Did some research on micro controllers to choose best micro controller that suitable for our project and we selected ESP32 microcontroller as our final choice.

Microcontroller Research for Smart Lighting System

To build a reliable and efficient smart lighting system, selecting the right microcontroller is crucial. Below is a detailed research report on various microcontroller options suitable for your project:

1. Key Requirements for the Microcontroller

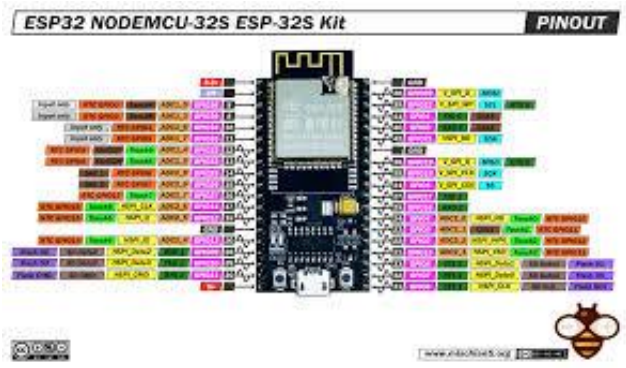
- **Connectivity:** Must support wireless communication (e.g., Wi-Fi, Bluetooth) for mobile app integration.
- **Low Power Consumption:** Essential for energy-efficient operation.
- **GPIO Availability:** Sufficient pins to connect sensors (ultrasonic) and control LED lights.
- **Processing Power:** Capable of real-time data processing for person detection and light control.
- **Compatibility:** Should easily integrate with sensors and external peripherals.
- **Cost Effectiveness:** Affordable for scalability across multiple nodes.

2. Microcontroller Options

Selected

a. ESP32

- **Features:**
  - Dual-core processor with clock speed up to 240 MHz.
  - Built-in Wi-Fi and Bluetooth connectivity.
  - GPIO pins: ~30 (varies by board version).
  - Ultra-low power consumption with sleep modes.
  - Compatible with ultrasonic sensors like HC-SR04.



- **Advantages:**
  - Ideal for IoT applications with integrated wireless features.
  - High processing power for handling multiple tasks.
  - Cost-effective and widely supported by community resources.
- **Disadvantages:**
  - Slightly more complex to program compared to simpler microcontrollers.
- **Best Fit For:**
  - Nodes in your system that control lights and communicate with the central hub.

b. ESP8266

- **Features:**
  - Single-core processor with clock speed of 80–160 MHz.
  - Built-in Wi-Fi.
  - GPIO pins: ~11.
  - Low power consumption.
- **Advantages:**
  - Extremely cost-effective for simple IoT projects.
  - Compact size makes it suitable for light nodes.
- **Disadvantages:**
  - Limited GPIO pins and processing power compared to ESP32.
  - No native Bluetooth support.
- **Best Fit For:**
  - Basic light nodes in a small-scale system.

c. Arduino Nano 33 IoT

- **Features:**
  - 32-bit ARM Cortex-M0+ processor.
  - Built-in Wi-Fi and Bluetooth Low Energy (BLE).
  - GPIO pins: ~20.
  - Onboard IMU for motion sensing (if needed for advanced features).
- **Advantages:**
  - Compatible with the Arduino ecosystem for easy programming.
  - Reliable wireless communication.
- **Disadvantages:**
  - Higher cost compared to ESP series.
  - Less community support than ESP32.
- **Best Fit For:**
  - Advanced light nodes where motion sensing or BLE is required.

d. Raspberry Pi Pico W

- **Features:**
  - Dual-core ARM Cortex-M0+ processor.
  - Built-in Wi-Fi (Pico W model).
  - GPIO pins: ~26.
  - Cost-effective and compact design.
- **Advantages:**
  - Versatile and powerful for various applications.
  - Supports MicroPython and C/C++ programming.
- **Disadvantages:**
  - No native Bluetooth support.

    - Requires external components for advanced IoT functionality.
- **Best Fit For:**
    - Central hub for coordinating multiple light nodes.

e. STM32 Series

- **Features:**
    - ARM Cortex-M processors (variety of models).
    - Extensive GPIO and peripheral support.
    - Low power options with sleep modes.
- **Advantages:**
    - High reliability and performance for real-time systems.
    - Wide range of models for different needs.
- **Disadvantages:**
    - Steeper learning curve.
    - Higher cost for advanced models with wireless capabilities.
- **Best Fit For:**
    - Central hub or specialized nodes requiring high precision.

3. For our Smart Lighting System

Node Microcontrollers:

- **Preferred Option:** ESP32
    - Reason: Combines Wi-Fi, Bluetooth, low power consumption, and sufficient GPIO for ultrasonic sensors and LED control.

Central Hub:

- **Preferred Option:** Raspberry Pi Pico W
    - Reason: Capable of handling communication between nodes, running algorithms, and acting as a bridge to the mobile app.