Electrical Network Analysis

EL-2004

Lab Project

Fall 2024

Construction a Digital Wattmeter using Arduino for given specifications

Submitted by:

Muhammad Ibrahim (23i – 6090)

Mohammad Ahtesam (23i – 6060)

Date:

4th January, 2024

Lab Instructor

Asad Jameel

Course Instructor:

Muhammad Jafar

National University of Computer and Emerging Sciences, Islamabad

Introduction and Description of the Project:

The AC Wattmeter project aims to measure the power consumption and various electrical parameters of appliances operating on AC (Alternating Current). This project is designed by using components such as resistors, capacitors, an ACS712 current sensor, Arduino Uno and a 16x2 LCD to process voltage and current signals. These signals are analysed to calculate and display multiple parameters, including RMS voltage, RMS current, frequency, power factor, and reactive power. By using Arduino and LCD, the wattmeter provides comprehensive insights into the circuit's performance. This design demonstrates the principles of power measurement.

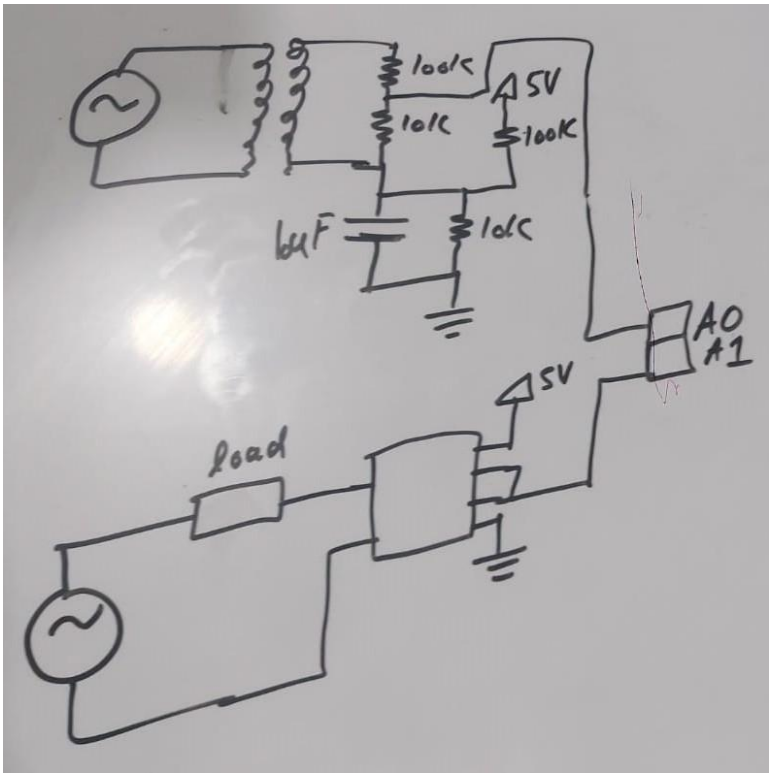
Block and Circuit diagram of the Project:

Block Diagram:

1. **Power Supply:** Provides DC power to the circuit.
2. **Voltage Measurement:** Rectifies and scales the AC voltage for RMS calculation.
3. **Current Measurement:** Rectifies and scales the AC current for RMS calculation.
4. **Phase Measurement:** Determines the phase difference between voltage and current signals.
5. **Calculations:** Computes power factor, real power, reactive power, and apparent power.
6. **LCD Display:** Displays the calculated parameters.

Circuit Diagram:

The circuit diagram includes components such as resistors, capacitors, the ACS712 current sensor, Arduino Uno, and a 16x2 LCD display. Below is the circuit diagram used in the Project:



List of Specifications:

- **Input Voltage Range:** 100V – 250V AC
- **Current Measurement Range:** 0A – 10A
- **Power Calculation Range:** 0W – 2500W
- **Display:** 16x2 LCD
- **Power Supply:** 5V DC
- **Measured Parameters:**
    1. RMS Voltage (V)
    2. RMS Current (A)
    3. Frequency (Hz)
    4. Real Power (W)
    5. Reactive Power (VAR)
    6. Apparent Power (VA)
    7. Phase Difference (Degrees)
    8. Power Factor (P.F)
    9. Value of Capacitor (in μF) to correct P.F to 1.

List of Components Used:

- **Passive Components:**
    1. 5 Resistors of 2(100kΩ), 2(10kΩ), 1(50Ω)
    2. 1 Capacitors 10μF
- **Active Components:**
    1. ACS712 Current Sensor
    2. Arduino Uno
- **Other Components:**
    1. Transformer
    2. LCD Display (16x2)
    3. Potentiometer
    4. Wires and connectors
    5. Breadboard
    6. Power supply (5V DC)

Calculations Performed to Design the Project:

1. **Voltage Divider for voltage Measurement:**
   Resistor values were chosen to step down the AC voltage safely for rectification and RMS calculation.
2. **Current Measurement Design:**
   ACS712 sensor is used to scale and convert AC current for RMS calculation.
3. **Power Calculation Formulas:**
   Real Power = P = Vrms * Irms * cos(θ)
   Reactive Power = Q = Vrms * Irms * sin(θ)
   Apparent Power = S = Vrms * Irms
4. **Power Factor (P.F):**
   Power Factor = P.F = cos(θ)

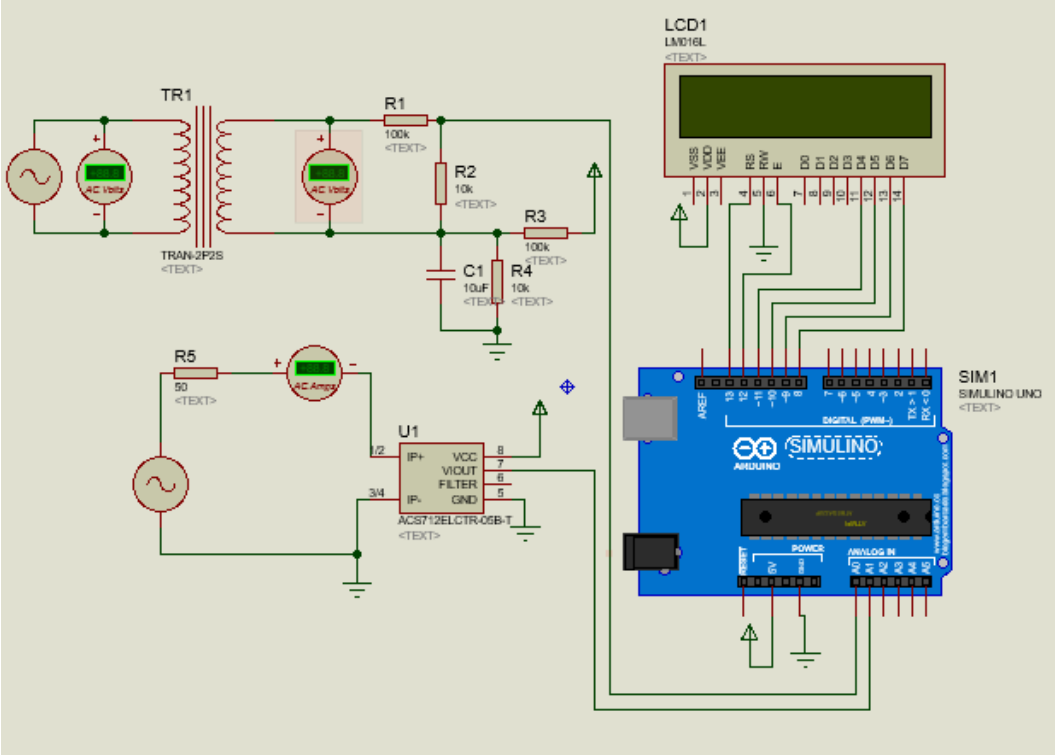
Arduino Code Used in Project:

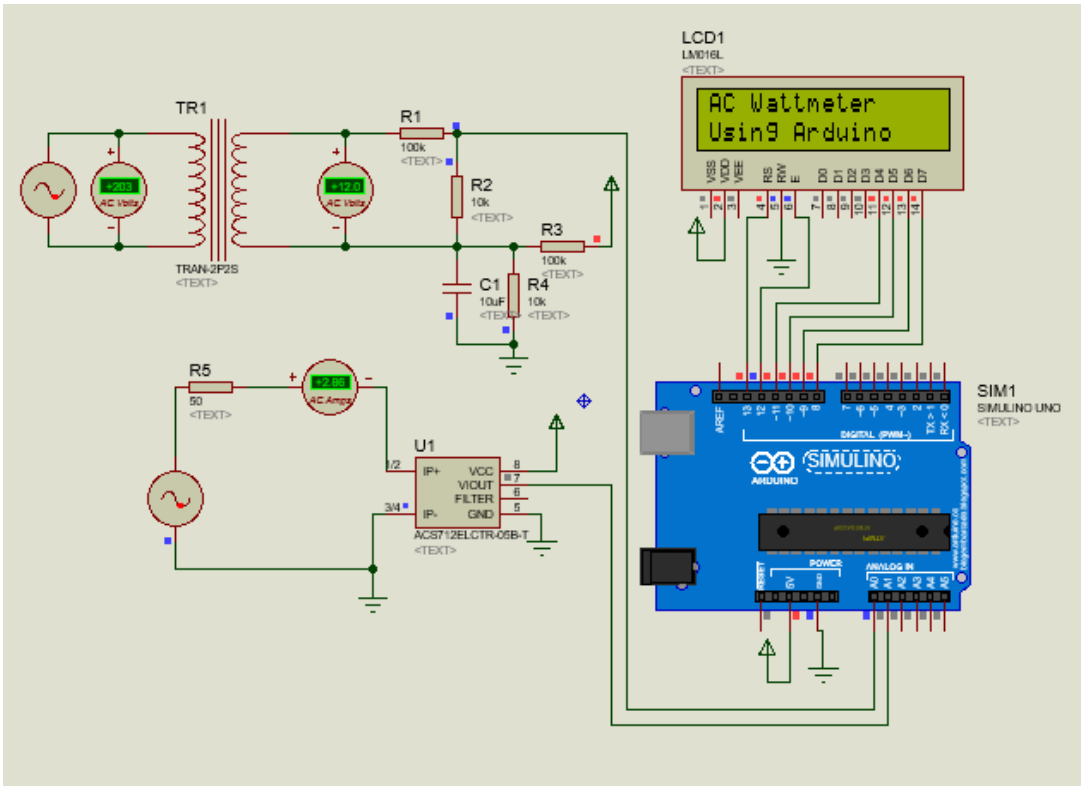
```
#include <LiquidCrystal.h>
#include "EmonLib.h"
#include <math.h>
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);
EnergyMonitor emon1;
const int Sensor_Pin = A1;
int sensitivity = 185;
int offsetvoltage = 2542;
const float freqNominal = 50.0;
float calculateFrequency() {
  unsigned long pulseTime = pulseIn(A0, HIGH);
  if (pulseTime > 0) {
    return 1.0 / (2.0 * pulseTime * 1e-6);
  }
  return freqNominal;
}
void setup() {
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("AC Wattmeter");
  lcd.setCursor(0, 1);
  lcd.print("Using Arduino");
  delay(200);
  lcd.clear();
  emon1.voltage(A0, 187, 1.7);
  emon1.current(A1, sensitivity);
}
void loop() {
  emon1.calcVI(20, 200);
  float Vrms = emon1.Vrms;
  float Irms = emon1.Irms;
  float realPower = emon1.realPower;
  float apparentPower = Vrms * Irms;
  float powerFactor = realPower / apparentPower;
  float reactivePower = sqrt(apparentPower * apparentPower - realPower * realPower);
  float phaseAngle = acos(powerFactor) * (180.0 / M_PI);
  float capacitance = (reactivePower / (2 * M_PI * freqNominal * Vrms * Vrms)) * 1e6;
  float frequency = calculateFrequency();
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("V:");
  lcd.print(Vrms, 1);
  lcd.print(" I:");
  lcd.print(Irms, 1);
  lcd.setCursor(0, 1);
  lcd.print("F:");
  lcd.print(frequency, 1);
  lcd.print("Hz P:");
```

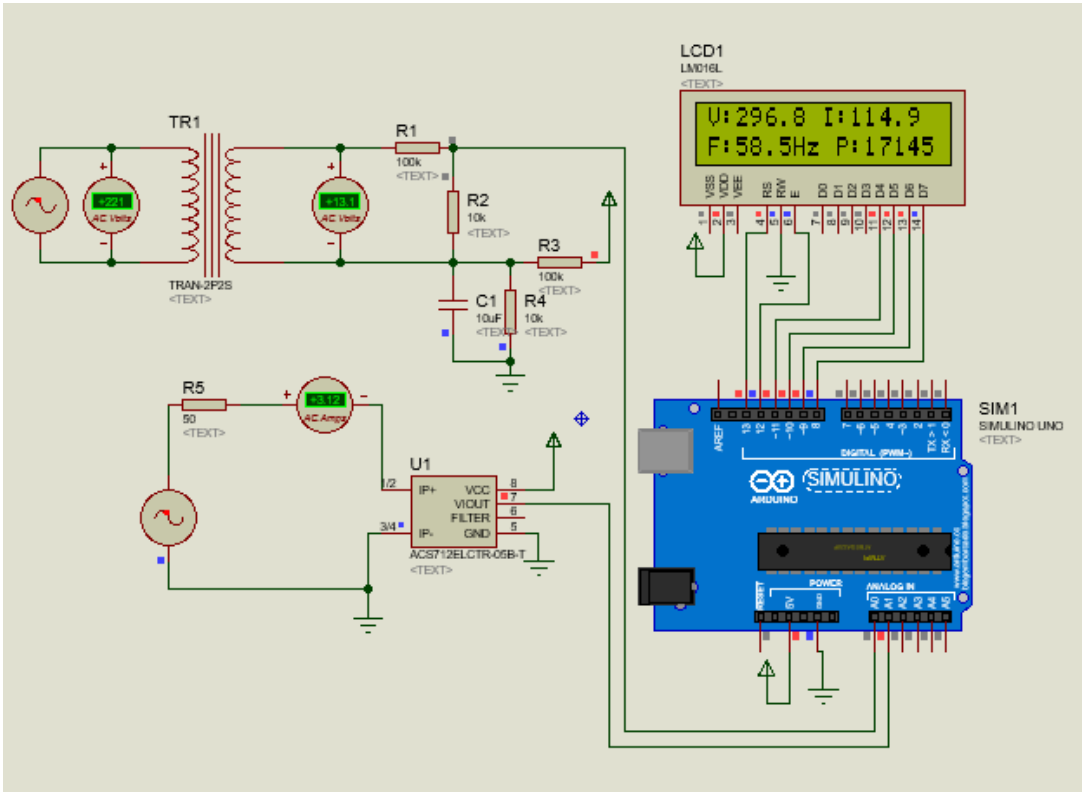
```
  lcd.print(realPower, 0);
  lcd.print("W");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("PF:");
  lcd.print(powerFactor, 2);
  lcd.print(" Ph:");
  lcd.print(phaseAngle, 1);
  lcd.setCursor(0, 1);
  lcd.print("Q:");
  lcd.print(reactivePower, 1);
  lcd.print("VAR C:");
  lcd.print(capacitance, 0);
  lcd.print("uF");
  delay(3000);
}
```

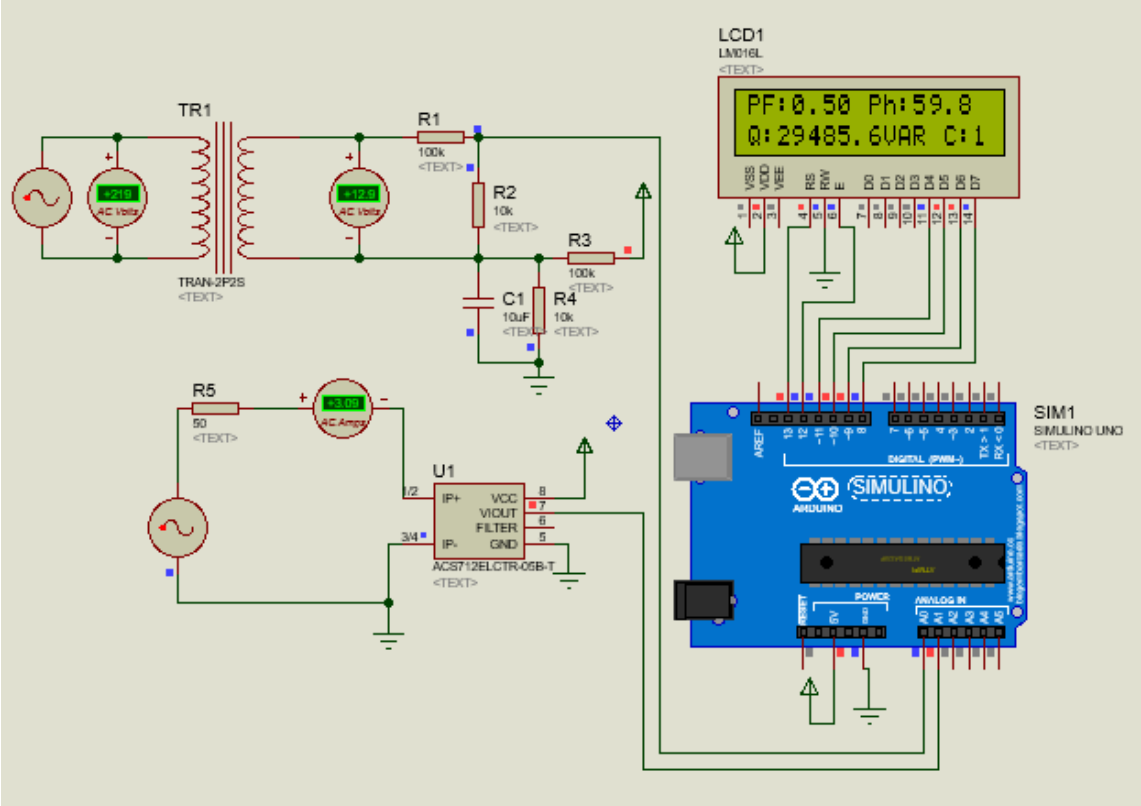
Proteus Simulation File:

The Proteus simulation File includes the circuit design and simulation of the wattmeter's functionality. The components were tested with simulated AC inputs to verify power calculations.

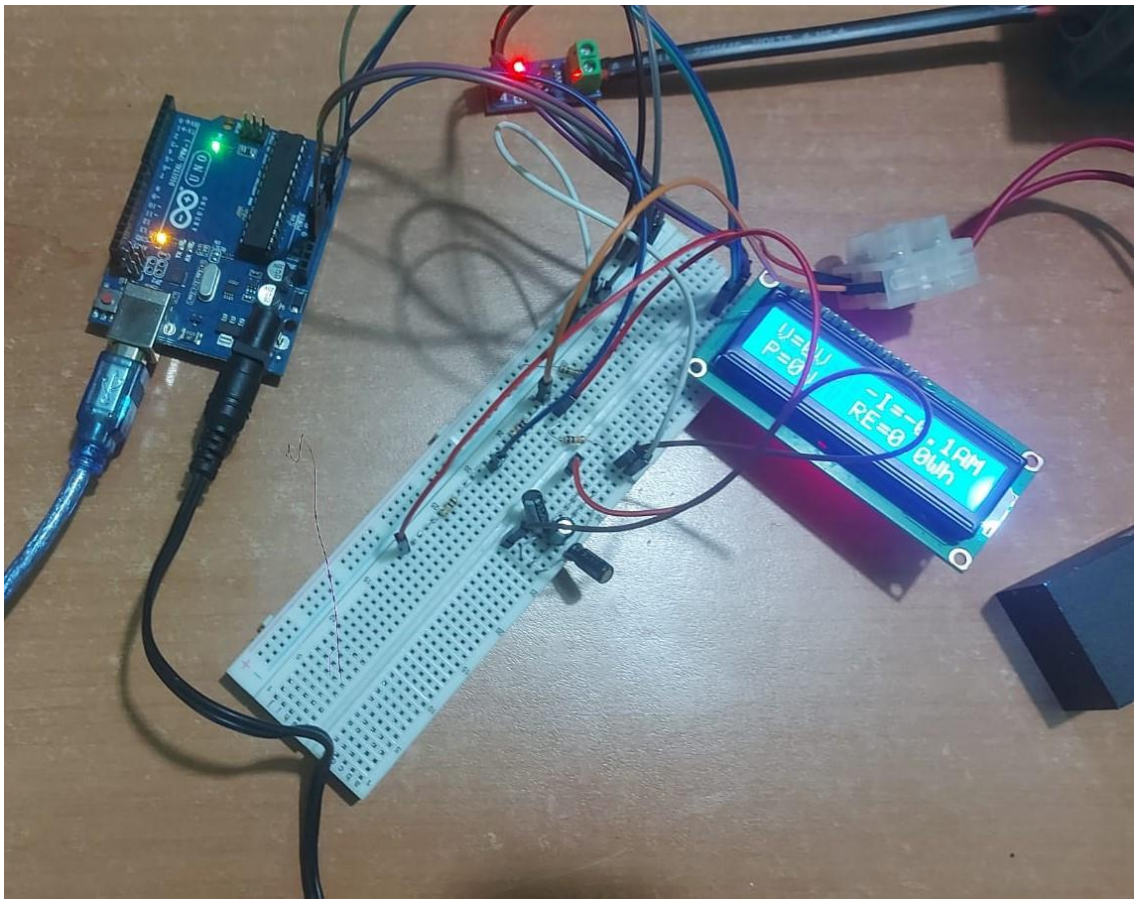


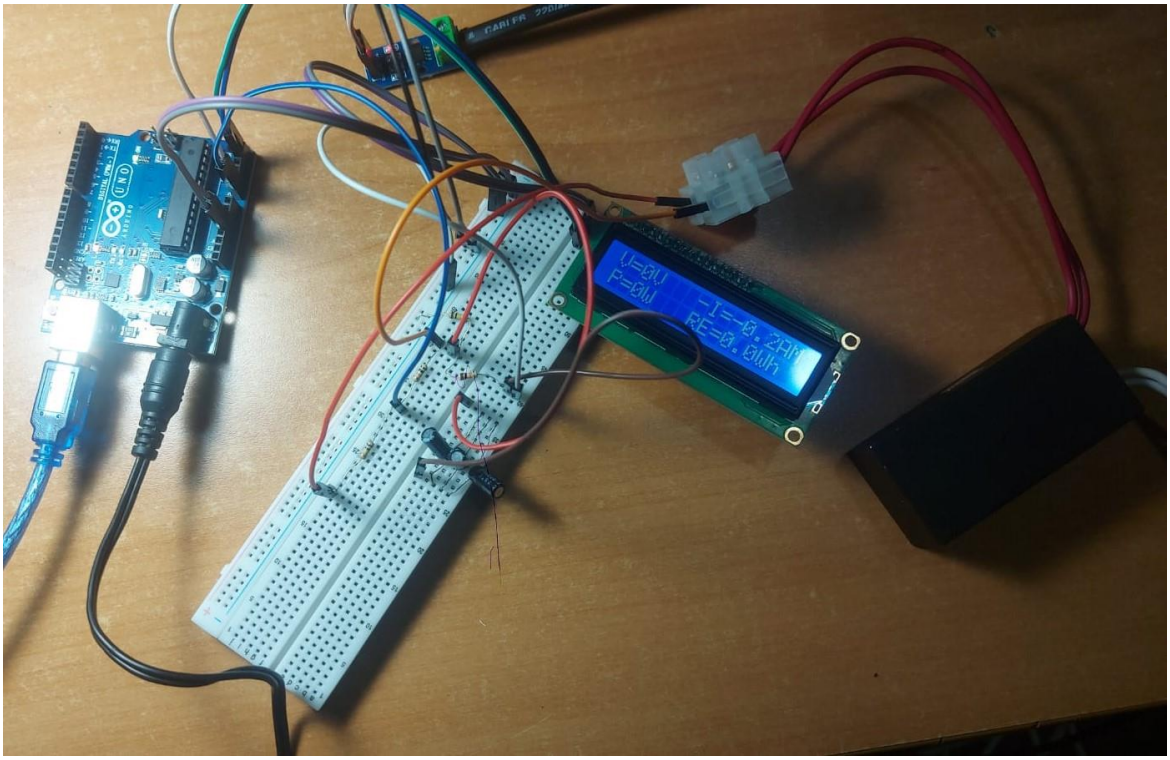

LCD1
LM016L
AC Wattmeter
Usin9 Arduino
TR1
TRAN-2P2S
R1
100k
R2
10k
R3
100k
C1
10uF
R4
10k
R5
50
U1
ACS712ELCTR-05B-T
SIM1
SIMULINO UNO

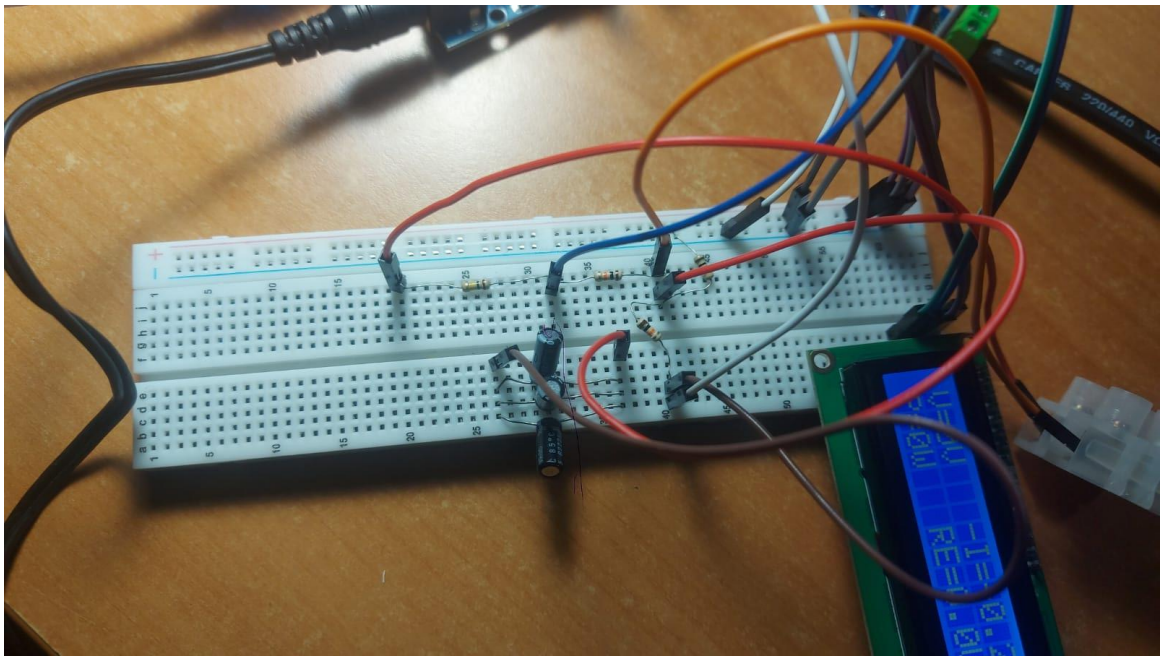

LCD1
LM016L
V:296.8 I:114.9
F:58.5Hz P:17145
TR1
TRAN-2P2S
R1
100k
R2
10k
R3
100k
C1
10uF
R4
10k
R5
50
U1
ACS712ELCTR-05B-T
SIM1
SIMULINO UNO



Practical Design:







Summary:

The AC Wattmeter project successfully demonstrates the ability to measure and display various electrical parameters of AC circuits, including RMS voltage, RMS current, real power, reactive power, apparent power, phase difference, power factor, and capacitor value for power factor correction. The design relies on components, showcasing the principles of power measurement.