SMART FARMING SYSTEM USING IOT

FOR EFFICIENT CROP GROWTH

A PROJECT REPORT

Submitted by

D.MOGANA PRIYA (310918104031)

N.PAVITHRA (310918104036)

P.SHREENITHI (310918104701)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

JERUSALEM COLLEGE OF ENGINEERING,

CHENNAI 600 100

MAY 2022

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**SMART FARMING SYSTEM USING IOT FOR EFFICIENT CROP GROWTH**" is the bonafide work of **MOGANAPRIYA.D(310917104031) PAVITHRA.N (310917104036) SHREENITHI.P(310918104701)** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Ms. H. Mercy,	**Dr. G.Sasikumar,**
Associate Professor,	**Professor & HOD- Incharge,**
Department of CSE,	Department of CSE,
Jerusalem College of Engineering,	Jerusalem College of Engineering,
Pallikaranai, Chennai – 600 100.	Pallikaranai, Chennai – 600 100

Submitted for the University examination held on ________________________

INTERNAL EXAMINER **EXTERNAL EXAMINE**

ACKNOWLEDGEMENT

We extended our warmest gratitude to **Dr.M.Mala, CEO**, Jerusalem College of Engineering for her enduring support.

We express our sincere thanks to **Dr.S.Ramesh, Principal,** Jerusalem College of Engineering for his kindness, which enabled us to do this project.

We express our sincere thanks to **Dr. G.Sasikumar, Professor & HOD-Incharge,** Computer Science and Engineering, Jerusalem College of Engineering for his support throughout the project.

We would like to take this opportunity to express our sincere thanks to our internal supervisor, **Mrs.H.Mercy, Associate Professor**, Department of Computer Science and Engineering, Jerusalem College of Engineering for her valuable guidance, inspirations and technical support throughout the project.

We thank our entire faculty and the supporting staff of the Department of Computer Science and Engineering, Jerusalem College of Engineering, for their co-operation and assistance in the successful completion of the project.

MOGANAPRIYA. D (310918104031)

PAVITHRA.N (310918104036)

SHREENITHI.P (310918104701)

ABSTRACT

One of the parameter of nature and an important aspect for living which is known as agriculture is being managed in a smart way by sensors and updated technology. Smart agriculture is a farming system which uses IoT technology. This emerging system increases the quantity and quality of agricultural products. IoT devices provide information about nature of farming fields and then take action depending on the farmer input. An IoT based advanced solution for monitoring the soil conditions and atmosphere for efficient crop grow this presented. The developed system is capable of monitoring temperature, humidity, soil moisture level using Node MCU and several sensors connected to it. Also, a notification in the form of SMS will be sent to farmer, phone using Wi-Fi about environmental condition of the field.

TABLE OF CONTENTS

LIST OF ABBREVATIONS

S.NO	ABBREVATION	EXPANSION
1.	IOT	Internet of Things
2.	GSM	Global System Mob Communication
3.	IR	Infrared
4.	DH11	Digital Temperature & Humidity
5.	PHP	Hypertext Preprocessor
6.	TX	Transmit
7.	RX	Receive
8.	LED	Light emitting Diode
9.	GND	Ground
10.	ADC	Analog to Digital Convertor

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

As the world is trending towards new technologies and implementations it is a necessary goal to trend up in agriculture too. Many researches are done in the field of agriculture and most of them signify the use of wireless sensor network that collect data from different sensors deployed at various nodes and send it through the wireless protocol. The collected data provide the information about the various environmental factors. Monitoring the environmental factors is not the complete solution to increase the yield of crops. There are number of other factors that decrease the productivity. Hence, automation must be implemented in agriculture to overcome these problems. In order to provide solution to such problems, it is necessary to develop an integrated system which wills improve productivity in every stage. But, complete automation in agriculture is not achieved due to various issues. Though it is implemented in the research level, it is not given to the farmers as a product to get benefitted from the resources. Hence, this paper deals about developing smart agriculture using IoT and given to the farmers.

1.1 GENERAL

Smart Farming System assisting farmers in getting Live Data (Temperature, Soil Moisture) for efficient environment monitoring which will enable them to expand their general surrounder and quality of products.

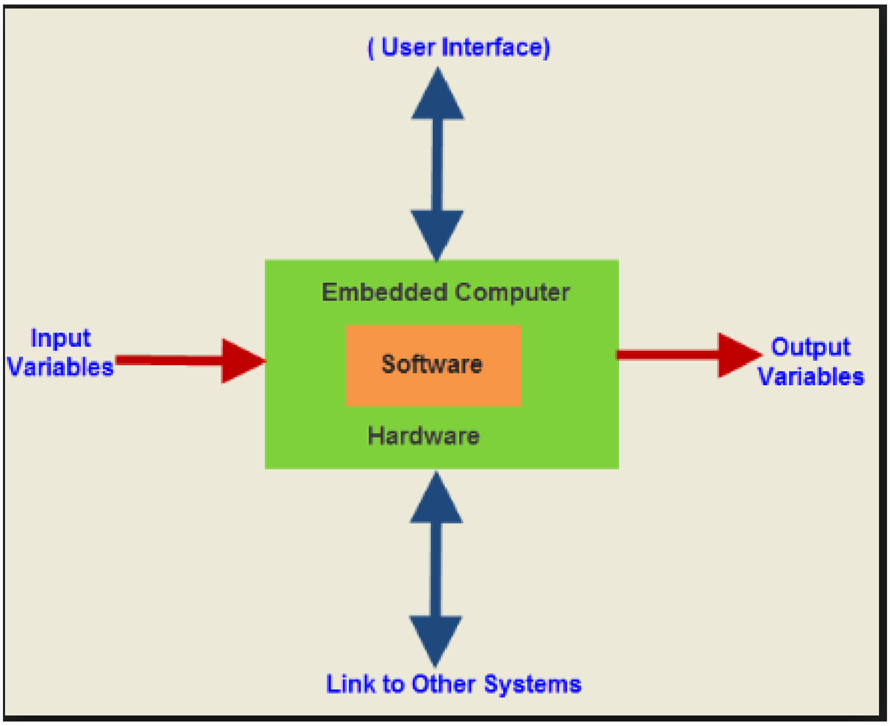
1.2 EMBEDDED SYSTEM:

An embedded system is a controller programmed and controlled by a real-time operating system (RTOS) with a dedicated function within a larger mechanical or electrical system, often with real-time consumption of embedded systems computing constraints.

It is embedded as part of a complete device often including hardware and mechanical parts. Embedded systems control many devices in common use today. Ninety-eight percent of all microprocessors are manufactured to serve as embedded system component.

Examples of properties of typical embedded computers when compared with general-purpose counterparts are low power consumption, small size, rugged operating ranges, and low per-unit cost. This comes at the price of limited processing resources, which make them significantly more difficult to program and to interact with.

However, by building intelligence mechanisms on top of the hardware, taking advantage of possible existing sensors and the existence of a network of embedded units, one can both optimally manage available resources at the unit and network levels as well as provide augmented functions, well beyond those available. For example, intelligent techniques can be designed to manage power.



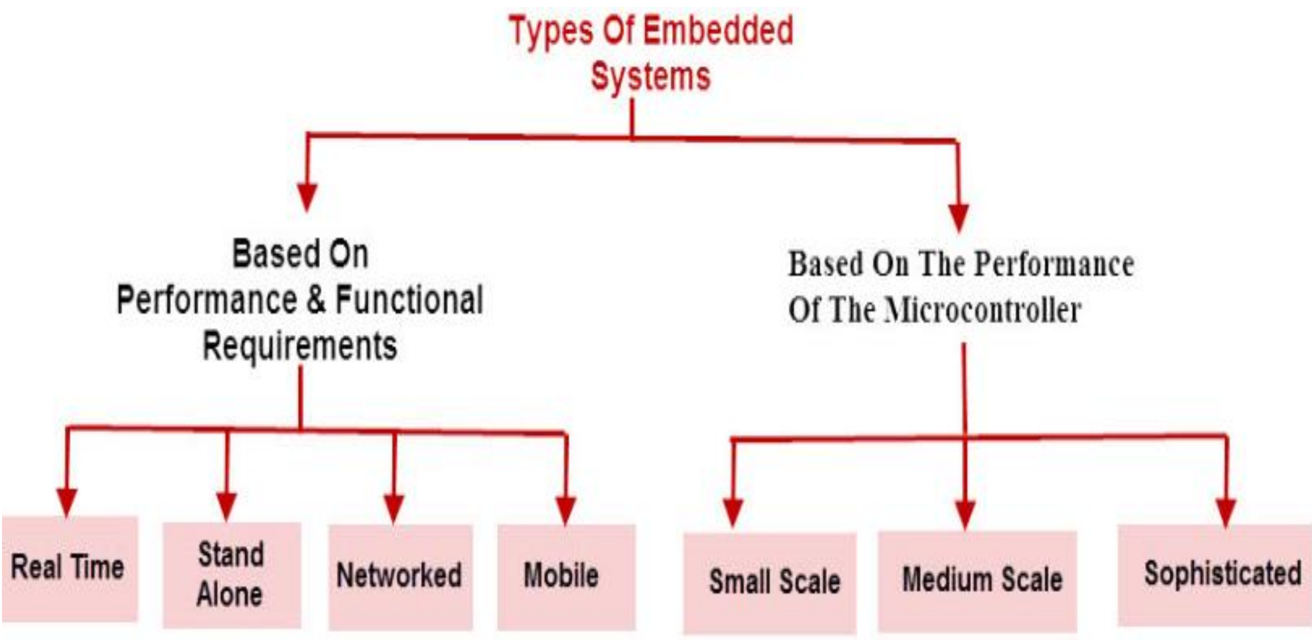
Embedded Computer Sub-Assembly for Electronic Voting Machine

Embedded systems are commonly found in consumer, industrial, automotive, medical, commercial and military applications.

Telecommunications systems employ numerous embedded systems from telephone switches for the network to cell phones at the end user. Computer networking uses dedicated routers and network bridges to route data.

Consumer electronics include MP3 players, mobile phones, video game consoles, digital cameras, GPS receivers, and printers. Household appliances, such as microwave ovens, washing machines and dishwashers, include embedded systems to provide flexibility, efficiency and features.

CLASSIFICATIONS OF EMBDDED SYSTEMS:

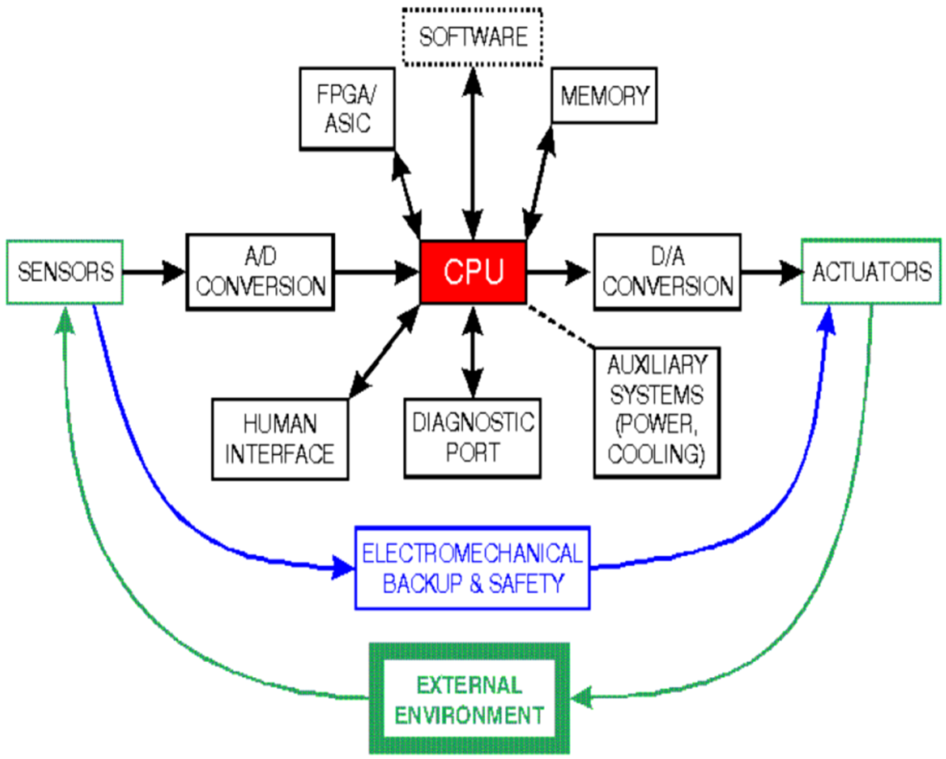


Advanced HVAC systems use networked thermostats to more accurately and efficiently control temperature that can change by time of day and season. Home automation uses wired- and wireless-networking that can be used to control lights, climate, security, audio/visual, surveillance, etc., all of which use embedded devices for sensing and controlling.

like traffic lights, factory controllers, and largely complex systems like hybrid vehicles, MRI, and avionics Embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations. Complexity varies from low, with a single microcontroller

Block diagram of an embedded system:

An embedded system usually contains an embedded processor. Many appliances that have a digital interface microwaves, VCRs, cars utilize embedded systems. Some embedded systems include an operating system. Others are very specialized resulting in the entire logic being implemented as a single program. These systems are embedded into some device for some specific purpose other than to provide general purpose computing.



Block diagram of a typical embedded system

EMBEDDED SYSTEMS APPLICATIONS:

Embedded systems in automobiles include motor control, cruise control, body safety, engine safety, robotics in an assembly line, car multimedia, car entertainment, E-com access, mobiles etc.

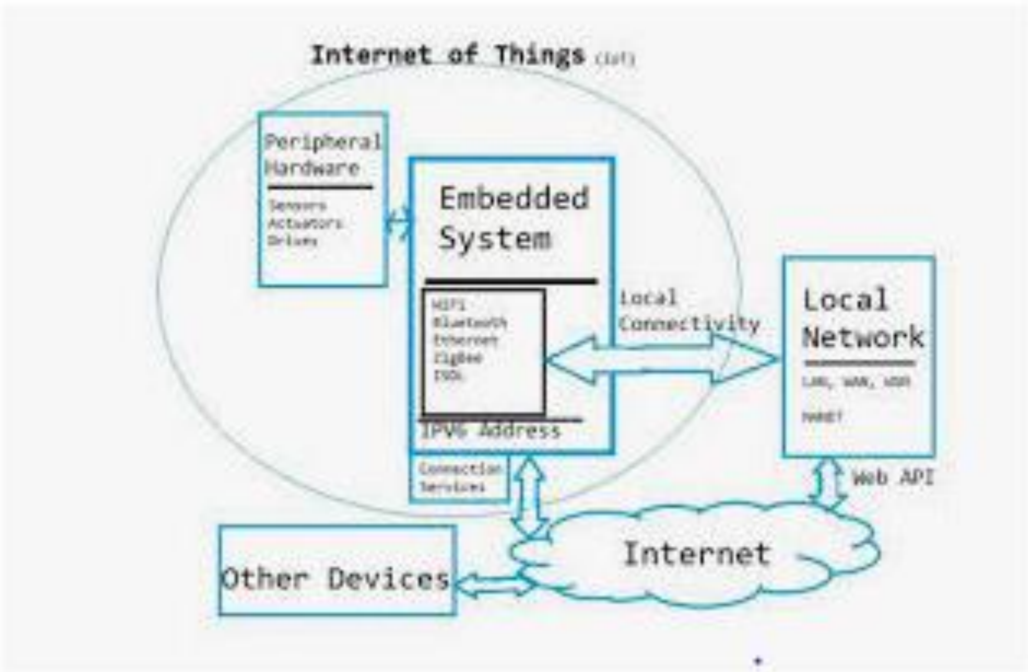
- Embedded systems in telecommunications include networking, mobile computing, and wireless communications, etc.
- Embedded systems in smart cards include banking, telephone and security systems.
- Embedded Systems in satellites and missiles include defense, communication, and aerospace
- Embedded systems in computer networking & peripherals include image processing, networking systems, printers, network cards, monitors and displays
- Embedded Systems in digital consumer electronics include set-top boxes, DVDs, high definition TVs and digital cameras.

1.3 INTERNET OF THINGS:

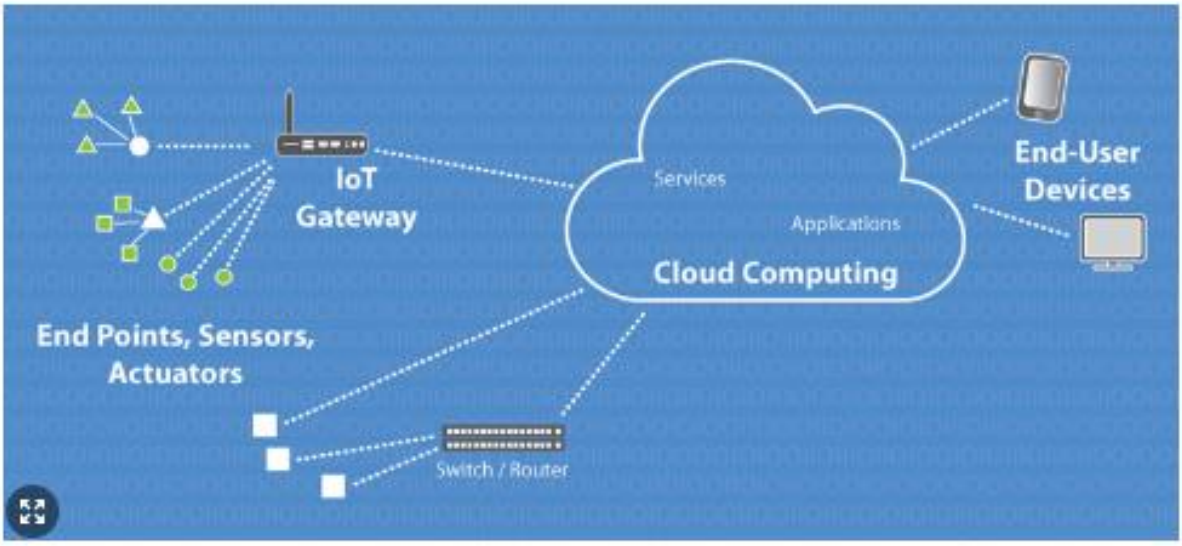
- The term Internet of Things generally refers to scenarios where network connectivity and computing capability extends to objects, sensors and everyday items not normally considered computers, allowing these devices to generate, exchange and consume data with minimal human intervention. There is, however, no single, universal definition.

- Enabling Technologies: The concept of combining computers, sensors, and networks to monitor and control devices has existed for decades. The recent confluence of several technology market trends, however, is bringing the Internet of Things closer to widespread reality. These include Ubiquitous Connectivity, Widespread Adoption of IP-based Networking, Computing Economics, Miniaturization, Advances in Data.

- Connectivity Models: IoT implementations use different technical communications models, each with its own characteristics. Four common communications models described by the Internet Architecture Board include: Device-to-Device, Device-to-Cloud, Device-to-Gateway, and Back-End Data-Sharing. These models highlight the flexibility in the ways that IoT devices can connect and provide value to the user.



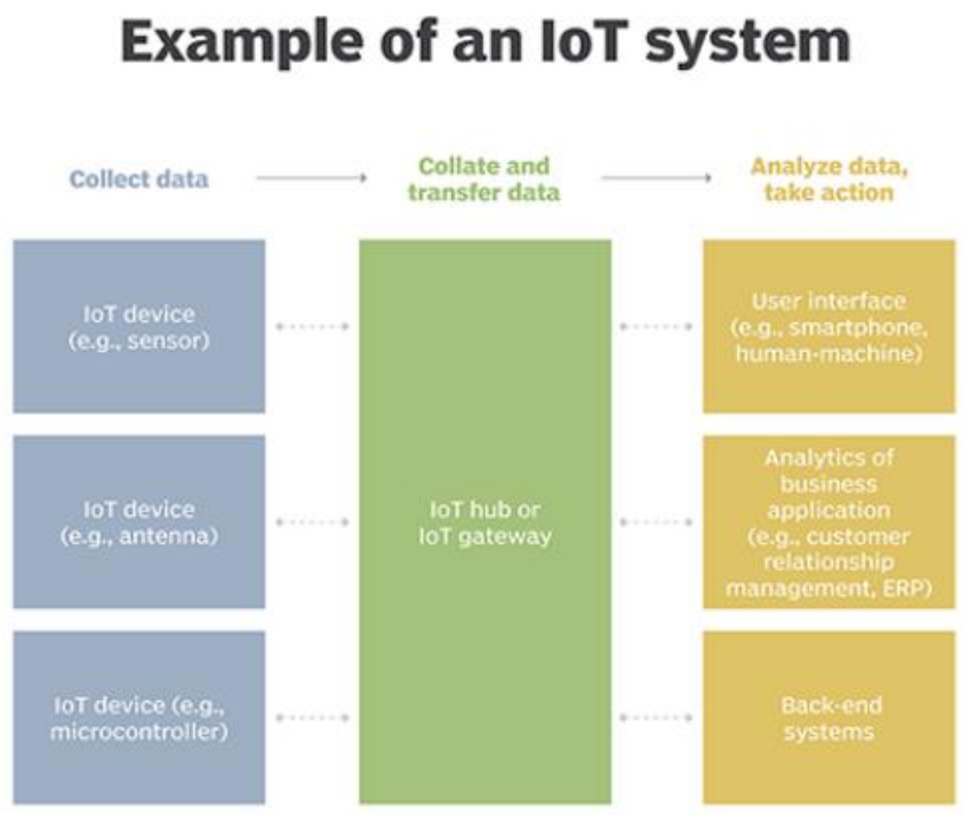
IoT devices are implemented using both hardware and software components. Dedicated hardware components are used to implement the interface with the physical world, and to perform tasks which are more computationally complex. Microcontrollers are used to execute software that interprets inputs and controls the system. This module discusses the roles of both the hardware and software components in the system. The functions of common hardware components are described and the interface between the software and hardware through the microcontroller is explained. IoT devices often use an operating system to support the interaction between the software and the microcontroller. We will define the role of an operating system in an IoT device and how an IoT operating system differs from a standard one.



How IoT works

An IoT ecosystem consists of web-enabled smart devices that use embedded processors, sensors and communication hardware to collect, send and act on data they acquire from their environments. IoT devices share the sensor data they collect by connecting to an IoT gateway or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices -- for instance, to set them up, give them instructions or access the data.

The connectivity, networking and communication protocols used with these web-enabled devices largely depend on the specific IoT applications deployed.



Benefits of IoT

The internet of things offers a number of benefits to organizations, enabling them to:

- monitor their overall business processes;
- improve the customer experience;
- save time and money;
- enhance employee productivity;
- integrate and adapt business models;
- make better business decisions; and
- generate more revenue.

IoT encourages companies to rethink the ways they approach their businesses, industries and markets and gives them the tools to improve their business strategies.

Consumer and enterprise IoT applications

There are numerous real-world applications of the internet of things, ranging from consumer IoT and enterprise IoT to manufacturing and industrial IoT (IoT). IoT applications span numerous verticals, including automotive, telco, energy and more.

Wearable devices with sensors and software can collect and analyze user data, sending messages to other technologies about the users with the aim of making users' lives easier and more comfortable. Wearable devices are also used for public safety -- for example, improving first responders' response times during emergencies by providing optimized routes to a location or by tracking construction workers' or firefighters' vital signs at life-threatening sites.

In healthcare, IoT offers many benefits, including the ability to monitor patients more closely to use the data that's generated and analyze it. Hospitals often use IoT systems to complete tasks such as inventory management, for both pharmaceuticals and medical instruments.

1.4 SUMMARY

This chapter deals with the introduction of Smart Farming System.

CHAPTER 2

LITERATURE SURVEY

2.1 GENERAL

Our literature survey focuses on papers related to Smart Farming System using IOT. Most papers mentioned here are focused on the way technology used to maintained the farm.

2.2 RELATED WORKS

AUTHOR: Nikesh Gondchawar, Dr. R.S. Kawitkar

TITLE:"IoT Based Smart Agriculture", International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)

DATE: 6, June 2018

DESCRIPTION

Internet of Things (IoT) plays a crucial role in smart agriculture. Smart farming is an emerging concept, because IoT sensors capable of providing information about their agriculture fields. The paper aims making use of evolving technology i.e. IoT and smart agriculture using automation. Monitoring environmental factors is the major factor to improve the yield of the efficient crops. The feature of this paper includes monitoring temperature and humidity in agricultural field through sensors using CC3200 single chip. Camera is interfaced with CC3200 to capture images and send that pictures through MMS to farmers mobile using Wi-Fi.

AUTHOR:S. R. Nandurkar, V. R. Thool, R. C. Thool,

TITLE:"Design and Development of Precision Agriculture System Using Wireless Sensor Network", IEEE International Conference on Automation, Control, Energy and Systems (ACES)

DATE:2017

DESCRIPTION

Crop farming in India is labour intensive and obsolete. Farming is still dependent on techniques which were evolved hundreds of years ago and doesn't take care of conservation of resources. The newer scenario of decreasing water tables, drying up of rivers and tanks, unpredictable environment presents an urgent need of proper utilization of water. We have the technology to bridge the gap between water usage and water wastage. Technology used in some developed countries is too expensive and complicated for a common farmer to understand. Our project is to give cheap, reliable, cost efficient and easy to use technology which would help in conservation of resources such as water and also in automatizing farms. We proposed use of temperature and moisture sensor at suitable locations for monitoring of crops. The sensing system is based on a feedback control mechanism with a centralized control unit which regulates the flow of water on to the field in the real time based on the instantaneous temperature and moisture values. The sensor data would be collected in a central processing unit which would take further action. Thus, by providing right amount of water we would increase the efficiency of the farm. The farmer can also look at the sensory data and decide course of action himself. In this paper we are proposed a low cost and efficient wireless sensor network technique to acquire the soil moisture and temperature from various locations of farm and as per the need of crop controller take the decision to make irrigation ON or OFF.

AUTHOR: Chetan Dwarkani M, Ganesh Ram R, Jagannathan S, R.Priyatharshini

TITLE:“Smart Farming System Using Sensors for Agricultural Task Automation”, IEEE International Conference on Technological Innovations in ICT for Agriculture and Rural Development

YEAR:2015.

DESCRIPTION

Agriculture is the broadest economic sector and plays an important role in the overall economic development of a nation. Technological advancements in the arena of agriculture will ascertain to increase the competence of certain farming activities. In this paper, we have proposed a novel methodology for smart farming by linking a smart sensing system and smart irrigator system through wireless communication technology. Our system focuses on the measurement of physical parameters such as soil moisture content, nutrient content, and pH of the soil that plays a vital role in farming activities. Based on the essential physical and chemical parameters of the soil measured, the required quantity of green manure, compost, and water is splashed on the crops using a smart irrigator, which is mounted on a movable overhead crane system. The detailed modeling and control strategies of a smart irrigator and smart farming system are demonstrated in this paper.

AUTHOR: Venkata Naga Rohit Gunturi

TITLE:"Micro Controller Based Automatic Plant Irrigation System" International Journal of Advancements in Research & Technology, Volume 2 **DATE**:April-2016.

DESCRIPTION

The main aim of this paper is to provide information about automatic irrigation to the plants which helps in saving money and water. The entire system is controlled using ATMEGA 328 microcontroller which is giving the interrupt signal to the motor. Temperature sensor and humidity sensor are connected to internal ports of micro controller via comparator, whenever there is a fluctuation in temperature and humidity of the environment these sensors sense the change in temperature and humidity and gives an interrupt signal to the micro-controller and thus the motor is activated, along with this buzzer is used to indicate that pump is on.

AUTHOR: Dr. V. Vidya Devi,G. Meena Kumari,

TITLE:"Real- Time Automation and Monitoring System for Modernized Agriculture" ,International Journal of Review and Research in Applied Sciences and Engineering (IJRRASE) Vol3 No.1. PP 7-12

YEAR:2017.

DESCRIPTION

The technological development in Wireless Sensor Networks made it possible to use in monitoring and control of greenhouse parameter in precision agriculture. Due to uneven natural distribution of rain water it is very crucial for farmers to monitor and control the equal distribution of water to all crops in the whole farm or as per the requirement of the crop. All the parameters of greenhouse require a detailed analysis in order to choose the correct method. With the evolution in wireless sensor technologies and miniaturized sensor devices, it is possible to use them for automatic environment monitoring and controlling the parameters of greenhouse, for Precision Agriculture (PA) application. In the Field bus concept, the data transfer is mainly controlled by a suitable wired communication system, now can be replaced with the hybrid system (wired and wireless) to extract the benefits of both and to automate the system performance and throughput. ZigBee protocols based on IEEE 802.15.4 – 2003 for wireless system are used.

AUTHOR: Joaquín Gutiérrez, Juan Francisco Villa-Medina, Alejandra Nieto-Garibay, and MiguelÁngel Porta Gándara”

TITLE: Automated Irrigation System Using a Wireless Sensor Network and GPRS module”

DATE: 1, January 2016.

DESCRIPTION

An automated irrigation system is developed to reduce the usage level of and to reduce power loss in agricultural fields. The system has of soil-moisture sensor, temperature sensor, water availability sensor, level sensor, EB power availability sensor. In addition, this automated system gives the sensor information, triggering signals to the actuators and also transmits the data through SMS to the farmer. A FUZZY based algorithm is developed with set values of temperature and soil moisture and level of water that is programmed into a microcontroller-based controller system to control the usage of water. A GSM modem is used to transmit about the condition of crop along with various sensors. Because of its energy efficiency and lowest cost, this automated system has to be useful in the areas where the ground water level is less like a geographically isolated area.

AUTHOR:S. Li, J. Cui, Z. Li,

TITLE:"Wireless Sensor Network for Precise Agriculture Monitoring," Fourth International Conference on Intelligent Computation Technology and Automation, Shenzhen, China,

DATE: March 28-29, 2016.

DESCRIPTION

Precision Agriculture Monitor System (PAMS) is an intelligent system which can monitor the agricultural environments of crops and provides service to farmers. PAMS based on the wireless sensor network (WSN) technique attracts increasing attention in recent years. The purpose of such systems is to improve the outputs of crops by means of managing and monitoring the growth period. This paper presents the design of a WSN for PAMS, shares our real-world experience, and discusses the research and engineering challenges in implementation and deployments.

2.3 SUMMARY

Our literature survey mostly focuses on the idea and papers regarding the farming system. Most of the research papers that we surveyed use motor systems for fetching water , as smart farming provides flexibility and has dedicated features for farming.

CHAPTER 3

3.1 GENERAL

IoT smart agriculture products are designed to help monitor crop fields using sensors and by automating irrigation systems. As a result, farmers and associated brands can easily monitor the field conditions from anywhere without any hassle.

3.2 EXISTING SYSTEM:

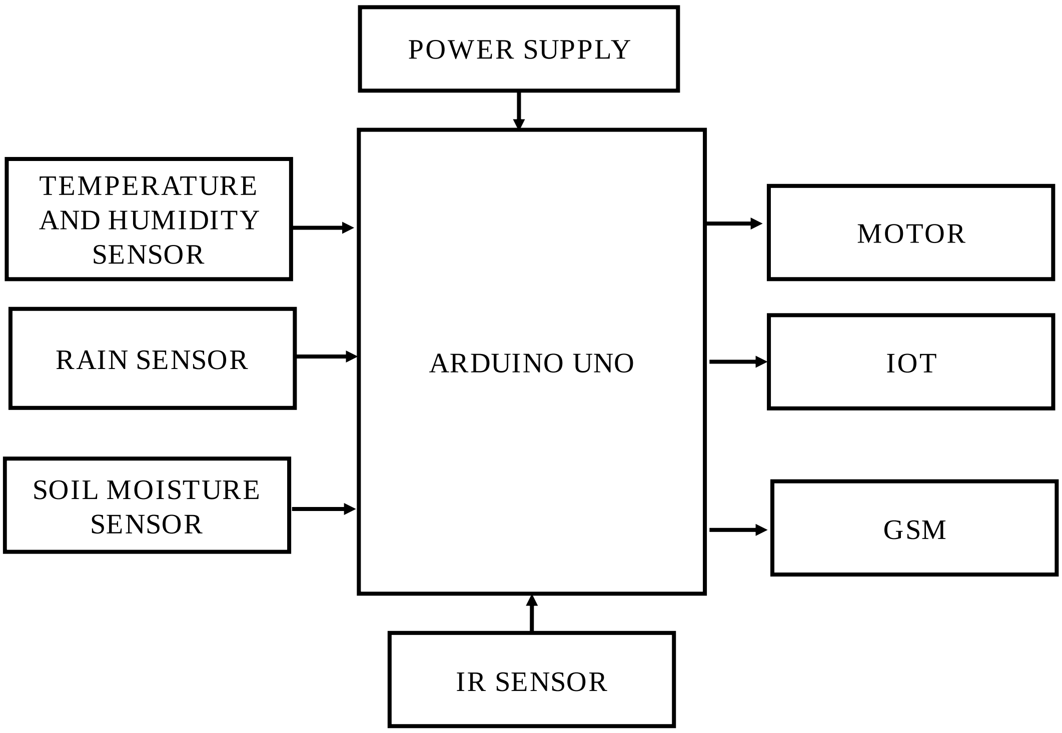
- The existing system also provides essential features for agricultural analysis and monitoring. But there does not present any wireless sensors that will update the values and condition of the agricultural field.
- This system consists only some of the parameters such as sensors and micro controllers. The micro controller used here is not as much as efficient.
- The main drawback of this system is that this is not connected to IOT. So, the values are not monitored properly.

3.3 PROPOSED SYSTEM;

- In the field section, various sensors are deployed in the field like temperature sensor, moisture sensor and water level sensor. The data collected from these sensors are connected to the microcontroller.
- The microcontroller gets switched ON and OFF automatically if the value exceeds the threshold point. Soon after the microcontroller is started, automatically the relay will be turned on and according to that the motor will be operated.
- Other parameters like the temperature, moisture shows the threshold value and the Rain sensor is used just to indicate the level of water inside a tank or the water resource.

- The proposed system will automatically monitor all the parameters of soil and the surrounding and the values are passed to the Arduino.
- Then the motor connected to the system will be operated according to the program written in that.
- And when the person detected in the field it will send an SMS through GSM.

3.4 BLOCK DIAGRAM



3.5 HARDWARE REQUIREMENTS:

1. Arduino UNO
2. Rain Sensor
3. DH11 Sensor
4. Soil moisture sensor
5. GSM module
6. IOT module

7. Motor
8. IR Sensor

3.6 SOFTWARE REQUIREMENTS

- Embedded C
- Arduino IDE
- PHP

3.7 SUMAARY

Our model provides better performance by exploiting the Smart Farming Technology. The performance depends on the throughput of the computer that runs our model and the version of the Embedded C that the computer uses.

CHAPTER-4

SYSTEM DESIGN

4.1 GENERAL

System design is the most important and vital part of any framework as it is used for the development of the system from its theory. This section includes the modules, architecture and various elements that are combined together to form the whole system"s framework.

4.2 MODULE DESCRIPTION:

4.2.1 ARDUINO UNO

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 a 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. Revision 3 of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.

- Stronger RESET circuit.

- Atmega 16U2 replace the 8U2. "Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

POWER:

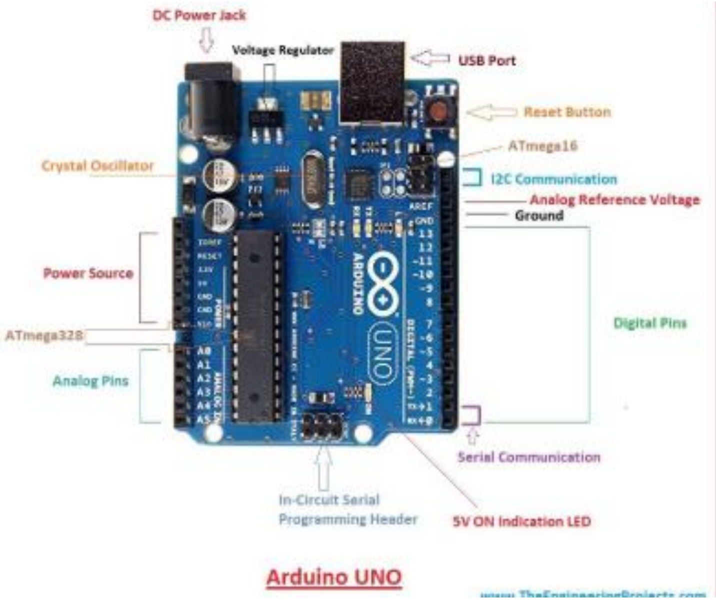
VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

- 5V. This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

- 3V 3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

- GND. Ground pins.

ARDUINO UNO:



Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library). Input and Output Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode (), digital Write (), and digital Read () functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms.

In addition, some pins have specialized functions:

• Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

• External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt () function for details.

- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write () function.

- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off. The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog Reference() function. Additionally, some pins have specialized functionality:

- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library. There are a couple of other pins on the board:

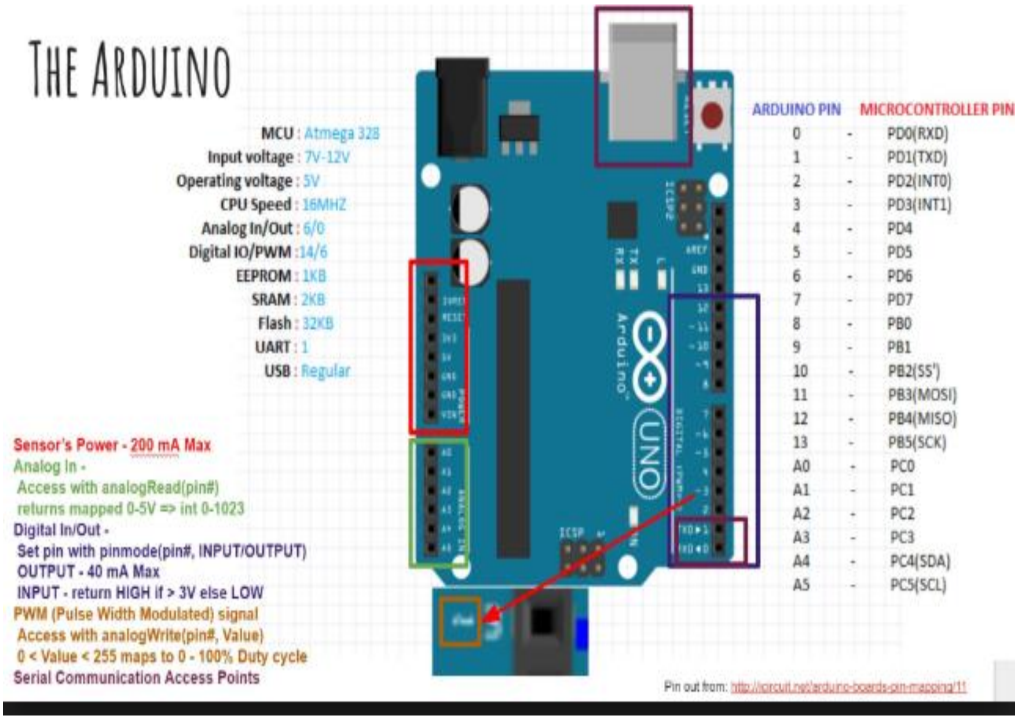
- AREF. Reference voltage for the analog inputs. Used with analog Reference().

- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board. See also the mapping between Arduino pins and ATmega328 ports. The mapping for the Atmega8, 168, and 328 is identical. Communication The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.

The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed.

However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

PIN CONFIGURATION:



USB Overcurrent Protection

The Arduino Uno has a resettable poly fuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than

500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100-mil spacing of the other pins.

4.2.2 SOILMOISTURE SENSOR:

Soil moisture sensors measure the volumetric water content in soil.[1] Since the direct gravimetric measurement of free soil moisture requires removing, drying, and weighting of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with neutrons, as a proxy for the moisture content.

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

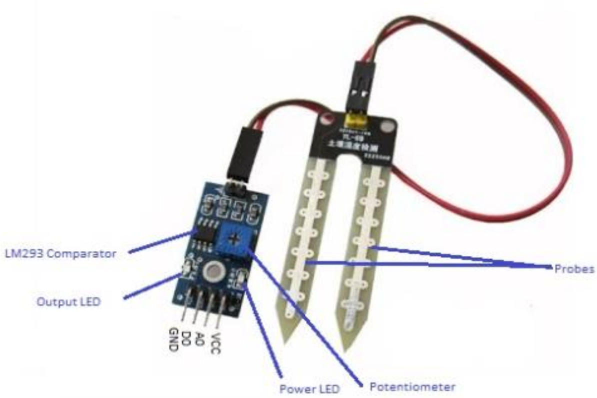
Soil moisture sensors typically refer to sensors that estimate volumetric water content. Another class of sensors measure another property of moisture in soils called water potential; these sensors are usually referred to as soil water potential sensors and include tensiometers and gypsum blocks

Working of Sensor

The soil moisture sensor consists of two probes which are used to measure the volumetric content of water. The two probes allow the current to pass through the soil and then it gets the resistance value to measure the moisture value.

When there is more water, the soil will conduct more electricity which means that there will be less resistance. Therefore, the moisture level will be higher. Dry soil conducts electricity poorly, so when there will be less water, then the soil will conduct less electricity which means that there will be more resistance. Therefore, the moisture level will be lower.

This sensor can be connected in two modes; Analog mode and digital mode. First, we will connect it in Analog mode and then we will use it in Digital mode.



Specifications

The specifications of the soil moisture sensor FC-28 are as follows

Input Voltage	3.3 – 5V
Output Voltage	0 – 4.2V
Input Current	35mA

Output Signal	Both Analog and Digital

Pin Out – Soil Moisture Sensor

The soil Moisture sensor FC-28 has four pins

- VCC: For power
- A0: Analog output
- D0: Digital output
- GND: Ground

The Module also contains a potentiometer which will set the threshold value and then this threshold value will be compared by the LM393 comparator. The output LED will light up and down according to this threshold value.

4.2.3 RAINDROP SENSOR:

Raindrop Sensor is a tool used for sensing rain. It consists of two modules, a rain board that detects the rain and a control module, which compares the analog value, and converts it to a digital value. The raindrop sensors can be used in the automobile sector to control the windshield wipers automatically, in the agriculture sector to sense rain and it is also used in home automation systems.

Raindrop Sensor Features:

Working voltage 5V

Output format: Digital switching output (0 and 1), and analog voltage output AO

Potentiometer adjust the sensitivity

Uses a wide voltage LM393 comparator

Comparator output signal clean waveform is good, driving ability, over 15mA

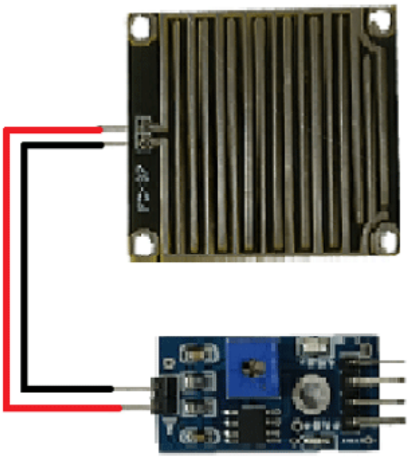
Anti-oxidation, anti-conductivity, with long use time

With bolt holes for easy installation

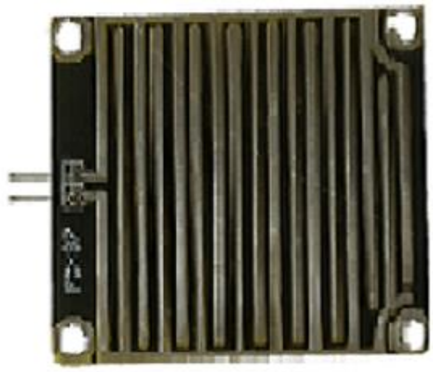
Small board PCB size: 3.2cm x 1.4cm

How to use Raindrop sensor:

Interfacing the raindrop sensor with a microcontroller like 8051, Arduino, or PIC is simple. The rain board module is connected with the control module of the raindrop sensor as shown in the below diagram.

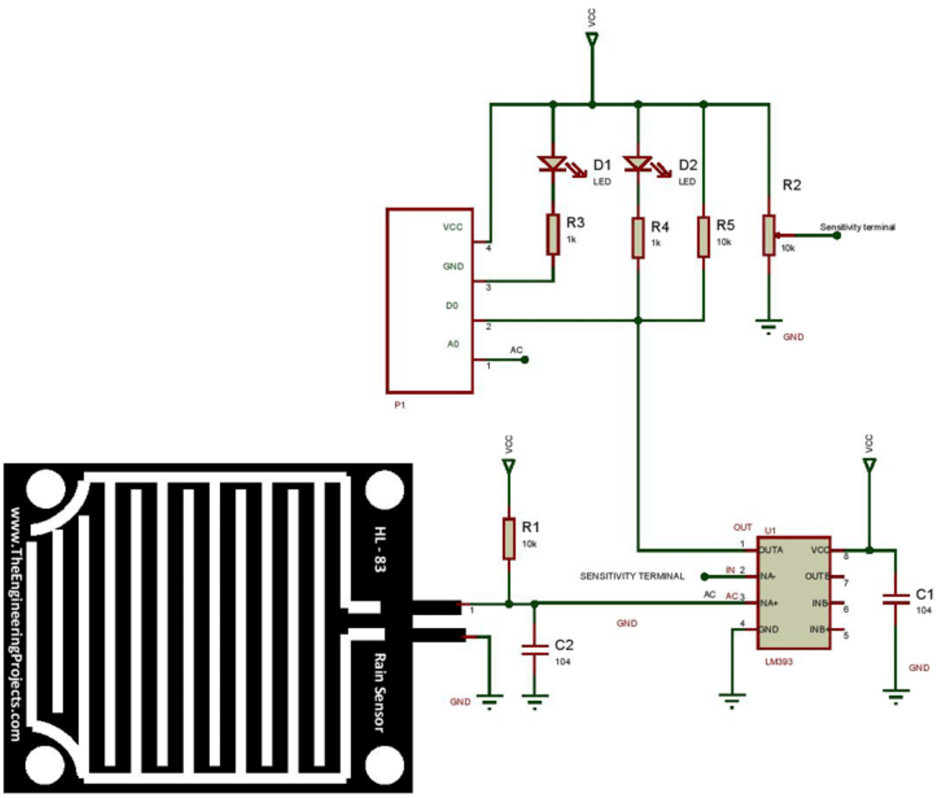


The control module of the raindrop sensor has 4 outputs. VCC is connected to a 5V supply. The GND pin of the module is connected to the ground. The D0 pin is connected to the digital pin of the microcontroller for digital output or the analog pin can be used. To use the analog output, the A0 pin can be connected to the ADC pin of a microcontroller. In the case of Arduino, it has 6 ADC pins, so we can use any of the 6 pins directly without using an ADC converter. The sensor module consists of a potentiometer, LN393 comparator, LEDs, capacitors and resistors. The pinout image above shows the components of the control module. The rain board module consists of copper tracks, which act as a variable resistor. Its resistance varies with respect to the wetness on the rain board. The below fig shows the rain board module.



Rain Board Module

The circuit diagram of a raindrop sensor module is given below.



Raindrop Sensor Module Circuit Diagram

As shown in the above figure, the R1 resistor and the rain board module will act as a voltage divider. Capacitors C1 and C2 are used as a biasing element. The input for the Non-inverting terminal is taken from the connection point of the R1, and rain board module. Another point is taken from this connection and connected to the A0 terminal of the control module.

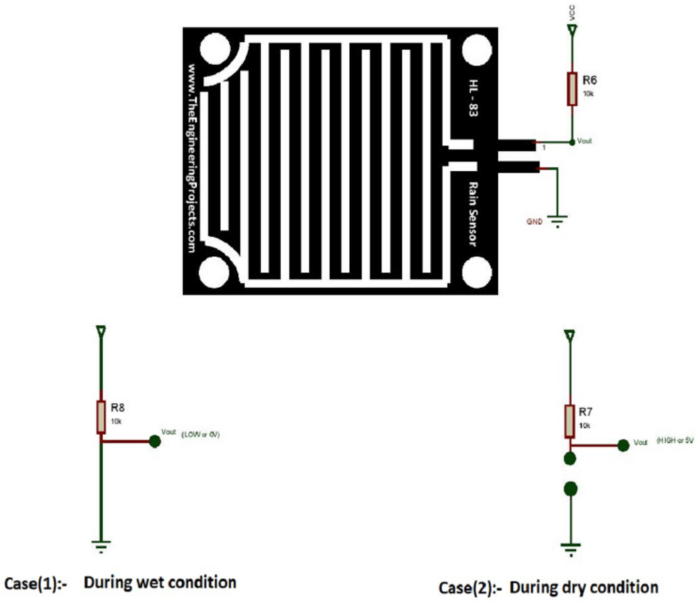
The input to the inverting terminal of the LM393 is taken from the potentiometer (R2). The R2 resistor acts as a voltage divider, and by varying R2 we can vary the input voltage to the inverting terminal, which in turn affects the sensitivity of the control module. The connections are shown in the above fig. The resistors R3 and R4 will act as current limiting resistors, while resistor R5 will act as a pull-up resistor to keep the bus in a high state when not in use.

Working of Rain Sensor:

Case1: When the input of the inverting terminal is higher than the input of the non-inverting terminal.

Case2: If the input of the inverting terminal is lower than the input of the non-inverting terminal.

The input to the inverting terminal is set to a certain value by varying the potentiometer and the sensitivity is set. When the rain board module's surface is exposed to rainwater, the surface of the rain board module will be wet, and it offers minimum resistance to the supply voltage. Due to this, the minimum voltage will be appearing at the non-inverting terminal of LM393 Op-Amp. The comparator compares both inverting and non-inverting terminal voltages. If the condition falls under case (1), the output of the Op-Amp will be digital LOW. If the condition falls under case (2), the output of the Op-Amp will be digital HIGH. The below diagram shows the equivalent circuit of both the conditions.



Raindrop Sensor Working

When the A0 pin is connected to the microcontroller, an additional analog to digital converter (ADC) circuit is used. In the case of Arduino, it consists of 6 ADC pins, which can be directly used for calculation purposes.

Applications of Rain sensor:

- Automatic windshield wipers
- Smart Agriculture
- Home-Automation

4.2.4 DHT11 SENSOR:

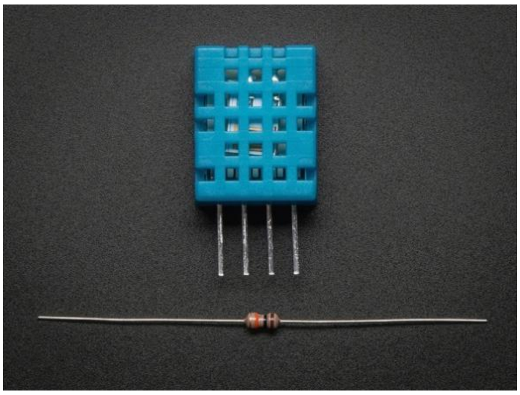
The DHT11 is a basic, ultra-low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It’s fairly simple to use, but requires careful timing to grab data. You can get new data from it once every 2 seconds, so when using the library from Adafruit, sensor readings can be up to 2 seconds old.

Comes with a 4.7K or 10K resistor, which you will want to use as a pullup from the data pin to VCC. Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor's internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up-to-20-meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect and special packages can be provided according to users' request

.

SPECIFICATIONS:

- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50 °C temperature readings +-2 °C accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- 4 pins with 0.1" spacing
- Adafruit Learning Documentation for DHTxx Sensors
- RoHS compliant



PIN IDENTIFICATION AND CONFIGURATION:

No:	Pin Name	Description
For DHT11 Sensor		
1	VCC	Power supply 3.5V to 5.5V
2	Data	Outputs both Temperature and Humidity through serial Data
3	NC	No Connection and hence not used
4	Ground	Connected to the ground of the circuit
For DHT11 Sensor module		
1	VCC	Power supply 3.5V to 5.5V
2	Data	Outputs both Temperature and Humidity through serial Data
3	Ground	Connected to the ground of the circuit

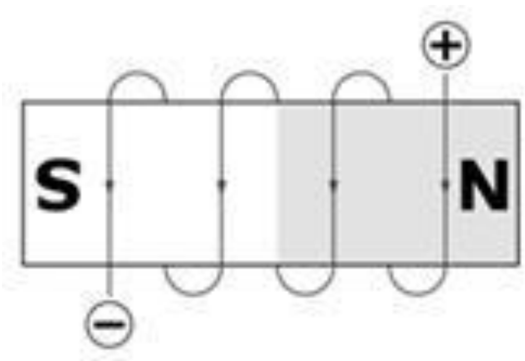
4.2.5 DC MOTOR:

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

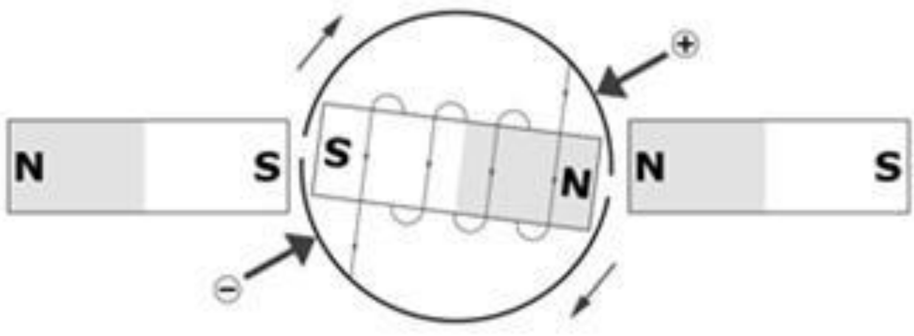
DC motors were the first type widely used, since they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC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

WORKING:

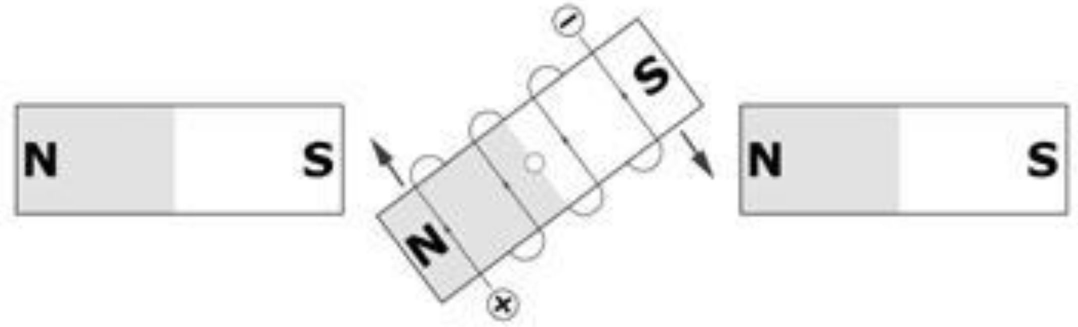
The DC motor is a machine that transforms electric energy into mechanical energy in form of rotation. Its movement is produced by the physical behavior of electromagnetism. DC motors have inductors inside, which produce the magnetic field used to generate movement. But how does this magnetic field changes if DC current is being used



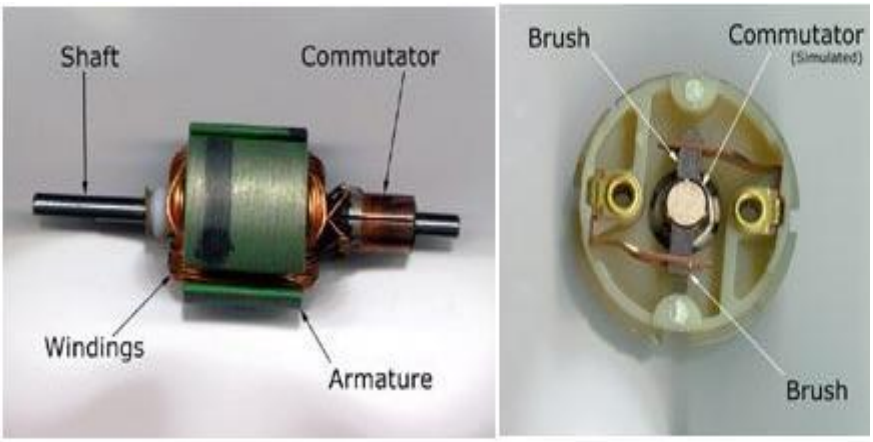
An electromagnet, which is a piece of iron wrapped with a wire coil that has voltage applied in its terminals. If two fixed magnets are added in both sides of this electromagnet, the repulsive and attractive forces will produce a torque.



Then, there are two problems to solve: feeding the current to the rotating electromagnet without the wires getting twisted, and changing the direction of the current at the appropriate time. Both of these problems are solved using two devices: a split-ring commutator, and a pair of brushes.



As it can be seen, the commutator has two segments which are connected to each terminal of the electromagnet, besides the two arrows are the brushes which apply electric current to the rotary electromagnet. In real DC motors it can be found three slots instead of two and two brushes.

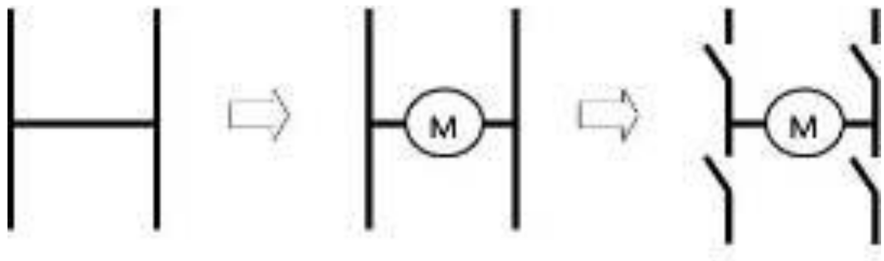


This way, as the electromagnet is moving its polarity is changing and the shaft may keep rotating. Even if it is simple and sounds that it will work great there are some issues which make these motors energy inefficient and mechanically unstable, the principal problem is due to the timing between each polarity inversion.

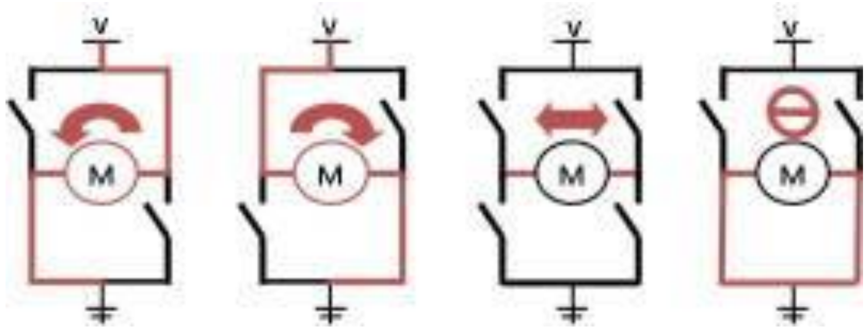
Since polarity in the electromagnet is changed mechanically, at some velocities polarity is changing too soon, which result in reverse impulses and sometimes in changing too late, generating instantaneous “stops” in rotation. Whatever the case, these issues produce current peaks and mechanical instability.

CONTROL OF DC MOTOR:

DC motors have only two terminals. If you apply a voltage to these terminals the motor will run, if you invert the terminals position the motor will change its direction. If the motor is running and you suddenly disconnect both terminals the motor will keep rotating but slowing down until stopping. Finally if the motor is running and you suddenly short-circuit both terminals the motor will stop. So there is not a third wire to control a DC motor, but knowing the previous behaviors it can be designed a way to control it, and the solution is an H-bridge.



Look at the last evolution of the DC Motor above, you can observe that there are four gates and a motor connected between them. This is the simplest H-bridge, where the four gates represent for transistors. By manipulating these gates and connecting the upper and lower terminals to a voltage supply, you can control the motor in all the behaviors as below.



Main Types of DC Motors

Understanding the different types of DC motors will also help you understand how they're used for different applications, and which type may apply to your application.

There are 4 main types of DC motors:

1. Permanent Magnet DC Motors

The permanent magnet motor uses a permanent magnet to create field flux. This type of DC motor provides great starting torque and has good speed regulation, but torque is limited so they are typically found on low horsepower applications.

2. Series DC Motors

In a series DC motor, the field is wound with a few turns of a large wire carrying the full armature current. Typically, series DC motors create a large amount of starting torque, but cannot regulate speed and can even be damaged by running with no load. These limitations mean that they are not a good option for variable speed drive applications.

3. Shunt DC Motors

In shunt DC motors the field is connected in parallel (shunt) with the armature windings. These motors offer great speed regulation due to the fact that the shunt field can be excited separately from the armature windings, which also offers simplified reversing controls.

4. Compound DC Motors

Compound DC motors, like shunt DC motors, have a separately excited shunt field. Compound DC motors have good starting torque but may experience control problems in variable speed drive applications.

ESP8266 MODULE:

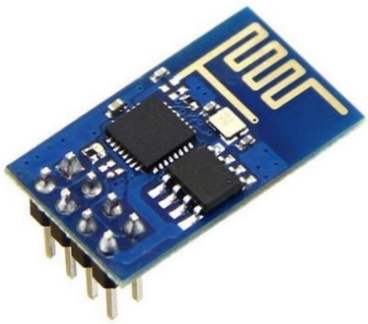
The **ESP8266** is a low-cost Wi-Fi microchip with full TCP/IP stack and microcontroller capability produced by manufacturer Espressif Systems in Shanghai, China.The chip first came to the attention of western makers in August 2014 with the **ESP-01** module, made by a third-party manufacturer Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at first there was almost no English-language documentation on the chip and the commands it accepted.[2] The very low price and the fact that there were very few external components on the module, which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, chip, and the software on it, as well as to translate the Chinese documentation

FEATURE:

- Memory:
  - 32 KiB instruction RAM
  - 32 KiB instruction cache RAM
  - 80 KiB user-data RAM
  - 16 KiB ETS system-data RAM
- External QSPI flash: up to 16 MiB is supported (512 KiB to 4 MiB typically included)
- IEEE 802.11 b/g/n Wi-Fi
  - Integrated TR switch, balun, LNA, power amplifier and matching network
  - WEP or WPA/WPA2 authentication, or open networks

- 16 GPIO pins
- SPI
- I²C (software implementation)[6]
- I²S interfaces with DMA (sharing pins with GPIO)
- UART on dedicated pins, plus a transmit-only UART can be enabled on GPIO2
- 10-bit ADC (successive approximation ADC)

ESP8266 is Wi-Fi enabled system on chip (SoC) module developed by Espressif system. It is mostly used for development of IoT (Internet of Things) embedded applications.



4.2.6 ESP8266-01 WIFI Module

ESP8266 comes with capabilities of

- 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2),
- general-purpose input/output (16 GPIO),
- Inter-Integrated Circuit (I²C) serial communication protocol,
- analog-to-digital conversion (10-bit ADC)
- Serial Peripheral Interface (SPI) serial communication protocol,
- I²S (Inter-IC Sound) interfaces with DMA (Direct Memory Access) (sharing pins with GPIO),

- UART (on dedicated pins, plus a transmit-only UART can be enabled on GPIO2), and
- pulse-width modulation (PWM).

It employs a 32-bit RISC CPU based on the Tensilica Xtensa L106 running at 80 MHz (or overclocked to 160 MHz). It has a 64 KB boot ROM, 64 KB instruction RAM and 96 KB data RAM. External flash memory can be accessed through SPI.

ESP8266 module is low-cost standalone wireless transceiver that can be used for end-point IoT developments.

To communicate with the ESP8266 module, microcontroller needs to use set of AT commands. Microcontroller communicates with ESP8266-01 module using UART having specified Baud rate.

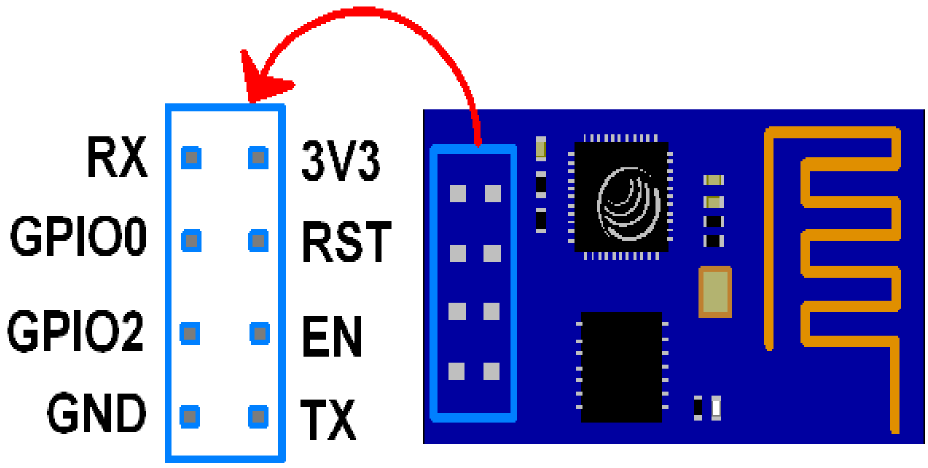
There are many third-party manufacturers that produce different modules based on this chip. So, the module comes with different pin availability options like,

- ESP-01 comes with 8 pins (2 GPIO pins) – PCB trace antenna. (shown in above figure)
- ESP-02 comes with 8 pins, (3 GPIO pins) – U-FL antenna connector.
- ESP-03 comes with 14 pins, (7 GPIO pins) – Ceramic antenna.
- ESP-04 comes with 14 pins, (7 GPIO pins) – No ant.

etc.

For example, below figure shows ESP-01 module pins

ESP8266-01 Module Pin Description



ESP8266-01 Module Pins

3V3: - 3.3 V Power Pin.

GND: - Ground Pin.

RST: - Active Low Reset Pin.

EN: - Active High Enable Pin.

TX: - Serial Transmit Pin of UART.

RX: - Serial Receive Pin of UART.

GPIO0 & GPIO2: - General Purpose I/O Pins. These pins decide what mode (boot or normal) the module starts up in. It also decides whether the TX/RX pins are used for Programming the module or for serial I/O purpose.

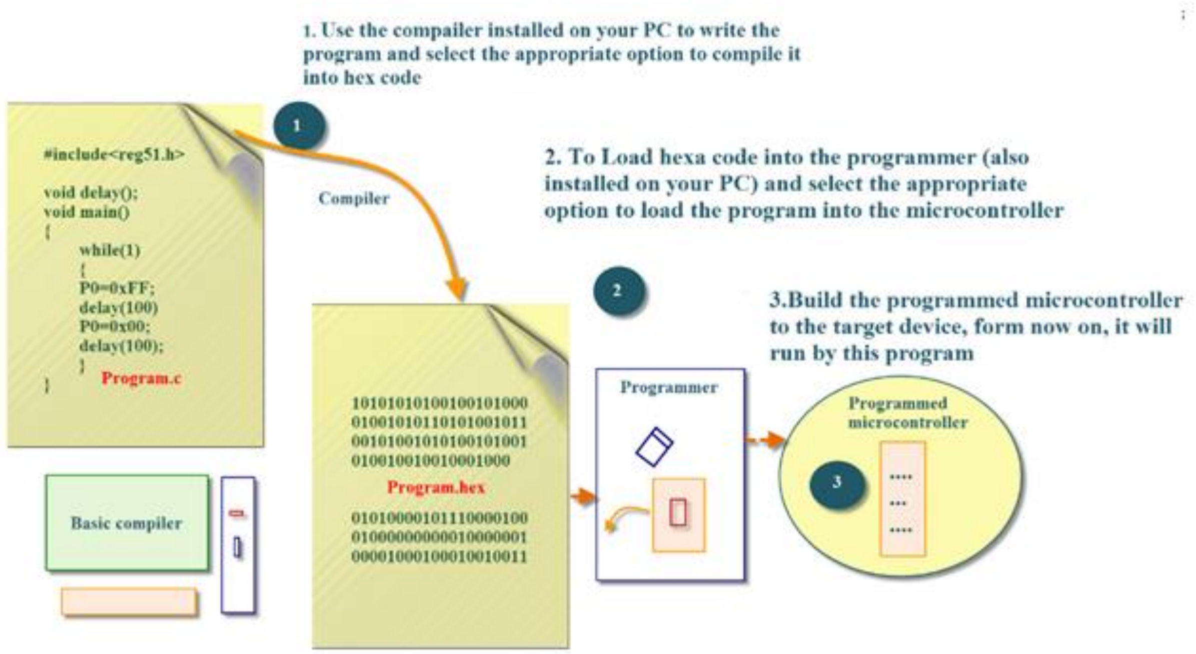
To program the module using UART, Connect GPIO0 to ground and GPIO2 to VCC or leave it open. To use UART for normal Serial I/O leave both the pins open (neither VCC nor Ground).

4.3 SOFTWARE DESCRIPTIONS

4.3.1 - EMBEDDED C:

Embedded C is most popular programming language in software field for developing electronic gadgets. Each processor used in electronic system is associated with embedded software.

Embedded C programming plays a key role in performing specific function by the processor. In day-to-day life we used many electronic devices such as mobile phone, washing machine, digital camera, etc. These all-device working is based on microcontroller that are programmed by embedded C.



The Embedded C code written in above block diagram is used for blinking the LED connected with Port0 of microcontroller.

In embedded system programming C code is preferred over other language. Due to the following reasons:

- Easy to understand
- High Reliability
- Portability
- Scalability

Function is a collection of statements that is used for performing a specific task and a collection of one or more functions is called a programming language.

Most consumers are familiar with application software that provide functionality on a computer. Embedded software however is often less visible, but no less complicated. Unlike application software, embedded software has fixed hardware requirements and capabilities, and addition of third-party hardware or software is strictly controlled.

Embedded software needs to include all needed device drivers at manufacturing time, and the device drivers are written for the specific hardware. The software is highly dependent on the CPU and specific chips chosen. Most embedded software engineers have at least a passing knowledge of reading schematics, and reading data sheets for components to determine usage of registers and communication system. Conversion between decimal, hexadecimal and binary is useful as well as using bit manipulation.

Web applications are rarely used, although XML files and other output may be passed to a computer for display. File systems with folders are typically absent as are SQL databases.

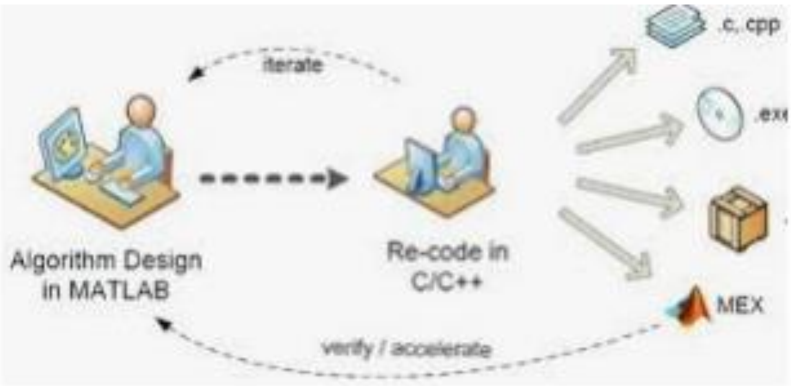
Software development requires use of a cross compiler, which runs on a computer but produces executable code for the target device. Debugging requires use of an in-circuit emulator, JTAG or SWD. Software developers often have access to the complete kernel (OS) source code.

Size of the storage memory and RAM can vary significantly. Some systems run in 16 KB of Flash and 4 KB of RAM with a CPU operating at 8 MHz, other systems can rival contemporary computers. These space requirements lead to more work being done in C or embedded C++, instead of C++. Interpreted languages like BASIC (while e.g. Parallax Propeller can use compiled BASIC) and Java (Java ME Embedded 8.3[9] is available for e.g. ARM Cortex-M4, Cortex-M7 microcontrollers and older ARM11used in Raspberry Pi and Intel Galileo Gen. 2) are not commonly used; while an implementation of the interpreted Python 3 language – MicroPython – is however available expressly for microcontroller use, e.g. 32-bit ARM-based (such as BBC micro:bit) and 16-bit PIC microcontrollers.

Communications between processors and between one processor and other components are essential. Besides direct memory addressing, common protocols include I²C, SPI, serial ports, and USB.
Communications protocols designed for use in embedded systems are available as closed source from companies including InterNiche Technologies and CMX Systems. Open-source protocols stem from uIP, lwip, and others

- This program explains how to use structure within structure in C using normal variable. “student_college_detail’ structure is declared inside “student_detail” structure in this program. Both structure variables Please note that members of “student_college_detail” structure are accessed by 2 dot(.) operator and members of “student are normal structure variables.

- _detail” structure are accessed by single dot(.) operator.



Embedded C Program with Examples ...

A Keyword is a special word with a special meaning to the compiler (a C Compiler for example, is a software that is used to convert program written in C to Machine Code). For example, if we take the Keil's Cx51 Compiler (a popular C Compiler for 8051 based Microcontrollers) the following are some of the keywords:

- bit
- sbit
- sfr
- small
- large

These are few of the many keywords associated with the Cx51 C Compiler along with the standard C Keywords.

4.3.2 ARDUINO IDE

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development

Environment) that runs on your computer, used to write and upload computer code to the physical board.

The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board – you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package.

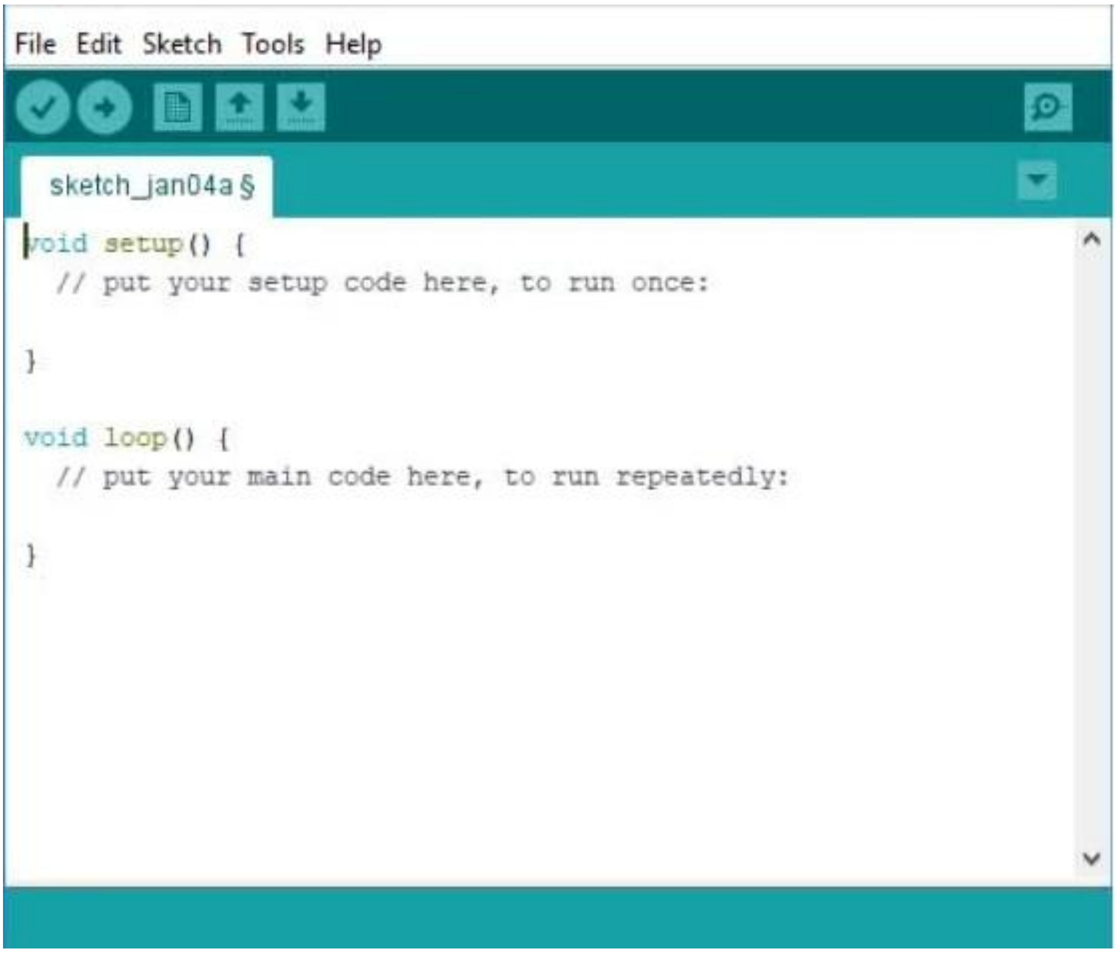
The Arduino hardware and software was designed for artists, designers, hobbyists, hackers, newbies, and anyone interested in creating interactive objects or environments. Arduino can interact with buttons, LEDs, motors, speakers, GPS units, cameras, the internet, and even your smart-phone or your TV! This flexibility combined with the fact that the Arduino software is free, the hardware boards are pretty cheap, and both the software and hardware are easy to learn has led to a large community of users who have contributed code and released instructions for a **huge** variety of Arduino-based projects

There are many varieties of Arduino boards (explained on the next page) that can be used for different purposes. Some boards look a bit different from the one below, but most Arduinos have the majority of these components in common:

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

The Arduino IDE is incredibly minimalistic, yet it provides a near-complete environment for most Arduino-based projects. The top menu bar has the standard options, including “File” (new, load save, etc.), “Edit” (font, copy, paste, etc.), “Sketch” (for compiling and programming), “Tools” (useful options for testing projects), and “Help”. The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that were found in the program, and various other useful messages.

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an “Arduino language.” However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

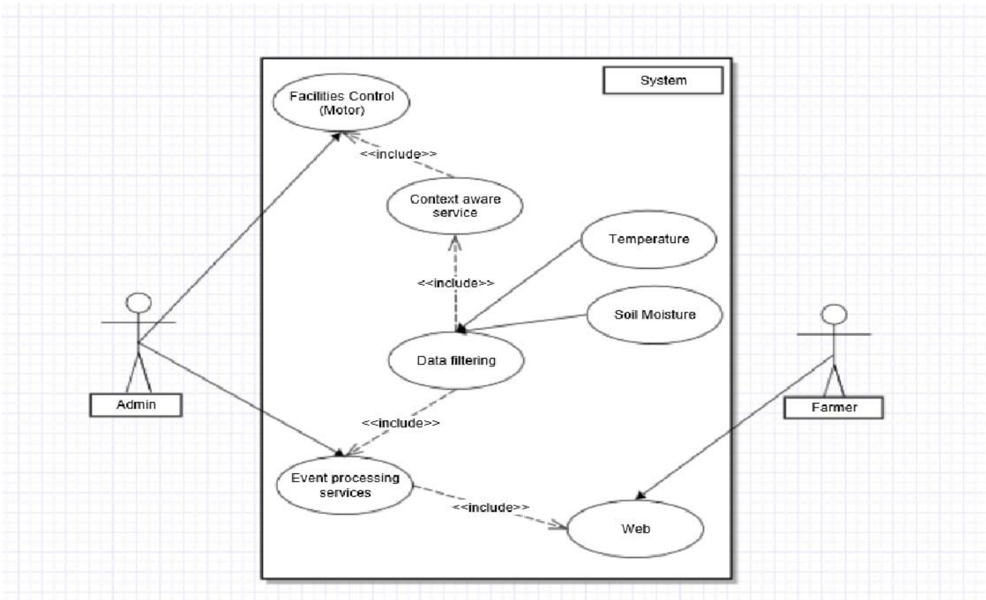


The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

4.4 UML DIAGRAMS

4.4.1- USE CASE DIAGRAM

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.



4.5 – CONCLUSION

The overview of the system design is included in the chapter which covers the architecture and uml diagrams for the system design that is to be developed To a farming system.

CHAPTER-5

IMPLEMENTATION AND RESULT

5.1 GENERAL

Implementation is the important part in achieving a successful system and ensuring confidence to the user that the system is working effectively. Implementation is the stage of the project when the theoretical design is turned out to a working system. This chapter explain about the working and the results of the work carried out in the project. The operations of our project is displayed along with the details and their screenshot.

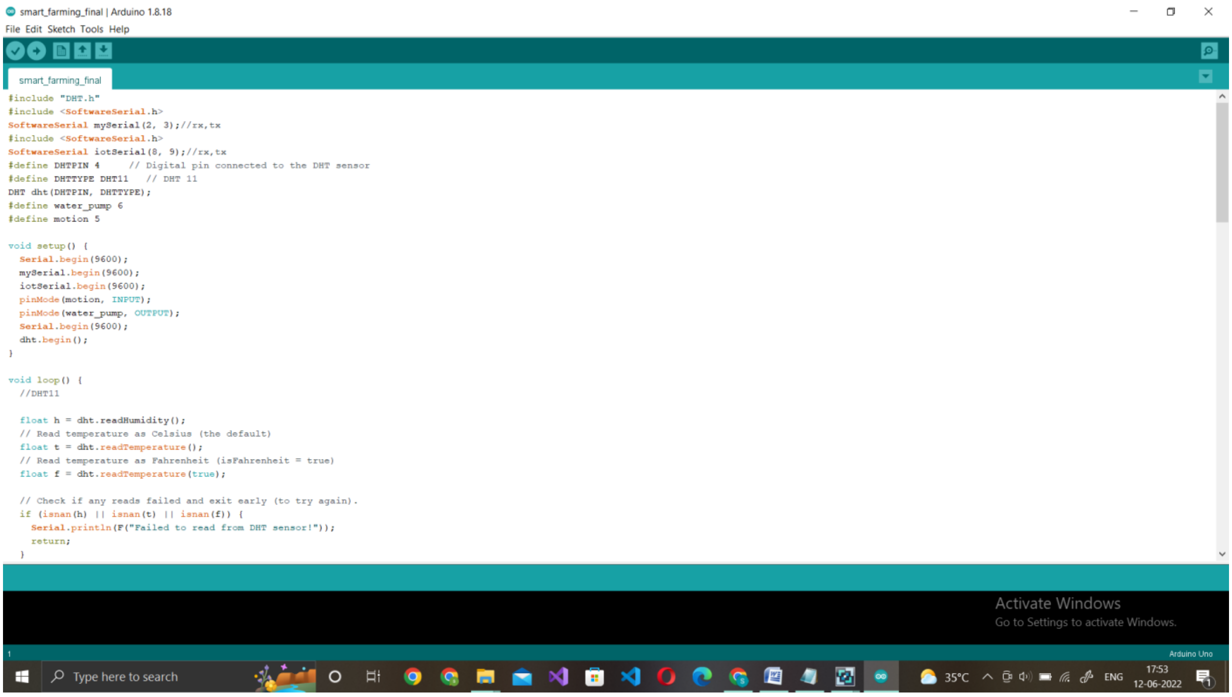
5.2-IMPLEMENTATION

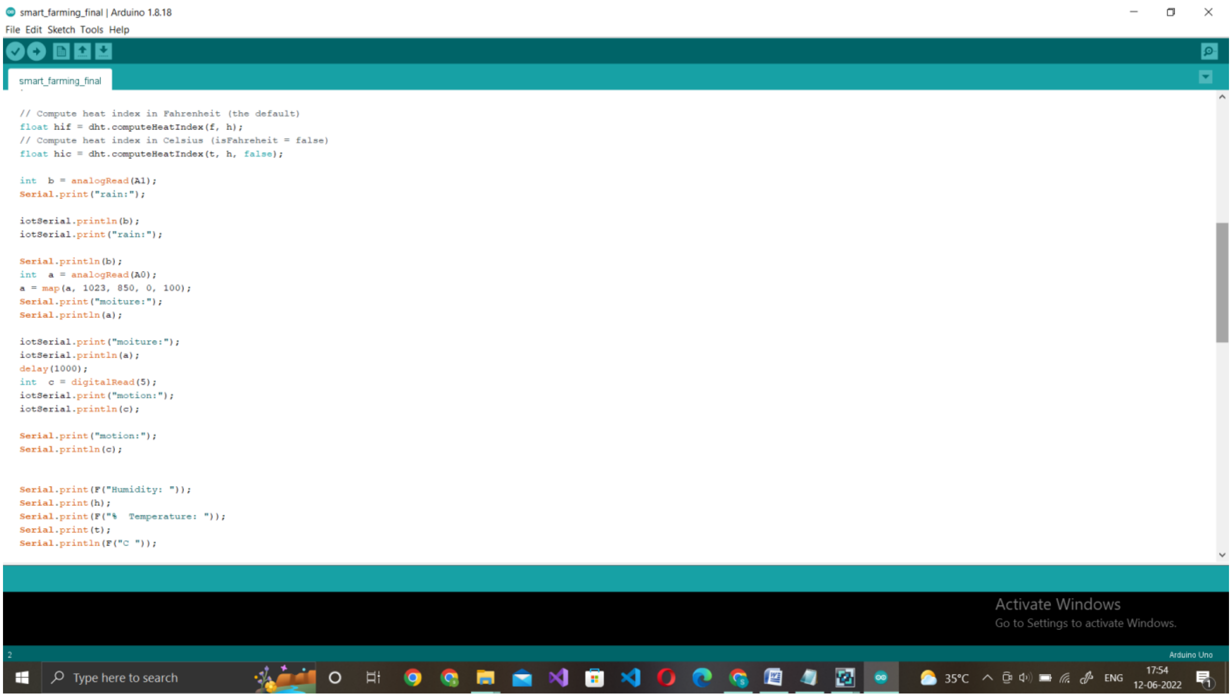
5.2.1-DOWNLOAD ARDUINO IDE

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.

5.2.2- READINGS CODE AND OUTPUT.

This picture shows the code for the readings of soil moisture sensor,rain,humidiy and also the detection of the motion,and according to the moisture value it displays whether the motor is ON or OFF.

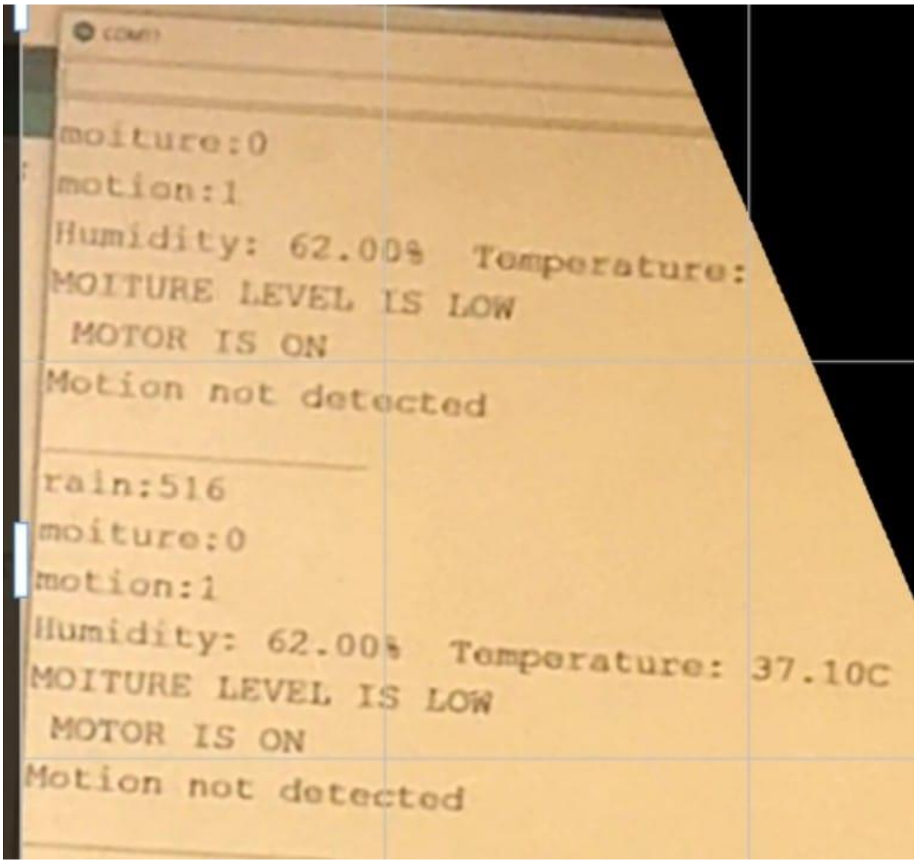




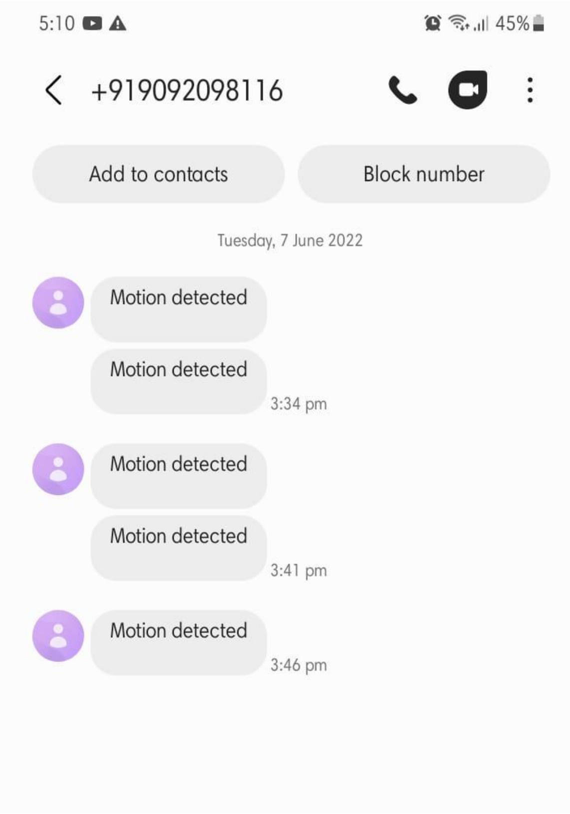
```
iotSerial.print(F("Humidity: "));
iotSerial.print(h);
iotSerial.print(F("%  Temperature: "));
iotSerial.print(t);
iotSerial.println(F("C "));

if ((a < 30) && (b > 500))
{
  Serial.println("MOITURE LEVEL IS LOW");
  Serial.println(" MOTOR IS ON");

  iotSerial.println("MOITURE LEVEL IS LOW");
  iotSerial.println(" MOTOR IS ON");

  digitalWrite(water_pump, HIGH);


}
else if ((a < 30) && (b < 500))
{
  Serial.println("MOITURE LEVEL IS LOW");
  Serial.println("RAINING SO MOTOR IS OFF");

  iotSerial.println("MOITURE LEVEL IS LOW");
  iotSerial.println("RAINING SO MOTOR IS OFF");
  digitalWrite(water_pump, LOW);
}
else
{
  Serial.println("MOITURE LEVEL IS OKAY");
  Serial.println(" MOTOR IS OFF");

  iotSerial.println("MOITURE LEVEL IS OKAY");
  iotSerial.println(" MOTOR IS OFF");
```

```
  digitalWrite(water_pump, LOW);
}
else
{
  Serial.println("MOITURE LEVEL IS OKAY");
  Serial.println(" MOTOR IS OFF");

  iotSerial.println("MOITURE LEVEL IS OKAY");
  iotSerial.println(" MOTOR IS OFF");
  digitalWrite(water_pump, LOW);
}
if (c == 0)
{
  Serial.println("Motion detected");
  iotSerial.println("Motion detected");
  mySerial.println("AT+CMGF=1");    //Sets the GSM Module in Text Mode
  delay(1000);
  mySerial.println("AT+CMGS=\"+917358474992\"\r"); // Replace x with mobile number
  delay(1000);
  mySerial.println("Motion detected");// The SMS text you want to send
  delay(1000);
  mySerial.println((char)26);// ASCII code of CTRL+Z for saying the end of sms to  the module
  delay(1000);
  Serial.println("done");
  delay(1000);
}
else
{
  Serial.println("Motion not detected");
  iotSerial.println("Motion not detected");
}

Serial.println("______________");
delay(1000);
}
```

Output of the code

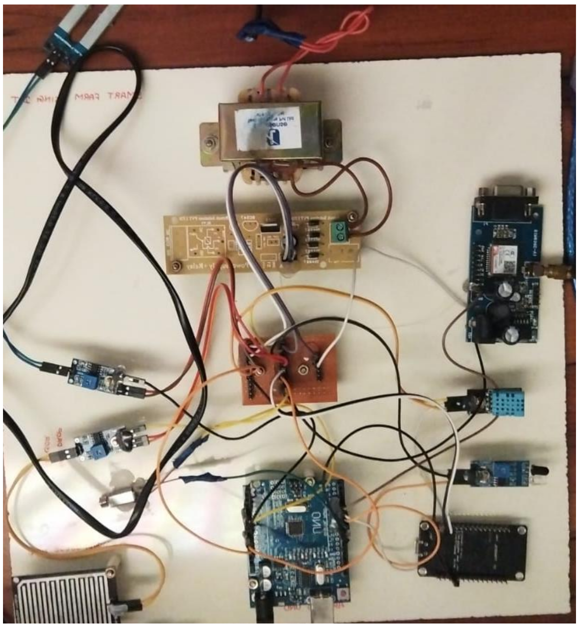


5.2.3-MOTION DETECTION



5.2.4-THE CIRCUIT BOARD

This shows the entire implementation of connected all the sensors .



5.3- SUMMARY

In this chapter, the step by step implementation of the project is illustrated.This section summarizes the internal working of each module.This shows how we can use the smart farming System.The screenshots shown is the intermediate result generated by each module.This chapter shows how each module interacts with the readings produced by them.

CHAPTER-6
CONCLUSION AND FUTURE SCOPE.

6.1-CONCLUSION

We can conclude with the fact that IoT applications are making it possible for farmers to collect meaningful data that is utilized to increase efficiency. Large landowners and small farmers must understand the potential of IoT-based smart farming and they must implement IoT solutions in a prosperous manner.

6.2-FUTURE SCOPE

For future developments it can be enhanced by developing this system for large acres of land. Also, the system can be integrated to check the quality of the soil and the growth of crop in each soil. The sensors and microcontroller are successfully interfaced and wireless communication is achieved between various nodes. All observations and experimental tests prove that this project is a complete solution to field activities and irrigation problems. Implementation of such a system in the field can definitely help to improve the yield of the crops and overall production.

6.3-SUMMARY

Smart Farming is focused on the use of data acquired through various sources (historical, geographical, and instrumental) in the management of farm activities. Technologically advanced doesn’t essentially mean that it is a smart system. Smart agriculture technologies differentiate themselves through their ability to record the data and make sense of it. Smart farming employs hardware (IoT) and software (Software as a Service or SaaS) to capture the data and give actionable insights to manage all the operations on the farm, both pre-and post-harvest. The data is organized, accessible all the time, and full of data on every aspect of finance and field operations that can be monitored from anywhere in the world.

REFERENCE

[1] Lopez Research LLC, "An Introduction to the Internet of Things. Part 1. of The IoT Series", 2003.

[2] E.Sowmiya, S.Sivaranjani, "Smart System Monitoring On Soil Using Internet of Things (IoT)". International Research Journal of Engineering and Technology (IRJET), 4,2, 1070, 2017.

[3] K. SravanthGoud and Abraham Sudharson, "Internet Based Smart Poultry Farm", Indian Journal of Science and Technology", Vol (19), pl101 2015.

[4] R. Brian "Farms of the Future: The Rise of IoT in Agriculture". Retrieved from https://www.link-labs.com/blog/rise-of-iot-inagriculture, 2016.

[5] N.Suma, S. R. Samson, S.Saranya, G.Shanmugapriya, R.Subhashri, "IoT Based Smart Agriculture Monitoring System". International Journal on Recent and Innovation Trends in Computing and Communication, 5, 2, 177-181, 2017.

[6] Xiaojing, Z., Yuangua, L., Zigbee implementation in intelligent agriculture based on internet of things, College of Electronics Information and engineering, Qiongzhou University, China, 2012

[7] Adinarayana, J., Huh , E., Wan, Y and Kiura, T., Becoming technological advanced -IOT applications in smart agriculture

[8] Agarwal, R. and Karahanna, E. Time flies when you're having fun:Cognitive absorption and beliefs about information technology usage, MIS Quarterly, vol. 24, no. 4, pp. 665-694, 2000.

[9] AgriSETA, Sector Analysis agriculture, Department of Higher Education and Training, South Africa, 2010

[10] S. R. Nandurkar, V. R. Thool, R. C. Thool, "Design and Development of Precision Agriculture System Using Wireless Sensor Network", IEEE

International Conference on Automation, Control, Energy and Systems (ACES), 2014.

[11] JoaquínGutiérrez, Juan Francisco Villa-Medina, Alejandra Nieto-Garibay, and Miguel Ángel PortaGándara, "Automated Irrigation System Using a Wireless Sensor Network and GPRS Module",IEEE TRANSACTIONS INSTRUMENTATION AND MEASUREMENT, 0018-9456,2013

[12] Dr. V .Vidya Devi,G. Meena Kumari, "Real- Time Automation and Monitoring System for Modernized Agriculture" ,International Journal of Review and Research in Applied Sciences and Engineering (IJRRASE) Vol3 No.1. PP 7-12, 2013

[13] Y. Kim, R. Evans and W. Iversen, "Remote Sensing and Control of an Irrigation System Using a Distributed Wireless Sensor Network", IEEE Transactions on Instrumentation and Measurement, pp. 1379–1387, 2008.

[14] Hemlata Sahu, Shalini Sharma, Seema Gondhalkar, "A Brief Overview on Data Mining Survey", International Journal of Computer Technology and Electronics Engineering (IJCTEE), Volume 1, Issue 3.

[15] D Ramesh, B. Vishnu Vardhan, "Data Mining Techniques and Applications to Agricultural Yield Data", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 9, September 2013.

[16] Meghali A. Kalyankar, S.J.Alaspurkar, "Data Mining Technique to Analyze the Metrological Data", Research Paper in International Journal of Advanced Research in Computer Science and Software Engineering.

[17] Sumitha Thankachan, S. Kirubakaran, "E-Agriculture Information Management System", International Journal of Advanced Research in

Computer Science and Software Engineering, Volume 3, Issue 2, February 2013.

[18] Aqueel-ur-Rehman, Abu Zafar Abbasi, Noman Islam, Zubair Ahmed Shaikh, "A review of wireless sensor and networks applications in agriculture", Computer Standards & Interfaces 36(2014) 263- 270.

[19] Sally Jo Cunningham, Geoffrey Holmes, "Developing innovative applications in agriculture using data mining", SEARCC'99 conference proceedings.