LAMPIRAN I
(LISTING PROGRAM)
#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

const int numReadings = 10;
int readings[numReadings]; // the readings from the analog input
int readIndex = 0; // the index of the current reading
int total = 0; // the running total
int average = 0; // the average
int inputPin = A1;
int sensorValue = 0; // variable to store the value coming from the sensor
const int buzzer = 13, ledPin2 = 1, ledPin3 = 2, ledPin4 = 3, ledPin5 = 11, ledPin6 = 12;

void setup() {
    pinMode(buzzer, OUTPUT);
    pinMode(ledPin2, OUTPUT);
    pinMode(ledPin3, OUTPUT);
    pinMode(ledPin4, OUTPUT);
    pinMode(ledPin5, OUTPUT);
    pinMode(ledPin6, OUTPUT);
    digitalWrite(ledPin6, HIGH);
    digitalWrite(ledPin5, HIGH);
    digitalWrite(ledPin4, HIGH);
    digitalWrite(ledPin3, HIGH);
    digitalWrite(ledPin2, HIGH);
    for (int thisReading = 0; thisReading < numReadings; thisReading++) {
        readings[thisReading] = 0;
    }
    lcd.begin(16, 2);
    lcd.setCursor(0, 0);
    lcd.print(" Metrologi");
    lcd.setCursor(0, 1);
    lcd.print("angkatan 2014");
    delay(1000);
    lcd.setCursor(0, 0);
    lcd.print(" Armin");
    lcd.setCursor(0, 1);
    lcd.print(" 142411079 ");
    delay(2000);
    lcd.clear();
    digitalWrite(ledPin6, LOW);
    digitalWrite(ledPin5, LOW);
    digitalWrite(ledPin4, LOW);
    digitalWrite(ledPin3, LOW);
    digitalWrite(ledPin2, LOW);
    digitalWrite(buzzer, LOW);
    delay(2000);
void loop() {

    total = total - readings[readIndex];
    readings[readIndex] = analogRead(inputPin);
    total = total + readings[readIndex];
    readIndex = readIndex + 1;
    if (readIndex >= numReadings) {
        readIndex = 0;
    }
    average = total / numReadings;

    int sensorValue1 = average;
    float voltage1 = (sensorValue1 * (5000 / 1023));
    float tekanan=((voltage1/9)*0.145)-2.3)*7;
    if (tekanan<=0) tekanan=0;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("nilai:"); lcd.print(voltage1); lcd.print(" mV");
    lcd.setCursor(0, 1);
    lcd.print("Pres:"); lcd.print(tekanan); lcd.print(" Psi");

    if(tekanan<28) {
        digitalWrite(ledPin6, LOW);
        digitalWrite(ledPin5, LOW);
        digitalWrite(ledPin4, LOW);
        digitalWrite(ledPin3, LOW);
        digitalWrite(buzzer, LOW);
    }
    else if(tekanan>=28 && tekanan<=29) {
        digitalWrite(ledPin6, HIGH);
        digitalWrite(ledPin5, LOW);
        digitalWrite(ledPin4, LOW);
        digitalWrite(ledPin3, LOW);
        digitalWrite(ledPin2, LOW);
    }
    else if(tekanan>=30 && tekanan<=31) {
        digitalWrite(ledPin6, HIGH);
        digitalWrite(ledPin5, HIGH);
        digitalWrite(ledPin4, LOW);
        digitalWrite(ledPin3, LOW);
        digitalWrite(ledPin2, LOW);
    }
    else if(tekanan=32) {
}
digitalWrite(ledPin6, HIGH);
digitalWrite(ledPin5, HIGH);
digitalWrite(ledPin4, HIGH);
digitalWrite(ledPin3, LOW);
digitalWrite(ledPin2, LOW);
}

else if(tekanan=33) {
    digitalWrite(ledPin6, HIGH);
digitalWrite(ledPin5, HIGH);
digitalWrite(ledPin4, HIGH);
digitalWrite(ledPin3, HIGH);
digitalWrite(ledPin2, LOW);
}

else if(tekanan>=34) {
    digitalWrite(ledPin6, HIGH);
digitalWrite(ledPin5, HIGH);
digitalWrite(ledPin4, HIGH);
digitalWrite(ledPin3, HIGH);
digitalWrite(ledPin2, HIGH);
}

if (tekanan>=28) digitalWrite(buzzer, HIGH);
if (tekanan<=28) digitalWrite(buzzer, LOW);
    delay(250);
}
LAMPIRAN II
(DATA SHEET ARDUINO)
The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

<table>
<thead>
<tr>
<th>Microcontroller Operating Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Voltage (recommended)</td>
</tr>
<tr>
<td>Input Voltage (limits)</td>
</tr>
<tr>
<td>Digital I/O Pins</td>
</tr>
<tr>
<td>Analog Input Pins</td>
</tr>
<tr>
<td>DC Current per I/O Pin</td>
</tr>
<tr>
<td>DC Current for 3.3V Pin</td>
</tr>
<tr>
<td>Flash Memory</td>
</tr>
</tbody>
</table>
SRAM
EEPROM
Clock Speed
ATmega328
5V
7-12V
6-20V
14 (of which 6 provide PWM output)
6
40mA
50mA
32 KB of which 0.5 KB used by bootloader
2 KB
1 KB
16 MHz
The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-watt) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.
LAMPIRAN III
(DATA SHEET SENSOR MPX 5700 AP)
Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated

The MPX5700 series piezoresistive transducer is a state-of-the-art monolithic silicon pressure sensor designed for a wide range of applications, but particularly those employing a microcontroller or microprocessor with A/D inputs. This patented, single element transducer combines advanced micromachining techniques, thin-film metallization, and bipolar processing to provide an accurate, high level analog output signal that is proportional to the applied pressure.

Features
1. 2.5% Maximum Error over 0° to 85°C
2. Ideally Suited for Microprocessor or Microcontroller-Based Systems
3. Available in Absolute, Differential and Gauge Configurations
4. Patented Silicon Shear Stress Strain Gauge
5. Durable Epoxy Unibody Element

ORDERING INFORMATION

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Options</th>
<th>Case Type</th>
<th>Order Number</th>
<th>Device Marking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Elements</td>
<td>Differential</td>
<td>8 67</td>
<td>MPX5700D</td>
<td>MPX5700D</td>
</tr>
<tr>
<td></td>
<td>Absolute</td>
<td>8 67</td>
<td>MPX5700A</td>
<td>MPX5700A</td>
</tr>
<tr>
<td>Ported Elements</td>
<td>Differential Dual Ports</td>
<td>86 7C</td>
<td>MPX5700D</td>
<td>MPX5700D</td>
</tr>
<tr>
<td></td>
<td>Gauge</td>
<td>86 7B</td>
<td>MPX5700G</td>
<td>MPX5700G</td>
</tr>
<tr>
<td></td>
<td>Gauge, Axial</td>
<td>86 7E</td>
<td>MPX5700D</td>
<td>MPX5700D</td>
</tr>
<tr>
<td></td>
<td>Absolute</td>
<td>86 7B</td>
<td>MPX5700A</td>
<td>MPX5700A</td>
</tr>
<tr>
<td></td>
<td>Absolute, Axial</td>
<td>86 7E</td>
<td>MPX5700A</td>
<td>MPX5700A</td>
</tr>
<tr>
<td></td>
<td>Absolute, Axial PC Mount</td>
<td>86 7F</td>
<td>MPX5700A</td>
<td>MPX5700A</td>
</tr>
</tbody>
</table>
LAMPIRAN IV
(DATA SHEET BUZZER)
**General Description:**

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke.

Buzzer is an integrated structure of electronic transducers, DC power supply, widely used in computers, printers, copiers, alarms, electronic toys, automotive electronic equipment, telephones, timers and other electronic products for sound devices. Active buzzer 5V Rated power can be directly connected to a continuous sound, this section dedicated sensor expansion module and the board in combination, can complete a simple circuit design, to "plug and play."

**Specifications:**

- 40 On-board passive buzzer
On-board 8550 triode drive
Can control with single-chip microcontroller IO directly
Working voltage: 5V
Board size: 22 (mm) x 12 (mm)

Pin Configuration:

50 VCC
51 Input
52 Ground