University of Asia Pacific

Department of Computer Science and Engineering

CSE 316: Microprocessors and Microcontrollers Lab

LAB REPORT

Experiment Number: 04

Experiment Title: Mini Project 4: Water Level based Automatic Pump Control

Date of Submission: 18.09.2025

Submitted by:

Name : M Sakib Rahman

Student ID : 22201240

Section : E-1

Submitted to:

Zaima Sartaj Taheri

Lecturer,

Department of Computer Science and Engineering

1. Experiment Name

Mini Project 4: Water Level based Automatic Pump Control

2. Objective

The main objective of this experiment is to design and implement an automatic water-level-based pump control system that:

- Uses an ultrasonic sensor to continuously measure the water level in a tank.
- Automatically turns the pump ON when the water level falls below a defined threshold (low level).
- Automatically turns the pump OFF when the tank is full (high level).
- Provides reliable switching using an L293D H-bridge motor driver for controlling the DC pump motor.

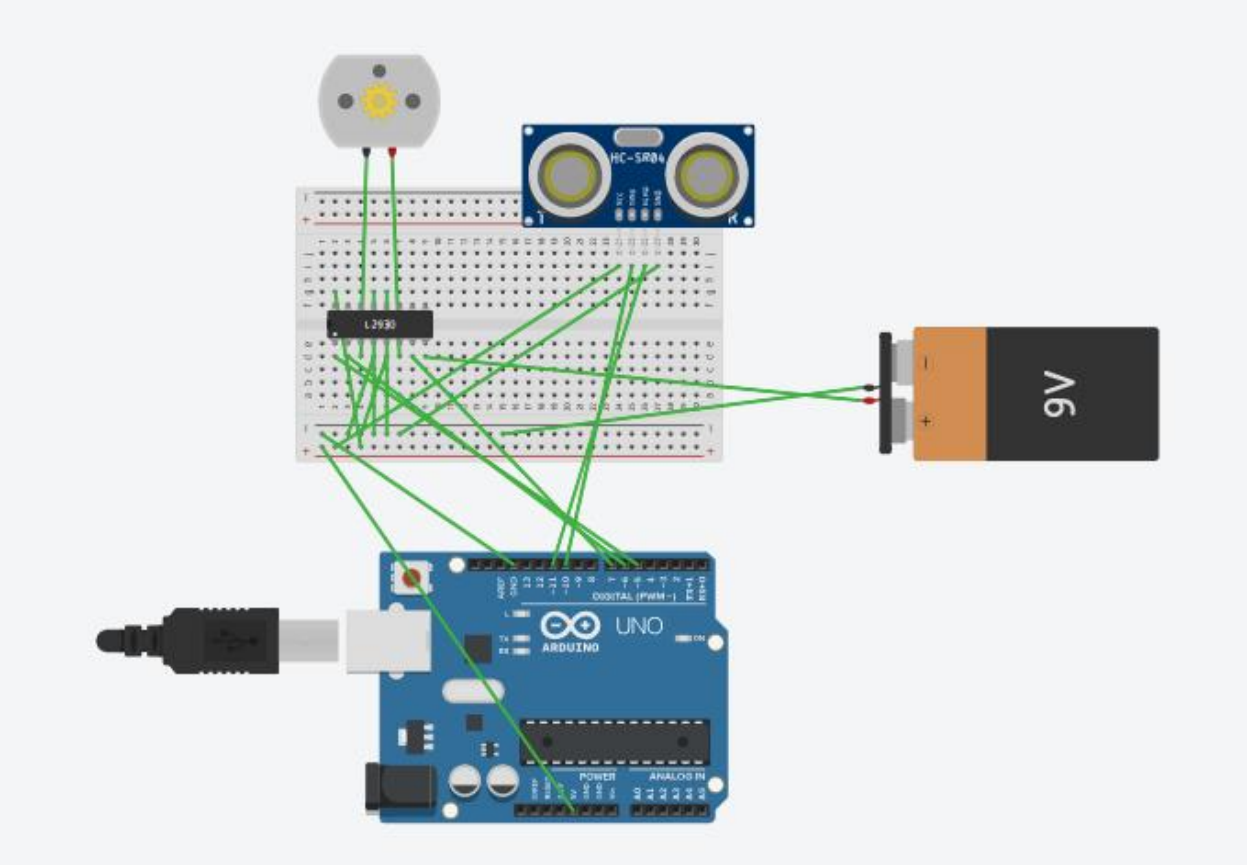
This system reduces human effort in monitoring and switching the pump, prevents overflow, and ensures efficient water management.

3. Apparatus / Hardware & Software Requirements

All required tools and components:

- Arduino Uno R3
- Ultrasonic Sensor HC-SR04
- L293D Motor Driver IC
- DC Motor / Water Pump
- 9V Battery (Power Source)
- Breadboard & Jumper Wires

4. Circuit Diagram / Schematic



5. Code / Assembly Program

```
#define TRIG_PIN 10
#define ECHO_PIN 11
#define EN_PIN 5
#define IN1_PIN 6
#define IN2_PIN 7

const unsigned long READ_INTERVAL = 500;
unsigned long lastRead = 0;
bool pumpOn = false;

const int FULL_DISTANCE = 5;      // cm - "tank full"
const int EMPTY_DISTANCE = 12; // cm - "tank low"
```

```
void setup() {
  Serial.begin(9600);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  pinMode(EN_PIN, OUTPUT);
  pinMode(IN1_PIN, OUTPUT);
  pinMode(IN2_PIN, OUTPUT);

  digitalWrite(TRIG_PIN, LOW);
  digitalWrite(IN1_PIN, LOW);
  digitalWrite(IN2_PIN, LOW);
  analogWrite(EN_PIN, 0);
  delay(50);
  Serial.println("Water-level pump control starting...");
}

void loop() {
  if (millis() - lastRead >= READ_INTERVAL) {
    lastRead = millis();
    long distance = readUltrasonicCm();
    Serial.print("Distance: ");
    if (distance >= 999) {
      Serial.print("no echo");
    } else {
      Serial.print(distance);
      Serial.print(" cm");
    }
    Serial.print("    | Pump: ");
    Serial.println(pumpOn ? "ON" : "OFF");

    // Control logic with hysteresis to avoid rapid toggling:
    if (!pumpOn && distance > EMPTY_DISTANCE) {
      startPump();
    } else if (pumpOn && distance <= FULL_DISTANCE) {
      stopPump();
    }
```

```
    }
}

long readUltrasonicCm() {
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);


    long duration = pulseIn(ECHO_PIN, HIGH, 30000);
    if (duration == 0) return 999;
    long distanceCm = duration * 0.034 / 2.0;
    return distanceCm;
}

void startPump() {
    Serial.println(">> START PUMP");
    digitalWrite(IN1_PIN, HIGH);
    digitalWrite(IN2_PIN, LOW);
    analogWrite(EN_PIN, 255);
    pumpOn = true;
}

void stopPump() {
    Serial.println(">> STOP PUMP");
    analogWrite(EN_PIN, 0);
    digitalWrite(IN1_PIN, LOW);
    digitalWrite(IN2_PIN, LOW);
    pumpOn = false;
}
```

Output / Observations

When the water level is low (distance from sensor > 12 cm):

→ Pump turns ON, and water starts filling the tank.

When the water level is high/full (distance ≤ 5 cm):

→ Pump turns OFF, preventing overflow.

Serial Monitor shows continuous output:

Distance: 14 cm | Pump: ON

Distance: 6 cm | Pump: OFF

The system works reliably with hysteresis control, preventing frequent ON/OFF switching.

7. Result

The automatic water level-based pump control system was successfully simulated and implemented using Arduino Uno, HC-SR04 ultrasonic sensor, L293D motor driver, and a DC pump.

The pump motor automatically started when the water level was low and stopped when the tank was full, as expected.

8. Conclusion

This experiment demonstrated a practical automation system for water tank management. The use of an ultrasonic sensor ensured accurate and non-contact level detection, while the Arduino controlled the motor driver to operate the pump reliably.

The system:

- Prevents water wastage by avoiding overflow.
- Eliminates the need for manual supervision.
- Is simple, low-cost, and energy-efficient.

Future Improvements:

- Adding an LCD/LED display to show water level percentage.
- Integrating Wi-Fi or GSM for remote monitoring and alerts.
- Using a relay driver instead of L293D for real high-power water pumps.
- Adding a dry-run protection system to avoid pump damage if water source is empty.