

```

#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <elapsedMillis.h>

elapsedMillis timeElapsed; //Create an Instance

// A4 SDA
// A5 SCL

LiquidCrystal_I2C lcd(0x3F, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

int ccc = 1;
int sensorPin = A1; //Poti
int sensorValue = 0;
int freq = 0;
const int freqOut = 11;
unsigned int rpm;

int analogInput = A0; // Batt. Spannung
float vout = 0.0;
float vin = 0.0;
float R1 = 99860.0; // resistance of R1 (100K) -see text!
//http://www.electroschematics.com/9351/arduino-digital-voltmeter/
float R2 = 9910.0; // resistance of R2 (10K) - see text!
int value = 0;

void playTone(int freqq)
{
    tone(freqOut, freqq);

    //lcd.clear(); Macht Probleme weil LCD flickert

    lcd.home ();
    lcd.setCursor(11, 0); //clear letzte Stelle von rpm
    lcd.print(" ");
    lcd.setCursor(0, 0);
    lcd.print("RPM:");
    lcd.setCursor(7, 0);
    lcd.print(rpm);
    lcd.setCursor(0, 1);
    lcd.print("BAT:");
    lcd.setCursor(7, 1);
    lcd.print(vin);
    lcd.setCursor(13, 1);
    lcd.print("V");
}

void setup() {

    lcd.begin(16, 2);

    pinMode(freqOut, OUTPUT);
    pinMode(analogInput, INPUT);

```

```

    //lcd.backlight();

    lcd.setCursor(2, 0);
    lcd.print("Y A M A H A");
    lcd.setCursor(2, 1);
    lcd.print("YPVS  Tester");

    delay(3000);

    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("Version");
    lcd.setCursor(10, 1);
    lcd.print("T101");
    delay(1000);
    lcd.clear();

    noTone(freqOut);
}

void loop() {

    if (timeElapsed > 1000) //<elapsedMillis.h> Timer Library, Spannung 1x
in der Sekunde abfragen
    {
        // read the value at analog input
        value = analogRead(analogInput);
        vout = (value * 5) / 1024.0;
        vin = vout / (R2 / (R1 + R2));
        if (vin < 0.09) {
            vin = 0.0; //statement to quash undesired reading !
        }
        timeElapsed = 0; //<elapsedMillis.h> Timer Library
    }

    sensorValue = analogRead(sensorPin);
    //freq = ceil(sensorValue/5)*2*2;
    //freq = ceil(sensorValue/20)*5*2;    //rpm 900 - 15000
    freq = map(sensorValue, 0, 1023, 50, 500); //rpm 1500 - 15000
    rpm = freq * 30;

    playTone(freq);

    delay(100);
}

```