Technical Documentation for the Weather Station

Project No. 23

Rangu Elena-Daniela

ICC 2.1

1. **Description**

The Weather Station project represents a robust and efficient solution for temperature monitoring, implemented on an Arduino microcontroller. Using a temperature sensor and standard electronic components, the station provides a simple and intuitive interface for the user, along with advanced monitoring and control functionalities.

The source code is optimized for performance and efficiency, ensuring stable and reliable operation of the system. The modular structure of the code facilitates understanding and extension of functionalities, allowing for the addition of additional features or customization of system behavior as needed.

The hardware components are integrated into a simple and coherent scheme, minimizing energy consumption and optimizing the use of available resources. The temperature sensor provides accurate measurements, while the LCD display and LEDs ensure an intuitive visual interface for the user.

1. **Software Implementation**

The code is written in the C++ programming language and is compatible with the Arduino IDE.

The LiquidCrystal.h library is used for controlling the LCD display.

Timers are utilized to read the temperature at regular intervals and to update the LCD display.

LEDs are controlled using digitalWrite() instructions based on the temperature readings.

1. **Hardware Implementation**

**The LCD is connected to the Arduino using the specified communication pins.**

The temperature sensor is connected to an analog pin.

The start/stop button is connected to a digital pin.

The LEDs are connected to digital pins.

1. **Hardware Components**

- Arduino Uno R3

Fig. 1 – Arduino Uno

-Temperature Sensor LM35

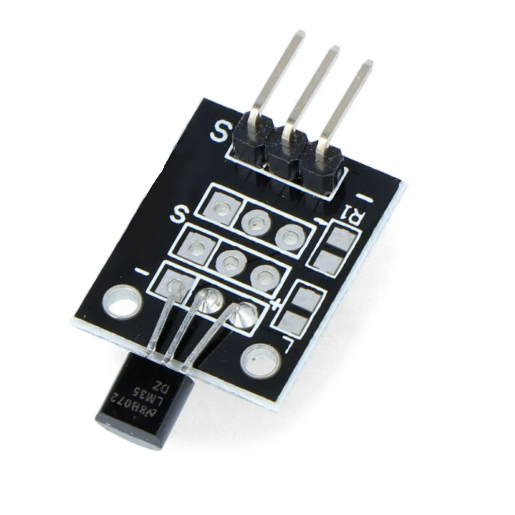


Fig. 2 --Temperature Sensor LM35

An analog temperature sensor that provides a voltage proportional to the temperature in degrees Celsius. It is accurate and easy to use, making it ideal for temperature monitoring applications.

- Tactile button



Fig. 3 - Tactile button

- LCD 16x2

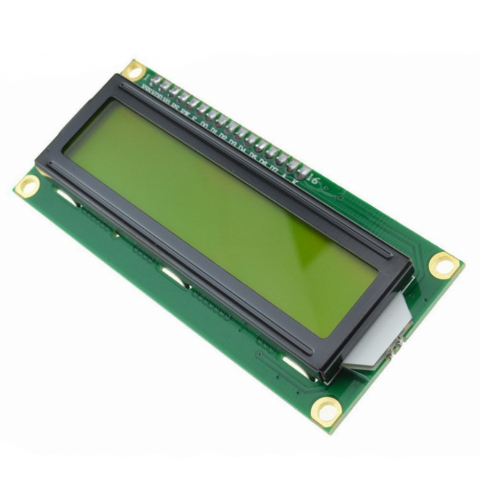


Fig. 4 – LCD 16x2

-LEDs

A row of different colored leds

Description automatically generated

Fig. 5 – LEDs

5. Connection Scheme

a. LM35 Temperature Sensor

- VCC pin connected to Arduino's 5V pin

- OUT pin connected to Arduino's A0 pin

- GND pin connected to Arduino's GND pin

b. LCD Display

- RS pin connected to Arduino's 3 pin

- EN pin connected to Arduino's 4 pin

- DB4-DB7 pins connected to Arduino's 10-13 pins

c. Tactile Button

- One end connected to Arduino's 2 pin

- Other end connected to Arduino's GND pin

d. LEDs

- One end of each LED connected to Arduino's 5V pin (through a 220 ohm resistor)

- The other end of the red LED connected to Arduino's 5 pin

- The other end of the yellow LED connected to Arduino's 6 pin

6. Code Description

Global Variables

- systemOn: A boolean indicating whether the station is turned on or off.

- minTemp: Variable to store the minimum temperature.

- maxTemp: Variable to store the maximum temperature.

- lastTempUpdate: Variable for the next temperature update.

- tempUpdateInterval: Temperature update interval, set to 1500 ms.

setup() Function

- Initializes ports, including the LCD and button.

loop() Function

- Checks the button state and changes the system state (on/off).

- Updates the temperature every 1.5 seconds and displays the values on the LCD.

- Updates the LEDs based on the read temperature.

readTemperature() Function

- Reads the analog value of the temperature sensor and converts it to Celsius.

updateMinMax(temp) Function

- Updates the minimum and maximum temperatures based on the current temperature.

updateLEDs(temp) Function

- Turns on the LEDs based on the current temperature: red for high temperatures, yellow for low temperatures.

7. Pin Declarations and Initializations:A screenshot of a computer

Description automatically generated

These lines define the pin numbers for different components such as the LED, button, and temperature sensor.

1. **Initializing the LCD:**

This line initializes the LCD object with the specified pins for communication with it.

1. **Reading the temperature and the analog value:**

A screen shot of a computer code

Description automatically generated

8. TinkerCAD schematic

A circuit board with wires and a display

Description automatically generated The functionality of the schematic when the Weather Station is turned off.

The functionality of the schematic when the Weather Station is turned on.

A circuit board with wires and a display

Description automatically generated

9. Code

A screenshot of a computer program

Description automatically generated

10. Conclusion and Possible Extensions

The Weather Station project represents a versatile and efficient temperature monitoring system, offering the possibility of extension to include other functions such as data logging or humidity display. Both the code and the connection scheme are designed to be easily understood and adaptable to various hardware configurations. With a modular and clear structure, this project can be customized and expanded according to the specific needs of the user.

To enhance the project, additional functions can be added, such as data logging in a local database or internet connectivity for transmitting data to a remote server. Integrating additional sensors, such as humidity or atmospheric pressure sensors, can provide additional and more precise information about local weather conditions. Additionally, optimizing the code to reduce energy consumption and improve system efficiency can be another aspect to consider in the development of this project.