Embedded Systems Intern Assignment - upliance.ai

Assignment: *Build a Basic Heater Control System*

Problem Statement:

Design and implement a simple heater control system using a temperature sensor and an actuating mechanism (simulated heating). The goal is to simulate or build a basic embedded system that turns a "heater" on or off based on temperature thresholds.

Keywords: embedded system, control system, sensors

PART 1- SYSTEM DESIGN

I. Minimum sensors required for heating detection and control

A temperature sensor is the minimum requirement for detecting heat levels and enabling the control logic of the heater system.

Temperature Sensor**:** DHT22 Temperature & Humidity Sensor (1 sensor handles both readings). Temperature is used for heater control**.** Humidity is optional now but allows for future features like air quality or moisture control.

Why this sensor is suitable:

- Provides accurate temperature readings (±0.5°C) for control logic
- Integrated humidity sensing enables future enhancements
- Uses a single digital data pin (efficient pin usage)
- Supported by Arduino libraries for easy integration

II. Recommended Communication Protocol

Protocol: Single-Wire Digital Communication (proprietary protocol used by DHT22)

Justification:

- Low pin usage**:** Only one data pin required
- Reliable for short-range indoor environments
- Arduino-compatible with well-supported libraries (Adafruit DHT)

III. Block Diagram

Overall System Architecture

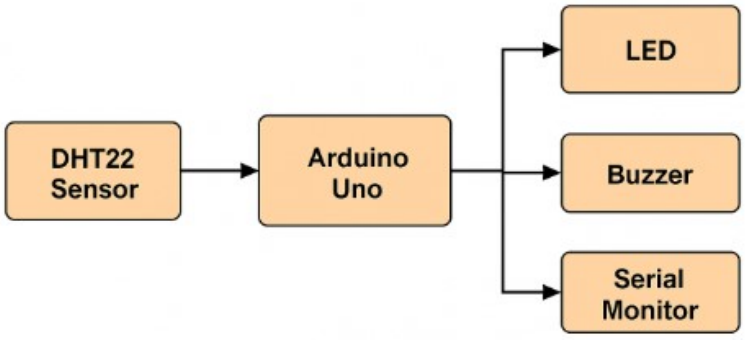

Fig 1. Block diagram and flow of heater control system

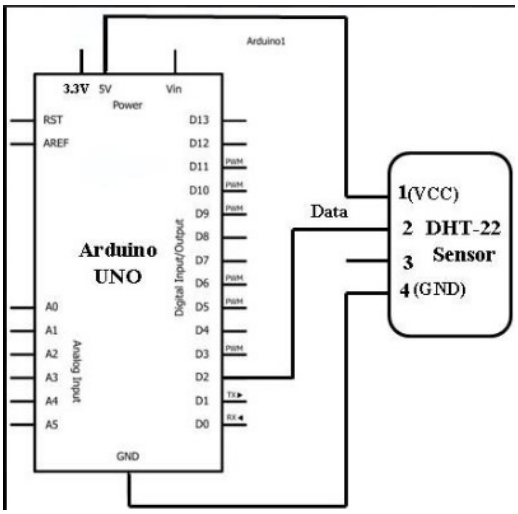

Fig.2 Pin diagram of Arduino UNO connected to DHT22

Based on Fig 1. the flow of block diagram is explained below:

Block Flow Explanation:

1. DHT22 Sensor → Measures temperature and sends data to Arduino via digital signal.
2. Arduino Uno → Processes data using a 5-state logic (state machine).
3. LED → Simulates heater ON/OFF control.
4. Buzzer → Alerts user if temperature exceeds overheat threshold.
5. Serial Monitor → Logs temperature, humidity, and system state in real-time.

IV. Operational States

The DHT22 sensor and Single-Wire Digital Communication protocol were chosen for their simplicity, accuracy, and scalability, ensuring minimal wiring and reliable data transfer. Table below shows the operational states of the system:

State	Description
Idle	System is passive; heater OFF, waiting for temp to fall below threshold.
Heating	Heater ON (LED ON); system actively heating
Stabilizing	Approaching target; careful monitoring to avoid overshoot
Target Reached	Heater OFF; desired temperature reached and maintained
Overheat	Heater OFF; buzzer ON; alerts user to shut down or intervene

V. Future Roadmap

A basic heater control system can be effectively designed using the Arduino UNO microcontroller and DHT22 temperature-humidity sensor. The following enhancements can be implemented in future versions:

1. Multiple Heating Profiles- User-selectable profiles for different temperature targets (e.g., slow heat vs. fast heat).
2. Mobile App Control (Future ESP32 Upgrade)-Add BLE/WiFi support for real-time control and monitoring.
3. Humidity Based Safety Cutoff- Turn off heater in low-humidity environments (e.g., dry air over heating risk).
4. Data Logging & Analytics- Store historical temperature/humidity trends in SD card or cloud.
5. PID Control Loop-Replace simple ON/OFF logic with smooth PID control for better stability.