**PRACTICAL –4**

**Aim: - Arduino programming with serial monitor, Temperature sensor.**

**Conclusion**

This practical session focused on consolidating and expanding Arduino programming skills through a series of experiments that utilized serial communication, temperature sensors, and LED control. Participants gained experience working with various Arduino functions specifically designed for serial communication and data manipulation.

The session began with a review of essential Arduino functions for serial communication, including:

* Serial.begin(): Initializes serial communication at a specified baud rate.
* Serial.end(): Terminates serial communication.
* Serial.read(): Reads a single byte of incoming serial data.
* Serial.write(): Writes a single byte of data to the serial port.
* Serial.print(): Prints a string or variable value to the serial monitor.
* Serial.println(): Prints a string or variable value followed by a newline character to the serial monitor.
* Serial.available(): Checks if data is available on the serial port for reading.

The practical exercises provided hands-on experience with these functions. The first experiment involved controlling LED brightness using a potentiometer while displaying the corresponding voltage level on the serial monitor. This exercise reinforced concepts of analog input, data conversion, and serial data transmission.

The second experiment explored temperature sensing with the Arduino. Participants interfaced a temperature sensor with the board and wrote code to read the temperature data. This data was then displayed on the serial monitor, allowing participants to observe the temperature readings in real-time.

The final experiment integrated temperature data with LED control, creating a more complex application. Participants wrote code to control an RGB LED based on the measured temperature. The code assigned specific colors (red, green, and blue) to temperature ranges, effectively creating a visual representation of the temperature using the LED.

By successfully completing these experiments, participants have honed their Arduino programming skills in the areas of serial communication, sensor data acquisition, and LED control. They have also gained experience in applying conditional statements (if-else) to create decision-making logic within their code. These combined skills will empower them to tackle more intricate projects involving sensor data processing and actuator control in future endeavors.