



Temperature Measurement using Arduino-Based Sensor

Ibragimova Nigora

Graduate of Webster University, Tashkent

Annotation

In this paper, an Arduino microcontroller and an electronic temperature sensor are used to design and implement a temperature measurement system. This work's primary goal is to show how contemporary microcontroller technology and the physical principles of temperature measurement can be combined to produce a straightforward, affordable, and precise measuring tool. Real-time ambient temperature readings are displayed on an LCD screen or serial monitor by the system. According to experimental findings, the instrument yields accurate and consistent readings, which makes it valuable for environmental monitoring, laboratory measurements, and educational experiments.

Keywords: Arduino, Temperature Measurement, LM35 Sensor, DHT11, Microcontroller, Physics Experiment, Electronic Measurement, Real-Time Data, Thermodynamics, Sensor Technology.



This is an open-access article under the [CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/) license

Introduction.

One of the most basic physical variables, temperature, affects practically all chemical, biological, and physical processes. For experimental analysis and theoretical model validation in physics, precise temperature measurement is essential. Modern electronic thermometers rely on sensors that translate temperature into electrical signals, whereas traditional thermometers are based on the expansion of liquids or metals. This project measures and displays temperature values using an Arduino Uno microcontroller and a temperature sensor (LM35 or DHT11). An open-source microcontroller platform called Arduino makes it simple to interface with a variety of sensors and conduct prototyping. The system bridges the gap between applied technology and physics theory by showing how digital electronics can be used to practically implement physical concepts.

The average kinetic energy of the particles in a substance is its temperature. Thermodynamic principles state that objects transfer heat when their temperatures differ. Sensors in electronic measurement systems pick up temperature variations and translate them into the appropriate electrical outputs.

With a sensitivity of 10 mV/°C, the LM35 temperature sensor generates an analog voltage output proportional to the temperature in Celsius. For instance, the output voltage is roughly 250 mV at 25°C. As an alternative, the DHT11 sensor measures humidity and temperature and offers a

digital output. After reading the sensor's signal and converting it to digital form (if analog), the Arduino microcontroller shows the outcome.

The relationship between the sensor's voltage output (V_{out}) and temperature (T) can be expressed as:

Materials and Methods

Components Used:

- ✓ Arduino Uno board
- ✓ LM35 or DHT11 temperature sensor
- ✓ Breadboard and jumper wires
- ✓ LCD (16x2) or computer serial monitor
- ✓ USB cable and Arduino IDE software

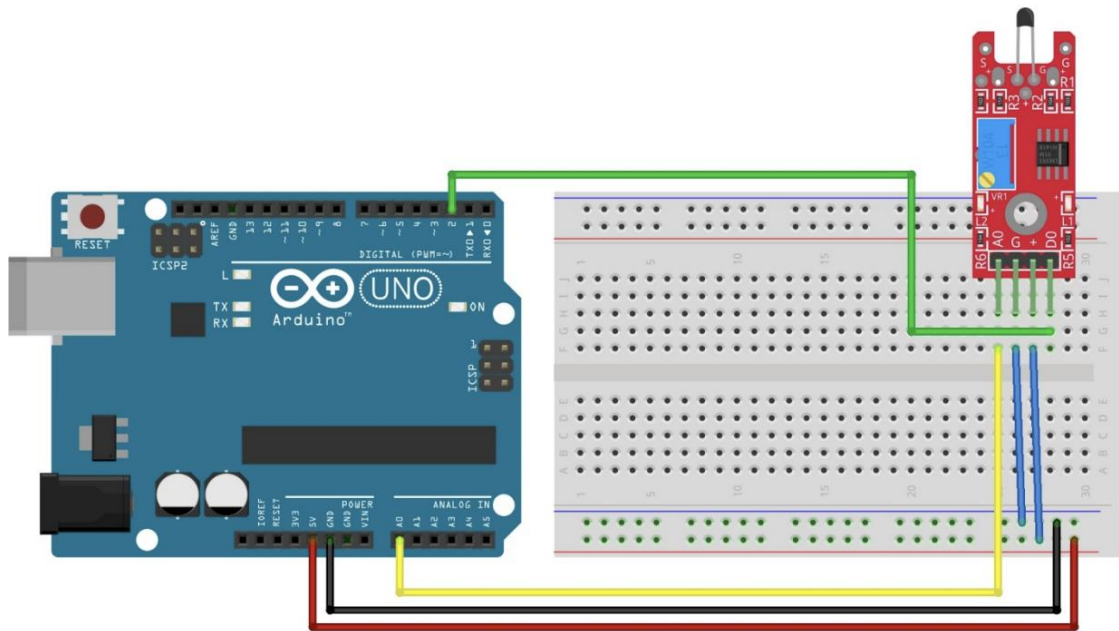
Methodology:

1. The LM35 sensor was connected to the Arduino board through the analog input pin (A0).
2. The Arduino IDE was used to program the board to read the sensor's output voltage and convert it into a temperature value.
3. The measured temperature was displayed on the serial monitor and updated every second.

Circuit Description The LM35 sensor has three pins: VCC, GND, and Output. The VCC is connected to the 5V pin of the Arduino, the GND to the ground, and the Output pin to the analog input A0. The Arduino reads the analog voltage and calculates the corresponding temperature.

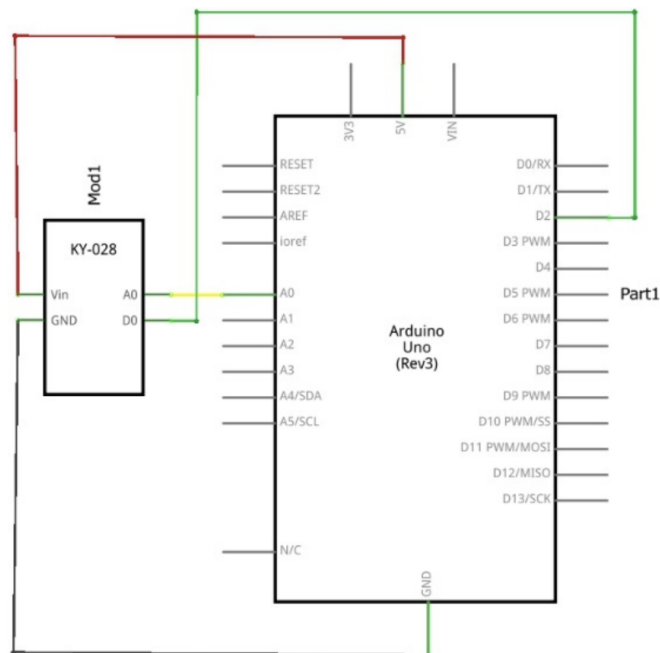
Arduino Code Example:

```
1  int sensorPin = A0;
2  float temperature;
3
4  void setup() {
5      Serial.begin(9600);
6  }
7
8  void loop() {
9      int sensorValue = analogRead(sensorPin);
10     temperature = (sensorValue * 5.0 * 100.0) / 1024.0;
11     Serial.print("Temperature: ");
12     Serial.print(temperature);
13     Serial.println(" °C");
14     delay(1000);
15 }
```



```
#include <ACI_10K_an.h>
#include <Aci_10K.h>
int led = 13;
int digitalPin = 2;
int analogPin = A0;
int digitalVal;
int analogVal;
void setup() {
pinMode(led, OUTPUT);
pinMode(digitalPin, INPUT);
Serial.begin(9600);
}
void loop() {
digitalVal = digitalRead(digitalPin);
if (digitalVal == HIGH) {
digitalWrite(led, HIGH);
} else {
digitalWrite(led, LOW);
}
analogVal = analogRead(analogPin);
Serial.print("digitalVal = ");
Serial.print(digitalVal);
Serial.print(" AnalogVal = ");
```

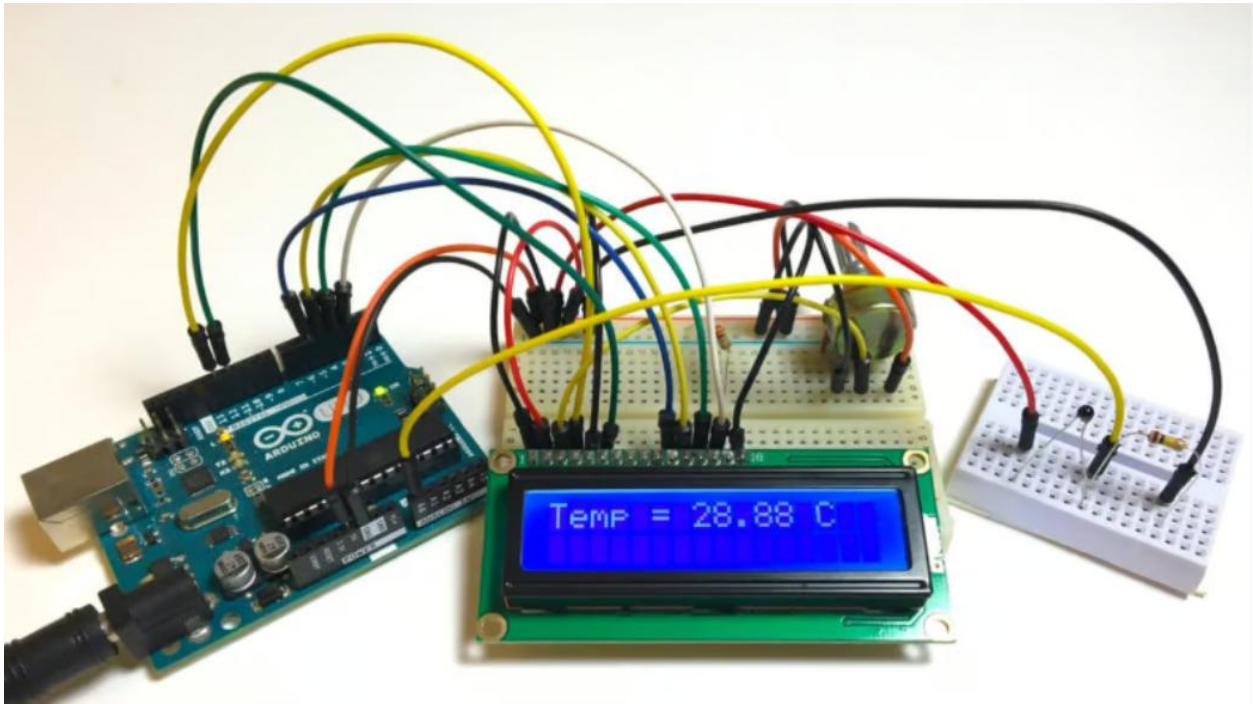
```
Serial.print(analogVal);  
Aci_10K an10k;  
float temperature = an10k.getTemp(analogVal);  
Serial.print(" temperature = ");  
Serial.println(temperature);  
delay(1000);  
}
```



The system was tested under different environmental conditions. The readings from the Arduino-based sensor were compared with a standard digital thermometer. The results showed a small deviation of $\pm 0.5^{\circ}\text{C}$, indicating good accuracy for general laboratory use. The device responded quickly to temperature changes and was able to measure in the range of 0°C to 100°C . The LM35 sensor proved to be stable, linear, and easy to interface with the Arduino board. The simplicity of the circuit and the low cost of components make this system ideal for physics experiments, student laboratories, and small-scale environmental monitoring projects.

Furthermore, the project demonstrates how physics concepts—such as temperature, energy transfer, and thermoelectric effects—can be explored through hands-on experimentation with microcontrollers.

The creation of a temperature measurement system based on Arduino marks a useful fusion of contemporary digital technology and physics. In addition to its usefulness in providing precise temperature readings, this project acts as a teaching tool by bridging the gap between abstract physics ideas and practical experimentation.



In addition to strengthening their grasp of thermal physics and measurement concepts, students acquire critical abilities in electronics, programming, and data analysis through the system's design, construction, and programming. Such interdisciplinary education fosters creativity, innovation, and problem-solving skills—qualities that are becoming more and more crucial in engineering practice and scientific research.

Conclusion.

More generally, this work demonstrates how basic physical quantities such as temperature can be investigated using inexpensive, easily accessible instruments. By turning theoretical concepts into practical experiments, these kinds of projects improve the teaching and learning process. They encourage a greater interest in experimental science and assist students in seeing how physics is applied in modern technology. Future advancements could involve adding wireless communication, connecting to data visualization platforms, or extending the system to measure additional environmental parameters. Even so, the Arduino-based temperature measurement project successfully illustrates how interactive, technologically driven learning experiences can advance physics education.

References:

1. Arduino Official Documentation. (2024). <https://www.arduino.cc>
2. Arduino for Windows 2.1.0
3. Fundamentals of Physics 10th Edition by Davi Halliday, RoberResnick, JearlWalker
4. Articles about Temperature and Humidity Sensor on an Arduino
5. DHT11 Sensor Datasheet. (2022). Aosong Electronics Co., Ltd.