CHAPTER 1

INTRODUCTION

Making the farmers' work much easier as they can concentrate on other farm activities. Agriculture is the unquestionably the largest livelihood provider in India. With rising population, there is a need for increased agricultural production. In order to support greater production in farms, the requirement of the amount of fresh water used in irrigation also rises. Currently, agriculture accounts 83% of the total water consumption in India. Unplanned use of water inadvertently results in wastage of water. This suggests that there is an urgent need to develop systems that prevent water wastage without imposing pressure on farmers. Over the past 15 years, farmers started using computers and software systems to organize their financial data and keep track of their transactions with third parties and also monitor their crops more effectively. In the Internet era, where information plays a key role in people's lives, agriculture is rapidly becoming a very data intensive industry where farmers need to collect and evaluate a huge amount of information from a diverse number of devices (eg., sensors, faming machinery etc.) in order to become more efficient in production and communicating appropriate information. With the advent of open-source Arduino boards along with cheap moisture sensors, it is viable to create devices that can monitor the soil moisture content and accordingly irrigating the fields or the landscape as an when needed. The proposed system makes use of microcontroller ATMEGA328P on Arduino uno platform and IOT which enable farmers to remotely monitor the status of sprinklers installed on the farm by knowing the sensor values thereby, making the farmers' work much easier as they can concentrate on other farm activities.

CHAPTER 2

SURVEY OF LITERATURE

2.1 LITERATURE SURVEY

In A Remote Measurement and Control System for Greenhouse Based on GSM-SMS 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) 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 Bluetooth 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, 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. This system covered lower range of agriculture land and not economically affordable.

2.2 EXISTING SYSTEM

In existing systems, there are DTMF BASED CONTROL in the system we have to control motor through phone without knowing the soil moisture status. And also, no Automation.

2.3 PROPOSED SYSTEM

In this project, the automated irrigation system based on low power microcontroller was developed and deployed.

In proposed system, we are using ARDUINO UNO, moisture sensor, LCD, Relay and water motor.

The required electrical and electronic components of this proposed system are moisture sensor and water motor.

Moisture sensor detects the humidity level in the soil, if humidity is low the water motor will be ON.

The status of sensor will be displayed on LCD and water motor will be controlled through the status of soil moisture sensor.

CHAPTER 3
SYSTEM AND SPECIFICATIONS

3.1 EMBEDDED SYSTEM

3.1.1 Definition 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. An embedded system is a microcontroller-based, software driven, reliable, real-time control system, autonomous, or human or network interactive, operating on diverse physical variables and in diverse environments and sold into a competitive and cost-conscious market.

An embedded system is not a computer system that is used primarily for processing, not a software system on PC or UNIX, not a traditional business or scientific application. High-end embedded & lower end embedded systems. High-end embedded system - Generally 32, 64 Bit Controllers used with OS. Examples Personal Digital Assistant and Mobile phones etc. Lower end embedded systems - Generally 8,16 Bit Controllers used with a minimal operating systems and hardware layout designed for the specific purpose.

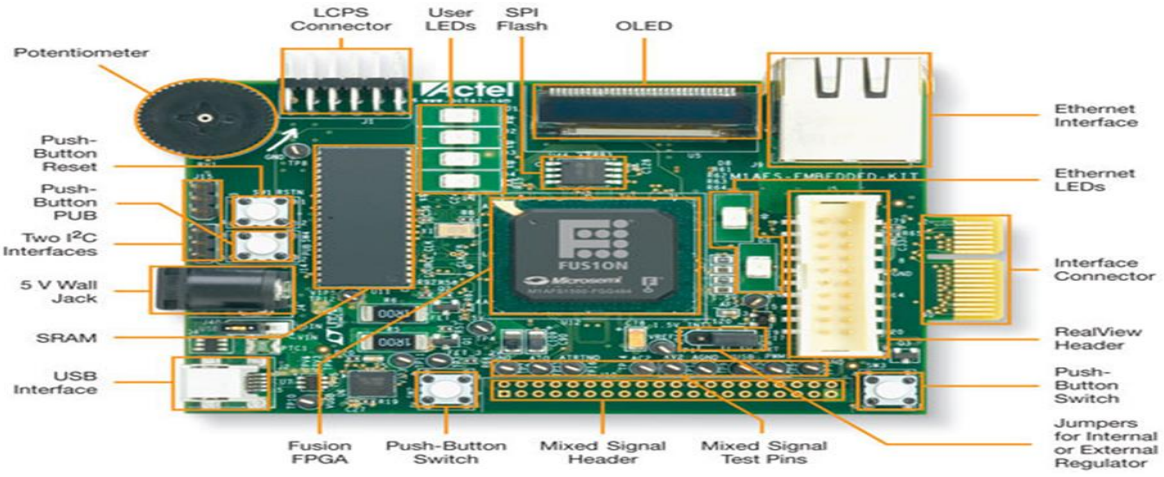


Figure 1.1 Basic Embedded System

3.1.2 Characteristics

An embedded system is any computer system hidden inside a product other than a computer. They will encounter a number of difficulties when writing embedded system software in addition

to those we encounter when we write applications. Throughput – Our system may need to handle a lot of data in a short period of time. Response–Our system may need to react to events quickly. Testability–Setting up equipment to test embedded software can be difficult. Debugability–Without a screen or a keyboard, finding out what the software is doing wrong (other than not working) is a troublesome problem. Reliability – embedded systems must be able to handle any situation without human intervention.

Memory space – Memory is limited on embedded systems, and you must make the software and the data fit into whatever memory exists. Program installation – you will need special tools to get your software into embedded systems. Power consumption – Portable systems must run on battery power, and the software in these systems must conserve power. Processor hogs – computing that requires large amounts of CPU time can complicate the response problem. Cost – Reducing the cost of the hardware is a concern in many embedded system projects; software often operates on hardware that is barely adequate for the job. Embedded systems have a microprocessor/ microcontroller and a memory. Some have a serial port or a network connection. They usually do not have keyboards, screens or disk drives.

3.1.3 Applications

- Military and aerospace embedded software applications
- Communication Applications.
- Industrial automation and process control software.
- Mastering the complexity of applications.
- Reduction of product design time.
- Real time processing of ever-increasing amounts of data.

3.2 SPECIFICATIONS

3.2.1 HARDWARE REQUIREMENTS

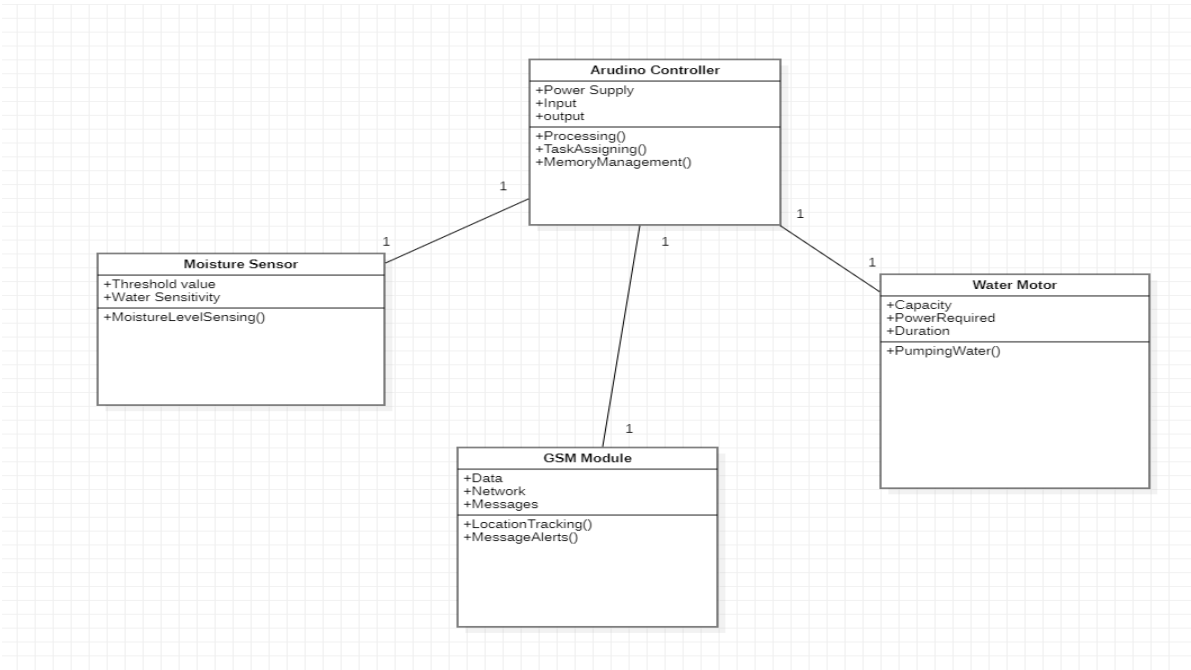
- Soil moisture sensor
- Power supply
- Micro Controller
- LCD
- Water Motor

3.2.2 SOFTWARE REQUIREMENTS

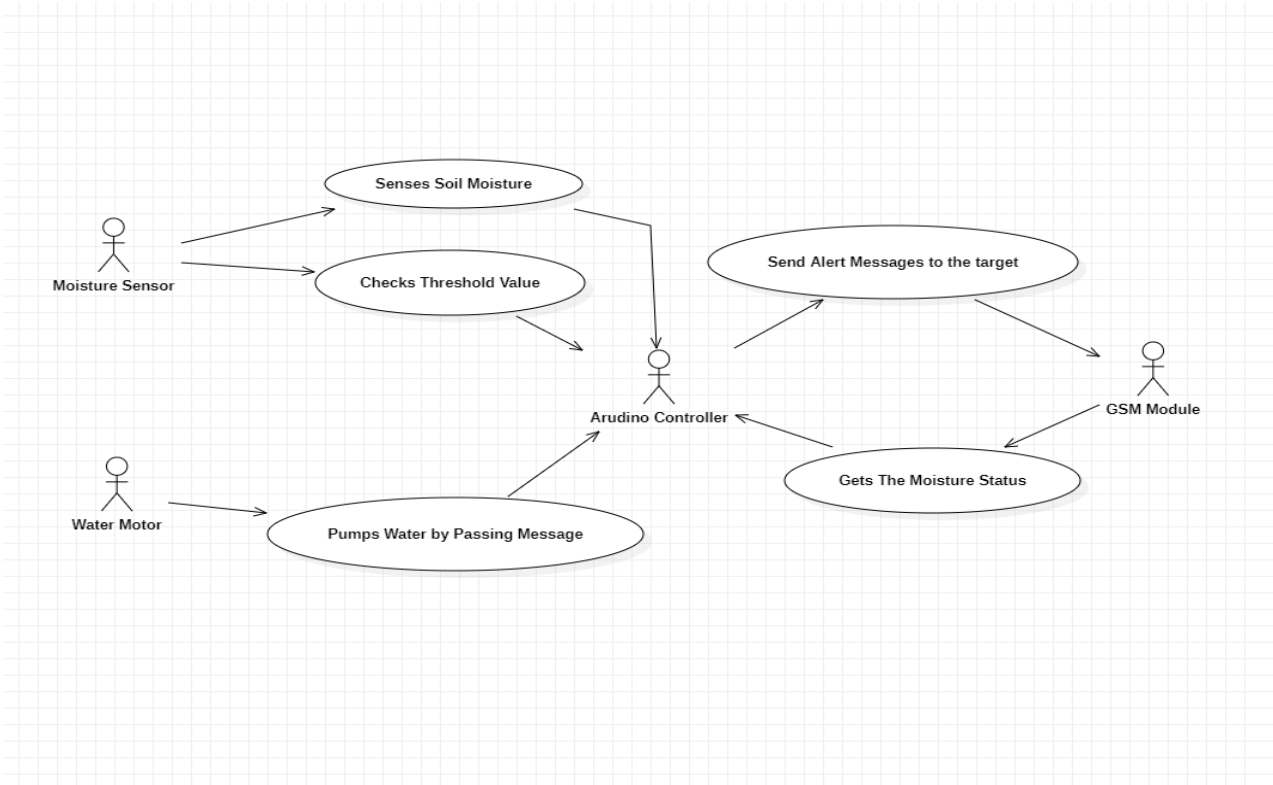
- ARDUINO IDE
- C LANGUAGE

3.2.3 UML DIAGRAMS

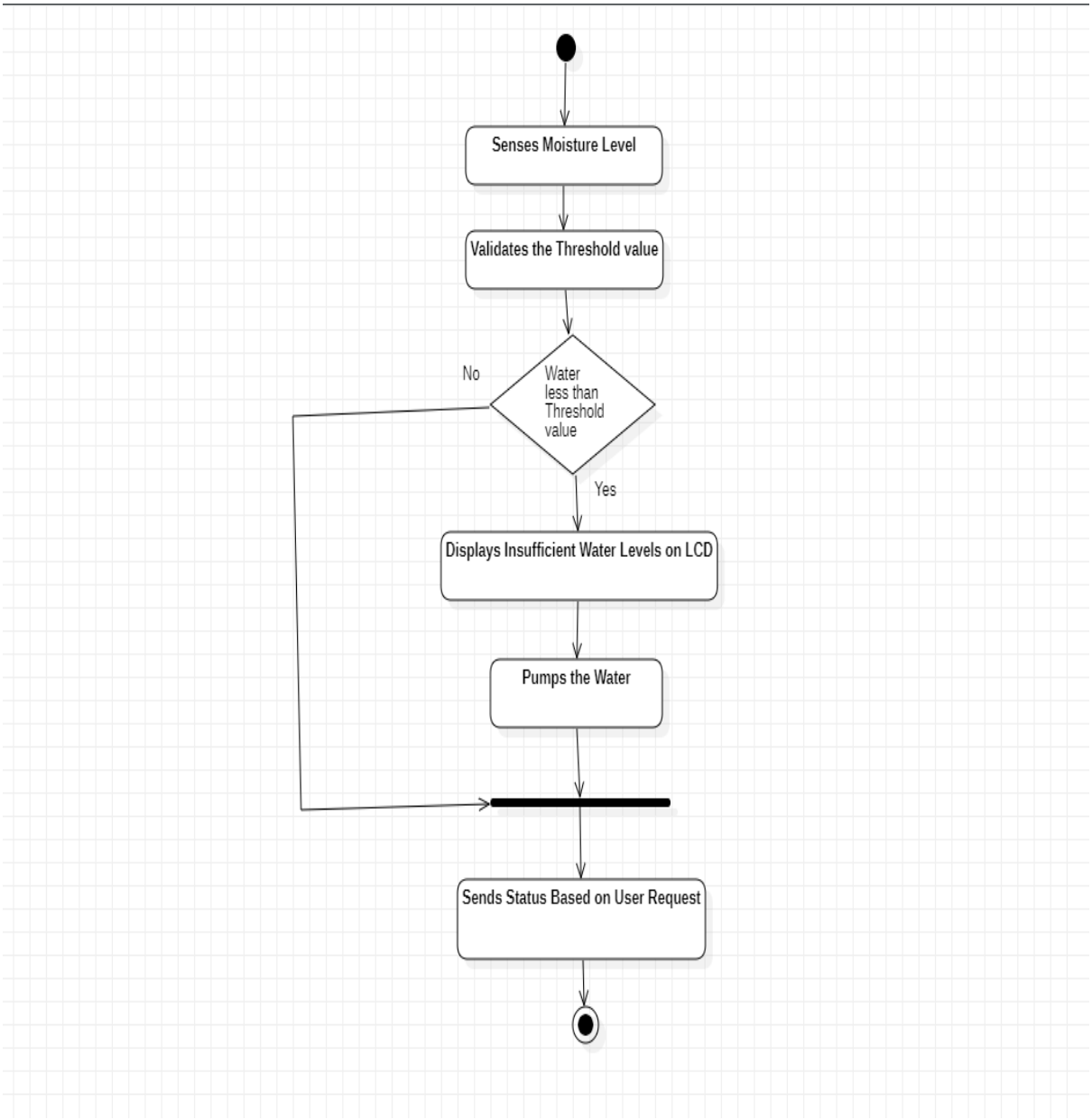
CLASS DIAGRAM



USE CASE DIAGRAM



ACTIVITY DIAGRAM



3.3 IOT SYSTEMS

The **Internet of things** (**IoT**) describes physical objects (or groups of such objects) with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems over the internet or other communications networks. Internet of things has been considered a misnomer because devices do not need to be connected to the public internet, they only need to be connected to a network and be individually addressable.

The field has evolved due to the convergence of multiple technologies, including ubiquitous computing, commodity sensors, increasingly powerful embedded systems, and machine learning. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), independently and collectively enable the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "smart home", including devices and appliances (such as lighting fixtures, thermostats, home security systems, cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. IoT is also used in healthcare systems.

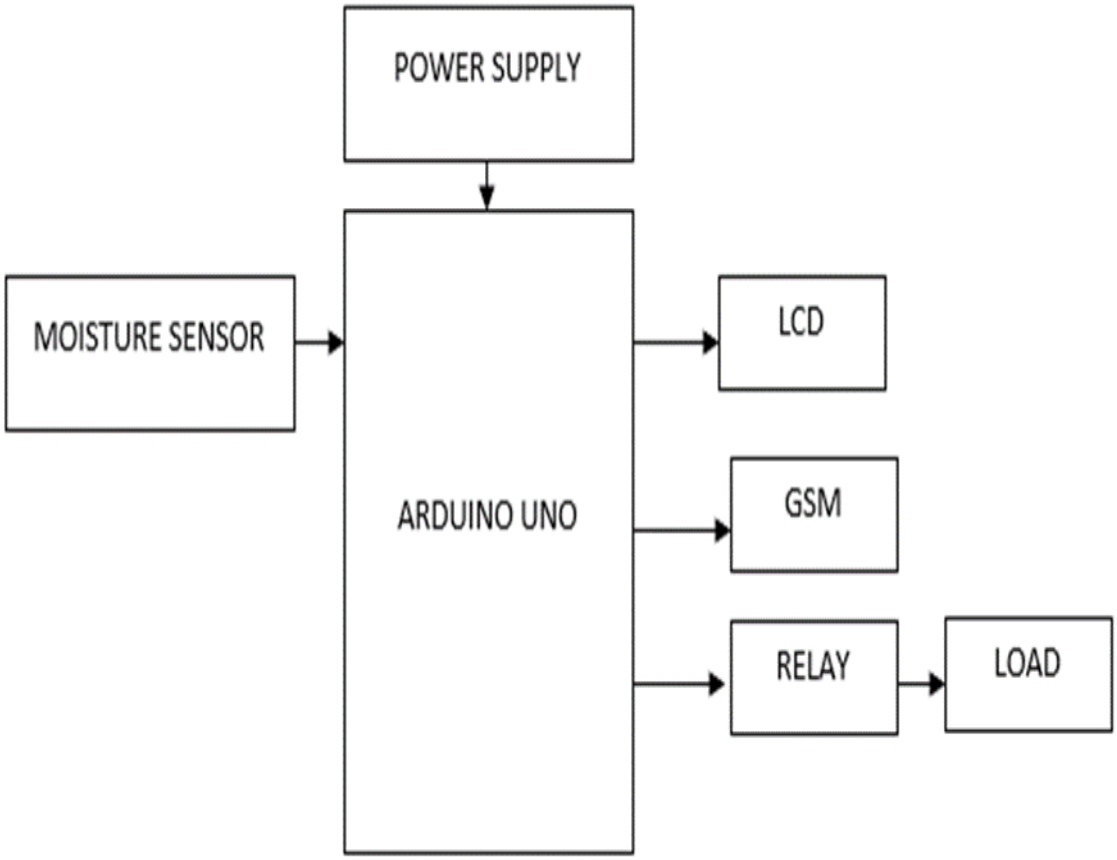
There are number of concerns about the risks in the growth of IoT technologies and products, especially in the areas of privacy and security, and consequently, industry and governmental moves to address these concerns have begun, including the development of international and local standards, guidelines, and regulatory frameworks.

The IoT can be described as an extension of the internet and other network connections to different sensors and devices — or “things” — affording even simple objects, such as lightbulbs, locks, and vents, a higher degree of computing and analytical capabilities.

Interoperability is one of the key aspects of the IoT that contribute to its growing popularity. Connected or “smart” devices — as “things” in the IoT are often called — have the ability to gather and share data from their environments with other devices and networks. Through the analysis and processing of the data, devices can perform their functions with little or no need for human interaction.

Given the ever-increasing number of connected devices, the IoT continues its path of evolution, adding different layers to the data that is already being shared and processed, and giving rise to sophisticated algorithms that result in improved levels of automation. And because of the variety of “things” that can be connected to it, the IoT has enabled diverse applications for individual users and entire industries alike.

3.4 SYSTEM ARCHITECTURE



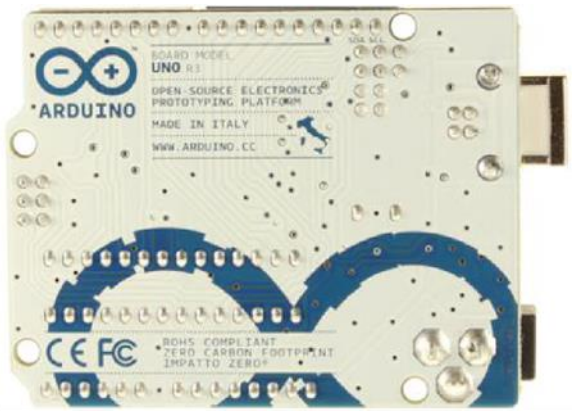
CHAPTER-4

HARDWARE AND SOFTWARE DESCRIPTION

4.1 HARDWARE DESCRIPTION

4.1.1 ARDUINO UNO

Arduino Uno R3 Front



Arduino Uno R3 Back

Arduino Uno R2 Front

Arduino Uno SM

Arduino Uno Front

Arduino Uno Back

Overview

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it

with an AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Revision 2 of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

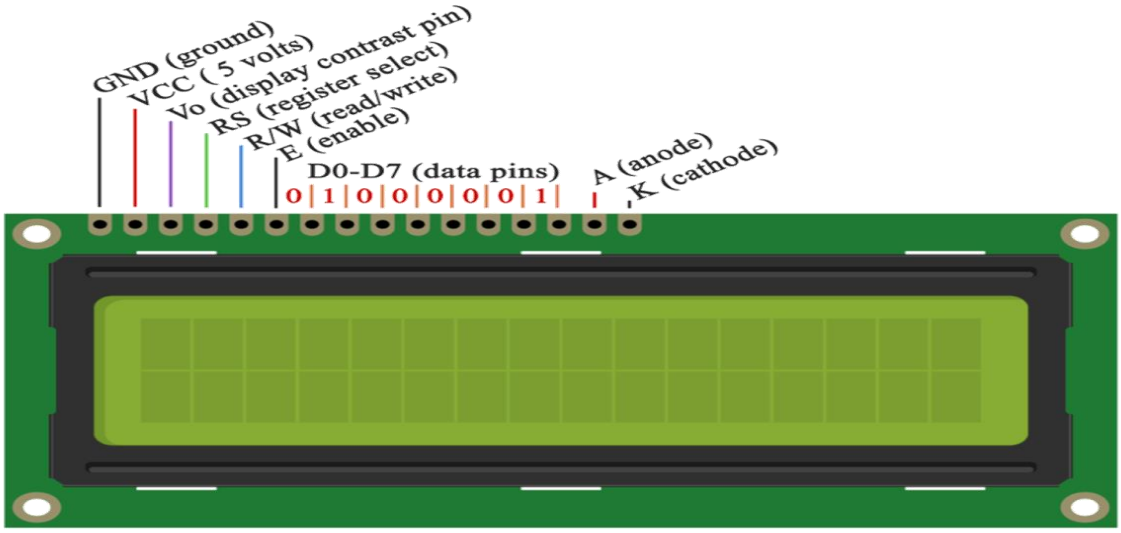
Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (limits)	6-20V
Digital I/O Pins	(Of which 6 provide PWM output)
Analog Input Pins	
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)

4.1.2 LCD

- A liquid crystal display (LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector.
- Each pixel consists of a column of liquid crystal molecules suspended between two transparent electrodes, and two polarizing filters, the axes of polarity of which are perpendicular to each other. Without the liquid crystals between them, light passing through one would be blocked by the other.
- The liquid crystal twists the polarization of light entering one filter to allow it to pass through the other.
- A program must interact with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to a controller is an LCD display.

The LCDs have a parallel interface, meaning that the microcontroller has to manipulate several interface pins at once to control the display. The interface consists of the following pi



- A **register select (RS) pin** that controls where in the LCD's memory you're writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.
- A **Read/Write (R/W) pin** that selects reading mode or writing mode
- An **Enable pin** that enables writing to the registers
- 8 **data pins (D0 -D7)**. The states of these pins (high or low) are the bits that you're writing to a register when you write, or the values you're reading when you read.

4.1.3 GSM MODULE

This is a GSM/GPRS-compatible Quad-band cell phone, which works on a frequency of 850/900/1800/1900MHz and which can be used not only to access the Internet, but also for oral communication (provided that it is connected to a microphone and a small loud speaker) and for SMSs. Externally, it looks like a big package (0.94 inches x 0.94 inches x 0.12 inches) with L-shaped contacts on four sides so that they can be soldered both on the side and at the bottom. Internally, the module is managed by an AMR926EJ-S processor, which controls phone communication, data communication (through an integrated TCP/IP stack), and (through an UART and a TTL serial interface) the communication with the circuit interfaced with the cell phone itself. The processor is also in charge of a SIM card (3 or 1,8 V) which needs to be attached to the outer wall of the module. In addition, the GSM900 device integrates an analog interface, an A/D converter, an RTC, an SPI bus, an I²C, and a PWM module. The radio section is GSM phase 2/2+ compatible and is either class 4 (2 W) at 850/ 900 MHz or class 1 (1W)at1800/1900MHz.The TTL serial interface is in charge not only of communicating all the data relative to the SMS already received and those that come in during TCP/IP sessions in GPRS (the data-rate is determined by GPRS class 10: max. 85,6 kbps), but also of receiving the circuit commands (in our case, coming from the PIC governing the remote control) that can be either AT standard or AT-enhanced SIM Com type. The module is supplied with continuous energy (between 3.4 and 4.5 V) and absorbs a maximum of 0.8 A during transmission.

What is GSM Technology?

GSM is a mobile communication modem; it stands for global system for mobile communication (GSM). The idea of GSM was developed at Bell Laboratories in 1970. It is a widely used mobile communication system in the world. GSM is an open and digital cellular technology used for transmitting mobile voice and data services operate at the 850MHz, 900MHz, 1800MHz, and 1900MHz frequency bands.

GSM technology was developed as a digital system using the time division multiple access (TDMA) technique for communication purposes. A GSM digitizes and reduces the data, then sends it down through a channel with two different streams of client data, each in its own particular time slot. The digital system has the ability to carry 64 kbps to 120 Mbps of data rates.

GSM Modem

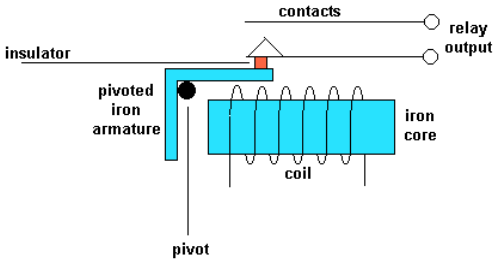
There are various cell sizes in a GSM system such as macro, micro, pico, and umbrella cells. Each cell varies as per the implementation domain. There are five different cell sizes in a GSM network macro, micro, pico, and umbrella cells. The coverage area of each cell varies according to the implementation environment.

The time division multiple access (TDMA) technique relies on assigning different time slots to each user on the same frequency. It can easily adapt to data transmission and voice communication and can carry 64kbps to 120Mbps of data rate.

4.1.4 RELAY

Introduction

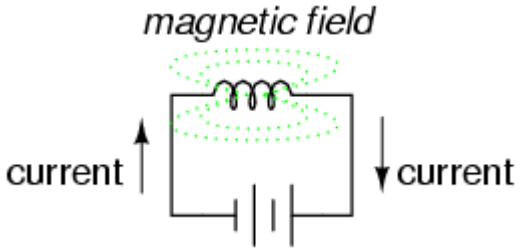
A relay is an electrical switch that opens and closes under the control of another electrical circuit. In the original form, the switch is operated by an electromagnet to open or close one or many sets of contacts. A relay is able to control an output circuit of higher power than the input circuit, it can be considered to be, in a broad sense, a form of an electrical amplifier.

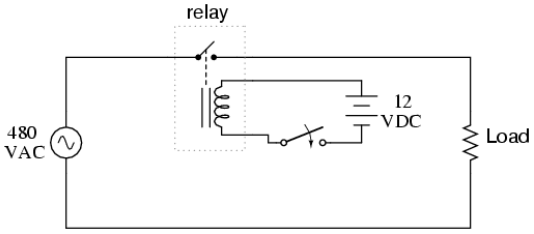


Relays are usually SPDT (single pole double through switch)or DPDT (double pole double through switch) but they can have many more sets of switch contacts, for example relays with 4 sets of changeover contacts are readily available.

Basic operation of a relay

An electric current through a conductor will produce a magnetic field at right angles to the direction of electron flow. If that conductor is wrapped into a coil shape, the magnetic field produced will be oriented along the length of the coil. The greater the current, the greater the strength of the magnetic field, all other factors being equal.





Inductors react against changes in current because of the energy stored in this magnetic field. When we construct a transformer from two inductor coils around a common iron core, we use this field to transfer energy from one coil to the other. However, there are simpler and more direct uses for electromagnetic fields than the applications we've seen with inductors and transformers. The magnetic field produced by a coil of current-carrying wire can be used to exert a mechanical force on any magnetic object, just as we can use a permanent magnet to attract magnetic objects, except that this magnet (formed by the coil) can be turned on or off by switching the current on or off through the coil.

If we place a magnetic object near such a coil for the purpose of making that object move when we energize the coil with electric current, we have what is called a *solenoid*. The movable magnetic object is called an *armature*, and most armatures can be moved with either direct current (DC) or alternating current (AC) energizing the coil. The polarity of the magnetic field is irrelevant for the purpose of attracting an iron armature. Solenoids can be used to electrically open-door latches, open or shut valves, move robotic limbs, and even actuate electric switch mechanisms and is used to actuate a set of switch contacts

4.1.5 SOIL MOISTURE SENSOR

Soil Moisture Sensor

This sensor can be used to test the moisture of soil, when the soil is having water shortage, the module output is at high level, else the output is at low level. By using this sensor one can automatically water the flower plant, or any other plants requiring automatic watering technique. Module triple output mode, digital output is simple, analog output more accurate, serial output with exact readings.

Features

- Sensitivity adjustable.
- Has fixed bolt hole, convenient installation.
- Threshold level can be configured.
- Module triple output mode, digital output is simple, analog output more accurate, serial output with exact readings.

Applications

- Agriculture
- Landscape irrigation

Specifications

Parameter	Value
Operating Voltage	+5v dc regulated
Soil moisture	Digital value is indicated by out pin

Pin Details

Pin	Name	Details
1	out	Active high output
2	+5v	Power supply
3	gnd	Power supply gnd
4	rx	receiver
5	tx	transmitter

Working

Soil moisture sensors measure the water content in soil. A soil moisture probe is made up of multiple soil moisture sensors. One common type of soil moisture sensors in commercial use is a Frequency domain sensor such as a capacitance sensor. Another sensor, the neutron moisture gauge, utilize the moderator properties of water for neutrons. Soil moisture content may be determined via its effect on dielectric constant by measuring the capacitance between two electrodes implanted in the soil. Where soil moisture is predominantly in the form of free water (e.g., in sandy soils), the dielectric constant is directly proportional to the moisture content. The probe is normally given a frequency excitation to permit measurement of the dielectric constant. The readout from the probe is not linear with water content and is influenced by soil type and soil temperature. Therefore, careful calibration is required and long-term stability of the calibration is questionable.

4.1.6 WATER MOTOR

Submersible Pump is a highly efficient tool for various machines that require a pump to function. This pump has an 18W motor that can lift water up to 1.85 m and is made of rust proof durable ceramic shaft. The motor submerges completely into the water as to out.

FEATURES

Capacity: 18W motor, can lift water up to 1.85 m. Water output 1100 L per hour (approx)

Includes motor and output pipe. completely submersible motor

Rust proof and durable quality

Water resistant and easy to install and handle

Low electricity consumption

4.2 SOFTWARE DESCRIPTION

4.2.1 ARDUINO IDE

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

Arduino IDE is the free software used to program Arduino circuit boards. Arduino IDE has a unique programming language to ensure that all the hardware products associated with it can be programmed in the same way. It's also open-source, and so many tech-savvy individuals have taken to creating boards of their own.

The open – source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Cross platform – The Arduino Software (IDE) runs on windows, Macintosh OSX, and Linux Operating systems. Most microcontroller systems are limited to windows.

CODE IMPLEMENTATION

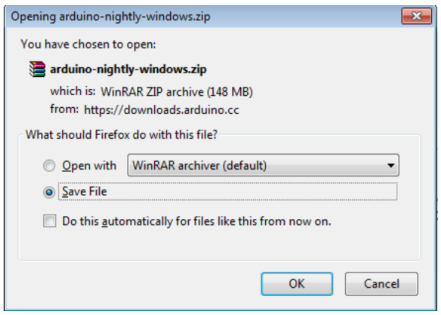
After learning about the main parts of the Arduino UNO board, we are ready to learn how to set up the Arduino IDE. Once we learn this, we will be ready to upload our program on the Arduino board.

In this section, we will learn in easy steps, how to set up the Arduino IDE on our computer and prepare the board to receive the program via USB cable.

Step 1 − First you must have your Arduino board (you can choose your favourite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), the kind you would connect to a USB printer as shown in the following image.

In case you use Arduino Nano, you will need an A to Mini-B cable instead as shown in the following image.

Step 2 − Download Arduino IDE Software.



You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.

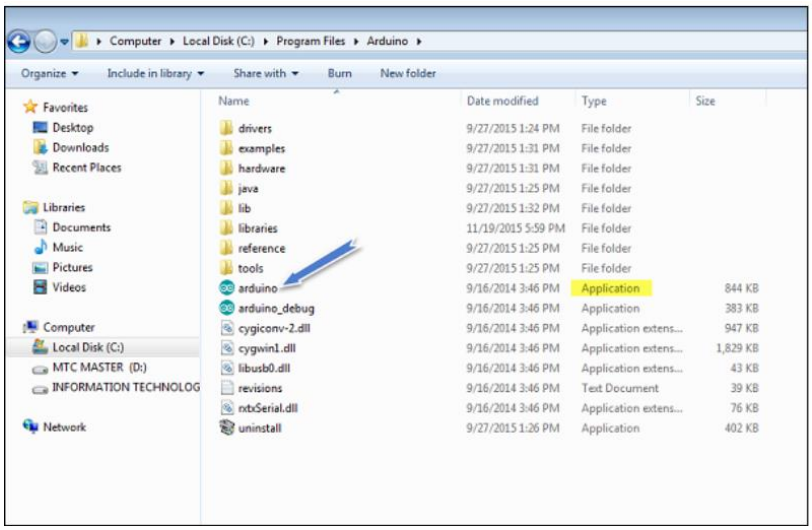
Step 3 − Power up your board.

The Arduino Uno, Mega, Duemilanove and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If you are using an Arduino Diecimila, you have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks. Check that it is on the two pins closest to the USB port.

Connect the Arduino board to your computer using the USB cable. The green power LED (labelled PWR) should glow.

Step 4 − Launch Arduino IDE.

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to start the IDE.

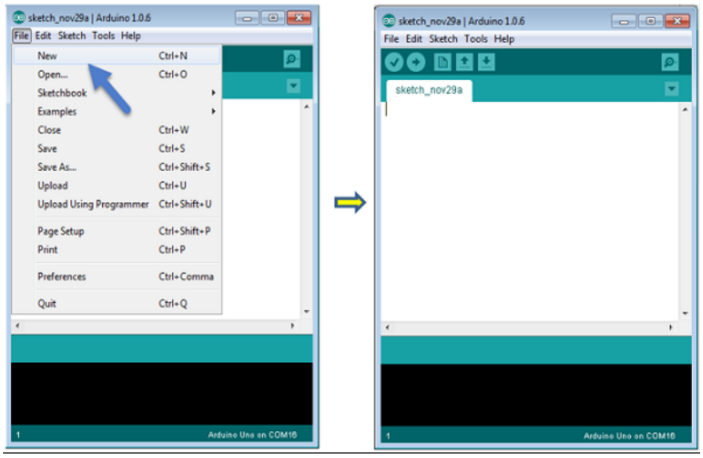


Step 5 − Open your first project.

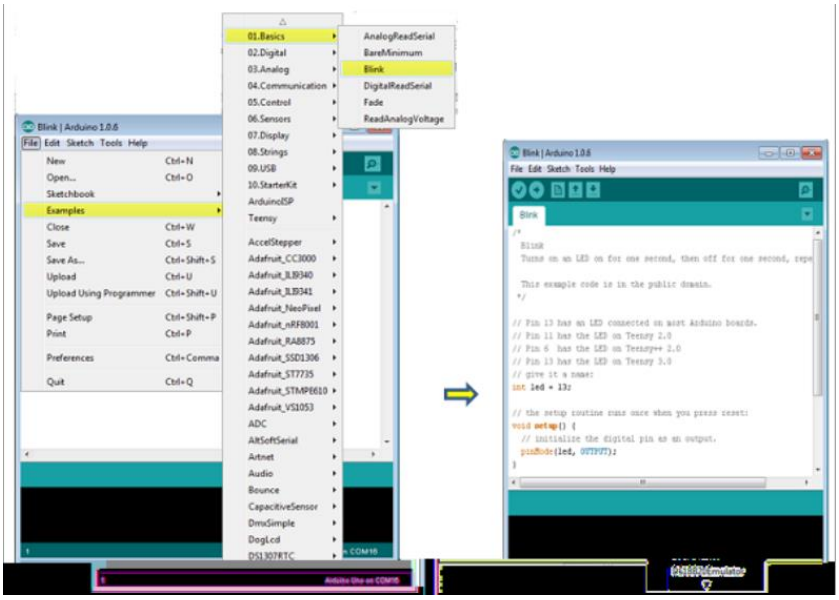
Once the software starts, you have two options −

- Create a new project.
- Open an existing project example.

To create a new project, select File → **New**.



To open an existing project example, select File → Example → Basics → Blink.

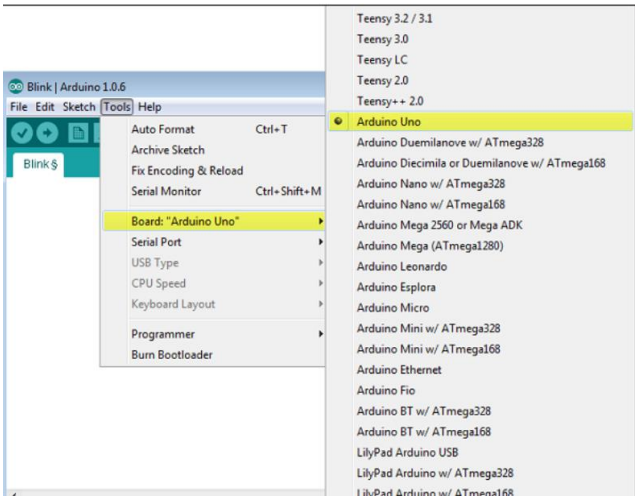


Here, we are selecting just one of the examples with the name **Blink**. It turns the LED on and off with some time delay. You can select any other example from the list.

Step 6 − Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

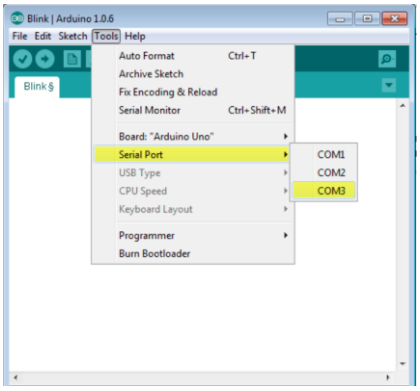
Go to Tools → Board and select your board.



Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

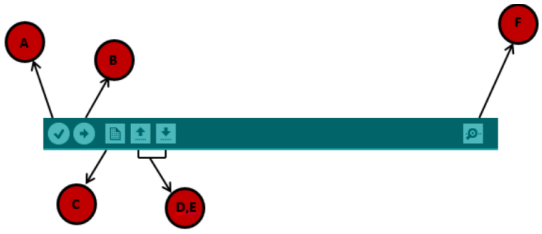
Step 7 − Select your serial port.

Select the serial device of the Arduino board. Go to **Tools → Serial Port** menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.



Step 8 − Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.



A − Used to check if there is any compilation error.

B − Used to upload a program to the Arduino board.

C − Shortcut used to create a new sketch.

D − Used to directly open one of the example sketches.

E − Used to save your sketch.

F − Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

Note − If you have an Arduino Mini, NG, or other board, you need to press the reset button physically on the board, immediately before clicking the upload button on the Arduino Software.

CHAPTER 5

CODE IMPLEMENTATION

5.1 SOURCE CODE

```
#include <LiquidCrystal.h>
SoftwareSerial mySerial(8,9);
const int rs = A0, en = A1, d4 = A2, d5 = A3, d6 = A4, d7 = A5;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
int temper = 0;
int i = 0;
int tmp;
char str[15];
String soil_status;
String getStr;
const int motor = 5;//pwm
const int soil_sen = 3;
const int buzzer = 13;
void setup() {
  Serial.begin(9600);
  pinMode(soil_sen, INPUT);
  pinMode(motor, OUTPUT);
  digitalWrite(motor, HIGH); // stop
  pinMode(buzzer, OUTPUT);
  digitalWrite(buzzer, LOW); // stop buzzing
  lcd.begin(16, 2);
```

```
 lcd.setCursor(0, 0);
 lcd.print("Agrilture Motor");
 lcd.setCursor(0, 1);
 lcd.print("Contl using GSM");
// mySerial.print(" Agriculture Motor");
 delay(1000);

 //----------------------gsm-----------------
 lcd.setCursor(0, 0);
 lcd.print("Gsm Module      ");
 lcd.setCursor(0, 1);
 lcd.print("Initilizing.... ");
 delay(1000);
 mySerial.begin(9600);
 delay(1000);
 mySerial.println("AT+CNMI=2,2,0,0,0");
 delay(3000);
 mySerial.println("AT+CMGF=1");
 delay(3000);
 mySerial.println("AT+CMGS=\"+917981711009\"\r"); // Replace x with mobile number
 delay(3000);
 mySerial.println("System is ready to send Messages.");// The SMS text you want to send
 delay(3000);
 mySerial.println((char)26); // ASCII code of CTRL+Z
 delay(3000);
 lcd.setCursor(0, 0);
 lcd.print("Gsm Module       ");
 lcd.setCursor(0, 1);
 lcd.print("Initilized..... ");
 Serial.println("Gsm Module Initilized.");
```

```
  delay(1000);
  lcd.clear();
}//setup
void loop()
{
  int soil_sen_data = digitalRead(soil_sen);
  if (soil_sen_data == 0)//wet
  {
    soil_status = "wet";
    digitalWrite(motor, HIGH)  ;//off
    Serial.println("off");
    lcd.setCursor(0, 0);
    lcd.print("Soil:");
    lcd.setCursor(5, 0);
    lcd.print(soil_status );
    lcd.setCursor(0, 1);
    lcd.print("MOTOR:");
    lcd.setCursor(6, 1);
    lcd.print("OFF");
    delay(1000);
    lcd.clear();
  }
  else
  {
    digitalWrite(motor, LOW)  ;//on
   Serial.println("ON");
    lcd.setCursor(0, 0);
    lcd.print("Soil:");
    lcd.setCursor(5, 0)
    lcd.print(soil_status );
```

```
  lcd.setCursor(0, 1);
  lcd.print("MOTOR:");
  lcd.setCursor(6, 1);
  lcd.print("ON");
  delay(1000);
  lcd.clear();
  soil_status = "Dry";
 }
//----------------------gsm-------------------------
 while (mySerial.available())
 {
  lcd.setCursor(0, 0);
  lcd.print("loop-1       ");
  delay(500);
  i = 0;
  if (mySerial.find('*'))
  {
   //delay(1000);
   while (mySerial.available())
   {
    char inChar =  mySerial.read();
    str[i++] = inChar;
    Serial.println(str[i]);
    if (inChar == '#')
    {
     temper = 1;
     str[i - 1] = '\0';
     return;
    }
   }
```

```
    Serial.println(str);
   }
////////////////////////////////////////
 if (temper == 1)
 {
  lcd.setCursor(0, 0);
  lcd.print("Msg Rcvd     ");
  lcd.setCursor(0, 1);
  lcd.print(str);
  Serial.println("Msg Rcvd");
  delay(1000);
  digitalWrite(buzzer, HIGH); // buzz
  delay(500);
  digitalWrite(buzzer, LOW); // stop buzzing
  delay(300);
  lcd.clear();
  temper = 0;
  getStr="Hi,Message from kit.Soil status is:";
  getStr+=soil_status;
  getStr+=".";
  //getStr+="Switch ON/Off the motor based on status.";
  //---------------- status sms----------------------------
  if (!(strncmp(str, "status", 6)))
  {
   lcd.print("  Reading     ");
   lcd.setCursor(0, 1);
   lcd.print("  Status..... ");
   delay(600);
   lcd.clear();
   //-------------------------------------
```

```
      Send_sms_status();
      //---------------------------------


    }
  }//temp

}//loop
void Send_sms_status()
{
  lcd.setCursor(0, 0);
  lcd.print("Sending Status");
  lcd.setCursor(0, 1);
  lcd.print("SMS..... ");
  mySerial.println("AT+CNMI=2,2,0,0,0");
  delay(3000);
  mySerial.println("AT+CMGF=1");
  delay(3000);
  mySerial.println("AT+CMGS=\"+917981711009\"\r"); // Replace x with mobile number
  delay(3000);
  delay(3000);
  mySerial.println((char)26); // ASCII code of CTRL+Z
  delay(3000);
  Serial.println("Status   sent");
  lcd.setCursor(0, 0);
  lcd.print("Message         ");
  lcd.setCursor(0, 1);
  lcd.print("Sent..... ");
  delay(1000);
  lcd.clear();
}
```

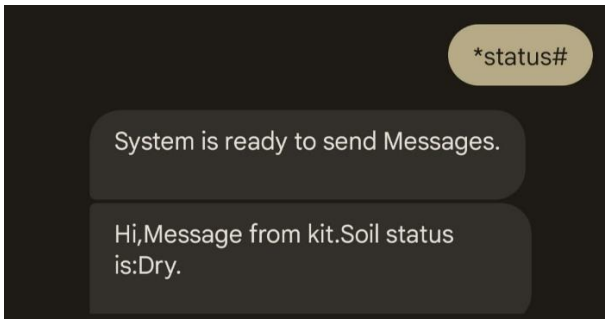
CHAPTER 6

RESULTS

6.1 SNAPSHOTS

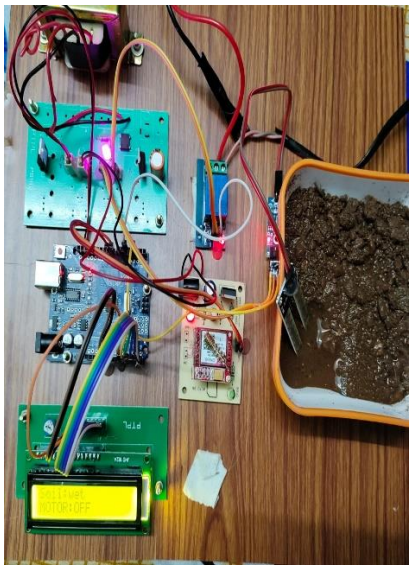
WHEN THE SOIL IS DRY

Kit

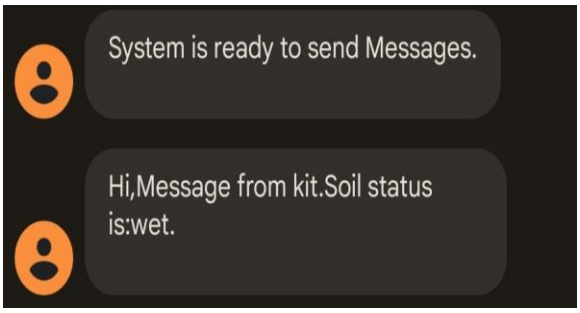


Message received to mobile

WHEN THE SOIL IS WET



Kit



Message received to mobile

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 CONCLUSION

An automatic plant irrigation system using Arduino is designed in this project. The prototype of the model worked properly when tested on different soils. The components that we use in the system are readily available and easy to operate.Thus, this system acts as an effectual method of irrigation. It is far better than themanual irrigation process which requires a lot of manpower and time. By using theapp, the farmer can operate the system from distant places. The farmer can utilizethis time in other significant activities. Also, the major issue of water scarcity is dealtwith. No amount of water is wasted in the process of irrigation. Thus, this system canbe very useful in areas where water is in short supply.Farmers can thus benefit fromthe enhanced crop yields. Irrigation play a vital role for economic in any developing countries like India. Over the years, professionals involved in irrigation implemented manual method of irrigation. The manual method has lots of drawbacks and is quite unreliable for irrigation of big areas. Irrigation has direct impact on cost and production of final product. This system aims to eradicate the traditional manual method of irrigation which needs to be improved over the time. This prototype has many advantages which make it a good alternative to the current approaches since it facilitates the farmers to assist them in daily needs of the monitoring and controlling the field environmental parameters with minimum cost and user friendliness.

7.2 FUTURE ENHANCEMENTS

A lot of future scope are available that can be used with this work to improve the efficiency and effectiveness of the system, the following recommendations can be put into consideration. The idea of using IOT for irrigation can be implemented with this system. Other activities in farming such as cattle management, fire detection and climate control can be introduced with this system. This system can be improvised by adding a Webscaper which can predict the weather and water the plants/crops accordingly. If rain is forecasted, less water is let out for the plants. The future work of the system can include the addition of temperature sensors and a more powerful motor to pump water to the fields. Thus,the large-scale implementation of the project can also be done.

BIBLIOGRAPHY

REFERENCES

- [1] Govt. of India, “Indian economic survey”, 2018. http://mofapp. nic.in:8080/economic survey/
- [2] Sharma, D., Bhondekar, A. P., Ojha, A., Shukla, A. K., & Ghanshyam, C. (2016). A Technical Assessment of IOT for Indian Agriculture Sector. In 47th Mid-Term Symposium on Modern Information and Communication Technologies for Digital India, Chandigarh.
- [3] K.N. Manjula Swathi and D. Sree Sandhya, Intelligent Automatic Plant Irrigation System
- [4] G. Vellidis, M. Tucker, C. Perry, C. Kvien, C. Bednarz, “A Real-Time Wireless Smart Sensor Array for Scheduling Irrigation”, National Environmentally Sound Production Agriculture Laboratory (NESPAL), 2007.
- [5] K.S.S. Prasad, Nitesh Kumar, Nitish Kumar Sinha and Palash Kumar Saha, 2012, Water-Saving Irrigation System Based on Automatic Control by Using GSM Technology, Middle-East Journal of Scientific Research 12 (12): 1824- 1827, ISSN 1990-9233, pp-1824-1827.
- [6] D Anurag, Siuli Roy and Somprakash Bandyopadhyay, "Agro-Sense- Precision Agriculture using Sensor-based Wireless Mesh Networks", ITU-T Innovation in NGN Kaleidoscope Conference, 12-13 May 2008.
- [7] C. Arun and K. Lakshmi Sudha, "Agricultural Management using Wireless Sensor Networks – A Survey", 2nd International Conference on Environment Science and Biotechnology IPCBEE, 2012.
- [8] Nazleeni Samiha Haron, Izzatdin Abdul Aziz, Mohd Hilmi Hasan, Mohd Jimmy Ismail and Mazlina Mehat, Remote Monitoring in Agricultural Greenhouse Using Wireless Sensor and Short Message Service (SMS), 2008.
- [9] Jeonghwan Hwang, Changsun Shin and Hyun Yoe, "Study on an Agricultural Environment Monitoring Server System using Wireless Sensor Networks", 2010.
- [10] Laxmi Shabadi, Nandini Patil, M Nikita., J Shruti., P Smitha. and C Swati., "Irrigation Control System Using Android and GSM for Efficient Use of Water and Power", International Journal of Advanced Research in Computer Science, vol. 4, no. 7, July 2014.
- [11] S. R. Kumbhar and Arjun P. Ghatule, "Microcontroller based Controlled Irrigation System for Plantation", Proceedings of the International Multiconference of Engineers and Computer Scientists, March 2013.
- [12] Mahir Dursun and Semih Ozden, "A wireless application of drip irrigation 88 Pavan Kumar Naik Arun Kumbi Kirthishree Katti and Nagaraj Telkar automation supported by soil moisture sensors", Scientific Research and Essays, vol. 6, no. 7, pp. 1573-1582, April 2011.

TEXTBOOKS

Microcontrollers [THEORY AND APPLICATIONS] By Ajay V Deshmukh.

PIC Microcontroller and Embedded System using Assembly and C for pic18 By Muhammad Ali Mazidi.

WEBSITES

- for information of gsm module, [online] Available: https-//www.elprocus.com/gsm-architecture-features-working/.
- for the info of Arduino ide, [online] Available: https-//en.wikipedia.org/wiki/Arduino.
- for the info of ArduinoUno, [online] Available: https-//en.wikipedia.org/wiki/Arduino Uno.
- About soil moisture sensor, Available: https//www.vernier.com/products/sensors/sms-bta/.
- www.engineersgarage.com
- http://www.way2project.com