

1)Objective: Design a temperature monitoring system that reads temperature data from a sensor and triggers an alarm if the temperature exceeds a predefined threshold.

Requirements:

Read temperature data from a temperature sensor at regular intervals.

- Compare the read temperature with a predefined threshold.
- If the temperature exceeds the threshold, activate an alarm (e.g., LED or buzzer).
- Include functionality to reset the alarm.

ALGORITHM

- 1.Start
- 2.Initialise the system Components
- 3.Initialise threshold Value
- 4.Continuously read temperature values from the sensor at predefined intervals.
- 5.If the read temperature exceeds the predefined threshold, activate the alarm.
- 6.Reset the alarm
- 7.Repeat steps 4 to 6
- 8.Stop

2)

Motor Control System

Objective: Implement a motor control system that adjusts the speed of a DC motor based on user input.

Requirements:

- Use a potentiometer to read user input for desired motor speed.
- Control the motor speed using PWM (Pulse Width Modulation).
- Display the current speed on an LCD.

ALGORITHM

- 1.Start
- 2.Initialize System Components
- 3.Read the input from user using potentiometer
- 4.Adjust the motor speed using PWM
- 5.Display the motor's current speed on the LCD.
- 6.Stop

3)Problem Statement 3: LED Blinking Pattern

Objective: Create an embedded system that controls an array of LEDs to blink in a specific pattern based on user- defined settings.

Requirements:

Allow users to define blink patterns (e.g., fast,slow).

- Implement different patterns using timers and interrupts.
- Provide feedback through an LCD or serial monitor.

ALGORITHM

- 1.Start
- 2.Initialize timers and interrupts
- 3.Create a list of patterns and define time interval for each pattern
- 4.execute the pattern and display the output through an LCD or serial monitor.
- 5.Stop

4)Objective: Develop a data Logger that collects sensor data over time and stores it in non-volatile memory.

Requirements:

- Read data from sensors(e.g., temperature,humidity) at specified intervals.
- Store collected data in EEPROM or flash memory.
- Implement functionality to retrieve and display logged data

ALGORITHM

- 1.start
- 2.enter the time interval
- 3.Read Data from sensors at specified time interval provided by the user
- 4.Store the data in EEPROM or flash memory.
- 5.Implement functionality to retrieve and display the logged data

5)Psuedocode for simple calculator

enter num1 and num2

enter operator

switch operator:

case '+':

result=num1+num2

print result

break

case '-':

result=num1-num2

print result

break

```

case '*':
    result=num1*num2
    print result
    break

case '/':
    if(num2!=0)
    then
        result=num1/num2
        print result
    else
        print invalid operation
Default:
    print invalid

end

```

6)Pseudocode for factorial of a number

```

enter num
if num<0
    enter positive integer
else
    fact=1
    for i=1 to num
        fact=fact*i
    end for
    print factorial of the number

```

7)Pseudocode for factorial of a number using recursion

```

Function Factorial(n)
    if n <= 1 then
        return 1
    else
        return n * Factorial(n - 1)
    endif
End function

```

8) Smart Irrigation System

Objective: Design a smart irrigation system that automatically waters plants based on soil moisture levels and environmental conditions. The system should monitor soil moisture and activate the water pump when the moisture level falls below a predefined threshold.

PUSEDOCODE

enter the threshold value(T)
enter the time of the day(time)
enter the current soil moisture in percentage(c)
if $c > T$ and $\text{time} \geq 6 \text{ AM}$ and $\text{time} \leq 6 \text{ PM}$
 activate pump
 print current moisture level
else
 print watering is not needed
 print current moisture level

