**SURFACE RUN OFF MANAGEMENT**

**A PROJECT REPORT**

***Submitted by***

**SHIVA HARINI S (310619205094)**

**SIDDHATARTH D (310619205099)**

**SWETHA S (310619205113)**

***in partial fulfilment for the award of degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**INFORMATION TECHNOLOGY**



**EASWARI ENGINEERING COLLEGE, CHENNAI**

**(Autonomous Institution)**

***affiliated to***

**ANNA UNIVERSITY: CHENNAI – 6000 025**

**April 2023**

**EASWARI ENGINEERING COLLEGE, CHENNAI**

**(AUTONOMOUS INSTITUTION)**

**AFFILIATED TO ANNA UNIVERSITY, CHENNAI 600025**

**BONAFIDE CERTIFICATE**

Certified that this project report **SURFACE RUN OFF MANAGEMENT** is the bonafide work **of SHIVA HARINI S (310619205094) , SIDDHTARTH** **(310619205099) , SWETHA S (310619205113)** Who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

**Dr.N.Ananthi , Ph.D ., Dr.V.Balaji**

**HEAD OF THE DEPARTMENT SUPERVISOR**

Associate Professor

Department of Information Department of Information

Technology, Technology,

Easwari Engineering College, Easwari Engineering College,

Ramapuram, Chennai 600089. Ramapuram,Chennai 600089.

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**VIVA VOCE EXAMINATION**

The viva voce examination of the project work, submitted by

**Shiva Harini S, Register Number: 310619205094**

**Siddhtarth D, Register Number: 310619205099 and**

**Swetha S, Register Number: 310619205113** is held on -----------------------

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**ACKNOWLEDGEMENT**

We hereby place our deep sense of gratitude to our beloved Founder Chairman of the institution**, DR.T.R.PAARIVENDHAR,** for providing us with the requisite infrastructure throughout the course. We would also like to express our gratitude towards our Chairman **DR.R.SHIVAKUMAR,** for giving the necessary facilities

We convey our sincere thanks to **DR.R.S.KUMAR,** Principal Easwari Engineering College, for his encouragement and support. We extend our hearty thanks to **DR.V.ELANGO,** Vice Principal (academics) and **DR.S.NAGARAJAN,**  Vice Principal (admin), Easwari Engineering College, for their constant encouragement.

We take the privilege to extend our hearty thanks to **DR.N.ANANTHI,** Head of the Department, Information Technology, Easwari Engineering College for her suggestions, support and encouragement towards the completion of the project with perfection.

We would like to express our gratitude to our Project Coordinator, **MRS.P.M LAVANYA** Assistant Professor, Department of Information Technology, Easwari Engineering College, for her constant support and encouragement.

We would also like to express our gratitude to our guide **DR.V.BALAJI,** Associate Professor, Department of Information Technology, Easwari Engineering College, for her constant support and encouragement.

Finally, we wholeheartedly thank all the faculty members of the Department of Information Technology for warm cooperation and encouragement.

**ABSTRACT**

We employ multi-parameter sensor-based automated farming since the existing method of manually monitoring agricultural fields has various limitations. IOT technology was used to develop the Smart Agriculture Monitoring System. This system is powered by an Arduino Uno, a temperature humidity sensor, a soil moisture sensor, a water level sensor, a barometric pressure sensor, water pumps, and DC motors. The sensors measure the field's water level and soil moisture level when the smart agricultural monitoring system is engaged. If the water level in the field falls below the level specified for a certain crop, the system will automatically engage the water pump. The LCD display module will display IOT alerts such as current water level, soil moisture level, and motor starting. We may also manually access the pump via a website. Farmers will also be able to get this information by cell phone. The user can abruptly stop the flow of water in the field by pressing the system's assigned button. This strategy, when applied, will assist farmers in maintaining correct soil moisture and water levels, hence increasing crop yield with minimum effort.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT**  **LIST OF FIGURES**  **LIST OF TABLES**  **LIST OF ABBREVIATIONS** | 5  8  10  11 |
| 1. | **INTRODUCTION**   * 1. GENERAL   2. OBJECTIVE   1.3 ORGANIZATION REPORT  1.4 SUMMARY | 12  12  19  20  20 |
| 2. | **LITERATURE SURVEY**  2.1 INTRODUCTION  2.2 RELATED WORKS  2.3 SUMMARY | 21  21  21  25 |
| 3. | **SYSTEM DESIGN**  3.1 INTRODUCTION  3.2 PROPOSED SYSTEM  3.3 SYSTEM ARCHITECTURE  3.4 SUMMARY | 26  26  26  27  28 |
| 4. | **SYSTEM IMPLEMENTATION**  4.1 INTRODUCTION  4.2 METHODOLOGIES  4.2.1 MODULES  4.3 HARDWARE AND SOFTWARE SPECIFICATIONS  4.3.1 Hardware Requirements  4.3.2 Software Requirements  4.4 IMPLEMENTATION  4.5 SUMMARY | 29  29  29  30  33  33  55  64  65 |
| 5. | **PERFORMANCE EVALUATION**  5.1 INTRODUCTION  5.2 COMPARISON  5.3 SUMMARY | 66  66  66  67 |
| 6. | **CONCLUSION AND FUTURE WORK**  6.1 CONCLUSION  6.2 FUTURE ENHANCEMENT | 68  68  68 |

**APPENDIX** 69

**REFERENCES** 72

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **TITLE** | **PAGE NO** |
| 3.1 | SYSTEM FLOW DIAGRAM | 27 |
| 3.2 | ARCHITECTURE DIAGRAM | 28 |
| 3.3 | BLOCK DIAGRAM | 28 |
| 4.1 | COLLECTION OF DATA FROM THE FIELD | 30 |
| 4.2 | MAINTENANCE OF CROPS USING THE SENSORS | 31 |
| 4.3 | MONITORING OF THE AGRICULTURAL FIELD | 32 |
| 4.4 | WATER MAINTAINING | 32 |
| 4.5 | ARDUINO MEGA 2560 | 35 |
| 4.6 | POWER SUPPLY | 40 |
| 4.7 | DHT11 SENSOR | 42 |
| 4.8 | FC-28 SOIL MOISTURE SENSOR | 43 |
| 4.9 | ARD2-3025 WATER LEVEL SENSOR | 48 |
| 4.10 | H BRIDGE MOTOR CONTROL CIRCUIT | 52 |
| 4.11 | PUMP MOTOR | 53 |
| 4.12 | ESP-12E BASED NODEMCU | 54 |
| 4.13 | EMBEDDED C PROGRAMMING DEVELOPMENT | 55 |
| 4.14 | EMBEDDED C PROGRAMMING STEPS | 56 |
| 4.15 | RTOS-REAL TIME OPERATING SYSTEM | 57 |
| 4.16 | STRUCTURE OF AN EMBEDDED SYSTEM | 58 |
| 4.17 | FLOW CHART | 59 |
| 4.18 | ARDUINO IDE SOFTWARE | 64 |
| 4.19 | SNAPSHOT | 64 |
| 4.20 | ADAFRUIT | 65 |
| 5.1 | EXPENDITURE VS TIME GRAPH | 66 |
| 5.2 | EXPENDITURE VS LABOUR GRAPH | 67 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **TABLE NO.** | **TITLE** | **PAGE NO.** |
| 1 | TECHNICAL SPECIFICATIONS OF ARDUINO MEGA 2560 | 36 |
| 2 | TECHNICAL SPECIFICATIONS OF DHT11 SENSOR | 41 |
| 3 | TECHNICAL SPECIFICATIONS OF WATER LEVEL SENSOR | 50 |

**LIST OF ABBREVIATIONS**

IOT Internet of Things

WSN Wireless Sensor Networks

M2M Machine to Machine

UV Ultraviolet

LCD Liquid Crystal Display

MCU Microcontroller Unit

SFT Smart Farming Technology

TAM Technology Acceptance Model

PU Perceived Usefulness

PEU Perceived Ease of Use

RTOS Real-Time Operating System

SNR Signal-Noise Ratio

**CHAPTER 1**

**INTRODUCTION**

* 1. **GENERAL**

Several studies examine the process of agricultural digitization, suggesting technology capable of meeting today's agri-food concerns. Existing bibliometric studies, however, do not completely use the complementarity of many new bibliometric approaches, such as performance analysis and scientific mapping, and do not take into account the entire research field time span.

As a result, the goal of this study is to supplement and update past works while also providing a comprehensive quantitative and qualitative picture of Agriculture 4.0 research through the use of synergistic performance analysis and scientific mapping. The study was carried out on a sample of 2334 publications using statistical frequency analysis and the VOS viewer. Key findings from performance analysis included the following indicators: research subject areas, publication trends, most productive journals, document types, author productivity, author and index keywords, most cited papers, most productive and influential institutions, country map collaboration, and documents by funding sponsor. Science mapping revealed key findings about the main thematic research field's clusters and their evolution over time: technology application in agricultural industry, data model for analysis and prediction, smart agriculture experimentation and application studies, decision support systems for crop monitoring.

Internet of Things:

The internet of things (IoT) is a network of physical devices, cars, buildings, and other items that are integrated with electronics, software, sensors, actuators, and network connectivity to gather and share data. The Global Standards Initiative on Internet of Things (IoT-GSI) described the Internet of Things (IoT) in 2013 as "the infrastructure of the information society." The Internet of Things enables things to be sensed and controlled remotely through existing network infrastructure, allowing for more direct integration of the physical world into computer-based systems and resulting in enhanced efficiency, accuracy, and economic advantage. When IoT is enhanced with sensors and actuators, it becomes a subset of the broader class of cyber-physical systems, which includes technologies such as smart grids, smart homes, intelligent transportation, and smart cities. Each object is uniquely identified by its embedded computing system yet may interact with the existing Internet infrastructure. According to experts, the Internet of Things will include around 50 billion items by 2020.

INFRASTRUCTURE:

The Internet of Things will become part of the fabric of everyday life. It will become part of our overall infrastructure just like water, electricity, telephone, TV and most recently the Internet. Whereas the current Internet typically connects full-scale computers, the Internet of Things (as part of the Future Internet) will connect everyday objects with a strong integration into the physical world.

1. Plug and Play Integration:

There is a great deal of variation in IoT-related technologies accessible today. It is often used for highly particular applications, and the configuration takes extensive technical knowledge and can be time-consuming. To build a genuine Internet of Things, we must abandon such small-scale, vertical application silos in favor of a horizontal infrastructure on which several apps may operate concurrently.

1. Infrastructure Functionality

The infrastructure must help apps locate what they need. An application may run everywhere, including on the things themselves. Finding items is not limited to the start-up time of an application. Automatic adaptation is required anytime relevant new items become accessible, things become unavailable, or the status of objects changes. The infrastructure must facilitate the monitoring of such changes as well as the adaption necessary as a result of the changes.

1. Physical Location and Position

Due to the fact that the Internet of things is deeply embedded in the real world, the concept of physical location and position is critical, not just for discovering objects, but also for gaining knowledge. As a result, the infrastructure must allow locating objects based on location (e.g., geo-location-based discovery). With mobility in mind, localisation technologies will play a significant role in the Internet of Things and may become incorporated in the Internet of Things architecture.

1. Security and Privacy

An infrastructure must also enable security and privacy capabilities such as identity, confidentiality, integrity, non-repudiation authentication, and authorisation. The variety and necessity for interoperability across different ICT systems deployed in infrastructure, as well as the resource limits of IoT devices (e.g., Nano sensors), must be considered here.

Data Management

Data management is an essential component of the Internet of Things. When contemplating a world of networked items continually sharing various sorts of information, the volume of created data and the procedures involved in its treatment become essential. The advent of Machine-to-Machine (M2M) computing, which is one of the enabling technologies for the Internet of Things, represents a long-term potential for wireless communications chip producers. This technology has a wide variety of uses. While analysts agree that M2M is a potential development area, expert estimates on the scale of the opportunity differ by a factor of four. Conservative estimates predict that 80 million to 90 million M2M devices will be sold in 2014, while more optimistic estimations predict 300 million units will be sold. According to previous estimates of adoption curves for comparable disruptive technologies, such as portable MP3 players and antilock braking systems for vehicles, unit sales in M2M might increase by a factor of ten over the next five years. Many technologies and elements are involved in "data management" in the context of IoT. Some of the most important notions that help us comprehend the problems and potential of data management are as follows.

• Data Collection and Analysis

• Big Data

• Semantic Sensor Networking

• Virtual Sensors

• Complex Event Processing.

APPLICATION AREAS:

The growth of markets and applications, and therefore their economic potential and effect in solving social trends and concerns over the future decades, has evolved considerably in recent years. Health and wellness, transportation and mobility, security and safety, energy and the environment, communication and e-society are the five societal trends. These changes open up enormous prospects in the consumer electronics, automotive electronics, medical applications, communication, and other areas. More and more semiconductor technology, communications, networks, and software improvements directly help these applications.

1. Cities

Smart Parking: Monitoring of parking spaces availability in the city.

Structural health**:** Monitoring of vibrations and material conditions in buildings, bridges and historical monuments.

Noise Urban Maps: Sound monitoring in bar areas and centric zones in real

time.

Traffic Congestion: Monitoring of vehicles and pedestrian levels to optimize

driving and walking routes.

Smart Lightning: Intelligent and weather adaptive lighting in street lights.

Waste Management: Detection of rubbish levels in containers to optimize

the trash collection routes.

Intelligent Transportation Systems: Smart Roads and Intelligent Highways

with warning messages and diversions according to climate conditions and unexpected events like accidents or traffic jams.

1. Environment

Forest Fire Detection: Monitoring of combustion gases and pre-emptive fire

conditions to define alert zones.

Air Pollution: Control of CO2 emissions of factories, pollution emitted by

cars and toxic gases generated in farms.

Landslide and Avalanche Prevention: Monitoring of soil moisture,

vibrations and earth density to detect dangerous patterns in land conditions.

Earthquake Early Detection: Distributed control in specific places of

tremors.

1. Water

Water Quality: Study of water suitability in rivers and the sea for fauna and

Eligibility for drinkable use.

Water Leakages: Detection of liquid presence outside tanks and pressure variations along pipes.

River Floods: Monitoring of water level variations in rivers, dams and reservoirs.

1. Energy Smart Grid, Smart Metering

Smart Grid: Energy consumption monitoring and management.

Tank level: Monitoring of water, oil and gas levels in storage tanks and cisterns.

Photovoltaic Installations: Monitoring and optimization of performance in solar energy plants.

Water Flow: Measurement of water pressure in water transportation systems.

Silos Stock Calculation: Measurement of emptiness level and weight of the goods.

1. Security & Emergencies

Perimeter Access Control: Access control to restricted areas and detection

` of people in non-authorized areas.

Liquid Presence: Liquid detection in data centres, warehouses and sensitive

building grounds to prevent break downs and corrosion.

Radiation Levels: Distributed measurement of radiation levels in nuclear power stations surroundings to generate leakage alerts.

Explosive and Hazardous Gases: Detection of gas levels and leakages in industrial environments, surroundings of chemical factories and inside mines.

1. Industrial Control

M2M Applications: Machine auto-diagnosis and assets control.

Indoor Air Quality: Monitoring of toxic gas and oxygen levels inside chemical plants to ensure workers and goods safety.

Temperature Monitoring: Control of temperature inside industrial and medical fridges with sensitive merchandise.

Ozone Presence: Monitoring of ozone levels during the drying meat process

in food factories.

Indoor Location: Asset indoor location by using active (ZigBee, UWB) and

passive tags (RFID/NFC).

Vehicle Auto-diagnosis: Information collection from CAN Bus to send real time alarms to emergencies or provide advice to drivers.

1. Agriculture

Wine Quality Enhancing: Monitoring soil moisture and trunk diameter in vineyards to control the amount of sugar in grapes and grapevine health.

Green Houses: Control micro-climate conditions to maximize the production of fruits and vegetables and its quality.

Golf Courses: Selective irrigation in dry zones to reduce the water resources

required in the green.

Meteorological Station Network: Study of weather conditions in fields to forecast ice formation, rain, drought, snow or wind changes.

Compost: Control of humidity and temperature levels in alfalfa, hay, straw, etc. to prevent fungus and other microbial contaminants.

1. Domestic & Home Automation

Energy and Water Use: Energy and water supply consumption monitoring

to obtain advice on how to save cost and resources.

Remote Control Appliances: Switching on and off remotely appliances to avoid accidents and save energy.

Intrusion Detection Systems: Detection of window and door openings and violations to prevent intruders.

Art and Goods Preservation: Monitoring of conditions inside museums and art warehouses.

1. eHealth

Fall Detection: Assistance for elderly or disabled people living independent.

Medical Fridges: Control of conditions inside freezers storing vaccines, medicines and organic elements.

Sportsmen Care: Vital signs monitoring in high performance centres and fields.

Patients Surveillance: Monitoring of conditions of patients inside hospitals and in old people’s home.

Ultraviolet Radiation: Measurement of UV sun rays to warn people not to be exposed in certain hours.

**1.2 OBJECTIVE**

* The aim of our project is to tackle surface run-off with minimal use of resources and with the use of automation.
* The agriculture fields are being monitored manually; thus, we developed a multi-parameter sensor-based automated farming, the smart agriculture monitoring system is designed utilizing IOT technology.
* The model uses Arduino uno, temperature humidity sensor, soil moisture sensor, water level sensor, barometric pressure sensor, water pumps, and DC motors power this system.
* When the smart agriculture monitoring system is activated, the sensors detect the field's water level and soil moisture level.
* If the water level falls below the level indicated for a certain crop grown in the field, the system will automatically activate the water pump.
* The IOT notifications regarding current water level, soil moisture level, and motor starting will be presented on the LCD display module. We may also access the pump manually via a website.
* Farmers will also receive this information through cell phone. By pushing the system's given button, the user can violently halt the water flow in the field.
  1. **ORGANIZATION REPORT**

The proposed work is divided into various chapters as mentioned below.

Chapter 1 is the introduction that explains the objective and scope of the proposed system.

Chapter 2 is the literature survey that elaborates on the research works on the existing system.

Chapter 3 gives details about proposed system, design and architecture of the roles played by various modules.

Chapter 4 gives details about the hardware and software requirements and the methodologies used.

Chapter 5 provides graphical description of the performance analysis with existing versus the proposed system design

Chapter 6 discusses about the conclusion and the possible future enhancements

* 1. **SUMMARY**

This chapter provides an overview of our domain. The needed principles used in our project are examined, and the goal of our project is also highlighted.

**CHAPTER- 2**

**LITERATURE REVIEW**

**2.1 INTRODUCTION**

A literature survey is an overview of a thorough assessment of published and unpublished work from secondary sources of data that pertain to areas of particular interest to the researcher.

**2.2 RELATED WORKS**

Smart Farming Technologies (SFTs) can increase productivity while lowering costs and conserving resources; nevertheless, farmers are reluctant to use them. A framework called technology acceptance model (TAM) is used to determine the factors affecting the farmer’s intentions to adopt the two types of SFT’s and analysed. As a result, the fast rise of SFTs has stimulated study in a variety of fields. These topics include the appraisal of these technologies, the creation of acceptable applications for them, demographic trends of their use, and the economic and environmental advantages they provide. There has been little investigation into the function of sources of information regarding SFTs and their relationship with the two essential components of the TAM, the perceived ease of use (PEU) and perceived usefulness (PU). The most successful way to improve the intention to use these technologies is to engage cooperative extension service agents and vendors, the most prevalent change agents in Italian agriculture, to promote information campaigns and training activities. [Federica Caffaro et. al 2020]

Recently, hydroponics refers to the art of growing plants in water (either saline) without soil (land). Nutrients for the plants are supplied to the roots in the form of solution that can be either in the form of static or flowing. Hydroponics can be cultivated both in green house and glass house environment. The limitation in green house environment is to maintain the temperature, pressure, humidity value at a particular level. In addition to that, monitoring on PH value and electrical conductivity in hydroponics is another challenge that has to be monitored and maintained. Manual monitoring is in practice which is a very trivial task else the plants may die out. This project, focuses on two tasks, the first one is to automate the greenhouse environment monitoring. The subsequent is automation of PH level and electrical conductivity maintenance. IOT is used to transfer the retrieved data to the internet (mass storage) and mobile app is used to communicate the current status to the user through the use of internet to their mobile phones, so that monitoring & maintenance will be easier. [Dr.D.Saraswathi et. al 2018]

Agriculture being a significant part of the economy across the world is entitled for a technological advancement. With the help of internet of things (IOT) and wireless sensor networks (WSNs), new methodologies of monitoring soil health, moisture content leading to the design and development of FarmFox-an Iot device, that analyses the information obtained from the sensors to the user via the internet. This is a cost effective and an economic alternative for sustainable agriculture that is fully automated such that the farmers can incorporate different agricultural parameters to make smart decisions. [Anirbit Sengupta et. al 2020]

Agriculture being a significant part of the economy across the world is entitled for a technological advancement. With the help of internet of things (IOT) and wireless sensor networks (WSNs), new methodologies of monitoring soil health, moisture content leading to the design and development of FarmFox-an Iot device, that analyses the information obtained from the sensors to the user via the internet. This is a cost effective and an economic alternative for sustainable agriculture that is fully automated such that the farmers can incorporate different agricultural parameters to make smart decisions. [ Anneketh Vij et. al 2020]

Rising water issues and the need for better methodologies is the utmost priority for farm maintenance. A monitoring system that tackles problems such as soil erosion, over irrigation and crop specific irrigation problem is developed for an easy and efficient management of agricultural practices. A distributed wireless sensor network that is spread across the form with different sensor modules that transmits data on a common server. Based on the Internet of Things, a sustainable and a computationally profitable strategy is proposed. [P. Visconti et. al 2020]

In order to improve the quality and productivity of agricultural produce an approach that utilizes smart sensors in monitoring, visualizing generating digital data thereby adding value in soil-less farming through automation. The fundamentals of smart sensors are explored, as well as the most practical sensors for monitoring soil and plant physicochemical parameters in field cultivation procedures, greenhouses, and indoor hydroponics. Precision farming, agro machinery automation, Precision Livestock Farming (PLF), TV White Spaces (TVWS) remote connectivity, Unmanned Aerial Vehicles (UAVs) based imagery, and IoT application can assist farming communities in using resources accurately based on real-time farm data acquired and improving crop yield without wastage. Issues such as lack of arable land, skilled manpower, shortage of water changing weather conditions leading to expensive, low yield and an unprofitable agricultural practice that is tackled by the use of data driven technologies in a digital-physical farm management cycle. [ Kenny Paul et. al 2022]

With the rise in world population, shortage of food being the most crucial problem that needs to be addressed by endorsing alternatives to improve the capacity of the soil and the safety of the environmental resources. The development of cost-effective miniscule sensors and processors resulting in a practical and an attainable smart farming practice. In conclusion, IoT-based farming has yet to be properly utilized to meet the difficulties of food to a rising global population with dwindling arable lands and thus limit human involvement and labour. IoT-based farming combined with big data has the potential to provide a worldwide solution to both indoor and arable farming. IoT-based farming represents a large development opportunity for one of the most advanced technologies. It combines sensor, embedded computer, current network, wireless communication, and distributed information processing technologies to give us with a new approach to acquire massive data for in-depth analysis and automation of the complete agricultural system for improved produce quality and quantity. [ Hira Farooq et. al 2020]

An extensive and scalable network architecture using cloud computing and wireless sensors based on throughput maximization, latency minimization, high signal-to-noise ratio (SNR), minimum mean square error, and improved coverage area corroborating to better performance than conventional strategies of farming. Based on the agriculture field situation, the parameter values of sensor conditions such as temperature, humidity, and wetness may be fixed. The suggested approach will maximize resource consumption and alleviate the problem of irrigation scarcity. The graphical depiction of WSN significant performance outperforms prior technologies that could be retrieved and visually shown. In the future, there will be animal incursion. To safeguard crops and farmers against animal assaults, a detection system can be deployed. Using IoT technology, real-time field monitoring is feasible. The suggested technique rigorously supervises agricultural resource waste. [ P. Sanjeevi et. al 2020]

Sensors capture real-time atmospheric data, which is then analysed using artificial intelligence (AI) algorithms to make gadgets act more intelligently. The revolution of modern agriculture techniques using IoT & AI over the traditional agriculture methodologies are making farming a profitable venture. Several issues go unnoticed in Indian farming such as, farmers still being dependent on seasonal rainfall and do not use any technological irrigational facility due to a lack of financial assistance. Farmers are also unable to implement automated tools for farming due to the high cost of hardware. Lack of smart and technological skills is also a major constraint in modern agriculture. The combination of this technological advancement will result in decrease of the farmer's crop production stress and reliability unanticipated weather forecasting patterns, as well as a decrease in the cost of human labour [N. Aggarwal et. al 2021]

To meet the needs of information monitoring for actual large-scale agricultural farms, a low-cost, low-power, and low-data-rate solution is provided. A small-scale farm is simple to operate. A large farm, on the other hand, will necessitate the automation of equipment that adds to agricultural production. Sensor-based soil property measurement is essential in creating a fully automated agricultural farm, and it also produces better results than any human technique. Existing information monitoring systems are inefficient in terms of greater deployment costs and a restricted communication range for large-scale agriculture fields. To increase the communication range, a tree-based communication method is used, with intermediary nodes added. A solar panel, a rechargeable cell, a microprocessor, a moisture sensor, and a communication unit are all included in each sensor node. This paper provides an information monitoring technique for gathering field data across vast distances in a fully automated agricultural farm. A network mechanism for the HC12 module is built in this system to extend the communication range. An experiment on a real farm demonstrates that the technology works better on soil measuring across a larger region. [ Muhammed Saqib et. al 2020]

**2.3 SUMMARY**

This chapter surveys the existing systems, their limitations, and their uses, which serve as references throughout the proposed system.

**CHAPTER-3**

**SYSTEM DESIGN**

**3.1 INTRODUCTION**

The system design chapter outlines the proposed system using an architecture design that includes the many components, modules, processes, and data needed to meet the system requirements.

**3.2 PROPOSED SYSTEM**

Four environmental parameters—temperature, humidity, ambient light intensity, and soil moisture content—were recorded for each individual location utilizing a total of three sensors, including an air temperature and humidity sensor, a light intensity sensor, and a soil moisture sensor.

The data collection method allows for a thorough examination of the agricultural sector's ecological circumstances.

The system's adaptability allows for the addition of additional monitors such as a pH sensor to estimate plant nutrients, a UV sensor to determine the intensity of ultraviolet light, a greenhouse gas and pressure sensor to track the composition of the surrounding air, a rain sensor to detect rainy weather, and a water level sensor to detect flooding in the area.

The sensors will detect based on their respective parameters whether to turn the pump motor ON/OFF and the readings are displayed on the LCD and recorded on the website via a common Wi-Fi using Node Mcu.

With our remote location, you may monitor the field area from anywhere on the planet. We may now get all of the sensor data from the webpage.

Sensor data will be used to manage the farm land. One pump motor is used to store excess water in the agricultural field, while another pump motor is utilized to provide water when needed.

**3.3 SYSTEM ARCHITECTURE**

AGRICULTURE FIELD

PUMP MOTOR

STORAGE TANK

PUMP MOTOR

Figure 3.1 System flow diagram of the proposed system

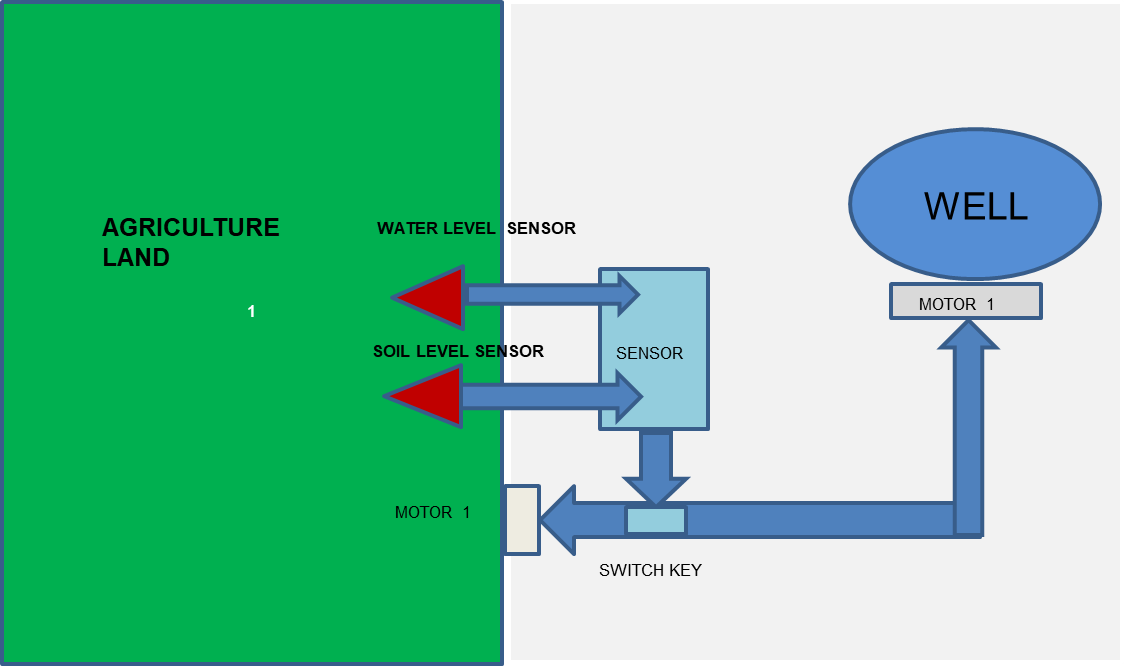
****

Figure 3.2 Architecture diagram of the proposed system

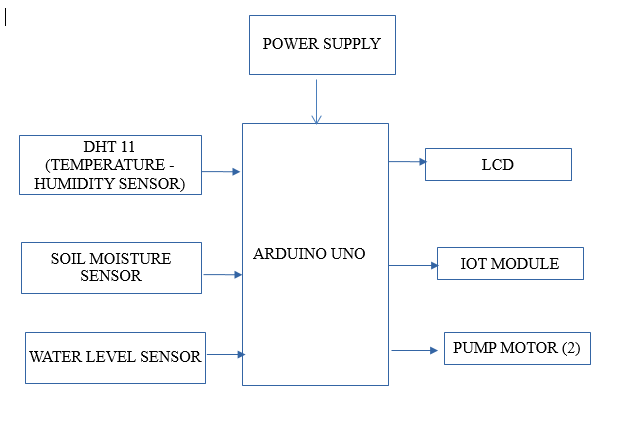
****

Figure 3.3 Block diagram of the proposed system

**3.4 SUMMARY**

This chapter summarizes the modules present in the system along with its design and implementation. The architectural design of the proposed system is drawn.

**CHAPTER-4**

**SYSTEM IMPLEMENTATION**

**4.1 INTRODUCTION**

The system implementation describes in detail the hardware, software implementations, and technologies utilized to construct the system. It aids in comprehending the operation of each module and their contribution to the overall system.

**4.2 METHODOLOGIES**

* COLLECT THE DATA FROM FIELD AREA
* MAINTAIN THE CROPS
* MONITOR THE FIELD AREA
* WATER MAINTAINING

**4.2.1 MODULES**

**MODULE 1- COLLECT THE DATA FROM FIELD AREA**

**ARDUINO**

**MEGA**

**POWER SUPPLY**

**TEMPRATURE**

**SENSOR**

**LCD**

**SOIL MOISTURE SENSOR**

**WATER LEVEL SENSOR**

Figure 4.1 Block diagram depicting collection of respective data from the field

Using sensor-based technology, we progressively collect the necessary information from the field area. The embedded system sensors and microcontroller assist the user in gathering information and updating the information on the LCD. Temperature, humidity, soil moisture, and water level sensors are used to collect data.

**MODULE 2- MAINTAIN THE CROPS**

**ARDUINO**

**MEGA**

**POWER SUPPLY**

**TEMPRATURE**

**SENSOR**

**LCD**

**SOIL MOISTURE SENSOR**

**WATER LEVEL SENSOR**

**DC MOTOR**

**MOTOR DRIVER**

Figure 4.2 Block diagram depicting the maintenance of crops using the sensors

Based on the soil moisture and water level values the pump motor condition ON/OFF. If the soil is dry the pump motor will turn on and fulfil the crop requirements. If the water is in the soil automatically get the moisture the pump motor will not run.

**MODULE 3- MONITOR THE FIELD AREA**

**ARDUINO**

**MEGA**

**POWER SUPPLY**

**TEMPRATURE**

**SENSOR**

**LCD**

**SOIL MOISTURE SENSOR**

**RAIN SENSOR**

**PUMP MOTOR**

**MOTOR DRIVER**

**IOT MODULE**

Figure 4.3 Block Diagram depicting the monitoring of the agricultural field

The field are can be monitored remotely. We can observe the information provided by the sensors from the webpage. According to the values provided by the sensor we can maintain the field.

**MODULE 4- WATER MAINTAINING**

STORAGE TANK

PUMP MOTOR

AGRICULTURE FIELD

PUMP MOTOR

Figure 4.4 Block diagram representing the working of the pumps in regulating water usage.

Out of the two motor pumps, one is responsible for storing the excess water from the agricultural field and the other provides water whenever the field requires.

* 1. **HARDWARE AND SOFTWARE SPECIFICATIONS:**
     1. **HARDWARE REQUIREMENTS**

**ARDUINO**

Arduino is a free and open-source electronics platform with simple hardware and software. are capable of reading inputs such as a light on a sensor, a finger on a button, or a Twitter post and converting them into outputs such as actuating a motor, turning on an LED, or publishing anything online. You may direct your board by delivering a series of instructions to the board's microcontroller.

Over the years, Arduino has served as the brain of hundreds of projects, ranging from simple household products to complicated scientific equipment. A global community of makers - students, amateurs, artists, programmers, and professionals - has collected around this open-source platform, and their contributions have built up to an astounding quantity of accessible information that may be of tremendous assistance to both novices and specialists. Arduino was created at the Ivrea Interaction Design Institute as a simple tool for rapid prototyping intended for students with no prior experience with electronics or programming. As the Arduino board gained popularity, it began to evolve to meet new demands and problems, transitioning from basic 8-bit boards to solutions for IoT applications, wearables, 3D printing, and embedded settings. All Arduino boards are totally open-source, allowing users to create them on their own and eventually customize them to their own requirements. The program is also open-source, and it is expanding thanks to the contributions of users all around the world.

**WHY ARDUINO?**

Arduino has been utilized in millions of different projects and applications because to its easy and accessible user experience. The Arduino software is simple to use for novices while yet being versatile enough for expert users. It is compatible with Mac, Windows, and Linux. It is used by teachers and students to create low-cost scientific equipment, to demonstrate chemistry and physics principles, and to get started with programming and robotics. Designers and architects use it to create interactive prototypes, while musicians and artists use it for installations and to test out new musical instruments. Makers, of course, utilize it to make many of the creations displayed at the Maker Faire. Arduino is an essential tool for learning new things.

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems.

* + In comparison to other microcontroller systems, Arduino boards are comparatively affordable. The cheapest Arduino module may be constructed by hand, and even pre-made Arduino modules cost less than $50.
  + Cross-platform - The Arduino Software (IDE) is available for Windows, Macintosh OSX, and Linux. Most microcontroller systems are only compatible with Windows.
  + Simple, straightforward programming environment - The Arduino Software (IDE) is simple enough for novices to use while yet being versatile enough for expert users to benefit from. It's built on the Processing programming environment, which is useful for teachers because students learning to code in that environment will be familiar with how the Arduino IDE works.
  + Open source and extendable software - The Arduino software is accessible as open-source tools for skilled programmers to extend.
  + Open source and extendable hardware - The Arduino board plans are given under a Creative Commons license, allowing skilled circuit designers to create their own version of the module, expanding and upgrading it. Even unskilled users may construct the breadboard version of the module to learn how it works and save money.

**ARDUINO MEGA**

The MEGA 2560 is designed for more complex projects. With 54 digital I/O pins, 16 analog inputs and a larger space for your sketch it is the recommended board for 3D printers and robotics projects.

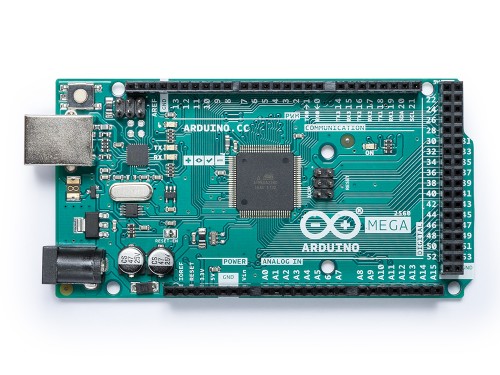


Figure 4.5 Arduino Mega 2560

The **Arduino Mega 2560** is a microcontroller board based on the [ATmega2560](http://www.atmel.com/Images/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf). It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started. The Mega 2560 board is compatible with most shields designed for the Uno and the former boards Duemilanove or Diecimila.

**Table 1 Technical Specifications Of Arduino Mega**

|  |  |
| --- | --- |
| **Microcontroller** | [ATmega2560](http://www.atmel.com/Images/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf) |
| **Operating Voltage** | 5V |
| **Input Voltage (recommended)** | 7-12V |
| **Input Voltage (limit)** | 6-20V |
| **Digital I/O Pins** | 54 (of which 15 provide PWM output) |
| **Analog Input Pins** | 16 |
| **DC Current per I/O Pin** | 20 mA |
| **DC Current for 3.3V Pin** | 50 mA |
| **Flash Memory** | 256 KB of which 8 KB used by boot loader |
| **SRAM** | 8 KB |
| **EEPROM** | 4 KB |
| **Clock Speed** | 16 MHz |
| **LED\_BUILTIN** | 13 |
| **Length** | 101.52 mm |
| **Width** | 53.3 mm |
| **Weight** | 37 g |

Although the hardware and software blueprints are publicly accessible under copyright licenses, the creators have requested that the name Arduino remain exclusive to the original product and not be used for derivative products without permission. The official policy paper on the usage of the Arduino name underlines that the project is open to integrate contributions by others into the official product. Several commercially available Arduino-compatible devices have evaded the project name by utilizing names that finish in -Duino. An early Arduino board with an RS-232 serial interface (upper left) and an Atmel ATmega8 microcontroller chip (black, lower right); the 14 digital I/O pins are at the top, the 6 analog input pins are at the lower right, and the power connector is at the bottom left.

Most Arduino boards are built on an 8-bit AVR microcontroller from Atmel (ATmega8, ATmega168, ATmega328, ATmega1280, ATmega2560) with variable amounts of flash memory, pins, and features. In 2012, the 32-bit Arduino Due, based on the Atmel SAM3X8E, was released. Single or double-row pins or female headers are used on the boards to simplify connections for programming and inclusion into other circuits. These can be linked to shields, which are add-on modules. An I2C serial bus can be used to separately address several and perhaps stacked shields. A 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator are standard on most boards. Due to form-factor constraints, certain designs, such as the Lilypad, operate at 8 MHz and do not have an integrated voltage regulator.

Arduino microcontrollers come pre-programmed with a boot loader that facilitates software uploading to the on-chip flash memory. The Opti boot bootloader is the Arduino UNO's default bootloader. A serial link to another computer is used to load software code onto boards. A level shifter circuit is used on certain serial Arduino boards to convert between RS-232 logic levels and transistor-transistor logic (TTL) level signals. Current Arduino boards are programmed over USB, which is accomplished with USB-to-serial adaptor chips such as the FTDI FT232. Some boards, such as later-model Uno boards, replace the FTDI chip with a separate AVR chip that has USB-to-serial firmware that can be reprogrammed through its own ICSP connector.

The Arduino board makes the bulk of the microcontroller's I/O pins accessible for use by other circuits. Each of the Diecimila, Duemilanove, and modern Uno has 14 digital I/O pins, six of which may create pulse-width modulated signals, and six analog inputs that can also operate as six digital I/O pins. These pins are accessed from the top of the board through female 0.1-inch (2.54 mm) headers. There are also plug-in application shields available for purchase. Male header pins on the underside of the Arduino Nano, as well as the Arduino-compatible Bare Bones Board and Boarduino boards, may be inserted into solderless breadboards.

### **PROGRAMMING:**

### **WARNINGS:**

The Mega 2560 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.

### **POWER:**

The Mega 2560 may be charged through USB or with an extra power source. The power source is automatically selected.

External (non-USB) power can be supplied by either an AC-to-DC adaptor (wall-wart) or a battery. Connect the adapter by inserting a 2.1mm center-positive connector into the board's power port. Battery leads can be placed into the POWER connector's GND and Vin pin headers.

The board may be powered by an external source ranging from 6 to 20 volts. However, if less than 7V is given, the 5V pin may deliver less than five volts, causing the board to become unstable. When more than 12V is applied, the voltage regulator may overheat and destroy the board. The suggested range is 7 to 12 Volts.

The power pins are as follows:

* **Vin**. The input voltage to the 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.
* **3V3**. A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
* **GND**. Ground pins.
* **IOREF**. This pin on the board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

### **MEMORY:**

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](https://www.arduino.cc/en/Reference/EEPROM)).

**INPUT AND OUTPUT:**

## ATmega2560-Arduino Pin Mapping.

Below is the pin mapping for the Atmega2560. The chip used in Arduino 2560. There are pin mappings to [Atmega8](https://www.arduino.cc/en/Hacking/PinMapping) and [Atmega 168/328](https://www.arduino.cc/en/Hacking/PinMapping168) as well.

**POWER SUPPLY**

This section describes how to generate +5V DC power supply

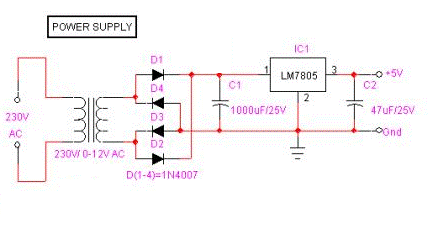


Figure 4.6 Power supply

The power supply section is essential as it should deliver constant output regulated power supply for successful working of the project. A 0-12V/1 mA transformer is used for this purpose. The primary component of this transformer is connected in to main supply through on/off switch& fuse for protecting from overload and short circuit protection. The secondary component is connected to the diodes to convert 12V AC to 12V DC voltage. And filtered by the capacitors, which is further regulated to +5v, by using IC 7805.

**LCD**

LCD screens are electronic display modules that have a broad variety of uses. A 16x2 LCD display is a relatively simple module that is widely utilized in a variety of devices and circuits. These modules are recommended over multi-segment LEDs with seven segments. The reasons for this are as follows: LCDs are inexpensive, readily programmable, and have no restrictions for showing unique and even bespoke characters (unlike in seven segments), animations, and so on. A 16x2 LCD can display 16 characters per line and has two such lines. Each character is presented in a 5x7 pixel matrix on this LCD. This LCD contains two registers: Command and Data. The command register holds the LCD's command instructions.

**DHT11 SENSOR:**

This DHT11 Temperature & Humidity Sensor has a temperature and humidity sensor complex as well as a calibrated digital signal output. It offers great dependability and outstanding long-term stability by employing an innovative digital-signal-acquisition approach as well as temperature and humidity sensor technology. This sensor consists of a resistive-type humidity measurement component and an NTC temperature measuring component that links to a high-performance 8-bit microcontroller, providing great quality, rapid response, anti-interference capability, and cost-effectiveness. Each DHT11 element is rigorously calibrated in a laboratory to provide exceptional accuracy in humidity calibration. The single-wire serial interface speeds up and simplifies system integration.

**Table 2 Technical Specifications Of DHT11 Sensor**

|  |  |  |
| --- | --- | --- |
|  | Humidity | Temperature |
| Resolution | 16 Bit | 16 Bit |
| Repeatability | ±1% RH | ±0.2 |
| Accuracy | At 25±5% RH | 1/e (63%) 10S |
| Interchangeability | Fully Interchangeable |  |
| Hysteresis | <±0.3%RH |  |
| Long-term stability | <±0.5% RH/yr. |  |

**Electrical Characteristics:**

* Power supply: DC 3.5-5.5V
* Supply Current: measurement 0.3mA standby 60μ A
* Sampling period: more than 2 seconds

**Features:**

* Simple design
* One wire interface
* High accuracy
* Internal pull out

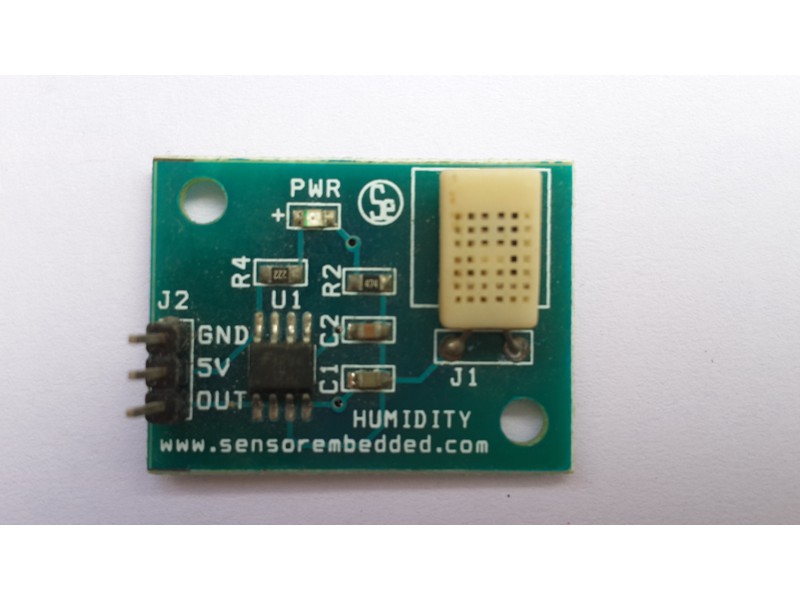


Figure 4.7 DHT 11 Sensor

**SOIL MOISTURE SENSOR**

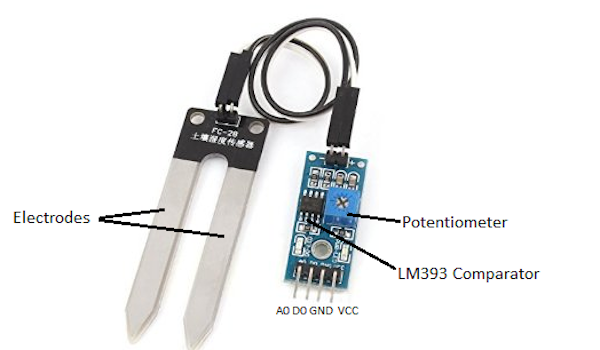


Figure 4.8 FC-28 soil moisture sensor

This sensor may be used to test the moisture of soil; when there is a water scarcity in the soil, the module output is high; otherwise, the output is low. Using this sensor, one may automatically water a flower plant or any other plant that requires automated watering. The volumetric water content of soil is measured by soil moisture sensors. Because direct gravimetric measurement of free-soil moisture requires removing, drying, and weighing a sample, soil moisture sensors indirectly measure the volumetric water content by using another property of the soil as a proxy for the moisture content, such as electrical resistance, dielectric constant, or interaction with neutrons. Module has three output modes: digital output, analog output, and mixed output.

**Description:**

This sensor is made up of two parts: a fork-shaped probe that must be nailed to the ground and an electrical module used to link the probe to Arduino. It requires a 3.3V or 5V power, which is controlled via the VCC and GND pins. When we are feeding, the red LED will light up. The pin AO will give us an analog signal between the supply value to 0V (for Arduino UNO 5v to 0V) to higher humidity value will be higher, this is due to the operation of the probe because the higher the humidity is greater conductivity of the soil and thus increase the value that we measure. The DO pin will output a digital signal of "1" when the humidity value is less than what we set on the potentiometer and "0" when it is greater. As the probe signal, the potentiometer is coupled to a comparator LM393. To utilize it appropriately, we must calibrate it using different soil samples with varying humidity levels. The green LED will assist us in this endeavour since it will light on when we have a 1 in DO.

**Weatherproofing**

If you want to use this sensor outside, we recommend adding some PCB shielding to make it stay longer. You could always use regular hot glue. Hot glue, on the other hand, does not fare well in the sun and should be reserved for projects that will not be subjected to high temperatures. Apply a conformal coating to the SMD parts on the PCB as well as the solder connections for projects that must withstand all elements.

**Calibration**

**System Calibration**

It is recommended that you calibrate your Soil Moisture Sensor to the soil you intend to monitor in order to obtain any usable data. The sensor can be affected by different types of soil, and you may receive different results from one composition to the next. Before you begin storing moisture data or triggering actions based on that number, you need first determine what values your sensor is producing. Using the schematic above, record the readings your sensor produces when it is entirely dry vs when it is completely submerged in a shallow cup of water. Your findings will vary depending on the microcontroller you're using, its operating voltage, and the resolution of its analog-to-digital converter.

**Soil Calibration**

Once you've determined what readings your sensor produces in fully dry and completely wet conditions, it's time to calibrate it for the exact soil you wish to monitor. Repeat the last test, but this time test your soil when it is as dry as possible before measuring it when it is entirely saturated with moisture. Obtaining these results and comparing them to the prior calibration will provide you with the most insight into what the data signify for your unique plant and soil. This exam may need some trial and error as well as patience. During these tests, take care not to over-water (or under-water) your plants.

**Working principle:**

This is a straightforward digital soil moisture sensor. Simply place the sensor into the soil and it will measure the moisture or water level content. It produces a digital output of 5V when the moisture level in the soil is high and 0V when the moisture level is low.

The sensor features a potentiometer for adjusting the moisture threshold. When the sensor detects more moisture than the preset threshold, the digital output rises and an LED illuminate. When the moisture content of the soil falls below the predetermined threshold, the output stays low. To sense the moisture level, connect the digital output to a microcontroller.

**Specifications: -**

* Operating voltage: 3.3V~5V
* Dual output mode, analog output more accurate
* A fixed bolt hole for easy installation
* With power indicator (red) and digital switching output indicator (green)
* Having LM393 comparator chip, stable
* Panel PCB Dimension: Approx.3cm x 1.5cm
* Soil Probe Dimension: Approx. 6cm x 3cm
* Cable Length: Approx.21cm
* VCC: 3.3V-5V
* GND: GND
* DO: digital output interface (0 and 1)
* AO: analog output interface
* Output Voltage: 0–4.2V
* Input Current: 35mA
* Output Signal: both analog and digital

**Connections: -**

* VCC  connect to 3.3V-5V
* GND connect to GND
* DO   digital value output connector（0 or 1）
* AO  analog value output connector

**Usage: -**

* Soil moisture module is most sensitive to the ambient, generally used to detect the moisture content of the soil.
* When the module cannot reach the threshold value, DO port output high, when the soil humidity exceeds a set threshold value, the module D0 output low;
* The small board digital output D0 can be connected directly to the MCU, MCU to detect high and low, to detect soil moisture;
* Small board digital output DO can directly drive the buzzer module or relay module in our store, which can form a soil moisture alarm equipment;
* Small board analog output AO and AD module connected through the AD converter; you can get more precise values ​​of soil moisture.

**Specifications**

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

* Input Voltage: 3.3–5V

**Pin-out**

The FC-28 soil moisture sensor has four pins:

* VCC: Power
* A0: Analog Output
* D0: Digital Output
* GND: Ground

**WATER LEVEL SENSOR**

.

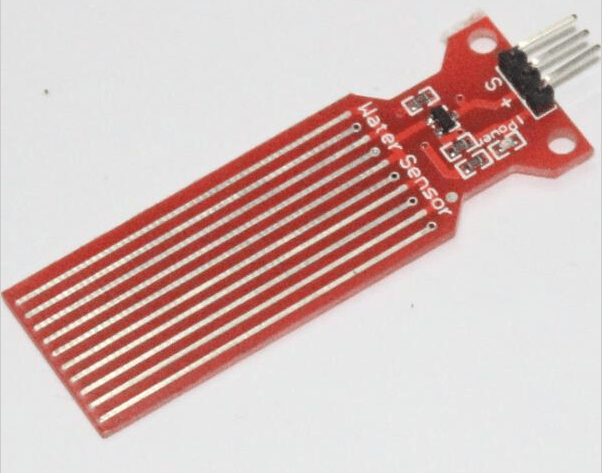


Figure 4.9 ARD2-3025 Water Level Sensor

A water level sensor is used to measure the amount of water in a tank or other piece of equipment. We need to detect the level of water in our daily lives, such as in a water tank on the roof that is not easily accessible, and in labs where a certain level of any liquid must be monitored. A water level sensor is extremely useful in such situations. In today's article, we'll look at how to connect a water level sensor to an Arduino and how to use this assembly to monitor certain water levels. The output of a water level sensor is voltage. This circuit will tell us approximately how much of the sensor is covered by liquid.

Liquid level sensors are utilized in a wide range of applications. They are widely used in vehicles, which rely on a large number of various fluids to operate in order to monitor the amount of petrol in the car, windshield washer fluid, and oil levels. In general, they are used to measure the level of any form of fluid in a system. As a result, being able to study, operate, and design circuits using them is tremendously useful.

The liquid level sensor we'll be using is an analog sensor, which means it emits an analog voltage proportional to the amount of liquid it's exposed to. We just connect the analog pin, denoted by an S, to an analog pin on the Arduino board.

As you can see from Figure 4.9, the sensor has a series of parallel wires across the board. These wires are what sense the liquid level that the board is exposed to.

If the board is submerged in water or another fluid, it will output the maximum analog value reading. Because analog values read by an Arduino range from 0 (lowest reading) to 1023 (highest reading), a board entirely submerged in liquid will give an Arduino a reading of 1023. The Arduino will read a reading of roughly 512 if the board is partly covered. If a liquid cover 1/4 of the board, the Arduino will read roughly 256. If there is no liquid on it at all, a near-zero reading should be obtained.

There are several things we can do with our liquid level sensor and an Arduino. Simply said, we may utilize the sensor to read and output the acquired analog value. Alternatively, we might design a form of alert indication status system. For example, when the sensor is entirely filled (submerged to the top with fluid), we may have a green LED light up to indicate that it is full. When the sensor's liquid level falls below 1/4, we may have a red LED light up, similar to how automobile dashboard LED indications notify us when our gas tank falls below E (empty).

So, with a microcontroller like the Arduino, we have virtually endless possibilities for incorporating the liquid level sensor.

## **WORKING PRINCIPLE**

This sensor operates by connecting a sequence of exposed traces to ground, with the sensing traces interlaced between the grounded traces. The sensor traces feature a 1 M pull-up resistor. The resistor will raise the sensor trace value until a drop of water shorts it to the grounded trace. This circuit, believe it or not, will operate with your Arduino's digital I/O pins or analog pins to measure the amount of water caused contact between the grounded and sensor traces.

Use as a tank level.

Install the sensor on the interior of the tank at the level where we want to manage the water level to use it as a level detector in a deposit. The sensor must be placed in such a way that parallel lines are perpendicular to the sensor's water level. As the sensor is submerged, the value on pin S will increase.

Use as a rain sensor

To detect rain with this sensor, set it horizontally so that droplets fall on the sensor; when raindrops fall on the sensor, a film of water on the surface is generated by raising the value of the pin S; this is how we can determine if it is raining.

**Table 3 Technical Specifications Of Water-level sensor**

|  |  |
| --- | --- |
| Working Voltage | DC 3-5V |
| Working Current | <20mA |
| Interface | Analog |
| Weight | 3g. |
| [Sensor](http://www.hotmcu.com/) Type | Simulation |
| Detection Area | 40 mm x 16 mm |
| Manufacturing Process | FR4 double spray tin |
| Fixed Hole Size | * 1. mm |
| Humanized Design | Half-moon sag non-skid treatment |
| Working Temperature | 10ºC to 30 °C |
| Work Humidity | 10% to 90% without condensation |
| Size | 65 mm x 20 mm x 8 mm |
| Output Voltage | 0- 4.2 v. |
| Optional Accessories | 3 pins [sensor](http://www.hotmcu.com/) connecting line |

**Applications:**

* Rainfall detecting
* Tank overflow detector
* Water level detection for the pan of a water heater. It’s very common for these heaters to leak when they get older. Placing a sensor in the pan to detect the presence of water where there should be none.
* Same goes for under sink links.  Especially in homes that are vacant for much of the year. Use an [Ethernet Shield](https://www.thegeekpub.com/16569/controlling-an-arduino-from-a-web-page/) (or Wi-Fi) to add this leak detection device to the internet and have it send an email if water is detected.

**MOTOR DRIVER IC:**

Common DC gear head motors need current above 250mA. There are many integrated circuits like ATmega16 [Microcontroller](https://www.elprocus.com/8051-microcontroller-architecture-and-applications/), [555 timers IC](https://www.elprocus.com/555-timer-circuits-for-engineering-students/). But, IC 74 series cannot supply this amount of current. When the motor is directly connected to the o/p of the above ICs then, they might damage. To overcome this problem, a motor control circuit is required, which can act as a bridge between the above motors and ICs ([integrated circuits](https://www.elprocus.com/different-types-of-integrated-circuits/)). There are various ways of making H-bridge motor control circuit such as using transistor, relays and using L293D/L298.

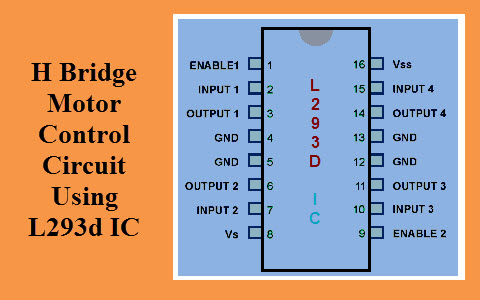
[](https://www.elprocus.com/wp-content/uploads/2015/03/H-Bridge-Motor-Control-Circuit-Using-L293d-IC.jpg)

Figure 4.10 H Bridge Motor Control Circuit Using L293d IC

## **H-BRIDGE CIRCUIT**:

An H bridge is an electronic circuit that allows a voltage to be applied across a load in any direction. H-bridge circuits are frequently used in robotics and many other applications to allow DC motors to run forward & backward. These motor control circuits are mostly used in different converters like DC-DC, DC-AC, AC-AC converters and many other types of [power electronic converters](https://www.elprocus.com/power-electronic-converters/). In specific, a bipolar stepper motor is always driven by a motor controller having two H-bridges.

An H-bridge is fabricated with[four switches](https://www.elprocus.com/switches-types-working/) like S1, S2, S3 and S4. When the S1 and S4 switches are closed, then a positive voltage will be applied across the motor. By opening the switches S1 and S4 and closing the switches S2 and S3, this voltage is inverted, allowing invert operation of the motor.

Generally, the H-bridge motor driver circuit is used to reverse the direction of the motor and also to break the motor. When the motor comes to a sudden stop, as the terminals of the motor are shorted. Or let the motor run free to a stop, when the motor is detached from the circuit. The table below gives the different operations with the four switches corresponding to the above circuit.

**WATER PUMP MOTOR**

Water pumps, as the name suggests, pump water. Customers may possibly find a water pump to fit their vehicle or to assist them pull water from the ground in a self-dug well to be utilized in pressure tanks inside the area, whether it is in a vehicle, at a company, in the house, or in a well.

Vehicle water pumps control the flow of water through a vehicle's cooling system; if the seal on one of these fails, the entire pump must be replaced. Pressure water pumps, which are located within the house or company, manage the water pressure year-round, managing water flow to various regions of the place. A direct current motor is a device that transforms direct current electrical power into mechanical power. A direct current (DC) motor operates on the principle that when a current carrying conductor is put in a magnetic field, it feels torque and has a propensity to move.

This is referred to as driving activity. Pumps use energy to produce mechanical work by moving fluid and are powered by some mechanism (usually reciprocating or rotational). Pumps employ a variety of energy sources, including manual labor, electricity, motors, or wind power, and they come in a variety of sizes, ranging from tiny for medical uses to enormous industrial pumps.



Figure 4.11 Pump motor

**ESP-12E BASED** **NODEMCU**

The ESP8266 is the name of a micro controller designed by Espressif Systems. The ESP8266 itself is a self-contained Wi-Fi networking solution offering as a bridge from existing micro controller to Wi-Fi and is also capable of running self-contained applications. This module comes with a built in USB connector and a rich assortment of pin-outs. With a micro-USB cable, you can connect NodeMCU devkit to your laptop and flash it without any trouble, just like Arduino. It is also immediately breadboard friendly.

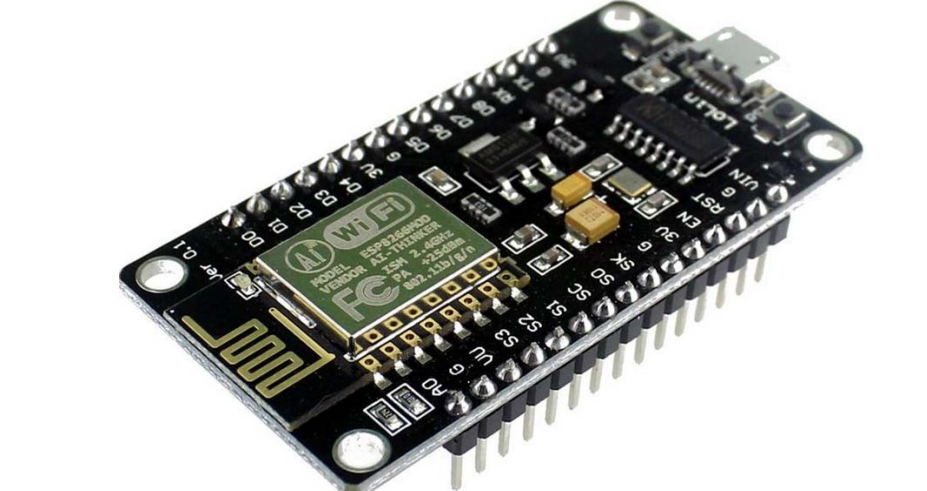


Figure 4.12 ESP-12E BASED NODEMCU

ESP-12E Wi-Fi module is developed by Ai-thinker Team. core processor ESP8266 in smaller sizes of the module encapsulates Tensilica L106 integrates industry-leading ultra-low power 32-bit MCU micro, with the 16-bit short mode, Clock speed support 80 MHz, 160 MHz, supports the RTOS, integrated Wi-Fi MAC/BB/RF/PA/LNA, on-board antenna. The module supports standard IEEE802.11 b/g/n agreement, complete TCP/IP protocol stack. Users can use the add modules to an existing device networking, or building a separate network controller. ESP8266 is high integration wireless SOCs, designed for space and power constrained mobile platform designers. It provides unsurpassed ability to embed Wi-Fi capabilities within other systems, or to function as a standalone application, with the lowest cost, and minimal space requirement.

The ESP8266EX is a self-contained Wi-Fi networking solution that may be used to host the application or to offload Wi-Fi networking functionalities from another application processor. When the ESP8266EX hosts the program, it boots from an external flash drive. It contains an embedded cache to increase system performance in such applications. Wireless internet access may also be introduced to any microcontroller-based design with basic connection (SPI/SDIO or I2C/UART interface) by acting as a Wi-Fi adaptor. The ESP8266EX is one of the most integrated Wi-Fi chips in the industry; it integrates antenna switches, RF baluns, power amplifiers, low noise receive amplifiers, filters, and power management modules, requires minimal external circuitry, and the entire solution, including the front-end module, is designed to occupy minimal PCB area.

**4.2.2 SOFTWARE REQUIREMENTS:**

**EMBEDDED C**

Embedded C is the most widely used programming language in the software industry for creating electrical devices. Every processor in an electronic system is linked to embedded software.

Embedded C programming is essential for the CPU to accomplish specified functions. We utilize various electronic equipment in our daily lives, such as a mobile phone, a washing machine, a digital camera, and so on. These all-device operations are based on microcontrollers coded in embedded C.

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

**EMBEDDED SYSTEM PROGRAMMING:**

Embedded C Programming with Keil Language2

Figure 4.13 Block diagram of Embedded C Programming development

## A function is a collection of statements that is used to complete a certain activity, and a programming language is a collection of one or more functions. Every language is made up of fundamental ingredients and grammatical rules. The C programming language is intended for function with variables, character set, data kinds, keywords, expressions, and so on.

## The embedded C programming language is a C language extension. In addition to the features listed above, embedded programming in C includes data types, keywords, and header files, which are represented by #includemicrocontroller name.h>.

## **BASIC EMBEDDED C PROGRAMMING STEPS:**

Embedded C Programming with Keil Language3

Figure 4.14 Block diagram representation of Embedded C Programming Steps

The microcontroller programming is different for each type of operating system. Even though there are many operating system is existed such as Windows, Linux, RTOS, etc. but RTOS has several advantages for embedded system development.

**EMBEDDED SYSTEMS:**

An embedded system is a combination of hardware, application software, and a real-time operating system. It might be a tiny self-contained system or a huge combinational system. Our Embedded System tutorial covers all aspects of the Embedded System, including characteristics, design, processors, microcontrollers, tools, addressing modes, assembly language, interrupts, embedded c programming, led blinking, serial communication, lcd programming, keyboard programming, project implementation, and so on. It may also be described as a method of functioning, arranging, or carrying out one or more tasks in accordance with a predetermined plan. An Embedded System is a system that has software embedded within computer hardware, resulting in a system devoted to a single application or product or part of a larger system. An embedded system might be a tiny self-contained unit or a huge combinational unit. It is a microcontroller-based control system that is used to complete a certain operation job. A system is a method of functioning, arranging, or doing one or more activities in accordance with a predetermined set of rules, program, or plan. An embedded system is a combination of three major components:

* **Hardware:** Hardware is physically used component that is physically connected with an embedded system. It comprises of microcontroller based integrated circuit, power supply, LCD display etc.
* **Application software:** Application software allows the user to perform varieties of application to be run on an embedded system by changing the code installed in an embedded system.
* **Real Time Operating system (RTOS):** RTOS supervises the way an embedded system work. It acts as an interface between hardware and application software which supervises the application software and provide mechanism to let the processor run on the basis of scheduling for controlling the effect of latencies.

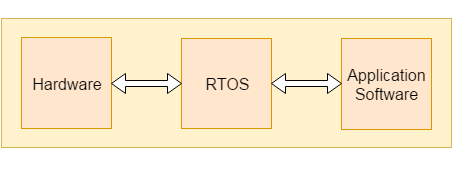


Figure 4.15 RTOS-Real time operating system

# **CHARACTERISTICS OF EMBEDDED SYSTEM:**

* An embedded system is software embedded into computer hardware that makes a system dedicated to be used for variety of application.
* Embedded system generally used for do specific task that provide real-time output on the basis of various characteristics of an embedded system.
* Embedded system may contain a smaller part within a larger device that used for serving the more specific application to perform variety of task using hardware-software intermixing configuration.
* It provides high reliability and real-time computation ability.

**BASIC STRUCTURE OF AN EMBEDDED SYSTEM:**

ES Design of ES

Figure 4.16 Block diagram shows the basic structure of an embedded system.

* **Sensor**: Sensor used for sensing the change in environment condition and it generate the electric signal on the basis of change in environment condition. Therefore, it is also called as transducers for providing electric input signal on the basis of change in environment condition.
* **A-D Converter**: An analog-to-digital converter is a device that converts analog electric input signal into its equivalent digital signal for further processing in an embedded system.
* **Processor & ASICs**: Processor used for processing the signal and data to execute desired set of instructions with high-speed of operation. Application specific integrated circuit (ASIC) is an integrated circuit designed to perform task specific operation inside an embedded system.
* **D-A Converter**: A digital-to-analog converter is a device that converts digital electric input signal into its equivalent analog signal for further processing in an embedded system.
* **Actuators**: Actuators is a comparator used for comparing the analog input signal level to desired output signal level for providing the error free output from the system.

**DESIGN STEPS REQUIRED FOR THE DEVELOPMENT OF EMBEDDED SYSTEM:**

Designing steps required for embedded system are different from the design process of another electronic system.

ES Design of ES

Figure 4.17 Flow chart represent the design steps required in the development of an embedded system

# **EMBEDDED SYSTEM TOOLS AND PERIPHERALS:**

**Compiler:**

Compiler is used for converting the source code from a high-level programming language to a low-level programming language. It converts the code written in high level programming language into assembly or machine code. The main reason for conversion is to develop an executable program.

Let's see the operations performed by compiler are:

* Code generation
* Code optimization
* Parsing
* Syntax direct translation
* Pre-processing

**Cross-Compiler:**

If a program compiled is run on a computer having different operating system and hardware configuration than the computer system on which a compiler compiled the program, that compiler is known as cross-compiler.

**Decompile:**

A tool used for translating a program from a low-level language to high-level language is called a decompile. It is used for conversion of assembly or machine code to high-level programming language.

**Assembler:**

Assembler is embedded system tool used for translating a computer instruction written in assembly language into a pattern of bits which is used by the computer processor for performing its basic operations. Assembler creates an object code by translating assembly language instruction into set of mnemonics for representing each low-level machine operation.

## **DEBUGGING TOOLS IN AN EMBEDDED SYSTEM:**

Debugging is a tool used for reducing the number of error or bugs inside a computer program or an assembled electronic hardware.

Debugging of a compact subsystem is difficult because a small change in one subsystem can create bugs in another system. The debugging used inside embedded system differs in terms of their development time and debugging features.

**Let's see the different debugging tools used in embedded system are:**

**Simulators:**

Simulator is a tool used for simulation of an embedded system. Code tested for microcontroller unit by simulating code on the host computer. Simulator is used for model the behavior of the complete microcontroller in software.

**Functions of simulators:**

Let's see the functions performed by simulator are:

* It defines the processing or processor device family with various version of target system.
* It monitors the detailed information of a source code and symbolic arguments as the execution goes for each single step of operation.
* It simulates the ports of target system for each single step of execution.
* It provides the working status of RAM.
* It monitors the response of system and determines the throughput.
* It provides the complete meaning of the present command.
* It monitors the detailed information of the simulator commands entered from the keyboard or selected from the menu.
* It facilitates synchronization of internal peripherals and delays.

**MICROCONTROLLER STARTER KIT:**

For developing an embedded system-based project a complete microcontroller starter kit is required. The major advantage of this kit over simulator is that they work in real-time operating condition. Therefore, it allows the easy input/output functional verification. Consider a microcontroller starter kit consists of:-

* Hardware Printed Circuit Board (PCB)
* In-System Programmer (ISP)
* Some embedded system tools like compiler, assembler, linker, etc
* Sometimes, there is a requirement of an Integrated Development Environment (IDE)

The above component available in microcontroller starter kit is completely enough and the cheapest option available for developing simple microcontroller projects.

**EMULATORS:**

An emulator is a software program or a hardware kit which emulates the functions of one computer system into another computer system. Emulators have an ability to support closer connection to an authenticity of the digital object.

It can also be defined as the ability of a computer program in electronic device to emulate another program or device. It focusing on recreating the original computer environment and helps a user to work on any type of application or operating system.

## **PERIPHERAL DEVICES IN EMBEDDED SYSTEMS:**

Communication of an embedded system with an outside environment is done by using different peripheral devices as a combination with microcontroller.

Let's see the different peripheral devices in embedded system are:-

* Universal Serial Bus (USB)
* Networks like Ethernet, Local Area Network (LAN) etc
* Multi Media Cards (SD Cards, Flash memory, etc)
* Serial Communication Interface (SCI) like RS-232, RS-485, RS-422, etc
* Synchronous Serial Communication Interface like SPI, SSC and ESSI
* Digital to Analog/ Analog to Digital (DAC/ADC)
* General Purpose Input/Output (GPIO)
* Debugging like In System Programming (ISP), In Circuit Serial Programming (ICSP), BDM Port, etc

## **CRITERIA FOR CHOOSING MICROCONTROLLER:**

Choosing a microcontroller is essential process in designing of embedded system. While selecting a microcontroller, make sure that it meets the system need and it must be cost effective. We need to decide whether an 8-bit, 16-bit or 32-bit microcontroller is best suitable for the computing needs of a task.

In addition to above, the following points need to be kept in mind while selecting a microcontroller:-

* **Speed**: The operational speed of the microcontroller or the highest speed microcontroller can support.
* **Packaging**: Packaging is important for improving the assembling, space and prototyping of an end-product.
* **RAM and ROM**: On the basis of operation of embedded system and memory need for storage data and programs the type of microcontroller required for designing system is decided.
* **Count of I/O pins**: The number of input and output devices connected with the system plays an essential role in choosing the type of microcontroller.
* **Cost per unit**: It is important in terms of final cost of the product in which the microcontroller is to be used.
* **Power consumption**: Power consumption plays an important role for maintaining the efficiency of an embedded system.

**ARDUINO SOFTWARE 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 Genuino hardware to upload programs and communicate with them.

### Image result for what is arduino ide software

Figure 4.18 Arduino IDE Software

**4.4 IMPLEMENTATION**

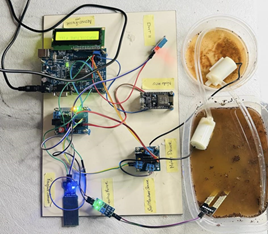
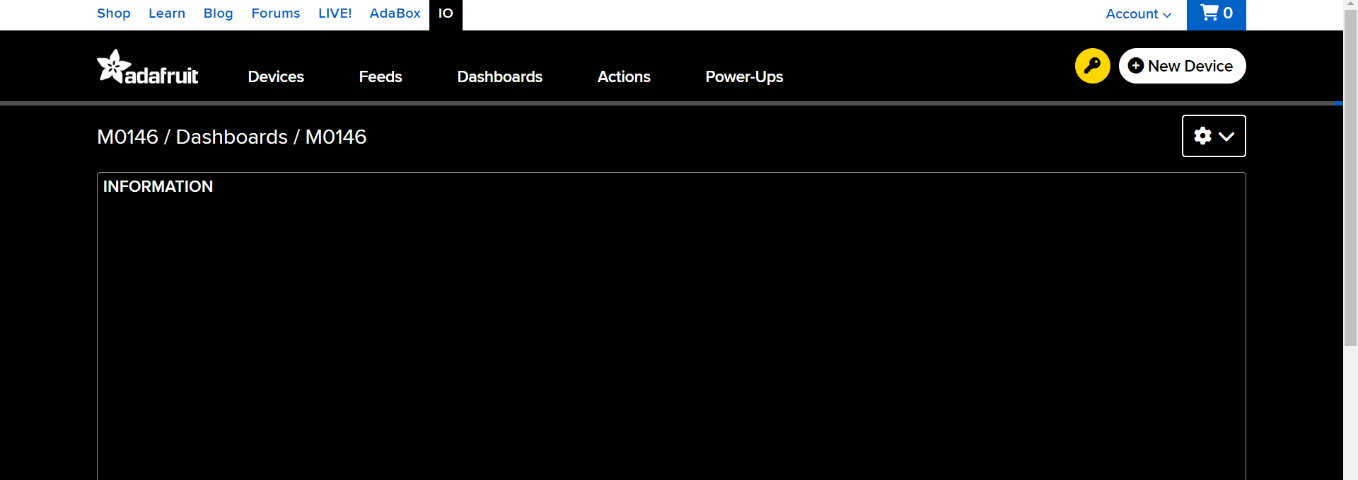
****

Figure.4.19 Device Photo



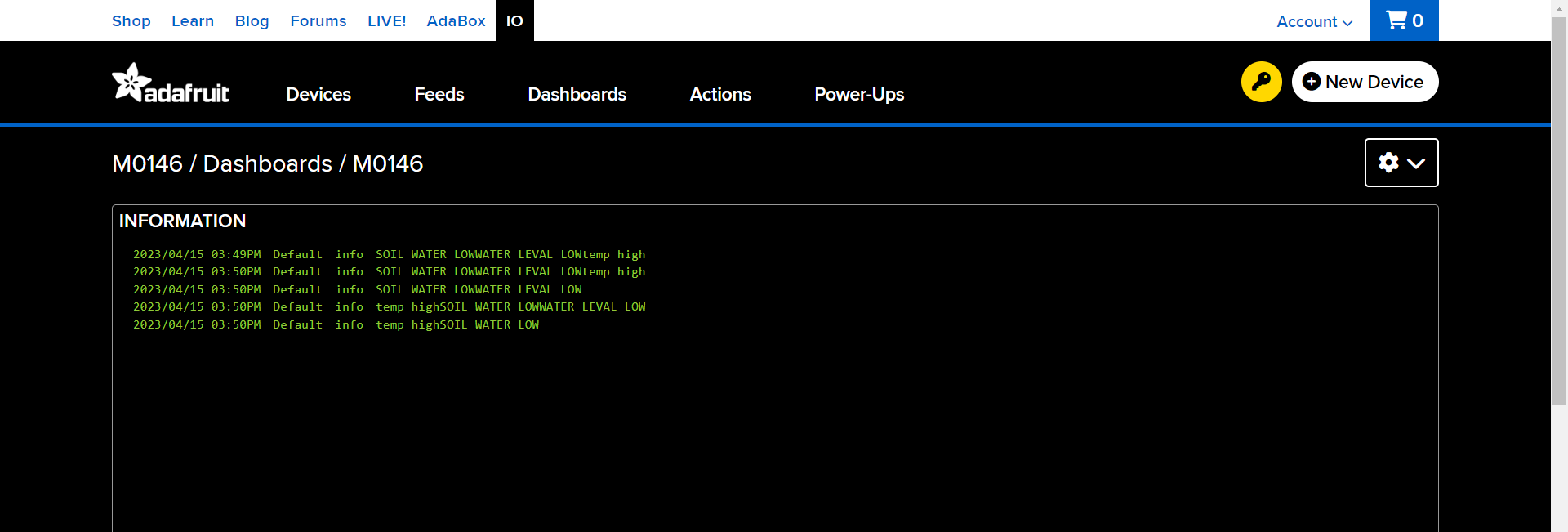


Figure 4.20 Adafruit Software

**4.5 SUMMARY**

This chapter describes the tools and libraries used in the system's implementation, as well as its reasoning and features. It also includes the technology, software, and hardware requirements for the system's effective deployment.

**CHAPTER- 5**

**PERFORMANCE EVALUATION**

**5.1 INTRODUCTION**

The process of analyzing the performance of software is referred to as performance analysis. Throughout the software program, check for correctness and integrity. The major goal of performance analysis is to monitor efficiency and guarantee that it is within the optimum efficiency range. The technique of analyzing the project's execution time is known as performance analysis. If the program operates well, it is operating at peak efficiency. If the software program is inefficient, it is in the negative efficiency range.

**5.2 COMPARISON**

Figure 5.1 depicts the expenditure versus time in implementing smart farming i.e., IoT sensor-based farming versus the traditional methodologies of farming. As observed, the cost increases at a significant rate in the conventional farming methods compared to the smart farming methodology

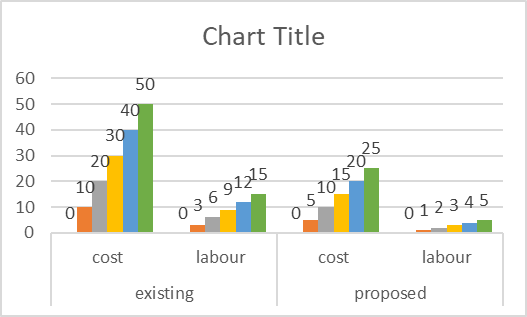


Figure 5.2 the graph represents the expenditure spent versus the amount of labour required. As you can see, the amount spent increases as the labour increases since the existing system is maintained manually. On the other hand, when using the smart iot based farming method the labour is less hence the cost decreases.

**5.3 SUMMARY**

This chapter mostly focuses on the software's performance analysis. A comparison of traditional ways and current agricultural methods is made, and it is discovered that sensor-based farming has more advantages and operates more effectively, and testing was performed to ensure its correctness. The suggested system's performance is assessed. When smart farming is utilized, it is shown that the accuracy increases significantly.

**CHAPTER-6**

**CONCLUSIONS AND FUTURE work**

**6.1 cONCLUSION**

We created an IoT-based automated soil health monitoring system that can cope with agricultural factors. Farmers will be able to incorporate many field parameters to make informed judgements based on the scenario. A is capable of remote data storage and retrieval and can be directly connected to the cloud. This study may be expanded to include AI/machine learning algorithms to assist agro-scientists and farmers in making wiser agricultural decisions and strategy creation.

**6.2 fUTURE ENHANCEMENTS**

In future, we would add a camera technology to interface with the controller and get live stream updates through IOT. Moreover, we could deploy different machine learning and deep learning techniques in prediction and maintenance of agricultural fields.

**APPENDIX:**

#include "dht11.h"

#include "iot.h"

DHT;

#include <LiquidCrystal.h>

void iot\_send(String s);

#define soil 3

#define water A1

#define Motor\_soil 5

#define Motor\_water 6

LiquidCrystal lcd(13, 12, 11, 10, 9, 8);

void setup() {

  Serial.begin(9600);

  pinMode(3, INPUT\_PULLUP);

  pinMode(A1, INPUT);

  pinMode(Motor\_soil, OUTPUT);

  pinMode(Motor\_water, OUTPUT);

  digitalWrite(Motor\_soil, LOW);

  digitalWrite(Motor\_water, LOW);

  lcd.begin(16, 2);

  lcd.setCursor(0, 0);

  lcd.print("AGRICULTURE FIELD");

  delay(1000);

}

void loop() {

  dht.dht\_read(2);

  Serial.println(dht.temperature);

  if (dht.temperature > 33) {

    iot\_send("\*temp normal#");

  } else {

    iot\_send("\*temp high#");

  }

  int soil\_pin = digitalRead(3);

  Serial.println(soil\_pin);

  int wate\_data = analogRead(A1);

  Serial.println(wate\_data);

  lcd.setCursor(0, 0);

  lcd.print("AGRICULTUR FIELD");

  lcd.setCursor(0, 1);

  lcd.print("SL Q:");

  lcd.setCursor(8, 1);

  lcd.print("WT L:");

  if (soil\_pin == 1) {

    lcd.setCursor(5, 1);

    lcd.print("LOW");

    iot\_send("\*SOIL WATER LOW#");

    digitalWrite(Motor\_soil, HIGH);

    digitalWrite(Motor\_water, HIGH);

  } else {

    lcd.setCursor(5, 1);

    lcd.print("OK");

    digitalWrite(Motor\_soil, LOW);

    digitalWrite(Motor\_water, LOW);

  }

  if (wate\_data <= 100) {

    lcd.setCursor(13, 1);

    lcd.print("LOW");

    iot\_send("\*WATER LEVAL LOW#");

    digitalWrite(Motor\_soil, HIGH);

    digitalWrite(Motor\_water, HIGH);

  } else {

    lcd.setCursor(13, 1);

    lcd.print("OK");

    digitalWrite(Motor\_soil, LOW);

    digitalWrite(Motor\_water, LOW);

  }

  delay(500);

}

**REFERENCES**

1. Aggarwal, N., & Singh, D. (2021). Technology assisted farming: Implications of IoT and AI. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1022, No. 1, p. 012080). IOP Publishing.
2. Caffaro, F., Cremasco, M. M., Roccato, M., & Cavallo, E. (2020). Drivers of farmers’ intention to adopt technological innovations in Italy: The role of information sources, perceived usefulness, and perceived ease of use. Journal of Rural Studies, Vol 76, (pp. 264-271).
3. Farooq, H., Rehman, H. U., Javed, A., Shoukat, M., & Dudley, S. (2020). A review on smart IoT based farming. *Annals of Emerging Technologies in Computing (AETiC), Print ISSN*, 2516-0281.
4. Gondchawar, N., & Kawitkar, R. S. (2016). IoT based smart agriculture. *International Journal of advanced research in Computer and Communication Engineering*, *5*(6), (pp. 838-842).
5. Krishna, K. L., Silver, O., Malende, W. F., & Anuradha, K. (2017, February). Internet of Things application for implementation of smart agriculture system. In *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)* (pp. 54-59). IEEE.
6. Patil, G. L., Gawande, P. S., & Bag, R. V. (2017). Smart agriculture system based on IoT and its social impact. *International Journal of Computer Applications*, *176*(1), 0975-8887.

# Paul, K., Chatterjee, S. S., Pai, P., Varshney, A., Juikar, S., Prasad, V., ... & Dasgupta, S. (2022). Viable smart sensors and their application in data driven agriculture. *Computers and Electronics in Agriculture*, Vol *198*, 107096.

1. Sanjeevi, P., Prasanna, S., Siva Kumar, B., Gunasekaran, G., Alagiri, I., & Vijay Anand, R. (2020). Precision agriculture and farming using Internet of Things based on wireless sensor network. *Transactions on Emerging Telecommunications Technologies*, *31*(12), e3978.
2. Saqib, M., Almohamad, T. A., & Mehmood, R. M. (2020). A low-cost information monitoring system for smart farming applications. *Sensors*,  *20*(8), 2367.
3. Saraswathi, D., Manibharathy, P., Gokulnath, R., Sureshkumar, E., & Karthikeyan, K. (2018, July). Automation of hydroponics green house farming using IoT. In *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA)* (pp. 1-4). IEEE.
4. Sengupta, A., Debnath, B., Das, A., & De, D. (2021). FarmFox: A quad-sensor-based IoT box for precision agriculture. *IEEE Consumer Electronics Magazine*, *10*(4), (pp. 63-68).
5. Suma, N., Samson, S. R., Saranya, S., Shanmugapriya, G., & Subhashri, R. (2017). IOT based smart agriculture monitoring system. *International Journal on Recent and Innovation Trends in computing and communication*, *5*(2), (pp. 177-181).
6. Sushanth, G., & Sujatha, S. (2018, March). IOT based smart agriculture system. In *2018 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (pp. 1-4). IEEE.
7. Vij, A., Vijendra, S., Jain, A., Bajaj, S., Bassi, A., & Sharma, A. (2020). IoT and machine learning approaches for automation of farm irrigation system. *Procedia Computer Science*, Vol *167*,(pp. 1250-1257).
8. Visconti, P., Giannoccaro, N. I., de Fazio, R., Strazzella, S., & Cafagna, D. (2020). IoT-oriented software platform applied to sensors-based farming facility with smartphone farmer app. *Bulletin of Electrical Engineering and Informatics*, *9*(3), (pp.1095-1105).