

## Automated Sludge Detection System for Water Tanks

---

### i) Problem Statement

In many water tanks, accumulated sludge over time poses serious hygiene risks and reduces water quality. Manual inspection is time-consuming and often overlooked. There is a need for an automated system to **detect sludge level** and **alert users** when the sludge exceeds acceptable limits.

---

### ii) Scope of the Solution

This project proposes a **microcontroller-based system** that monitors sludge levels using simulated conductivity detection (via push buttons). The system **displays sludge level** (CLEAN / LOW / MEDIUM / HIGH) on a 16x2 LCD and activates a **buzzer and LED alert** when sludge is high.

Key features:

- Real-time sludge level display
  - Visual (LED) and audible (buzzer) alert for high levels
  - Simulated using Proteus, Arduino Uno, and push buttons
- 

### iii) Required Components to Develop the Solution

#### Hardware (Simulated in Proteus):

- Arduino Uno
- 3 Push Buttons (used as level sensors at 2mm, 4mm, 6mm)
- 16x2 LCD (no I2C)
- Red LED
- Buzzer
- Pull-down resistors for button logic

#### Software Tools:

- **Arduino IDE** (for coding and uploading)
  - **Proteus Design Suite** (for simulation and circuit testing)
-

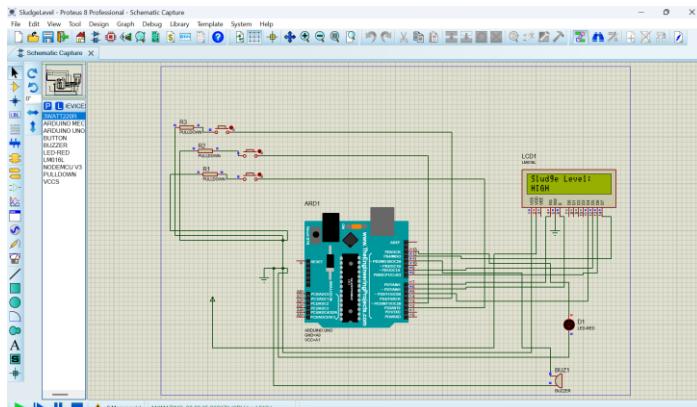
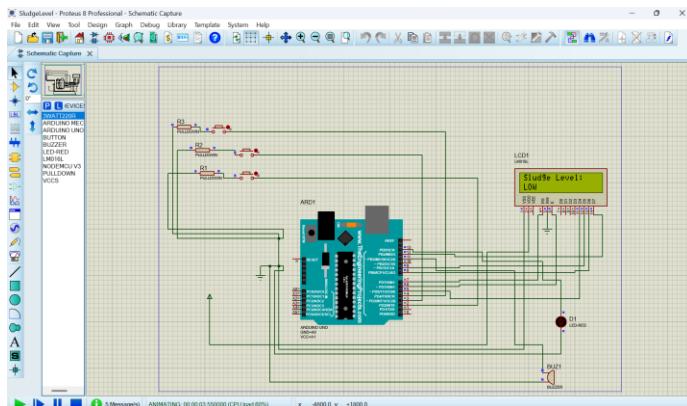
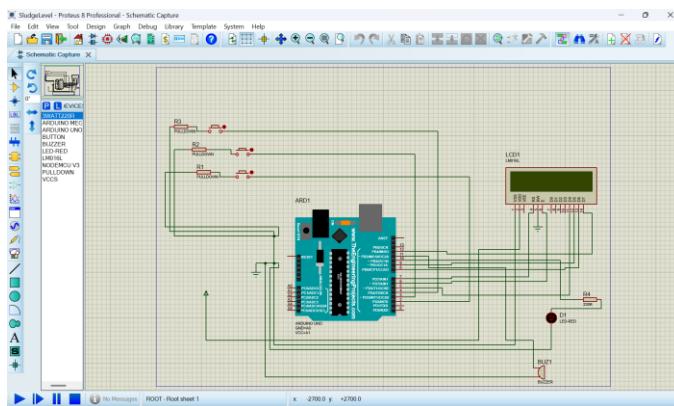
#### iv) Simulated Circuit

**Software Used:** Proteus 8 Professional

**File Type:** SludgeLevel.pdsprj (Proteus project)

Components connected as follows:

- Push buttons: D2, D3, D4 (as sludge sensors)
- LCD: D7, D6, D5, D9, D8, D12 (RS, E, D4–D7)
- LED: D10
- Buzzer: D11
- All buttons with pull-down resistors



---

## v) Video of the Demo

[https://drive.google.com/file/d/1Yage4JN7oy9OqOaHzadzkA4l9jja\\_Wy5/view?usp=sharing](https://drive.google.com/file/d/1Yage4JN7oy9OqOaHzadzkA4l9jja_Wy5/view?usp=sharing)

---

## vi) Gerber File

Since this prototype was simulated and not fabricated physically, Gerber files are not generated. If hardware fabrication is planned, PCB can be designed using Altium.

---

## vii) Code for the Solution

// Pin Definitions

```
const int level1 = 2; // Push Button 1 - 2mm
```

```
const int level2 = 3; // Push Button 2 - 4mm
```

```
const int level3 = 4; // Push Button 3 - 6mm
```

```
const int ledPin = 13;
```

```
const int buzzerPin = 11;
```

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(7, 6, 5, 9, 8, 12); // RS, E, D4, D5, D6, D7
```

```
void setup() {
```

```
  pinMode(level1, INPUT); // Pull-down logic (matches your Proteus wiring)
```

```
  pinMode(level2, INPUT);
```

```
  pinMode(level3, INPUT);
```

```
  pinMode(ledPin, OUTPUT);
```

```
  pinMode(buzzerPin, OUTPUT);
```

```
lcd.begin(16, 2);
```

```
lcd.print("Sludge Monitor");
```

```
delay(2000);
```

```
lcd.clear();
```

```
}s
```

```
void loop() {  
    bool l1 = digitalRead(level1);  
    bool l2 = digitalRead(level2);  
    bool l3 = digitalRead(level3);  
  
    lcd.setCursor(0, 0);  
    lcd.print("Sludge Level:");  
  
    if (l3) {  
        lcd.setCursor(0, 1);  
        lcd.print("HIGH      ");  
        digitalWrite(ledPin, HIGH);  
        tone(buzzerPin, 1000); // Buzz continuously  
    } else if (l2) {  
        lcd.setCursor(0, 1);  
        lcd.print("MEDIUM      ");  
        digitalWrite(ledPin, LOW);  
        noTone(buzzerPin);  
    } else if (l1) {  
        lcd.setCursor(0, 1);  
        lcd.print("LOW      ");  
        digitalWrite(ledPin, LOW);  
        noTone(buzzerPin);  
    } else {  
        lcd.setCursor(0, 1);  
        lcd.print("CLEAN      ");  
        digitalWrite(ledPin, LOW);  
        noTone(buzzerPin);  
    }  
    delay(500);}
```

---

