



This Arduino-based security alert system demonstrates how software logic controls real-world hardware using a PIR motion sensor. The code is structured around a simple yet powerful idea: continuously monitor the environment and react instantly when motion is detected.

First, the program clearly defines the connection between the Arduino and the hardware components. By assigning fixed pin numbers to the PIR sensor, LED, and buzzer, the code ensures clean communication between the microcontroller and external devices. This approach improves readability and makes the system easy to modify or expand later.

During initialization, the Arduino is configured to understand the role of each component. The PIR sensor is set to provide input signals, while the LED and buzzer are prepared to respond as output devices. This setup step is crucial because it establishes the direction of data flow in the system.

Once the system starts running, the Arduino enters a continuous monitoring mode. It repeatedly checks the PIR sensor to determine whether motion is present. This real-time sensing allows the system to act immediately, making it suitable for security and automation applications.

When motion is detected, the code triggers a clear alert response. The LED turns on to provide a visual indication, and the buzzer emits a sound to signal intrusion or activity. If no motion is detected, the system remains silent but alert, conserving energy while staying ready for the next event.

Overall, this project is a practical example of the Input → Processing → Output model in embedded systems. Although the logic is simple, the same structure is used in advanced

applications such as smart home security, surveillance systems, and industrial monitoring. This project strengthens the understanding of sensor integration, decision-making logic, and real-time system behavior.