**CHAPTER 3**

**PROPOSED SYSTEM**

**3.1 Introduction**

The increasing demand for automation in household and industrial cleaning applications has led to the development of smart floor cleaning systems. Traditional floor cleaning methods require manual effort and are often time-consuming, leading to inefficiencies in maintaining hygiene. Additionally, labour-intensive methods may result in inconsistent cleaning quality and increased physical strain on users. To overcome these challenges, we propose a Smart Floor Cleaning System that integrates automation, mobility, and user control for efficient cleaning operations.

This system is designed using an Arduino UNO microcontroller, a motor driver L298N, a Bluetooth module HC-05, servo motors for arm lifting, a water pump for spraying water, and is controlled via a mobile application. The Smart Floor Cleaning System enhances cleaning efficiency, reduces human effort, and ensures a systematic cleaning process through real-time control and automation. The mobile application interface allows users to select cleaning modes, start or stop cleaning, and monitor real-time progress. This makes the system more convenient and accessible for users across different environments, including homes, offices, and public spaces.

The system operates efficiently on both hard and smooth surfaces such as tile, marble, and wood, ensuring a versatile cleaning experience. It integrates a combination of wet and dry cleaning functionalities, allowing users to select the appropriate mode based on the level of dirt and surface type. The cleaning mechanism consists of rotating brushes for sweeping dust, a water pump for spraying, and a suction unit for drying the floor, ensuring a spotless finish. The system’s modular design enables easy maintenance and upgrades, ensuring long-term usability and flexibility.

Furthermore, the system is adaptable and scalable, making it suitable for a wide range of applications. It can be modified to include additional features such as obstacle detection, automated path planning, and sensor-based navigation, allowing it to function autonomously in future iterations. Integration with artificial intelligence and IoT technology can further improve its efficiency, enabling smart scheduling and voice control compatibility through virtual assistants like Google Assistant and Alexa. The Smart Floor Cleaning System represents a step toward a more advanced, automated, and intelligent approach to maintaining cleanliness in residential and commercial spaces.

**3.2 Advantages**

* Automated Cleaning Process: The system automates floor cleaning, reducing manual effort and ensuring consistent and efficient cleaning operations.
* Remote Control and Operation: Users can control the system wirelessly via a mobile application using Bluetooth, providing flexibility and ease of use.
* Dual-Mode Cleaning: Supports both wet and dry cleaning modes, making it suitable for different floor types and levels of dirt accumulation.
* Time and Energy Efficiency: Reduces cleaning time while optimizing energy consumption, making it an efficient alternative to manual cleaning.
* Customizable Cleaning Modes: Users can select different cleaning modes based on surface type and cleaning requirements.
* Enhanced Hygiene and Health Benefits: Automated cleaning ensures thorough and regular maintenance, reducing dust, allergens, and bacterial growth.
* User-Friendly Interface: The mobile application offers an intuitive and easy-to-use interface for controlling the system.
* Low Maintenance and Cost-Effective: Designed with affordable and replaceable components, making it easy to maintain and repair.
* Smart Data Logging and Performance Tracking: The system records cleaning activities, helping users track performance and optimize future cleaning sessions.
* Adaptability and Scalability: Can be upgraded with additional features like obstacle detection, automated navigation, and AI-based cleaning optimization.

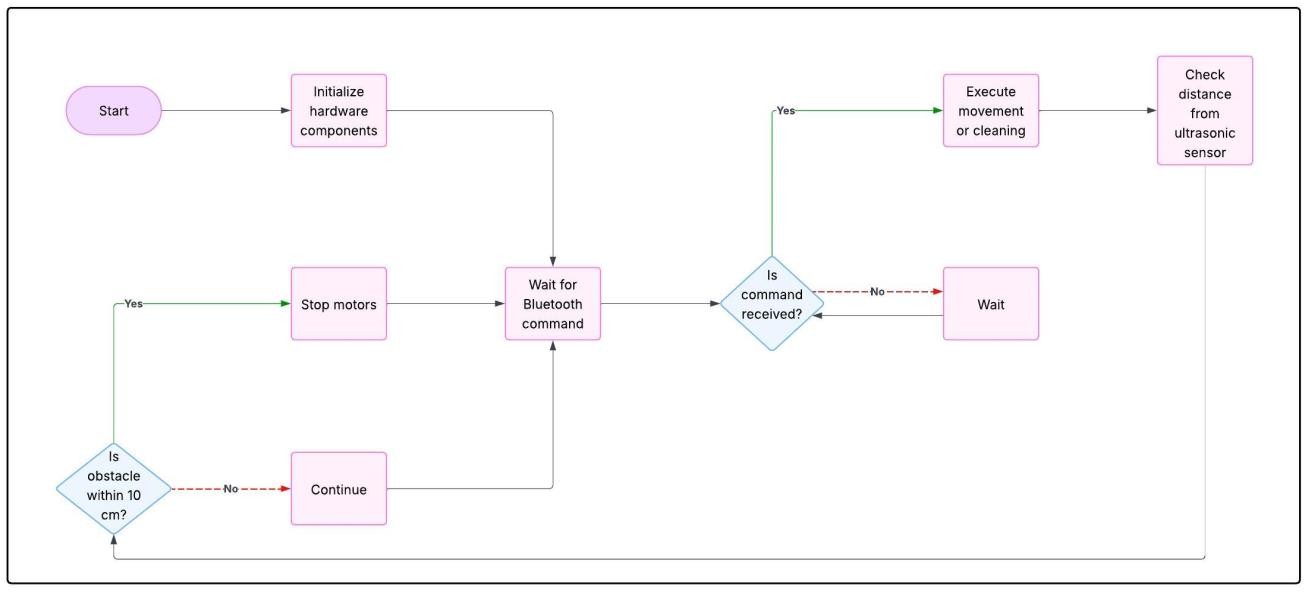
**3.3 Work flow of the system**

The Smart Floor Cleaning System follows a structured workflow to ensure efficient and automated floor cleaning. The process begins with the power supply and system initialization, where a rechargeable battery powers the entire system, including the Arduino UNO, motor driver, ECU (Electronic Control Unit), and various sensors. Once the system is powered on, the Arduino initializes all components and establishes communication with the Bluetooth module, sensors, and motors.Next, the user controls the system via the Bluetooth module, which allows for wireless communication between a mobile application and the Arduino UNO. Through the mobile app, users can send commands to start or stop the cleaning operation, control movement, and select different cleaning modes. This enhances user convenience and flexibility in managing the cleaning process.

Once activated, the system collects sensor data and detects obstacles using multiple ultrasonic sensors and an IR sensor. The ultrasonic sensors help in avoiding obstacles and navigating the environment, while the IR sensor aids in detecting surfaces and ensuring smooth operation without collisions. This real-time feedback allows the system to make necessary adjustments for effective cleaning.

For movement, the motor driver controls the motion of the Smart Floor Cleaner by operating four motors (M1, M2, M3, and M4). The front motors (M1 and M2) and rear motors (M3 and M4) work together to ensure smooth navigation across the floor. The movement is coordinated based on user commands and sensor feedback to prevent accidents and optimize coverage.As the system moves, the cleaning mechanism is activated. The ECU regulates the Brushless DC (BLDC) motor, which drives the cleaning brush or suction mechanism. If wet cleaning is selected, the water pump sprays water before the brushes start scrubbing the floor, ensuring effective dirt and stain removal. The servo motors also adjust the cleaning arm height to adapt to different surfaces, improving efficiency.

During operation, the system continuously processes data for real-time navigation and adjustments. If an obstacle is detected, the system alters its path or stops to prevent damage. The motor speed and cleaning functions are dynamically adjusted based on environmental conditions.

****Finally, the system completes the cleaning process and notifies the user. Once the cleaning cycle is finished, the Smart Floor Cleaning System stops operation and sends a notification to the user through the Bluetooth module. The system logs cleaning data for future reference, allowing users to track performance and maintenance needs.This structured workflow ensures that the Smart Floor Cleaning System operates efficiently, providing an automated, user-friendly, and effective cleaning solution for homes, offices, and public spaces.

**Figure 3.1 : Flow Chart Of The Proposed System**

**3.4 Methodology**

**3.4.1 System Design and Architecture**

This phase lays the groundwork for the entire project by outlining the components and how they interact.

* Microcontroller/Processor: Acts as the robot’s “brain.” Popular choices like Arduino or ESP32 handle input from sensors, control the motors, and execute cleaning algorithms. Arduino is often preferred for simpler setups, while ESP32 offers more processing power and Wi-Fi/Bluetooth capabilities.
* Motors and Actuators: These provide mechanical motion. DC motors drive the wheels, allowing the robot to navigate the environment. Servo motors or stepper motors handle more precise tasks like lifting arms, rotating brushes, or adjusting the mopping head.
* Sensors: Ultrasonic sensors help measure distances to obstacles, IR sensors detect nearby objects, and optionally, cameras or visual sensors help the robot identify dirty spots or map the surroundings.
* Power Supply: A rechargeable lithium-ion (Li-ion) battery pack powers the entire system. The capacity is chosen based on the energy requirements of the motors, sensors, and processor.
* User Interface (UI): This can be a physical control panel with buttons or more advanced interfaces like a mobile app or web dashboard, allowing users to start, stop, schedule, or monitor cleaning tasks.

**3.4.2. Path Planning and Navigation**

Efficient movement is key for cleaning coverage and battery conservation.

* Mapping: Sensors like ultrasonic or LIDAR scan the room and create a digital map, enabling the robot to know where it has cleaned and where it needs to go.
* Localization: The robot constantly updates its position relative to the environment using techniques like SLAM (Simultaneous Localization and Mapping). SLAM lets it adapt to changes, like a chair being moved or a new obstacle appearing.
* Obstacle Detection and Avoidance: Algorithms ensure the robot doesn’t collide with objects. A\* or Dijkstra’s algorithm helps find the shortest path around detected obstacles. This step is crucial for smooth navigation in dynamic, cluttered spaces.

**3.4.3 Cleaning Mechanism**

The robot’s hardware must deliver effective floor cleaning.

* Brushes or Mopping Mechanism: Rotating brushes sweep dirt into a collection bin, while a mop or microfiber pad wipes surfaces. Some designs also incorporate suction for vacuuming.
* Water or Cleaning Solution Dispensing: A small pump system sprays water or cleaning fluid onto the surface before the mop engages. Controlled dispensing prevents excess water usage and ensures even coverage.
* Dirt Detection and Adaptive Cleaning: Capacitive or optical sensors can detect particularly dirty areas (e.g., mud spots), prompting the robot to perform focused cleaning or multiple passes in those regions.

**3.3.4 Energy Management**

Without good energy planning, the robot may stop mid-task or fail to return to its charger.

* Energy-Efficient Design: Prioritizing low-power sensors, lightweight materials, and efficient motor control reduces battery consumption. Smart path planning also minimizes unnecessary movement.
* Solar Assistance (optional): Integrating solar panels (e.g., on top of the robot) can help recharge batteries between tasks, especially useful in bright environments like office lobbies or sunlit rooms.
* Energy Monitoring: Battery voltage sensors and power management ICs track battery levels and predict when the robot should return to its charging dock.

**3.4.5 Control System and Software Implementation**

The “intelligence” of the robot lives here.

* Microcontroller Programming: Using tools like Arduino IDE, developers write code that connects all components: reading sensor inputs, calculating paths, controlling motors, and triggering the cleaning mechanisms.
* Algorithm Implementation: Path-planning strategies (like random, spiral, or systematic grid cleaning) and obstacle-avoidance routines are implemented to guide the robot through its tasks efficiently.
* Communication Protocols: For user interaction, Bluetooth allows local wireless control (e.g., from a smartphone), while Wi-Fi enables broader control, including integration with smart home systems or cloud services.

**3.4.6 Testing and Validation**

Before releasing the system for real-world use, thorough testing ensures reliability.

* Simulation: Software tools simulate the environment and the robot’s actions, helping spot issues in navigation logic or sensor interpretation before building the prototype.
* Prototype Testing: A physical robot is tested in various real-world settings (homes, offices, uneven floors) to observe cleaning performance, navigation accuracy, battery life, and sensor reliability.
* Iterative Improvement: Feedback from testing is used to refine the design — optimizing code, tweaking hardware setups, or improving mechanical components.

**3.4.7 Integration with IoT and Remote Monitoring**

Adding smart features elevates the system’s functionality.

* Sensor Data Upload: The robot periodically uploads status reports — such as battery level, cleaning completion, or error logs — to a cloud server or local database.
* Remote Control: Users can schedule cleanings, change modes, or check progress from anywhere via a mobile app or web portal.
* Real-Time Feedback: Push notifications or app updates inform users about tasks (e.g., “cleaning complete,” “low battery,” “obstacle detected”) to improve user interaction.

**3.4.8 Final Evaluation and Optimization**

In this last stage, the system is polished for long-term use.

* Performance Evaluation: Data from multiple cleaning cycles is analyzed to assess how efficiently and thoroughly the robot cleans, how much energy it uses, and how satisfied users are with its operation.
* Optimization: Hardware (like motor speed), software (like better pathfinding algorithms), or user experience (like app interface improvements) are fine-tuned. Over-the-air software updates can be delivered if Wi-Fi capability is included.

**3.5 Summary**

The proposed Smart Floor Cleaning System is an automated, Bluetooth-controlled robot built using an Arduino UNO, motor driver L298N, HC-05 Bluetooth module, servo motors, and a water pump, designed to efficiently clean various surfaces with both wet and dry functionalities. Its system architecture integrates key components such as microcontrollers, sensors, motors, and a mobile app interface for remote operation, allowing users to select cleaning modes, control movement, and monitor progress in real time. The robot employs path planning and navigation techniques, including obstacle detection and algorithms like SLAM and A\*, to ensure thorough and energy-efficient coverage. The cleaning mechanism combines brushes, mopping, and controlled water dispensing, while energy management focuses on low-power design and smart battery monitoring. The software side involves microcontroller programming for sensor integration, movement control, and communication protocols, with extensive testing through simulations and real-world trials to refine performance. With potential IoT integration, users can remotely schedule and monitor cleaning, receiving real-time feedback, and the system is optimized through iterative improvements for enhanced efficiency, scalability, and long-term usability across home and commercial environments.