

**MID TERM PROJECT**  
**L&T Edutech**

**TITLE: Automated Slug Detection System for Water Tanks**

**Submitted by: Emil Alexander George**

## INDEX

Sl.no	CONTENT
1.	Problem Statement
1.1	Aim
1.2	Hardware
1.3	Working
1.4	Circuit
1.5	Gerber File
1.6	Code

1. **PROBLEM STATEMENT:** To develop a microcontroller based prototype which alerts the user to clean the tank when sludge level increases.

1.1 Aim: The aim of the project is to design an automated slug detector.

1.2 Hardware:

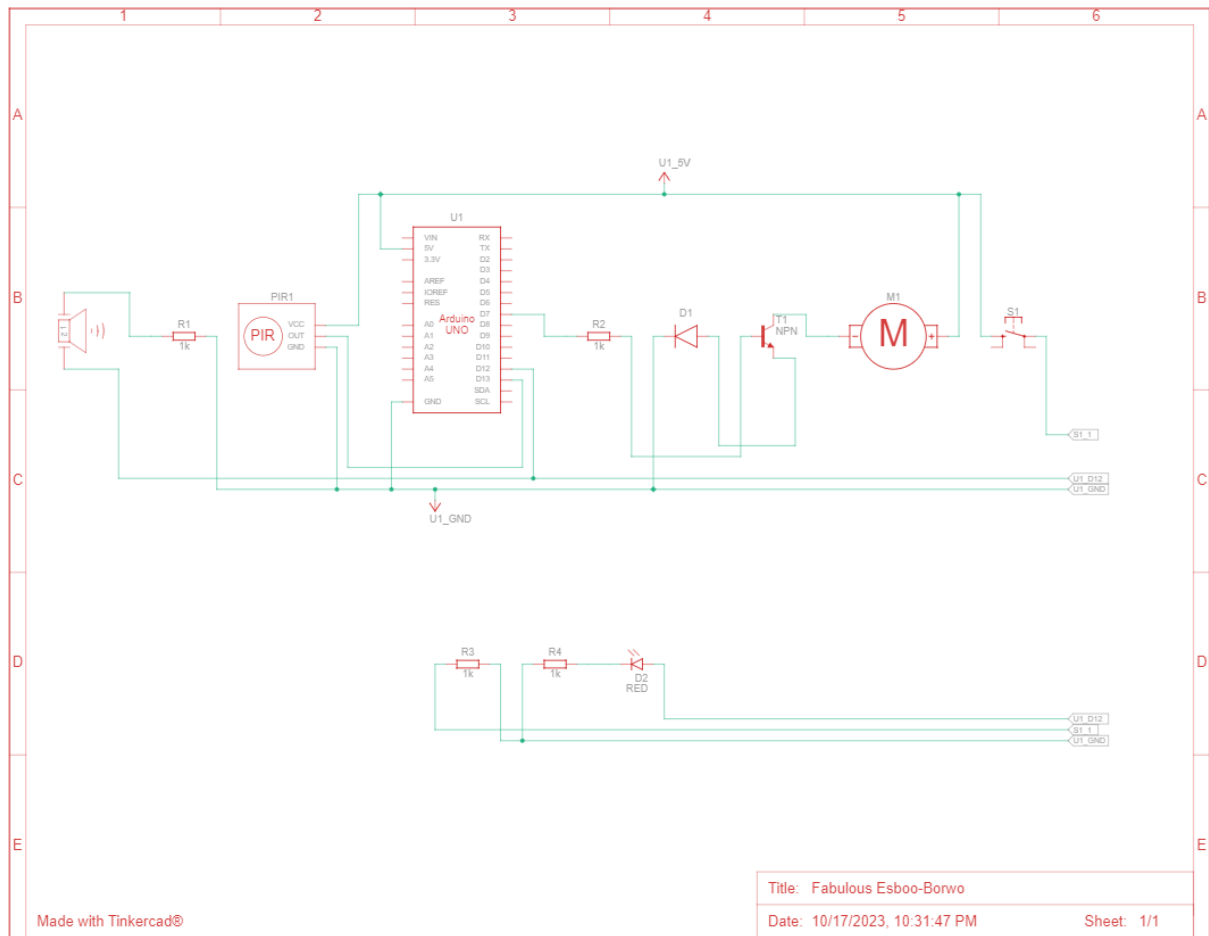
- a. Arduino uno R3
- b. DC motor
- c. PIR sensor
- d. Piezo(buzzer)
- e. Transistor
- f. Push button
- g. Resistors

1.3 Working: The given prototype consists of a PIR sensor, which is an infrared sensor. The model is designed in such a way that it will be installed below the water level in a tank and it detects any form of movement in the field of view of the sensor and alerts the user through an alarm indicating the presence of slugs in the water. The PIR sensor is tuned to detect the smallest movement to make it compatible to the movement of slugs. When detected, the PIR sensor will set off the alarm. The prototype has a button through which one can stop the alarm. This hence notifies the user and enable the user to clean the tank.

The diagram illustrates the following connections:

- Power:** A USB cable is connected to the Arduino Uno's USB port. The board's 5V and GND pins are connected to the breadboard's power rails.
- Sensors:**
  - A potentiometer is connected to the 5V and GND rails of the breadboard. Its wiper is connected to the Arduino's A0 pin.
  - An ultrasonic sensor (HC-SR04) is connected with its VCC to 5V, GND to GND, and TRIG to digital pin 2. The ECHO pin is connected to digital pin 3.
- Actuators:**
  - A servo motor is connected with its VCC to 5V, GND to GND, and signal to digital pin 4.
  - A DC motor is connected with its VCC to 5V, GND to GND, and a control wire to digital pin 5.
  - A fan is connected with its VCC to 5V, GND to GND, and a control wire to digital pin 6.

## 1.5 Gerber file:



## 1.6 Code for the microcontroller:

```
int pirPin=13;
int pirStatus=0;
int piezoPin=12;

int buttonPin=6;
int buttonStatus=0;
int motorPin=7;

void setup()
{
  pinMode(pirPin, INPUT);
  pinMode(piezoPin,OUTPUT);
  pinMode(motorPin,OUTPUT);
}

void loop()
{
  pirStatus=digitalRead(pirPin);
  if (pirStatus ==HIGH){
    tone(piezoPin,5000);
  } else {
    noTone(piezoPin);
  }

  buttonStatus=digitalRead(buttonPin);
  if (buttonStatus==HIGH){
    digitalWrite(motorPin,HIGH);
  } else {
    digitalWrite(motorPin,LOW);
  }
}
```