Journal on Electronic and Automation Engineering
Vol: 4(2), June 2025
REST Publisher; ISSN: 2583-6951 (Online)
Website: https://restpublisher.com/journals/jeae/
DOI: https://doi.org/10.46632/jeae/4/2/40

Smart Irrigation System for Precision Farming using Agribot

***P. Laxman, K. Saiveena, K. Vaishnavi, P. Pramod**
Nalla Malla Reddy Engineering College, Hyderabad, Telangana, India.
*Corresponding Author Email: plaxman.ece@nmrec.edu.in

Abstract: *Agriculture is crucial for food production, and effective irrigation is crucial for ensuring healthy crop growth. Smart Irrigation System Using Agribot is an autonomous robotic solution designed to optimize water use based on real-time soil moisture data. Built on a mobile robot platform and powered by a Raspberry Pi, the system manages movement using DC motors and avoids obstacles using an ultrasonic sensor. A servo motor inserts a soil moisture sensor into the ground, and if the moisture level is detected to be low, a relay triggers a water pump to irrigate the area. The system includes a DHT11 sensor to monitor temperature and humidity, and a pH sensor to assess soil acidity – providing critical insights for precision farming. Remotely controlled via a Bluetooth module and powered by a dedicated 12V battery, this automated system significantly reduces water wastage, increases efficiency, and reduces the need for manual intervention, making it a highly effective tool for smart and precision farming.*

1. INTRODUCTION

Smart irrigation systems are transforming modern agriculture by optimizing water use and reducing manual labor. Traditional irrigation methods often lead to water waste and inefficient crop management, making automation an important step towards sustainable agriculture. The smart irrigation system using Agribot is designed to address these challenges by integrating advanced technologies such as Raspberry Pi, soil moisture sensors, and automatic water control mechanisms. The system operates on a mobile robotic platform that not only detects soil moisture levels, but also monitors environmental factors such as temperature, humidity, and soil pH with a DHT11 sensor and a pH sensor. With features such as obstacle detection, remote control via Bluetooth, and automatic water distribution through a relay-controlled pumping system, Agribot improves agricultural efficiency and precision. By reducing human intervention and optimizing irrigation based on real-time soil conditions, the system promotes water conservation and supports the development of smart and sustainable agricultural practices.

2. EMBEDDED SYSTEM IMPLEMENTATION

An embedded system is a type of computer system that is mainly designed to perform multiple tasks such as accessing, processing, and storing and controlling data in various electronic based systems. Embedded systems are a combination of hardware and software where the software is usually called firmware embedded in the hardware. One of the most important characteristics of these systems is that it provides o/p within time constraints. Embedded systems help to make work more perfect and convenient. Therefore, we often use embedded systems in both simple and complex devices. The applications of embedded systems mainly include many devices in our real life such as microwaves, calculators, TV remote controls, home security, and ambient traffic control systems.

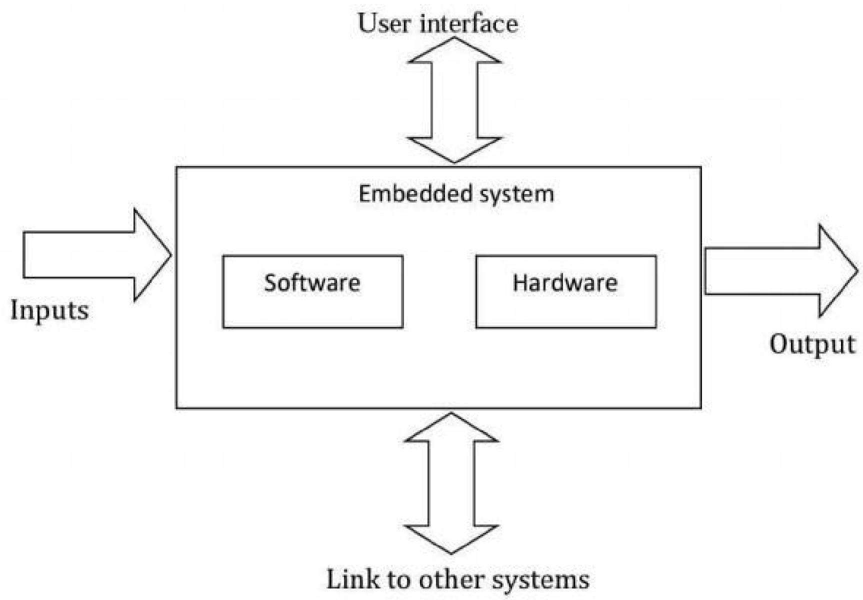

FIGURE 1. Overview of Embedded System

Embedded System**:** It includes mainly two sections, they are 1. Hardware 2. Software

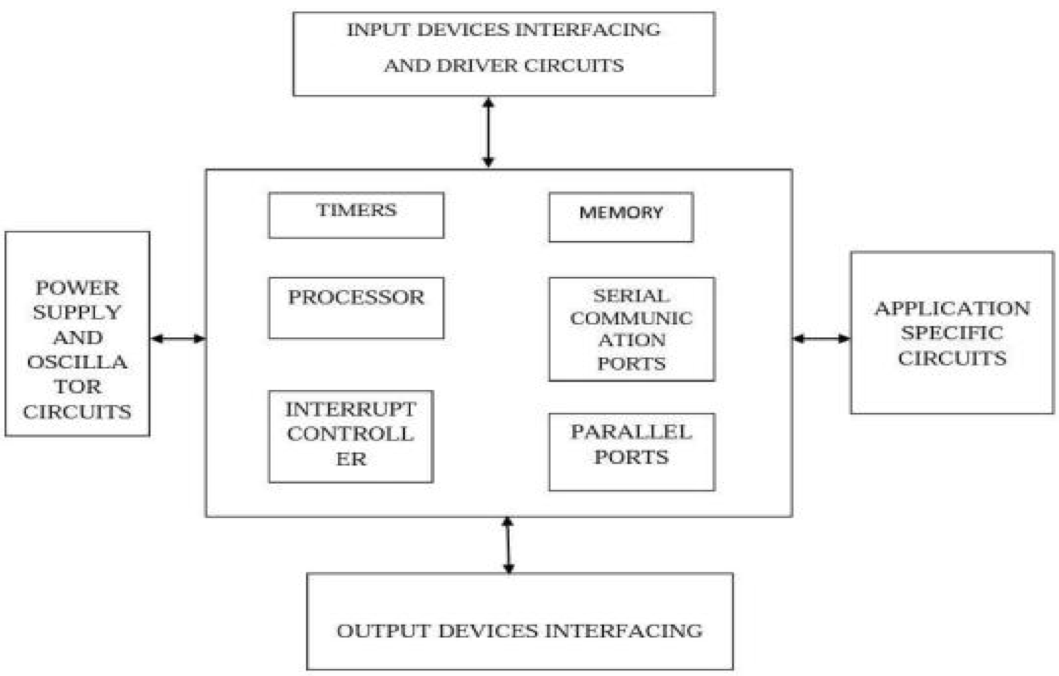

FIGURE 2. Block Diagram of Embedded System

Embedded System Hardware

As with any electronic system, an embedded system requires a hardware platform on which it performs the operation. Embedded system hardware is built with a microprocessor or microcontroller. The embedded system hardware has elements like input output (I/O) interfaces, user interface, memory and the display. Usually, an embedded system consists of:

- Power Supply

- Processor
- Memory
- Timers
- Serial communication ports
- Output/Output circuits
- System application specific circuits

Embedded System Software: The embedded system software is written to perform a specific function. It is typically written in a high-level format and then compiled down to provide code that can be lodged within anon-volatile memory within the hardware. An embedded system software is limited to:

- Availability of system memory
- Availability of processor's speed
- When the system runs continuously,

There is a need to limit power dissipation for events like stop, run and wake up.

Bringing Software and Hardware Together for Embedded system

To develop software for embedded systems, it's essential to integrate both software and hardware components. This involves programming a microprocessor or microcontroller—a hardware device—with the source code, enabling it to perform all embedded system tasks as instructed by the code. Typically, source code for embedded systems is written in assembly language; however, processors can only execute binary files. Converting source code into an executable binary image involves three main stages:

1. Each source file must first be compiled or assembled into an object file.
2. These object files are then linked together to create a single object file known as the relocatable program.
3. The final step, called relocation, involves assigning actual physical memory addresses to the relative locations in the re-locatable program.

The output of this process is a binary executable file, ready to be loaded and executed on the embedded hardware.

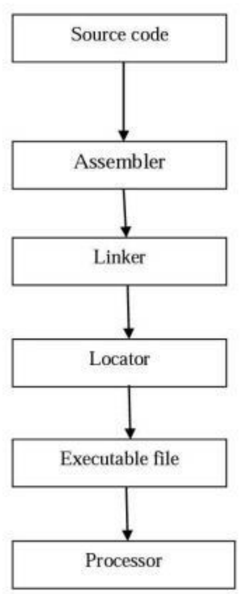

FIGURE 3. Flow of burning source code to Processor

Existing Methods: Conventional irrigation techniques such as manual irrigation, fixed sprinkler systems and flood irrigation often result in inefficient water use and significant wastage. These methods do not take into account real-time soil moisture levels, which can lead to over-watering – causing water shortages and crop damage – or under-watering, which hinders plant growth and yield. Furthermore, traditional systems lack automation and require constant human supervision, which can be labor-intensive and time-consuming. The lack of sensors to monitor key environmental parameters such as temperature, humidity and soil pH further limits their effectiveness in precision agriculture. In addition, fixed irrigation systems cannot adapt to changing field conditions, and physical obstacles can block water delivery. These limitations highlight the need for a more advanced, automated irrigation solution that uses real-time data to save water, reduce manual labor and increase agricultural productivity.

Proposed Method: The proposed smart irrigation system using Agri Bot addresses these shortcomings by combining automation and real-time environmental monitoring for efficient water management. Built on a mobile robotic platform and powered by a Raspberry Pi, the system autonomously navigates the field, using an ultrasonic sensor to detect and avoid obstacles. A servo motor-controlled soil moisture sensor measures moisture levels at various locations, and when low moisture is detected, a relay activates a water pump to precisely irrigate the area. The system includes a DHT11 sensor for monitoring temperature and humidity, and a pH sensor for assessing soil pH – providing critical insights for precision farming. Powered by a 12V battery and remotely controlled via Bluetooth, the system automates irrigation based on real-time soil and environmental data. This approach saves water, reduces the need for manual intervention, and improves farming efficiency, making it a smart and sustainable solution for modern farming practices.

3. BLOCK DIAGRAM

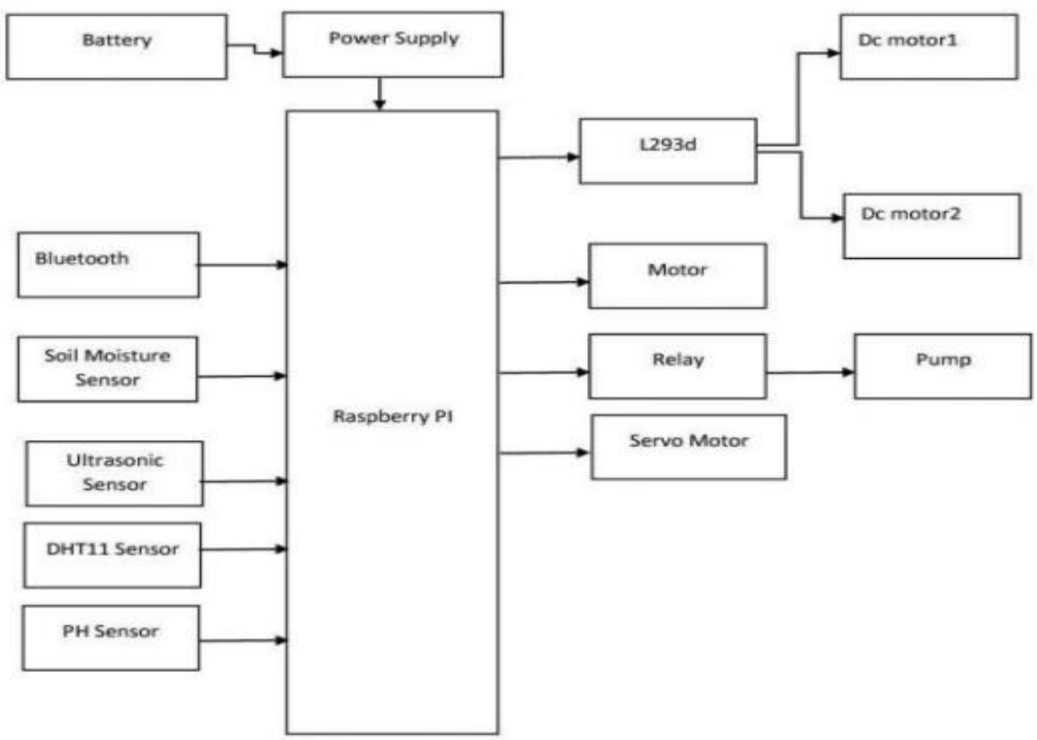

FIGURE 4. Block Diagram of Smart Irrigation System using Agribot

Block Diagram Description

The block diagram illustrates a Raspberry Pi-based Smart Agriculture System powered by a battery. This battery is connected to a power supply unit that ensures regulated power delivery to all system components.

Input Components (Sensors and Communication):

- **Bluetooth Module:** Facilitates wireless communication with external devices such as smartphones or computers.
- **Soil Moisture Sensor:** Detects the moisture content present in the soil.
- **Ultrasonic Sensor:** Measures distance, typically used for obstacle detection or monitoring levels.
- **DHT11 Sensor:** Gathers data on ambient temperature and humidity.
- **pH Sensor:** Checks the pH value of soil or irrigation water.

All input sensors are connected to the Raspberry Pi, which functions as the central processing and control unit of the system.

Output Components (Actuators):

- L293D Motor Driver: Interfaces with the Raspberry Pi to control two DC motors:
  - DC Motor 1
  - DC Motor 2
- Relay Module: Managed by the Raspberry Pi to switch a water pump on or off.
- Motor: May serve other mechanical functions such as activating sprayers or operating valves.
- Servo Motor: Provides accurate control, potentially for positioning sensors or rotating devices like a camera.

System Functionality Overview:

The Raspberry Pi collects real-time environmental data through the connected sensors. Based on this data and user commands received via the Bluetooth module, it can execute actions such as:

- Activating or deactivating the water pump
- Navigating the robotic system using DC motors

4. BRIEF DESCRIPTION OF SYSTEM ON CHIP (SOC)

A System on Chip (SoC) is a highly integrated circuit that combines several critical components of a computer or electronic system onto a single chip. These typically include a Central Processing Unit (CPU), Graphics Processing Unit (GPU), RAM, storage controllers, power management units, and a variety of input/output interfaces such as USB, Wi-Fi, and Bluetooth. SoCs are commonly found in devices like smartphones, tablets, Internet of Things (IoT) gadgets, and embedded systems due to their small form factor, energy efficiency, and strong performance capabilities. Unlike traditional setups that use multiple separate chips, SoCs help reduce power usage, cut manufacturing costs, and boost processing speed, making them well-suited for today's computing demands. Notable examples of SoCs include the Broadcom BCM2711 in the Raspberry Pi 4, Qualcomm Snapdragon chips in smartphones, and Apple's M-series processors in MacBooks.

FIGURE 5. System on chip (SoC)

PH Sensor: A pH sensor, also referred to as a pH probe or electrode, is an instrument used to determine the acidity or alkalinity of a solution. The pH value reflects the concentration of hydrogen ions and ranges from 0 to 14, with 7 representing a neutral state. Values below 7 indicate an acidic solution, while values above 7 indicate an alkaline one. pH sensors are widely utilized in scientific laboratories, various industries, and environmental monitoring due to their importance in assessing chemical properties.

Working Principle: PH sensors function based on electrochemical principles. They typically include two main components: a glass electrode and a reference electrode. The glass electrode reacts to the hydrogen ion concentration in the solution, while the reference electrode provides a constant potential. The voltage difference between these two electrodes is measured to determine the pH level.

Key Components:

- Glass Electrode: Acts as the sensing element, specifically responsive to hydrogen ions. It produces a voltage that correlates with the pH of the solution.
- Reference Electrode: Offers a stable reference voltage for accurate comparison. A commonly used type is the silver/silver chloride (Ag/AgCl) electrode.

Calibration:
To maintain measurement accuracy, pH sensors require regular calibration. This process involves placing the sensor in buffer solutions with known pH values and adjusting the sensor's readings accordingly. A two-point calibration using two different buffer solutions is typically employed.
Maintenance:
Routine maintenance is critical to the sensor's reliability. This includes gently cleaning the sensor with a mild detergent, rinsing it with distilled water, and storing it in an appropriate storage solution when not in use. Over time, the electrodes may degrade and require replacement to maintain consistent performance.

Applications: PH sensors are utilized across a wide range of industries such as water treatment, food and beverage manufacturing, pharmaceuticals, agriculture, environmental monitoring, and scientific research laboratories. They play a crucial role in monitoring and regulating pH levels during processes, maintaining product quality, and evaluating environmental parameters.

Types of pH sensors: Combination pH electrodes: Combine both the glass and reference electrodes into a single probe. Separate pH electrodes: Combine the glass and reference electrodes as separate entities and connect them with a cable. Online monitoring: pH sensors are used for online monitoring in industrial processes. They are integrated into control systems to maintain optimal pH conditions for various applications. Advanced pH sensors: Advances in sensor technology include the development of solid-state pH sensors, which eliminate the need for a liquid electrolyte and provide improved stability and durability.

5. SOFTWARE REQUIREMENTS

Raspbian is the officially recommended operating system for general use on the Raspberry Pi. It is a free, Debian-based OS specifically optimized for Raspberry Pi hardware. Raspbian includes over 35,000 precompiled packages—organized for easy installation—making it highly accessible and user-friendly. As a community-driven project, Raspbian is actively maintained with a focus on enhancing the stability and performance of as many Debian packages as possible. Steps to Install Raspbian OS: Step 1: Download the Raspbian operating system. Step 2: Extract the image file. The Raspbian image is compressed and must be unzipped before use. It uses the ZIP64 format, which may require specific tools if your system's built-in utilities are outdated. If you encounter issues during extraction, consider using the tools recommended by the Raspberry Pi community. For Windows users, it's best to use 7-Zip to extract the Raspbian image file. Mac users should opt for The Unarchiver, while Linux users can simply use the built-in Unzip tool. Step 3: Write the Disk Image to Your microSD Card Insert your microSD card into your computer, and use a suitable application to write the Raspbian image onto it: Windows users can use Win32 Disk Imager. Mac users may utilize the built-in Disk Utility. Linux users, as well as others, are recommended to use Etcher, which is compatible with all major operating systems. While the steps might vary slightly between programs, the overall process is straightforward. Select the target device (make sure it's your microSD card) and the unzipped Raspbian image file. Once confirmed, initiate the write process. Connect all necessary peripherals and the power supply. Upon booting, the latest version of Raspbian will launch directly into the desktop environment. The default login credentials are: Username: pi Password: raspberry.

6. PYTHON

Python is a high-level, general-purpose, dynamically typed, and interpreted programming language. It supports object-oriented programming, making it suitable for developing various types of applications. Python is known for its simplicity, readability, and ease of learning, while offering a wide range of built-in high-level data structures. Its clear syntax and dynamic typing, along with its interpreted execution, make it especially well-suited for scripting and rapid application development. Python accommodates multiple programming paradigms, including object-oriented, procedural, and functional styles. Although it is not limited to any specific domain like web development, it is considered a multipurpose language due to its applicability in areas such as web development, enterprise applications, and 3D CAD. Since Python uses dynamic typing, variables do not need explicit type declarations. For example, you can assign a value with a = 10 without declaring the data type. This feature, combined with the absence of a compilation step, allows for a fast development cycle with quick editing, testing, and debugging.

7. RESULTS

The Agribot system efficiently activates the water pump and delivers water accurately to areas where it is needed.

- Water Conservation:
- The system demonstrated a 30–40% decrease in water usage compared to conventional irrigation methods by utilizing real-time soil moisture data to guide watering.
- Water was applied specifically to dry zones, minimizing excess use and runoff.
- Enhanced Crop Health:
- Crops irrigated using the Agribot exhibited more uniform growth and improved yield stability.
- There was a 10–15% increase in crop yield compared to plots irrigated manually.
- Reduced Labor Dependency:
- Irrigation-related manual work was reduced by over 70%, allowing labor resources to be redirected to other agricultural activities.

FIGURE 6. Smart Irrigation System Using Agribot

8. ADVANTAGES AND APPLICATIONS

Benefits:

- High Accuracy
- Automated Operation
- Improved Efficiency
- Dependable Performance
- Environmentally Sustainable
- User-Friendly
- Mobile Functionality
- Flexible and Adaptable
- Precision Control
- Optimized Resource Use

Use Cases:

- Farming and Agriculture
- Robotic Applications
- Obstacle Avoidance
- Water Resource Management

- Industrial Process Automation
- Intelligent Irrigation Systems
- Environmental Data Monitoring
- Surveillance and Security
- Navigation Technologies
- Self-Driving and Autonomous Vehicles

9. CONCLUSION

The Smart Irrigation System using Agribot provides an advanced and automated alternative to conventional irrigation methods by incorporating real-time monitoring and precise control. Leveraging components like the Raspberry Pi, soil moisture sensors, a pH sensor, and a DHT11 sensor, the system ensures efficient water usage and maintains optimal growing conditions for crops. Features such as obstacle detection, Bluetooth-enabled remote control, and automated water distribution enhance the system's overall effectiveness by minimizing water waste and reducing the need for manual labor. This innovative solution not only boosts agricultural productivity but also promotes sustainable farming through resource conservation and data-informed decision-making. As technology continues to advance, smart irrigation systems like this one will be essential in tackling global agricultural issues and supporting food security.

10. FUTURE SCOPE

The future potential of smart irrigation systems using Agribots powered by Raspberry Pi lies in significant improvements in water conservation and agricultural efficiency. With the integration of advanced sensors that monitor soil moisture, temperature, and weather conditions, these systems can optimize irrigation schedules, ensuring crops receive the appropriate amount of water at the right time, thereby minimizing water waste. Solar power can be used to operate these systems, making them energy-efficient and suitable for remote areas. Additionally, integration with cloud-based platforms enables remote monitoring and control, while the system's scalability allows it to be tailored for various crop types and field sizes. This technology will enable farmers to make informed, data-driven decisions, promoting more sustainable farming practices.

REFERENCES

[1]. Amrita Sneha. A, Abirami. E, Ankita. A, Mrs. R. Praveen, Mrs. R. Srimeena, "Agricultural Robot for Automatic Ploughing and Seeding," 2015 IEEE International Conference on Technological Innovations in ICT (TIAR 2015).

[2]. P. Usha, V. Maheswari, Dr. V. Nandagopal, "Design and Implementation of Seeding Agricultural Robot," JIRAS.

[3]. Abdullah Tanveer, Abhishek Choudhary, Divya Pal, Rajani Gupta, Farooq Husain, "Automated Farming Using Microcontroller and Sensors," IJSRMS, ISSN: 2349-3371.

[4]. Nikesh Gondchawar, Prof. Dr. R. S. Kawitkar, "IOT Based Smart Agriculture," IJARCCE, June 2016.

[5]. Sachin D. Khirade, A. B. Patil, "Plant Disease Detection Using Image Processing," 2015 International Conference on Computing, Communication, Control, and Automation, IEEE.

[6]. Vijai Singh, Varsha, A. K. Mishra, "Detection of Unhealthy Regions of Plant Leaves Using Image Processing and Genetic Algorithm," ICA CEA, India, 2015.

[7]. Monica Jhuria, Ashwani Kumar, Rushikesh Borse, "Image Processing for Smart Farming, Detection of Disease, and Fruit Grading," Proceedings of the 2013 IEEE Second International Conference on Image Information Processing.

[8]. Mrunalani R. Badnakhe, Prashant R. Deshmukh, "Infected Leaf Analysis and Comparison by Otsu Threshold and K-Means Clustering," International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 3, March 2012.

[9]. C. Sridhathan and M. Senthil Kumar, *"Plant Infection Detection using Image Processing",* International Journal of Modern Engineering Research (IJMER). (Vol.8, Issue 7, 2018, PP 13-16, ISSN: 2249 – 6645).

[10]. Amit Gupta, M. Senthil Kumar, M. Raman Kumar and D. Hemanth Kumar, "Deep Learning Technique Used for Tomato and Potato Plant Leaf Disease Classification and Detection", 2023 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES), July 2023.

[11]. Amit Gupta, Abha Dargar, Abdul Majid, Shashi Kant Dargar, M. Senthil Kumar and M. Raju, "Markov Chain Model Used in Agricultural Yield Predictions Utilizing on Indian Agriculture", 2023 IEEE World Conference on Applied Intelligence and Computing (AIC), July 2023.

[12]. C.I. Vimalarani and M. Senthil Kumar, "Energy Efficient PCP Protocol for k-Coverage in Sensor Networks", *IEEE International Conference on Computational Intelligence and Computing Research, IEEE Proceedings, 2010.*

[13]. Senthilkumar Meyyappan, A. Bharath Naik, A. Uma Sai and Ch. Keerthi, "Improving Weather Forecasting Accuracy Using Machine Learning", *Journal on Electronic and Automation Engineering,* Vol. 2(4), December 2023, pp. 9-18.

[14]. Senthilkumar Meyyappan, K. Susmitha, K. Vaishnavi and M. Sai Rao, "Condition Based Monitoring and Maintenance System for Underground Metro Stations", *Journal on Electronic and Automation Engineering*, Vol. 4(1), March 2025, pp. 175-182.

[15]. Senthilkumar Meyyappan and N. Selvamuthukumaran, "Network Selection in Heterogeneous Wireless Systems using GRA Method", *Journal on Electronic and Automation Engineering*, Vol. 4(1), March 2025, pp. 127-132.

[16]. M. Senthil Kumar and C. Sridhathan, *"Impact of Mobility on the Routine of Enhanced – DSDV Protocol in Mobile Ad-hoc Networks",* International Journal of Applied Engineering Research (IJAER). (Vol.13, No.14, 2018, PP 11674-11679, ISSN: 0973-4562).

[17]. M. Senthil Kumar, R. Karthik and I. Rabeek Raja, *"An Efficient Approach for Increasing Power Optimization in Mobile Ad-Hoc Networks"*, International Journal of Engineering Research and Technology (IJERT). (Vol.3, Issue 2, February 2014).

[18]. M. Senthil Kumar, "Energy Efficient Techniques for Transmission of Data in Wireless Sensor Networks", *Journal of Computing Technologies (JCT)*. (Vol.5, Issue 2, February 2016, ISSN: 2278 – 3814).

[19]. M. Kavitha, T. Maheshwaran and M. Senthil Kumar, "Secure Routing in MANETs with Key Management", *International Journal on Engineering Technology and Sciences (IJETS).* (Vol.1, Issue 6, October 2014, ISSN (P): 2349 – 3968, ISSN (O): 2349 - 3976).

[20]. Senthilkumar Meyyappan, Kalyan Kasturi, G. Vijaya Lakshmi, J. Srinija Reddy and K. Grace Sampoorna, *"Improvement of LEACH Protocol for Enhancing Features of WSN",* Journal on Electronic and Automation Engineering, Vol. 2(4), December 2023, pp. 19-26.

[21]. K. Arutselvan, C. Sridhathan and M. Senthil Kumar *"Unlocking Mobile Devices using Improved Face Recognition and Eye Blinking Technique",* International Journal of Applied Engineering Research (IJAER). (Vol.13, No.24, 2018, PP 16907-16909, ISSN: 0973-4562).

[22]. M. Senthil Kumar and Ashish Chaturvedi, "Energy-Efficient Coverage and Prolongs for Network Lifetime of WSN using MCP*", European Journal of Scientific Research (EJSR).* (Vol.95, No.2, January 2013, ISSN: 1450 – 216X / 1450 – 202X).

[23]. M. Senthil Kumar and Ashish Chaturvedi, "A Novel Enhanced Coverage Optimization Algorithm for Effectively Solving Energy Optimization Problem in WSN", *Research Journal of Applied Sciences, Engineering and Technology (RJASET).* (Issue 4, Vol.7, January 2014, ISSN: 2040 – 7459 & e-ISSN: 2040 – 7467).

[24]. M. Kavitha, T. Maheshwaran and M. Senthil Kumar, "Ensure Data Transmission in Mobile Ad-Hoc Networks", *International Journal on Engineering Technology and Sciences (IJETS)*. (Vol.2, Issue 4, April 2015, ISSN (P): 2349 – 3968, ISSN (O): 2349 - 3976).

[25]. Senthilkumar Meyyappan, G. Lava Kumar, G. Niharika and G. Chakradhar, "Cellular Network Signal Strength Analyser", *Journal on Electronic and Automation Engineering*, Vol. 4(1), March 2025, pp. 165-174.

[26]. M. Senthil Kumar and M. Gopinath, "An Efficient Polynomial Pool-Based Scheme for Distributed Heterogeneous WSNs", *International Journal of Modern Engineering Research (IJMER)*. (Vol.3, Issue 6, Nov-Dec.2013, PP 3328-3335, ISSN: 2249-6645).

[27]. M. Senthil Kumar and L. Praveen, "An Assuring Approach for Tree-Based Routing Topology in WSNs", *International Journal of Emerging Trends in Engineering and Development (IJETED).* (Issue 3, Vol.6, November 2013, ISSN: 2249 – 6149).

[28]. R. Kathiresh, V.M. Ramprasath and M. Senthil Kumar, *"A Systematic Approach for Design of Compressed Test Data in SOC"*, CiiT International Journal of Software Engineering and Technology. (Vol.4, No.4, Issue: April 2012, Print: ISSN 0974 – 9748 & Online: ISSN 0974 – 9632).

