SoilMind: Real Sensor Integration Guide

Overview

This guide covers transitioning from potentiometer simulation to real agricultural sensors for the Smart Irrigation system. By the end, you'll have a fully functional edge-AI irrigation controller.

1. Hardware Requirements

1.1 Recommended Sensors

Sensor	Model	Purpose	Interface	Price Range
Soil Moisture	**Capacitive v1.2/v2.0**	Soil moisture level	Analog	$2-5
Temperature & Humidity	**DHT22 / AM2302**	Air temp & humidity	Digital (1-wire)	$3-6
Alternative Temp	**DS18B20** (waterproof)	Soil temperature	Digital (1-wire)	$2-4

> ⚠️ **Avoid resistive soil moisture sensors** - they corrode quickly and give inconsistent readings.

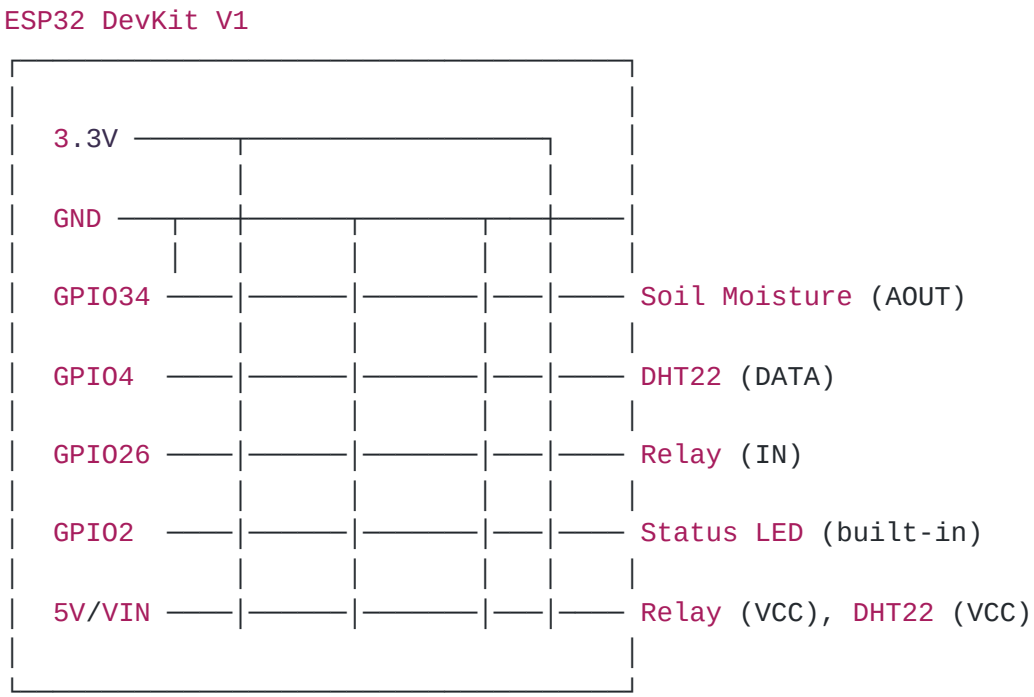
1.2 Bill of Materials

Component	Quantity	Notes
ESP32 DevKit	1	Any variant with ADC pins
Capacitive Soil Moisture Sensor	1	v1.2 or v2.0
DHT22 Sensor	1	With breakout board
5V Relay Module	1	For pump control
Water Pump (5V/12V)	1	Submersible or diaphragm
10kΩ Resistor	1	Pull-up for DHT22
Jumper Wires	~15	Male-to-female recommended

Component	Quantity	Notes
Breadboard	1	For prototyping
Power Supply	1	5V 2A minimum

2. Wiring Diagram

2.1 Pin Connections



2.2 Detailed Connections

Capacitive Soil Moisture Sensor:

Sensor		ESP32
VCC	→	3.3V (NOT 5V - can damage ADC!)
GND	→	GND
AOUT	→	GPIO34

DHT22 Temperature/Humidity Sensor:

Sensor		ESP32
VCC	→	5V (or 3.3V)

```
GND     →    GND
DATA    →    GPIO4 (with 10kΩ pull-up to VCC)
```

Relay Module:

```
Relay       ESP32
─────       ─────
VCC     →   5V
GND     →   GND
IN      →   GPIO26
COM     →   Pump power (+)
NO      →   Pump (+) terminal
```

3. Sensor Calibration

3.1 Soil Moisture Calibration

The capacitive sensor outputs analog values. You need to map these to match your training data range (50-450).

Calibration Procedure:

1. **Dry Reading (Air):**

```
Remove sensor from soil, wipe clean, let dry
Record value: DRY_VALUE = _______ (typically 3200-3600)
```

2. **Wet Reading (Water):**

```
Submerge sensor up to the line in water
Record value: WET_VALUE = _______ (typically 1400-1800)
```

3. **Calculate Mapping:**

```
// In your code:
const int DRY_VALUE = 3400;   // Your measured dry value
const int WET_VALUE = 1600;   // Your measured wet value

// Map to training data range (50-450)
float mapMoisture(int raw) {
    // Invert because lower ADC = wetter soil
    float normalized = (float)(DRY_VALUE - raw) / (DRY_VALUE - WET_VALUE);
    normalized = constrain(normalized, 0.0f, 1.0f);
```

```
    return MOISTURE_MIN + normalized * (MOISTURE_MAX - MOISTURE_MIN);
}
```

3.2 Temperature Verification

DHT22 is factory-calibrated, but verify readings:

1. Compare with a known thermometer
2. If offset exists, add correction factor:

```
float temp = dht.readTemperature() + TEMP_OFFSET;
```

4. Software Implementation

4.1 Required Libraries

Install via Arduino Library Manager:

```
- DHT sensor library (by Adafruit)
- Adafruit Unified Sensor
- ArduTFLite (already installed)
```

4.2 Code Structure

```
#include <ArduTFLite.h>
#include <DHT.h>
#include "irrigation_model.h"

// ============ PIN DEFINITIONS ============
#define SOIL_MOISTURE_PIN   34
#define DHT_PIN             4
#define RELAY_PIN           26
#define LED_PIN             2

// ============ SENSOR CONFIG ============
#define DHT_TYPE            DHT22

// Calibration values (UPDATE THESE!)
const int DRY_VALUE = 3400;
const int WET_VALUE = 1600;

// Mapping ranges (match training data)
const float MOISTURE_MIN = 50.0f;
const float MOISTURE_MAX = 450.0f;
```

```
// ============ OBJECTS ============
DHT dht(DHT_PIN, DHT_TYPE);
```

4.3 Sensor Reading Functions

```
float readSoilMoisture() {
    // Average multiple readings for stability
    long sum = 0;
    for (int i = 0; i < 10; i++) {
        sum += analogRead(SOIL_MOISTURE_PIN);
        delay(10);
    }
    int raw = sum / 10;

    // Map to training data range (inverted - lower ADC = wetter)
    float normalized = (float)(DRY_VALUE - raw) / (DRY_VALUE - WET_VALUE);
    normalized = constrain(normalized, 0.0f, 1.0f);

    return MOISTURE_MIN + normalized * (MOISTURE_MAX - MOISTURE_MIN);
}

float readTemperature() {
    float temp = dht.readTemperature();

    // Check for read errors
    if (isnan(temp)) {
        Serial.println("[ERROR] DHT read failed!");
        return -999.0f;  // Error indicator
    }

    return temp;
}
```

4.4 Setup Function

```
void setup() {
    Serial.begin(115200);
    delay(1000);

    // Initialize pins
    pinMode(SOIL_MOISTURE_PIN, INPUT);
    pinMode(RELAY_PIN, OUTPUT);
    pinMode(LED_PIN, OUTPUT);
    digitalWrite(RELAY_PIN, LOW);

    // Initialize DHT sensor
    dht.begin();
    Serial.println("[INIT] DHT22 initialized");
```

```
    // Initialize ML model
    modelReady = modelInit(irrigation_model, tensorArena, kTensorArenaSize);
    Serial.printf("[INIT] Model: %s\n", modelReady ? "OK" : "FAILED");

    // Initialize history
    history.init();

    // Wait for DHT to stabilize
    delay(2000);
}
```

5. Sampling Strategy

5.1 Recommended Intervals

Parameter	Interval	Reason
Sensor Reading	30-60 seconds	Soil moisture changes slowly
ML Inference	Every 4th reading	Need trend data
Pump Check	After each inference	React to predictions

5.2 Reading Interval Configuration

```
const unsigned long READING_INTERVAL = 30000;  // 30 seconds
const unsigned long MIN_PUMP_ON_TIME = 10000;  // Minimum 10s pump runtime
const unsigned long PUMP_COOLDOWN = 60000;     // Wait 60s before re-checking
```

6. Error Handling

6.1 Sensor Failure Detection

```
bool sensorsOK = true;

void checkSensors() {
    // Check DHT
    float temp = dht.readTemperature();
    if (isnan(temp)) {
        Serial.println("[WARN] DHT not responding!");
        sensorsOK = false;
    }
```

```
    // Check soil moisture (should be within calibration range)
    int raw = analogRead(SOIL_MOISTURE_PIN);
    if (raw < WET_VALUE - 500 || raw > DRY_VALUE + 500) {
        Serial.println("[WARN] Soil sensor out of range!");
        sensorsOK = false;
    }

    sensorsOK = true;
}
```

6.2 Failsafe Logic

```
void controlIrrigation(float probability) {
    // FAILSAFE: If sensors fail, turn OFF pump
    if (!sensorsOK || !modelReady) {
        digitalWrite(RELAY_PIN, LOW);
        irrigationActive = false;
        Serial.println("[SAFETY] Pump OFF - sensor/model error");
        return;
    }

    // Normal ML-based control
    bool shouldIrrigate = (probability >= IRRIGATION_THRESHOLD);
    // ... rest of control logic
}
```

7. Testing Checklist

7.1 Pre-Deployment Tests

- [ ] **Sensor Readings Stable**
  - Moisture readings vary < ±5 units when stable
  - Temperature readings vary < ±0.5°C when stable
- [ ] **Calibration Verified**
  - Dry soil → High moisture value (~50-100)
  - Wet soil → Low moisture value (~350-450)
  - Values within training data range
- [ ] **ML Predictions Sensible**
  - High moisture + stable → NO irrigation
  - Low moisture + dropping → IRRIGATE

    - Rising moisture → Eventually stops irrigation

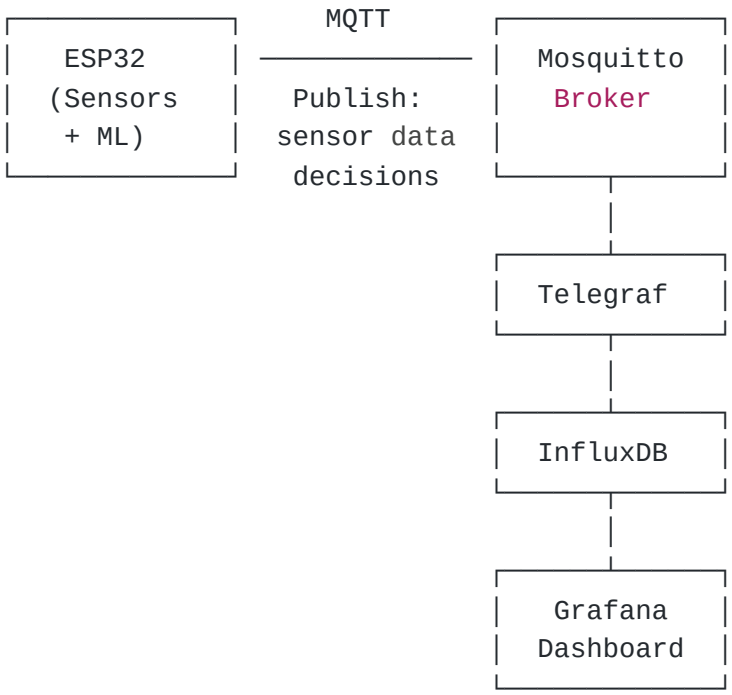
- [ ] **Relay/Pump Works**
    - Relay clicks when activated
    - Pump runs when relay closes
    - No false triggers
- [ ] **Long Duration Test (24h)**
    - No crashes or memory leaks
    - Reasonable irrigation cycles
    - Log data for review

7.2 Field Deployment Checklist

- [ ] Waterproof enclosure for ESP32
- [ ] Sensor cables secured and protected
- [ ] Power supply adequate and weatherproof
- [ ] Manual override switch installed
- [ ] Water supply connected and tested

8. Next Steps: MQTT Integration

Once sensors work locally, integrate with your IoT stack:



MQTT Topics Structure:

```
soilmind/sensors/temperature
soilmind/sensors/moisture
soilmind/sensors/moisture_trend
soilmind/ml/probability
soilmind/ml/decision
soilmind/pump/status
```

9. Troubleshooting

Problem	Possible Cause	Solution
Moisture always 0 or 450	Wrong calibration values	Re-calibrate DRY/WET values
DHT returns NaN	Wiring issue or no pull-up	Check wiring, add 10kΩ pull-up
Model says IRRIGATE always	Values outside training range	Verify mapping matches training data
ESP32 crashes	Arena too small	Increase `kTensorArenaSize`
Pump never turns off	Threshold too low	Increase `IRRIGATION_THRESHOLD`
Erratic readings	Power noise	Add 100µF capacitor on sensor VCC

10. Complete Code Template

exist in attached .ino file