2) / Pin definitions

const int sensorPin = A0; // Analog input pin for pressure sensor

const int pumpPin = 9; // PWM output pin for controlling pump speed

// Constants for pressure thresholds and speed adjustments

const int minPressure = 600; // Minimum pressure threshold

const int midPressure1 = 650; // Mid pressure threshold 1

const int midPressure2 = 700; // Mid pressure threshold 2

const int maxPressure = 1023; // Maximum pressure threshold

const int minDutyCycle = 100; // Initial duty cycle (in percentage)

const int maxDutyCycle = 200; // Maximum duty cycle (in percentage)

const int dutyCycleChange = 50; // Change in duty cycle percentage

void setup() {

pinMode(sensorPin, INPUT);

pinMode(pumpPin, OUTPUT);

// Initialize Serial communication

Serial.begin(9600);

}

void loop() {

// Read pressure sensor value

int pressureValue = analogRead(sensorPin);

// Convert analog reading to pressure (assuming 5V reference and 10-bit ADC)

float pressure = map(pressureValue, 0, 1023, 0, 5000); // Convert to mV

pressure = ((pressure - 200) / 4) + 0; // Convert to kPa

// Print pressure value for debugging

Serial.print("Pressure: ");

Serial.print(pressure);

Serial.println(" kPa");

// Adjust pump speed based on pressure

if (pressure >= minPressure && pressure < midPressure1) {

analogWrite(pumpPin, minDutyCycle);

}

else if (pressure >= midPressure1 && pressure < midPressure2) {

analogWrite(pumpPin, minDutyCycle + dutyCycleChange);

}

else if (pressure >= midPressure2 && pressure <= maxPressure) {

analogWrite(pumpPin, minDutyCycle + 2 \* dutyCycleChange);

}

else if (pressure < minPressure && pressure >= 0) {

analogWrite(pumpPin, minDutyCycle);

}

else {

// Handle out of range values

Serial.println("Pressure out of range.");

}

// Delay before next iteration

delay(1000); // Adjust delay as needed

}