiSanduleac, Procedia Engineering,Volume 181,2017, Pages 12-19,The Smart City Concept in the 21Stcentury

- Sujatha Joshi,SakshamSaxena,TanviGodbole,Shreya,Volume 93,2016, Pages 902-909,Developing Smart Cities: An Integrated Framework

- Mark Deakin, Smart cities: the state-of-the-art and governance challenge

- Jason Repko,Smart Cities Literature Review and Analysis• Iker Zubizarreta;Alessandro Seravalli; andSaioa Arrizabalaga,Ph.D.Volume142, Issue1 ,May 18, 2015,Smart City Concept: What It Is and What It Should Be

- SrishtiRawal,https://www.researchgate.net/publication/313779760_IOT_based_Smart_Irrigation_System