**Soil Sensor Application Documentation**

**1. Overview**

This project demonstrates a Raspberry Pi application that monitors soil moisture using a digital soil sensor (LM393).

The application is capable of:

• Printing real-time terminal output that shows whether the soil is “DRY” or “WET.”

• Optionally, lighting up an LED when the soil is dry.

• Saving readings into a CSV file for later analysis.

This setup is designed as a basic laboratory demo to illustrate how students can wire a sensor, program GPIO pins, and collect data.

**2. Hardware Configuration**

Required Components

• Raspberry Pi (Model 3, 4, or later)

• Soil Moisture Sensor (LM393 Digital Output type)

• Breadboard and jumper wires

• (Optional) LED and 220 Ω resistor

Wiring Setup

1. Connect the sensor’s VCC pin to Raspberry Pi pin 2 (5V).

2. Connect the sensor’s GND pin to Raspberry Pi pin 6 (Ground).

3. Connect the sensor’s DO pin to Raspberry Pi pin 11 (GPIO17, BCM=17).

4. If you want to add an LED indicator:

• Connect Raspberry Pi GPIO18 (pin 12) to a 220 Ω resistor.

• Connect the resistor to the LED’s long leg (anode).

• Connect the LED’s short leg (cathode) to Raspberry Pi Ground.

Calibration: Insert only the black fork probe into soil or water. Adjust the blue screw on the sensor board until the onboard D0 LED flips on and off between wet and dry conditions.

**3. GitHub Repository Link**

Repository: Insert your GitHub link here once you publish the project

**4. Required Software Libraries**

System Preparation

Update and upgrade Raspberry Pi OS by typing:

**sudo apt update && sudo apt -y upgrade**

Install required system packages:

**sudo apt -y install python3-pip python3-venv git**

Python Virtual Environment

Create a project folder called soil-sensor-app and inside it create a virtual environment:

**mkdir ~/soil-sensor-app && cd ~/soil-sensor-app**

**python3 -m venv .venv**

Activate the environment:

**source .venv/bin/activate**

Upgrade pip and install the required library:

**pip install --upgrade pip RPi.GPIO**

**5. Key Parameters**

• SENSOR\_PIN is the GPIO pin connected to the sensor DO output (default is 17).

• LED\_PIN is the GPIO pin connected to the LED (default is 18, set to None if not used).

• CSV\_LOG enables or disables CSV logging (default is True).

• CSV\_PATH specifies the CSV filename (default is soil\_log.csv).

• PRINT\_INTERVAL sets how many seconds to wait between readings (default is 1.0).

• INVERT\_LOGIC can be set to True if the module behaves in reverse logic.

**6. Project File Structure**

The folder soil-sensor-app contains:

• .venv (the Python virtual environment)

• soil\_test.py (a simple test script to check sensor readings)

• monitor\_soil.py (the main monitoring script with logging and optional LED)

• soil\_log.csv (log file that is created while the monitor runs)

• README.md or this documentation file

**7. Step-by-Step Guide to Run Project**

Raspberry Pi Setup with Raspberry Pi Imager

Before using the soil sensor application, the Raspberry Pi must be prepared with Raspberry Pi OS on a microSD card. This is done using the official Raspberry Pi Imager tool.

Step 1: Download Raspberry Pi Imager

• Go to the official Raspberry Pi website: https://www.raspberrypi.com/softwarse/

• Download and install Raspberry Pi Imager for your operating system (Windows, macOS).

Step 2: Select OS and Device

1. Open Raspberry Pi Imager.

2. Under Choose Device, select your Raspberry Pi model (e.g., Raspberry Pi 4 or Raspberry Pi 5).

3. Under Choose OS, select Raspberry Pi OS (64-bit).

4. Under Choose Storage, select your microSD card.

Step 3: Configure Settings

Click on the gear icon (⚙️) in Raspberry Pi Imager to configure settings before flashing.

General Tab

• Set hostname: (optional, e.g., raspberrypi).

• Set username: enter your chosen username (example: ankita).

• Set password: enter your chosen password (example: raspberry).

Services Tab (important for SSH access)

• Enable SSH → checked

• Authentication Method → select Use password authentication.

• Allow public-key authentication only → not selected (leave unchecked).

This ensures you can log in with username + password, not just an SSH key.

Step 4: Write OS to the SD Card

• After saving the settings, return to the main Imager screen.

• Click Write to flash the OS to the microSD card.

• Wait until the process finishes and the card is verified.

Step 5: Boot the Raspberry Pi

1. Safely eject the microSD card from your computer.

2. Insert the card into the Raspberry Pi.

3. Connect power, monitor (if using), and peripherals.

4. The Pi will boot up with your chosen username, password, and SSH settings enabled.

At this point, your Raspberry Pi is ready for either direct monitor login or SSH remote login.

Option A: Running Without SSH (Direct Monitor, Keyboard, and Mouse)

1. Connect the Raspberry Pi to a monitor using HDMI, and plug in a keyboard and mouse.

2. Power up the Pi and boot into Raspberry Pi OS.

3. Open the Terminal application from the desktop.

4. Navigate into the project folder by typing:

**cd ~/soil-sensor-app**

5. Activate the Python virtual environment by typing:

**source .venv/bin/activate**

6. Run the sensor test script by typing:

**python3 soil\_test.py**

• The output will show “WET” when the probe is in water and “DRY” when it is in air.

• Stop the script by pressing Ctrl + C

7. Run the main monitor script by typing:

**python3 monitor\_soil.py.**

• This script will print soil states once per second.

• If the LED is connected, it will light up when the soil is dry.

• A file named **soil\_log.csv** will be created and updated automatically.

Option B: Running With SSH (Remote Access)

Step 1: Prepare the Raspberry Pi

1. Make sure the Raspberry Pi is connected to Wi-Fi or Ethernet.

2. Enable SSH by typing: sudo raspi-config.

3. Navigate to “Interface Options,” select “SSH,” and choose “Enable.”

Step 2: Connecting From a Mac

1. On your Mac, open the **Terminal** application.

2. Connect to the Pi using its hostname by typing:

**ssh pi@raspberrypi.local**

3. When prompted, enter the Raspberry Pi’s password (default is raspberry if unchanged).

4. Once logged in, type:

**cd ~/soil-sensor-app**

5. Activate the virtual environment by typing:

**source .venv/bin/activate**

6. Run **python3 soil\_test.py** to test the sensor.

7. Run **python3 monitor\_soil.py** to start the main monitoring program.

Step 3: Connecting From a Windows PC

1. On Windows, open PowerShell (or use PuTTY if preferred).

2. Connect to the Raspberry Pi by typing: ssh pi@raspberrypi.local

3. Enter the Raspberry Pi’s password when asked.

4. Once connected, navigate to the project folder with: cd ~/soil-sensor-app

5. Activate the Python environment with: source .venv/bin/activate

6. Run the scripts with python3 soil\_test.py and python3 monitor\_soil.py.

**8. Viewing Logs**

When the main monitor script runs, it automatically creates a CSV file named **soil\_log.csv**

To view the log in real time, type: **tail -f soil\_log.csv**

Each log entry contains the timestamp, the soil state (DRY or WET), and the raw digital value (0 or 1).