© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716107

Automatic Soil Irrigation System

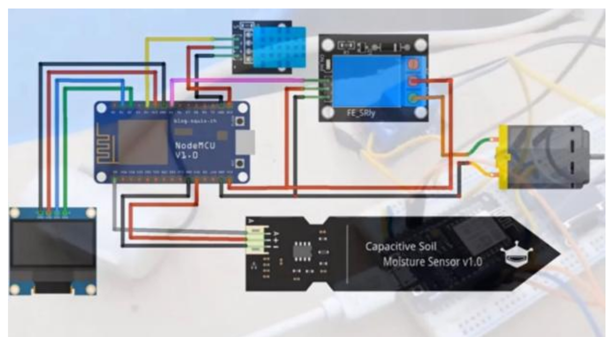
PRANAV A. CHAVAN[1], VISHAL N. RATHOD[2], TEJAS V. MORE[3], KUNAL K. RAJWADE[4], S. S. SATHE[5]
[1,2,3,4,5]*JSPM's RSCOE Polytechnic*

Abstract- Efficient water management is a critical challenge in modern agriculture due to increasing water scarcity and the need for sustainable farming practices. The Automatic Soil Irrigation System is designed to optimize water usage by delivering water to crops based on real-time soil moisture conditions. This system eliminates the need for manual irrigation and reduces water wastage, thereby improving crop productivity and resource efficiency. The system primarily consists of soil moisture sensors, a microcontroller (such as Arduino or similar embedded systems), a water pump, and a control unit. The soil moisture sensor continuously monitors the moisture level in the soil and sends the data to the microcontroller. When the moisture level falls below a predefined threshold, the controller automatically activates the irrigation pump to supply water to the soil. Once the desired moisture level is reached, the system switches off the pump, ensuring optimal water usage. Additionally, advanced versions of the system may incorporate features such as wireless communication, mobile monitoring, weather forecasting integration, and solar power support for enhanced efficiency and sustainability. These features enable remote monitoring and control, making the system suitable for smart farming applications. The Automatic Soil Irrigation System not only conserves water but also reduces human labor and minimizes the risk of over-irrigation or under-irrigation. It plays a significant role in promoting precision agriculture and can be effectively implemented in gardens, farms, and greenhouse environments. Overall, this system provides a cost-effective, reliable, and eco-friendly solution to modern irrigation challenges.

I. INTRODUCTION

Agriculture plays a vital role in the economy and food production of any country. It is the backbone of many developing nations and provides employment to a large portion of the population. The growth and productivity of crops mainly depend on proper irrigation, soil fertility, and environmental conditions. Among these, water is one of the most essential resources required for plant growth. Without adequate and timely water supply, crops cannot achieve their full yield potential.

In traditional farming practices, irrigation is mostly done manually or by using conventional methods such as flood irrigation. In manual irrigation, farmers have to monitor the condition of the soil regularly and supply water based on their observation and experience. This process is time-consuming, labor-intensive, and often inaccurate. On the other hand, flood irrigation leads to excessive use of water, uneven distribution, and wastage of valuable resources. It can also cause problems such as soil erosion, waterlogging, and nutrient loss.



1.2 PROBLEM STATEMENT

Agriculture is highly dependent on proper irrigation, and water management plays a crucial
role in determining crop productivity. In traditional irrigation systems, water is supplied to crops without accurate knowledge of soil moisture conditions. This leads to inefficient use of water and creates several problems for farmers as well as the environment.

One of the major issues in conventional irrigation methods is over-irrigation. Farmers often supply excess water to ensure that crops do not suffer from dryness. However, this practice leads to wastage of a large amount of water. Over-irrigation also causes soil erosion, where the top fertile layer of soil is washed away. This reduces soil fertility and affects plant growth. Additionally, excessive water can lead to nutrient leaching, where essential nutrients are washed away from the root zone, making them unavailable to plants.

On the other hand, under-irrigation is another serious problem. In many cases, farmers are unable to supply sufficient water due to lack of proper monitoring or limited water resources. This results in dry soil conditions, poor root development, and reduced crop yield. Plants may become weak and more susceptible to diseases. In extreme cases, crops may fail completely, leading to economic losses for farmers.

Another major challenge is the dependency on manual labor. Traditional irrigation systems require farmers to physically visit the fields, check soil conditions, and operate water pumps.

This process is time-consuming and labor-intensive. In large agricultural fields, it becomes very difficult to monitor all areas properly. Farmers may not always be available to irrigate

1.3 LITERATURE REVIEW

A detailed study of various research papers has been carried out to understand the design, performance, and feasibility of hybrid renewable energy systems. The literature review helps in identifying suitable technologies, materials, and design approaches for the proposed system.

1. Design and Performance Analysis of Savonius Vertical Axis Wind Turbine

Author: R. Gupta and R. Das
Year: 2021
This research focuses on the aerodynamic performance of the Savonius Vertical Axis Wind Turbine (VAWT), especially under low wind speed conditions. The Savonius turbine is widely known for its simple construction and ability to operate in any wind direction, making it suitable for urban and low-wind regions.
The study highlights that the performance of the turbine largely depends on parameters such as blade shape, overlap ratio, and number of stages. It was observed that a two-stage rotor design significantly improves torque generation compared to a single-stage rotor. The overlap ratio of 0.15 was found to be optimal, allowing a small amount of air to pass between the blades, which reduces negative drag and improves efficiency.

Another important finding is the self-starting capability of the turbine. The Savonius turbine can start rotating at wind speeds as low as 3 m/s, which makes it highly suitable for decentralized and small-scale power generation.

2. Performance Evaluation of Hybrid Solar-Wind Power Systems

Author: S. Kumar and V. Tyagi
Year: 2020
This study examines the advantages of combining solar and wind energy into a hybrid system. It explains that solar and wind energy sources are complementary in nature. Solar energy is available during sunny daytime conditions, whereas wind energy is often stronger during night, monsoon, and winter seasons. The researchers conducted seasonal analysis and found that hybrid systems can provide more consistent power output throughout the year. The system used a battery storage unit and charge controller to manage energy from both sources effectively.

The study reported a 25% increase in reliability compared to standalone solar systems. It also reduced the chances of power failure, especially in off-grid areas.

3. Optimization of Gear Ratio for Small Wind Turbines

Author: M. Al-Abadi and J. Thompson
Year: 2019
This paper focuses on improving the efficiency of small wind turbines by optimizing the gear ratio. One of the main challenges in small-scale turbines is the mismatch between the low rotational speed of the turbine and the high speed required by generators.
The study tested different mechanical transmission systems and found that a belt and pulley system with a 5:1 ratio provides the best balance between speed increase and energy loss.

This system helps in converting low-speed rotation into higher speed required for electricity generation.
The researchers also observed that this method improves generator performance without significantly increasing mechanical complexity or cost.

A properly designed gear ratio is essential for efficient energy conversion in small wind
turbines.

4. Design of Portable Charging Stations for Urban Spaces

Author: L. Chen and H. Wong
Year: 2022
This research focuses on the structural and functional design of portable charging stations. The study emphasizes the importance of using strong and durable materials such as mild steel for outdoor installations. The authors also highlight the need for modern charging technologies such as USB ports and Type-C Power Delivery (PD). These features allow fast charging of modern electronic devices like smartphones and tablets.

The study also considers environmental factors such as weather resistance, durability, and user safety.
Charging stations must be designed with strong materials and modern charging features to ensure reliability and user convenience.

5. Efficiency of Overlap in Savonius Rotor

Author: B. Wahyudi and S. Soekardi
Year: 2018
This research analyzes the effect of blade overlap in Savonius turbines. The overlap between blades allows air to pass through the center, reducing resistance on the returning blade.JSPM'S R.S.C.O.E. POLYTECHNIC TATHAWADE. The study found that proper overlap improves airflow and increases the power coefficient

1.4 SUMMARY
In this chapter, a detailed explanation of all the hardware components used in the Automatic Soil Irrigation System has been provided. Each component plays a vital role in the overall functioning of the system, and their proper selection and integration are essential for achieving desired results.

The soil moisture sensor acts as the input device, continuously monitoring the condition of the soil and providing real-time data. The Arduino controller processes this data and makes decisions based on the programmed logic. The relay module acts as an interface between low-power control signals and high-power devices, ensuring safe and efficient operation.

The water pump serves as the output device, delivering water to the plants when required. The integration of these components creates an intelligent system capable of automatic decision-making. This automation reduces the need for manual labor and minimizes human error. Farmers or users do not need to constantly check soil conditions, as the system performs this task efficiently.

One of the major advantages of this system is water conservation. Water is supplied only when the soil moisture level falls below a certain limit, avoiding unnecessary wastage. This is especially important in regions facing water scarcity. In addition, the system helps in maintaining optimum soil moisture, which leads to better plant growth and increased agricultural productivity

1.5 MECHANICAL METHODOLOGY

System Design
The system is designed using a soil moisture sensor, Arduino microcontroller, relay module, and water pump. The sensor is placed in the soil to measure moisture level.
Circuit Development
All components are connected properly. The sensor sends signals to the Arduino, and the relay controls the pump.

Programming
The Arduino is programmed to read sensor values and control the pump based on predefined moisture levels.

Implementation

The system is installed in soil, and real-time readings are taken to observe the working of the system The system is designed using a soil moisture sensor, Arduino microcontroller, relay module, and water pump. The sensor is placed in the soil to measure moisture.

- Design Parameters

```
#include <WiFi.h>
#include <WebServer.h>
#include <HardwareSerial.h> // usually not needed, but explicit is fine
// ────────────────────────────────────────────────
// WiFi credentials
const char* ssid = "Teja";
const char* password = "123456789";
// ────────────────────────────────────────────────
// For ESP32 most people use: GPIO17 = RX, GPIO16 = TX (or vice versa)
// You can also use GPIO4 & GPIO5, etc. Just avoid pins used by flash/boot
#define RX_PIN 16
#define TX_PIN 17
WebServer server(80);
HardwareSerial& arduinoSerial = Serial2; // Most readable way on ESP32
// Latest values
int moisture = 0;JSPM'S R.S.C.O.E.POLYTECHNIC TATHAWADE.
39
bool pumpOn = false;
unsigned long lastArduinoMsg = 0;
// ────────────────────────────────────────────────
// Very compact & fast HTML + CSS + JS (unchanged)
const char INDEX_HTML[] PROGMEM = R"=====(
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Soil Monitor</title>
<style>
body {
margin:0; padding:0; font-family: system-ui, sans-serif;
background: #0f0f17; color: #e0e0ff; min-height:100vh;
display:flex; flex-direction:column; align-items:center;
justify-content:center;
}
h1 {
font-size: 2.1rem; margin: 20px 0 8px; color: #aaffaa;
text-shadow: 0 0 10px #4f4;
}
card {
background: #1a1a2e; border-radius: 16px;
padding: 24px 32px; width: 90%; max-width: 360px;
box-shadow: 0 8px 30px rgba(0,0,0,0.6);
text-align: center;JSPM'S R.S.C.O.E.POLYTECHNIC TATHAWADE.
40
}
.value {
font-size: 4.8rem; font-weight: 700; margin: 12px 0;
line-height: 1;
}
.moisture { color: #55ff99; }
.status { font-size: 1.4rem; margin: 16px 0 8px; font-weight: 500; }
.pump { font-size: 1.3rem; margin-top: 12px; }
.pump.on { color: #ff5555; font-weight: bold; }
.pump.off { color: #88ff88; }
.info { color: #8888aa; font-size: 0.95rem; margin-top: 24px; }
.error { color: #ff7777; }
</style>
</head>
<body>
<h1>Soil Irrigation</h1>
<div class="card">
<div class="value moisture" id="moisture">--</div>
<div class="status" id="status">Waiting for data...</div>
<div class="pump" id="pump">Pump: —</div>
</div>
```

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716107

```
<div class="info" id="info">Updating every
2s</div>
<script>
const moistureEl =
document.getElementById('moisture');
const statusEl = document.getElementById('status');
const pumpEl = document.getElementById('pump');
const infoEl =
document.getElementById('info');JSPM'S
R.S.C.O.E.POLYTECHNIC TATHAWADE.
41
function update() {
fetch('/data')
then(r => r.json())
then(d => {
moistureEl.textContent = d.m;
pumpEl.textContent = "Pump: " + (d.p ? "ON" :
"OFF");
pumpEl.className = "pump " + (d.p ? "on" : "off");
if (d.m < 30) {
statusEl.textContent = "LOW MOISTURE –
WATERING";
statusEl.style.color = "#ff6666";
} else {
statusEl.textContent = "Soil is OK";
statusEl.style.color = "#88ff88";
}
infoEl.textContent = "Last update: just now";
infoEl.className = "info";
})
catch(() => {
infoEl.textContent = "No connection to ESP";
infoEl.className = "info error";
});
}
update();
setInterval(update, 2000);
</script>
</body>
</html>JSPM'S R.S.C.O.E.POLYTECHNIC
TATHAWADE.
42
)=====";
//
────────────────────────────────────────
──────────
void handleRoot() {
server.send_P(200, "text/html", INDEX_HTML);
}
void handleData() {
String json = "{\"m\":" + String(moisture) +
",\"p\":" + (pumpOn ? "true" : "false") + "}";
server.send(200, "application/json", json);
}
//
────────────────────────────────────────
──────────
void setup() {
Serial.begin(115200);
delay(100);
Serial.println("\nSoil Monitor starting... (ESP32)");
// UART2 — most common free pins: RX=16,
TX=17
arduinoSerial.begin(9600, SERIAL_8N1, RX_PIN,
TX_PIN);
// Alternative syntax: arduinoSerial.begin(9600); if
using default UART2 pins
WiFi.mode(WIFI_STA);
WiFi.begin(ssid, password);
Serial.print("Connecting to WiFi ");
unsigned long t0 = millis();
while (WiFi.status() != WL_CONNECTED) {
delay(400);
Serial.print(".");
if (millis() - t0 > 15000) {
Serial.println("\nWiFi timeout → rebooting in
3s...");JSPM'S R.S.C.O.E.POLYTECHNIC
TATHAWADE.
43
delay(3000);
ESP.restart();
}
}
Serial.println("\nConnected! IP = " +
WiFi.localIP().toString());
server.on("/", handleRoot);
server.on("/data", handleData);
server.begin();
Serial.println("HTTP server started");
Serial.println("Open in browser: http://" +
WiFi.localIP().toString());
}
void loop() {
server.handleClient();
// ── Non-blocking read from Arduino
────────────────────────────────────────
─
while (arduinoSerial.available()) {
```

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716107

```
String line = arduinoSerial.readStringUntil('\n');
line.trim();
if (line.length() == 0) continue;
// Accept only clean numeric values 0–100
if (line.toInt() == 0 && line != "0") continue; // reject garbage
int val = line.toInt();
if (val >= 0 && val <= 100) {
moisture = val;
lastArduinoMsg = millis();
// Same threshold logic as Arduino side (assumed)
pumpOn = (val < 30);JSPM'S R.S.C.O.E.POLYTECHNIC TATHAWADE.
44
Serial.printf("Moisture: %d%% Pump: %s\n", moisture, pumpOn ? "ON" : "OFF");
}
}
// Optional: timeout warning (once)
static bool warned = false;
if (millis() - lastArduinoMsg > 12000UL && !warned) {
Serial.println("Warning: No valid data from Arduino for >12 seconds");
warned = true;
}
}
```

1.6 Working:

The Automatic Soil Irrigation System operates based on real-time monitoring of soil moisture and automatic control of water supply. Its functioning involves sensing, processing, and actuation, which together ensure efficient irrigation.

1. Soil Moisture Detection

The system uses a soil moisture sensor placed in the soil near plant roots. This sensor measures the water content in the soil by detecting electrical conductivity or resistance.

- When the soil is dry, resistance is high (low moisture level).
- When the soil is wet, resistance is low (high moisture level).

The sensor continuously sends analog or digital signals representing moisture levels.

2. Signa Processing (Microcontroller Unit)

The sensor output is sent to a microcontroller (such as Arduino). The microcontroller:

- Reads the moisture data from the sensor
- Compares it with a predefined threshold value
- Decides whether irrigation is required or not

The threshold value is set based on the type of crop and soil requirements.

3. Decision Making

- If the soil moisture is below the threshold (dry condition), the system determines that irrigation is needed.
- If the moisture is above the threshold (sufficient water), no action is taken.

4. Activation of Water Pump

When irrigation is required:

- The microcontroller sends a signal to a relay module
- The relay acts as a switch and turns ON the water pump or solenoid valve
- Water starts flowing into the field or plant area

5. Automatic Cut-off

As water is supplied:

- The soil moisture level gradually increases
- The sensor continuously monitors this change
- Once the moisture reaches the desired level, the microcontroller turns OFF the pump via the relay

This prevents overwatering.

6. Continuous Monitoring

The system works in a loop:

- It continuously checks soil moisture at regular intervals
- Automatically controls irrigation without human intervention

7. Optional Advanced Features

In advanced systems, additional components improve functionality:

- LCD display: Shows moisture level and system status
- Wi-Fi/Bluetooth module: Enables remote monitoring via mobile apps
- Rain sensor/weather data: Prevents irrigation during rainfall

- Solar power supply: Makes the system energy-efficient and eco-friendly

1.7 RESULT:

The Automatic Soil Irrigation System was successfully designed and implemented, and its performance was tested under different soil moisture conditions. The system responded accurately to variations in soil moisture levels and controlled the irrigation process effectively.

During testing, it was observed that:

- When the soil moisture level dropped below the preset threshold, the system automatically activated the water pump and supplied water to the soil.
- As the moisture level increased and reached the desired value, the system turned OFF the pump without any delay.
- The system operated continuously and reliably without the need for manual intervention.

The results demonstrate that the system:

- Reduces water wastage by supplying only the required amount of water
- Improves irrigation efficiency compared to traditional methods
- Minimizes human effort through automation
- Prevents over-irrigation and under-irrigation
- Provides consistent soil moisture levels, which is beneficial for plant growth

In conclusion, the Automatic Soil Irrigation System proved to be a cost-effective, efficient, and reliable solution for smart irrigation. It can be effectively used in agricultural fields, gardens, and greenhouse environments to enhance productivity and conserve water resources.

1.8 CONCLUSION:

The Automatic Soil Irrigation System provides an efficient and smart solution for modern irrigation challenges. It reduces water wastage, minimizes manual effort, and improves crop productivity by supplying water based on real-time soil moisture conditions.

The system uses a soil moisture sensor to detect the water content in the soil and sends signals to the controller. Based on these signals, the controller operates the relay to turn the water pump ON or OFF. This ensures that plants receive the required amount of water at the right time.

The project demonstrates that automation in agriculture can significantly improve efficiency and resource management. The use of technologies like Arduino and ESP32 makes the system reliable, cost-effective, and easy to implement.

Although there are some limitations such as dependency on power and sensor accuracy, the system overall is a practical solution for smart farming. With further improvements, it can be widely used in agriculture, gardens, and greenhouses.

REFERENCES

[1] K. K. Chitkara, Water Resources Engineering, Tata McGraw-Hill Education.
[2] B. C. Punmia and Pande B. B. Lal, Irrigation and Water Power Engineering, Laxmi Publications.
[3] A. K. Sawhney, Electrical and Electronics Measurements and Instrumentation, Dhanpat Rai Publications.
[4] Muhammad Ali Mazidi, The 8051 Microcontroller and Embedded Systems, Pearson Education.JSPM'S R.S.C.O.E. POLYTECHNIC TATHAWADE. 70
[5] Simon Monk, Programming Arduino: Getting Started with Sketches, McGraw-Hill Education.

B) Research Articles / Journals

[1] R. Gupta and R. Das, "Design and Performance Analysis of Savonius Vertical Axis Wind Turbine," Journal of Renewable Energy, 2021.
[2] S. Kumar and V. Tyagi, "Performance Evaluation of Hybrid Solar-Wind Power

Systems," International Journal of Green Energy, 2020.

[3] M. Al-Abadi and J. Thompson, "Optimization of Gear Ratio for Small Wind Turbines," Renewable Energy Research, 2019.

[4] L. Chen and H. Wong, "Design of Portable Charging Stations," Energy Systems Journal, 2022.

[5] P. Meena and K. Reddy, "Power Electronics in Hybrid Systems," Electrical Engineering Review, 2023.

C) Websites

[1] Arduino Official Website – https://www.arduino.cc

[2] ESP32 Official Documentation – https://www.espressif.com

[3] Electronics Tutorials – https://www.electronics-tutorials.ws

[4] Circuit Digest – https://circuitdigest.com

[5] ResearchGate – https://www.researchgate.net

D) Online Resources / Tutorials

[1] YouTube tutorials on Arduino-based irrigation systems

[2] Online IoT project guides and blogs

[3] Technical forums for troubleshooting (Stack Overflow, Arduino Forum)

[4] Open-source project documentation and GitHub repositories

[5] Educational websites for sensor interfacing and embedded systems